Design and FPGA Implementation of OFDM System with Channel Estimation and Synchronization

Hongyan Zhou

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montréal, Québec, Canada

June 2013

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Hongyan Zhou

Entitled: "Design and FPGA Implementation of OFDM System with Channel Estimation and Synchronization"

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science

Complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Chair
Dr. M. Z. Kabir

_____ Examiner, External to the Program
Dr. B. Fung (CIISE)

_____ Examiner
Dr. Y. R. Shayan

_____ Supervisor
Dr. M. N. S. Swamy

_____ Supervisor
Dr. W.-P. Zhu

Approved by: _____
Dr. W. E. Lynch, Chair
Department of Electrical and Computer Engineering

_____ _____
Dr. C. W. Trueman
Interim Dean, Faculty of Engineering
and Computer Science

ABSTRACT

Design and FPGA Implementation of OFDM System

with Channel Estimation and Synchronization

Hongyan Zhou

In wireless and mobile communications, multipath fading severely degrades the quality of information exchange. The orthogonal frequency division multiplexing (OFDM) technology is able to provide a high transmission data rate with enhanced communication performance at a relatively small bandwidth cost, together with proper estimation and compensation of channel effects. Therefore, it has been widely applied in many wireless and mobile networks, especially for the state-of-the-art communication standards. The unique structure of OFDM signals and the application of discrete Fourier transform (DFT) algorithm have significantly simplified the digital implementation of OFDM system. Among different kinds of implementations, field programmable gate array (FPGA) is a very cost-effective and highly flexible solution, which provides superior system performance and enables easy system upgrade.

In this thesis, a baseband OFDM system with channel estimation and timing synchronization is designed and implemented using the FPGA technology. The system is prototyped based on the IEEE 802.11a standard and the signals is transmitted and received using a bandwidth of 20 MHz. With the help of the quadrature phase shift keying (QPSK) modulation, the system can achieve a throughput of 24 Mbps. Moreover, the least squares (LS) algorithm is implemented and the estimation of a frequency-selective fading channel is demonstrated. For the coarse estimation of timing, a modified

maximum-normalized correlation (MNC) scheme is investigated and implemented. Starting from theoretical study, this thesis in detail describes the system design and verification on the basis of both MATLAB simulation and hardware implementation. Bit error rate (BER) verses bit energy to noise spectral density (E_b/N_0) is presented in the case of different channels. In the meanwhile, comparison is made between the simulation and implementation results, which verifies system performance from the system level to the register transfer level (RTL).

First of all, the entire system is modeled in MATLAB and a floating-point model is established. Then, the fixed-point model is created with the help of Xilinx's System Generator for DSP (XSG) and Simulink. Subsequently, the system is synthesized and implemented within Xilinx's Integrated Software Environment (ISE) tools and targeted to Xilinx Virtex-5 board. What is more, a hardware co-simulation is devised to reduce the processing time while calculating the BER for the fixed-point model.

The present thesis is an initial work on the implementation part of an collaborative research and development (CRD) project of the Natural Sciences and Engineering Research Council of Canada (NSERC) sponsored by the WiTel Technologies, Ontario. It is the first and foremost step for further investigation of designing innovative channel estimation techniques towards applications in the fourth generation (4G) mobile communication systems.

ACKNOWLEDGEMENTS

First of all, I would like to express my sincere gratitude to my thesis supervisors, Dr. M. N. S. Swamy and Dr. Wei-Ping Zhu, who offered me an opportunity to pursue the master studies at Concordia University and directed me towards academic research. Without their consistent support, continuous inspiration and motivating guidance, this thesis could not have been completed.

I am very grateful to Mr. Tadeusz Obuchowics and Dr. Chengjun Zhang for their selfless help in resolving hardware problems. I would also like to thank my colleagues, in particular Miss Guanqun Gu and Mr. Yang Fu, for their individual help.

I would like to express my appreciation to the members of the examining committee for reviewing my thesis and providing precious advice.

My special thanks goes to the *Fonds de recherché sur la nature et les technologies* (*FQRNT*) and the *Regroupement stratégique en microsystème du Québec* (*ReSMiQ*) which sponsored my international internship at the Santa Clara University (SCU) during my master study. I am also indebted to Dr. Tokunbo Ogunfunmi at the SCU, for providing me the opportunity for the exchange study in his lab. My gratitude extends to all the lab members, especially Dr. JeoongSung Park and Dr. Thomas Paul.

My deepest gratitude goes to my parents who gave me birth and brought me up, as well as my sister. In addition, I am very thankful to my husband, Dr. Liang Han, who has been providing continuous inspiration and support throughout my study. Last but not

least, I would like to dedicate this thesis to my father who passed away during my graduate studies.

Table of Contents

List of Figures

List of Tables

List of Acronyms

3GPP	Third Generation Partnership Project
4G	Fourth Generation
ADSL	Asymmetric Digital Subscriber Line
ASIC	Application Specific Integrated Circuit
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CIC	Cascaded Integrator-Comb
CLB	Configurable Logic Block
CP	Cyclic Prefix
DAB	Digital Audio Broadcasting
DD	Decision-Directed
DFT	Discrete Fourier Transform
DSP	Digital Signal Processor
DVB	Digital Video Broadcasting
FDM	Frequency Division Multiplexing
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
HDSL	High-bit-rate Digital Subscriber Line
IC	Integrated Circuit
ICI	Inter-Carrier Interference

IDFT	Inverse Discrete Fourier Transform
IFFT	Inverse Fast Fourier Transform
IP	Intellectual Property
ISI	Inter-Symbol Interference
LCA	Logic Cell Array
LOS	Light-of-Sight
LS	Least Squares
LTE	Long Term Evolution
LTS	Long Training Symbol
MAC	Medium Access Control
Mbaud	Mega Symbol per Second
MC	Maximum-Correlation
MIMO	Multiple-Input, Multiple-Output
ML	Maximum-Likelihood
MMSE	Minimum-Mean-Square-Error
MNC	Maximum-Normalized Correlation
MSE	Mean Square Error
MUX	Multiplexer
NLOS	Non-Line-of-Sight
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
PA	Pilot-Assisted
PDF	Probability Density Function

PHY	Physical Layer
PLCP	Physical Layer Convergence Protocol
PLD	Programmable Logic Device
PMD	Physical Medium Dependent
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RAM	Random-Access Memory
RF	Radio Frequency
ROM	Read-Only Memory
RTL	Register Transfer Level
SNR	Signal to Noise Ratio
STS	Short Training Symbol
VDSL	Very-high-speed Digital Subscriber Line
VHDL	VHSIC hardware description language
VHSIC	Very-High-Speed Integrated Circuits
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Access Networks
ZF	Zero Forcing

List of Symbols

β	Roll-off factor of raised cosine filter
N	Number of subcarriers
T	Effective OFDM symbol period
T_g	Guard interval
T_{sym}	OFDM symbol period
B_w	Available channel bandwidth
σ_τ	Delay spread of the channel
Δf	Subcarrier spacing
N_g	Number of guard samples
N_d	Number of data subcarriers
N_p	Number of pilot subcarriers
N_u	Number of used subcarriers
E_b	Bit energy of transmitted signal
E_s	OFDM symbol energy
N_0	Noise spectral density
P_b	Probability of bit error
σ_z^2	Power of AWGN noise
σ_s^2	Power of signal
c	Speed of light
v	Velocity of the moving receiver
λ	Wavelength of the signal

f_c	RF carrier frequency
T_c	Coherence time
B_d	Doppler spread
f_d	Maximum Doppler shift
B_c	Coherence bandwidth
T_s	Sampling period
B_s	Sampling rate
h_l	Multipath channel tap gain
L	Channel length
K	Rician factor
S_t	Pilot spacing in time domain
S_f	Pilot spacing in frequency domain
D	Length of the delay window
W	Length of the sliding window
$\mathbf{S}$	Frequency response of the transmitted signal
$\mathbf{R}$	Frequency response of the received signal
$\mathbf{H}$	Frequency impulse response of the fading channel
$\mathbf{Z}$	Frequency response of the additive noise
$\hat{\mathbf{H}}$	Estimate of the fading channel response
$I_0(\bullet)$	Zeroth-order modified Bessel function of the first kind
$J(\bullet)$	Cost function
$C(\bullet)$	Correlation function

$P(\bullet)$	Power of delayed signal
$M(\bullet)$	Timing metric
$\delta(\bullet)$	Dirac delta function
$(\bullet)^*$	Complex conjugation
$[\bullet]^T$	Matrix transpose
$[\bullet]^H$	Conjugate transpose
$E[\bullet]$	Expectation
$Var[\bullet]$	Variance
$diag[\bullet]$	Diagonal matrix

Chapter 1 Introduction

1.1 Research Background

In order to meet the unprecedented requirements for high quality of service (QoS) and high data rate communication as well as emerging multimedia services, telecommunication professionals are currently working towards the fourth generation (4G) wireless communication systems. The orthogonal frequency division multiplexing (OFDM), one of the most promising technologies, has raised a great deal of attention in view of the rapid development of digital signal processing techniques and circuits in recent years [1]-[3].

OFDM was originally proposed in 1960s as a parallel data transmitting scheme. The basic idea is to divide a single high rate data stream into a number of lower-rate data streams. Each of these data streams is modulated on a specific carrier, which is called subcarrier, and transmitted simultaneously. Such a kind of multi-carrier modulation preserves the robustness against the effects of multipath fading. Moreover, these subcarriers are orthogonal to one another so as to enhance the spectrum efficiency compared to that in conventional multi-carrier transmission. Since the data streams carried by each subcarrier are separated by frequencies, OFDM is also considered as a frequency division multiplexing (FDM) scheme [3].

At the early development stage [4]-[7], traditional methods as that used in single-carrier modulation were applied to implement OFDM modem, which require a number of sinusoidal subcarrier oscillators and multipliers in the modulator and banks of correlators

in the demodulator. The implementation complexity limited the development of OFDM until 1971, when the discrete Fourier transform (DFT) was applied to this technology [8]. The DFT significantly simplified the modulation and demodulation processes and made it practical to implement the baseband OFDM modem in a digital manner. From then on, more and more applications of OFDM have been investigated in practice.

In 1980s, OFDM was widely studied in such areas as high-density recording, high-speed modems, and digital mobile communications [9]-[12]. Since 1990s, OFDM has been employed in wideband data transmission. Applications of OFDM technology include asymmetric digital subscriber line (ADSL), high-bit-rate digital subscriber line (HDSL), and very-high-speed digital subscriber line (VDSL) in wired systems, and digital audio broadcasting (DAB), digital video broadcasting (DVB) in wireless systems. Furthermore, it has also been recognized as the basis of the wireless local area network (WLAN) standards [13]-[14], among which the IEEE 802.11a standard is one of the most important ones.

Recently, high data rate and high QoS have been two main topics in wireless and mobile communications, which require communication systems to be capable of adapting to fast varying channel conditions and providing a steady communication environment to various kinds of users at a high speed of data transmission. Due to its advantages of providing high data rate and less sensitivity to fast channel fading, OFDM technology, in combination with other powerful techniques such as the multiple-input, multiple-output (MIMO) technique, has been the mainstream of wireless and mobile networks. It has been used or at least considered as a viable candidate in various state-of-the-art applications, such as wireless fidelity (Wi-Fi), worldwide interoperability for microwave

access (WiMAX), and the third generation partnership project (3GPP) long term evolution (LTE). For example, orthogonal frequency division multiple access (OFDMA), a multiple-access scheme, in which each subcarrier is assigned to deliver information for an individual user, has been employed in 3GPP LTE and WiMAX [1].

With the advanced development of digital integrated circuits (ICs), the high flexibility and low complexity of digital implementation of OFDM modem has boosted its application. Among numerous sorts of the digital IC technologies, field programmable gate array (FPGA) has attracted the most attention in recent years due to its superior performance and high flexibility. As a general-purpose IC, FPGA is an array of gates that can be reconfigured by the designer as a versatile design platform. It is developed based on the programmable logic devices (PLDs) and the logic cell array (LCA) concept. By providing a two-dimensional array of configurable logic blocks (CLBs) and programming the interconnection that connects the configurable resources, FPGA can implement a wide range of arithmetic and logic functions [16]. Compared to other popular IC technologies such as application specific integrated circuits (ASICs) and digital signal processors (DSPs), FPGA has the following advantages [15]-[18]:

- **Performance**

Through its inherent high parallel architecture, FPGA has the ability to exceed the speed limit of sequential execution technologies and is able to process data at a much higher speed than DSP processors and microprocessors whose performance is determined by the system clock rate. In addition, with the development of IC fabrication and the degree of circuit integration, the scale of gates in FPGA has become larger and larger.

Therefore, it can achieve much higher performance in various applications that requires large arithmetic resources, such as OFDM.

- **Cost**

Because of its reprogrammable nature, FPGA provides a cost-effective solution for system development. It can be easily customized and reconfigured so that effectively versatile functionality can be realized using FPGA and there is no need to kick off design for each application, compared to some application specific devices, such as the ASIC. Therefore, the non-recurring engineering cost and the development cycle of FPGA is relatively low. Moreover, the power consumption of FPGA can be minimized if it is designed appropriately.

- **Reliability**

The FPGA resources can be separated by task requirements. The tasks interact through the well-defined interfaces between one another, which greatly reduces unexpected interaction. The change for one task will not affect the execution of other tasks. This high isolation and high parallelism mechanism not only minimize the reliability concerns, but also reduce the deterministic hardware dedicated to every task.

- **Flexibility**

Since the functionality of FPGA is not fixed by the manufacturer, but defined by the designer, it allows easy system maintenance and quick upgrade with simple prototype changes. By reconfiguration even on site, FPGA requires very low prototyping costs and avoid the compatibility problem.

- **Time to market**

Due to the reconfiguration and programming flexibility, as well as fast prototyping, it is easy for the designers to develop, verify and debug the devices without complex modeling and fabrication process. With the help of some popular development tools, the learning and developing period is further shortened and the products using FPGA are getting to market quickly.

Especially, the enrichment of intellectual property (IP) core libraries as well as advancement of design tools have simplified the development process for communication and signal processing engineers and also provided an environment where they only need to focus on system development at a high level.

All in all, FPGA has become the most important candidate for implementing OFDM technology in wireless and mobile communication systems.

1.2 Motivation

In practical wireless communication systems, the signal is not transmitted over ideal channels. It is attenuated and especially distorted by multipath propagation through transmission. However, the fading effects cannot be compensated by increasing the transmitted signal power. Therefore, in order to realize reliable communication, it is very critical to estimate the fading channel and equalize the channel effects in wireless communication systems. The objective of this thesis is to design and implement an OFDM system with channel estimation and synchronization using the FPGA technology.

Design and implementation of OFDM-based wireless communication system has been studied for many years. Most of the works have been focusing on specific areas of the implementation of OFDM system using FPGA. In [19], Chris overviewed the implementation of an OFDM transceiver at a high level with focus on certain topics in the receiver design, such as the synchronization, packet detection, channel estimation and equalization. [20], [21] and [22] are respectively focused on the development of OFDM modulator, transmitter and receiver. In addition, OFDM transceivers design for the AWGN channel have been presented in [23]-[25]. However, there hasn't been a comprehensive work presenting a complete development of OFDM system with channel estimation and synchronization using the FPGA technology.

The main objective to this work is to design and implement a complete baseband OFDM system using a top-down approach, and demonstrate the system performance for various sorts of channel conditions. System synchronization will also be discussed in this thesis. Moreover, it concentrates on the design and implementation of channel estimation and equalization, while a verification at system level is performed. The detailed objectives include:

- To design, model and implement a baseband OFDM system including both the transmitter and the receiver, and to analyze the system performance.
- To prototype an OFDM system based on a specific wireless communication standard.
- To implement the synchronization and channel estimation system for the receiver and provide system evaluation under different channel conditions.

The present thesis is an initial work on the implementation part of an collaborative research and development (CRD) project of the Natural Sciences and Engineering Research Council of Canada (NSERC) sponsored by the WiTel Technologies, Ontario. This project will lay a solid foundation for further investigation of innovative channel estimation techniques towards applications in 4G wireless communication systems.

1.3 General Design and Implementation Methodology

The proposed system is designed using a top–down system design approach and targeted to the IEEE 802.11a standard. System performance will be presented and compared between different channel models. Figure 1.1 shows the design flow, which includes four major steps.

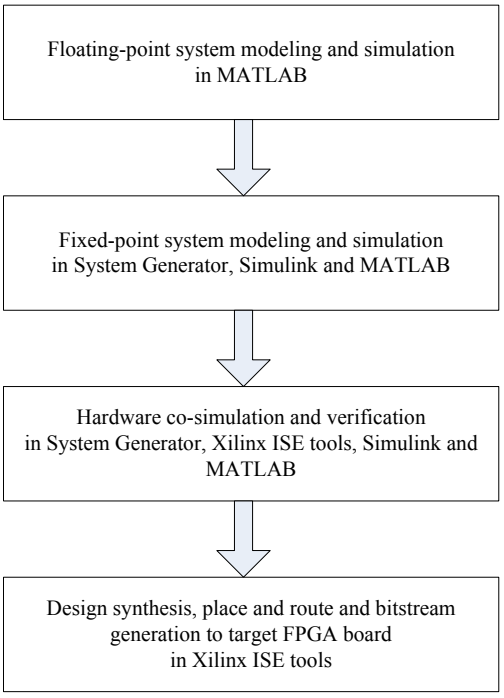


Figure 1.1 System design flow.

1) Floating-point system modeling and simulation

First of all, a general OFDM system is designed and verified with a floating-point model based on the mathematical analysis using MATLAB. System modeling is composed of two steps. Firstly, the model for a basic OFDM system including the transmitter and the receiver is established and the bit error rate (BER) under an ideal channel with additive white Gaussian noise (AWGN) is calculated. According to the physical layer (PHY) specification of the IEEE 802.11a, the information to be transmitted is modulated using a quadrature phase shift keying (QPSK) scheme. Secondly, a number of synchronization and channel estimation techniques are studied and the BERs under Rayleigh fading channel are calculated and compared. A modified maximum-normalized correlation (MNC) scheme that is suitable for burst OFDM applications like the IEEE 802.11a is chosen for system synchronization. Considering application scenarios and the implementation complexity, the least squares (LS) algorithm is adopted for channel estimation and compensation.

2) Fix-point system modeling and simulation

After MATLAB simulation, the OFDM system is converted into hardware models with fixed-point representation. This step is completed within a visualized platform embedded in the Simulink, namely the Xilinx's System Generator for DSP (XSG). The system modeling starts from a general system architecture. Each functional block of the system is designed by a relatively independent subsystem, and therefore this platform is versatile and can be applied to various standards by modifying specific subsystems slightly, if necessary. Then, system functions are verified through comprehensive

simulations. In this case, the BERs under AWGN and Rayleigh fading channels are calculated and compared with those of the floating-point model.

3) Hardware co-simulation and verification

In order to verify the system performance at a hardware level, a co-simulation of software and hardware platforms, which is called hardware co-simulation, is performed to simplify the hardware verification and accelerate simulation process. This methodology provides a hardware-in-the-loop verification, in which the inputs and test vectors are generated by Simulink and the system simulation is actually carried out on the FPGA platform. On the other hand, VHDL (Very-High-Speed Integrated Circuits, or VHSIC hardware description language) codes are generated and then analyzed with Xilinx's Integrated Software Environment (ISE) tools to make further system verification at a register transfer level (RTL).

4) Design synthesis, place and route and bitstream generation

Design synthesis is performed with Xilinx's XST when the generation of VHDL codes is completed. After place and route, a gate-level netlist is generated and can be back-annotated to perform system optimization and verification. Finally, a bitstream is generated to program the target FPGA board [26]. The hardware co-simulation and implementation process will be discussed in detail later in Chapter 3.

1.4 Contributions of the Thesis

In this thesis, a complete and comprehensive development procedure of an OFDM system with focus on channel estimation is presented in a systematic and top-down manner. The major contributions include the following aspects.

- A baseband QPSK OFDM system including both the transmitter and the receiver is designed and implemented using the FPGA technology. The system is developed by a number of function blocks based on a mathematical model of the OFDM system. Various modulation schemes can be directly applied without modifying other function blocks.

- The OFDM system is designed based on the IEEE 802.11a standard, and it is very flexible and can be used to prototype other standards that employ the OFDM technology with slight modifications.

- Channel estimation and equalization for receiver is implemented. The LS algorithm is performed for the purpose of both ensuring a good system performance and reducing circuit complexity. The BER performance is compared between MATLAB simulation and hardware co-simulation under the AWGN channel as well as the Rayleigh channel.

- Various synchronization techniques are introduced. Also, a modified MNC algorithm is implemented and tested for a packet transmission scenario.

- Hardware co-simulation is used to provide a cost-effective method for system verification and performance evaluation.

1.5 Organization of the Thesis

This thesis is organized as follows.

Chapter 2 introduces the system design and MATLAB simulation of a basic OFDM system. The fundamentals of OFDM are first introduced through theoretical derivations, and it is then followed by the system parameters specified in the IEEE 802.11a PHY, which is used in the proposed design. This chapter ends with the MATLAB simulation results for the floating-point model.

Chapter 3 presents the FPGA implementation for a baseband OFDM system. It starts with the system modeling of the basic OFDM functional blocks at the transmitter and the receiver in XSG. For each function block, simulation results are presented for system verification. Moreover, the BER performance under AWGN channel for this hardware model is calculated and compared with that of the floating-point model in Chapter 2. Finally, the implementation results and resource consumption are summarized.

Chapter 4 focuses on the design and implementation of the most important part of this work, the channel estimation and equalization subsystem. Several kinds of channel estimation techniques for OFDM systems are presented. The LS algorithm is applied and implemented at the receiver for its low complexity. For system validation, the BER performance under various channel models is simulated in MATLAB and compared. In addition, the fixed-point model is designed and the BER results under the IEEE 802.11 channel is compared between MATLAB simulation and hardware implementation. This chapter concludes with a summary of the hardware resource consumption.

Chapter 5 introduces another important tpoic for OFDM communication, namely, synchronization. It begins with the introduction and comparison of several frame detection and time synchronization methods. Then, the modified MNC scheme is investigated, and the hardware modeling and implementation is proposed. The simulation results are compared between various communication environment for both floating-point and fixed-point models. The summary of implementation results and resource consumption are also given.

Chapter 6 concludes the thesis and provides an outlook for future work.

Chapter 2 Design and Simulation of a Fundamental OFDM System

This chapter gives a brief review of the fundamentals of OFDM modulation scheme and describes the system design process. First, the principle and mathematical model of a baseband OFDM system are studied. Then, the specifications about OFDM modulation technique for the PHY of IEEE 802.11a standard and its architecture are introduced. On the basis of this, the floating-point model is finally described and simulation results are provided.

2.1 Fundamentals of OFDM Systems

2.1.1 OFDM v.s. FDM

As mentioned in Chapter 1, OFDM can be considered as a special case of the FDM scheme. Figure 2.1 shows the typical spectrums of FDM and OFDM systems. In an FDM system, the entire signal band is partitioned into several sub-bands. Each low-rate data stream carrying information for an individual user is modulated onto one sub-band, which is also called a subcarrier. Then all the subcarriers are combined and a high rate signal is achieved. Since the symbol duration increases significantly, the signal is more robust against the time dispersion of fading channels. This is also the basic idea of OFDM.

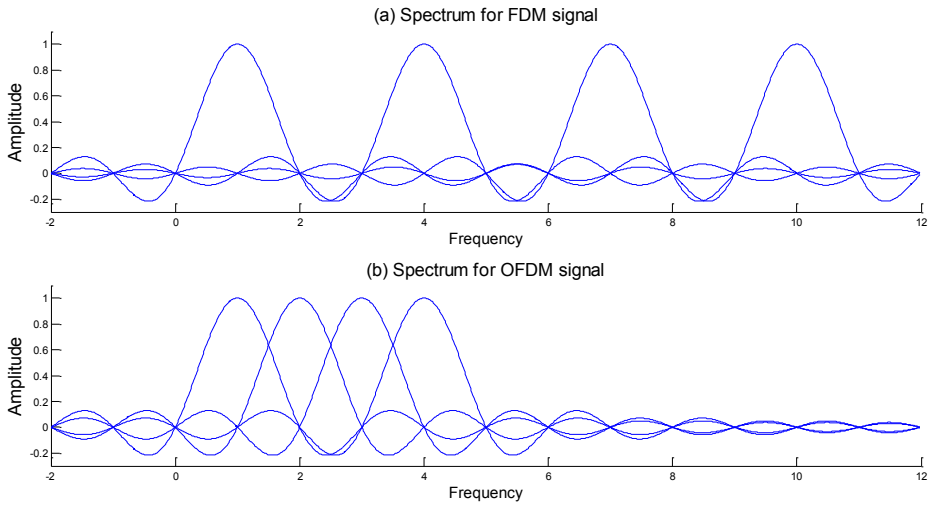


Figure 2.1 Subcarriers in FDM and OFDM systems.

For avoiding interference between sub-bands and allowing the bandpass filters at the receiver to easily separate the indicated user bands, guard bands are inserted between the adjacent sub-bands. The theory of FDM is shown in Figure 2.1(a), where the spectra of each subcarrier, or the mainlobe of it, is separated from one another. This method achieves network security at a high cost of bandwidth efficiency. OFDM, is a good solution that not only ensures the orthogonality between the subcarriers and preserves the advantages of FDM, but also deals with the problem of bandwidth inefficiency.

As indicated in Figure 2.1(b), the subcarriers in an OFDM signal are arranged in such a way that the peak of each subcarrier exactly shows up at the point where all the others have zero amplitude. Unlike the FDM, the subcarriers are overlapped so that the bandwidth is saved. From Figure 2.1, it can be observed that OFDM technique can reduce the bandwidth to at least half of that of FDM [2]. However, they do not interfere with one another due to the inherent orthogonality between these subcarriers. This orthogonality property is satisfied when the spacing between the center frequencies of

two adjacent subcarriers is the reciprocal of the OFDM symbol period. In the frequency domain, the subcarriers should be integer multiples of the spacing frequency, i.e., for an OFDM symbol of period T, any two subcarriers follow

$$\int_0^T \exp\left(j2\pi \frac{i-m}{T} t \right) dt = \begin{cases} 0 & i \neq m \\ T & i = m \end{cases} \tag{2.1}$$

and are orthogonal to each other within the symbol period T. A typical OFDM waveform with four subcarriers is shown in Figure 2.2.

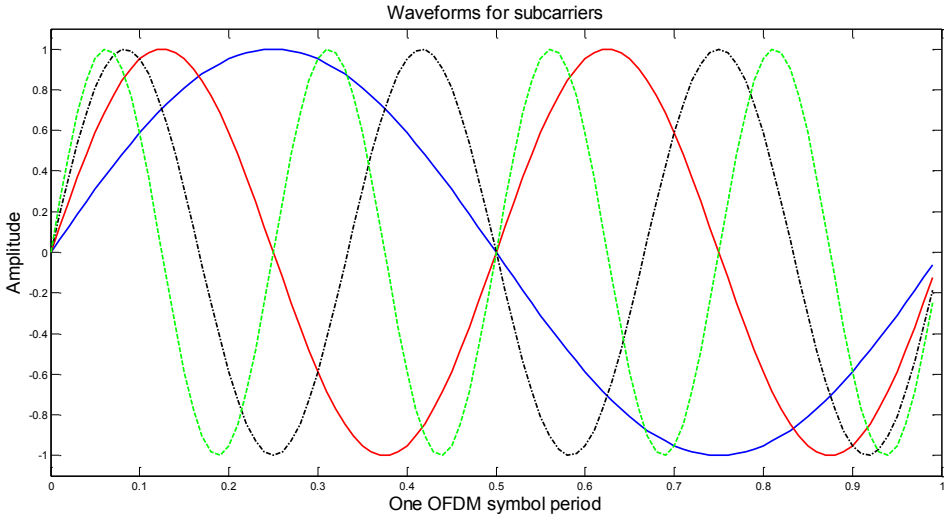


Figure 2.2 OFDM waveform in time domain.

As described before, the basic idea of OFDM is to divide and transmit a single high-rate data stream over a number of low-rate subcarriers, where all the subcarriers are orthogonal to one another. Assume each subcarrier applies the same modulation method, in the case of the same data rate, the OFDM signal with N subcarriers elongates the signal duration to N times of that for a single carrier system, while the signal bandwidth for these two systems are identical. On the other hand, in the case of the same signal

duration, the data rate of an OFDM system increases to N times at a very low cost of the bandwidth increase.

In practical wireless communication systems, signals should arrive at the receiver from numerous different paths because of the existence of obstacles along its propagation. As mentioned before, one important reason why OFDM is so popular is that it deals with the delay spread caused by multipath fading and efficiently eliminates inter-symbol interference (ISI) and inter-carrier interference (ICI). To achieve this goal, a cyclically extended prefix is introduced at the transmitter with a low cost of bandwidth efficiency degradation.

Therefore, OFDM holds the following advantages:

- High data rate by transmitting on a number of subcarriers simultaneously.
- High spectral utilization by spacing the subcarriers very closely.
- High robustness against time dispersion of multipath channel by elongating symbol duration.
- High resistance to ISI and ICI by introducing cyclic prefix (CP).
- Less implementation complexity by using forward and inverse fast Fourier transform (FFT/IFFT) pair.
- Simple equalization to remove channel effects in the frequency domain.

2.1.2 OFDM System Architecture

In general, an OFDM signal can be considered as a summation of many OFDM symbols, which are continuous in the time domain. It has the following general form [27]:

$$s(t) = \sum_{k=-\infty}^{+\infty} s_k(t) \tag{2.2}$$

where $s_k(t)$ is the *k-th* OFDM symbol which starts at time $t = t_s$. As a multi-carrier transmission scheme, the mathematical model of an OFDM signal is generalized by summing a series of digitally modulated subcarriers. Each individual subcarrier is modulated using the phase shift keying (PSK) or quadrature amplitude modulation (QAM) and transmitted in parallel. Therefore, an OFDM symbol can be expressed by [2]:

$$s_k(t) = \begin{cases} \mathrm{Re}\left\{ \sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} d_{i+\frac{N}{2}} \exp\left[j2\pi \left(f_c - \frac{i+0.5}{T} \right)(t - t_s) \right] \right\}, & t_s \le t < t_s + T \\ 0, & otherwise \end{cases} \tag{2.3}$$

where T is the symbol duration, N is the number of subcarriers, f_c is the signal carrier frequency on the radio frequency (RF) band, and d_i is the complex value for PSK or QAM modulated symbol. If A_i and ϕ_i are the amplitude and phase of the *i-th* subcarrier respectively, d_i is defined as [3]:

$$d_i = A_i exp(j\phi_i) = I_i + jQ_i \tag{2.4}$$

where

$$\begin{cases} I_i = A_i \cos(\phi_i) \\ Q_i = A_i \sin(\phi_i) \end{cases} \tag{2.5}$$

I_i and Q_i being the in-phase and quadrature part of d_i, respectively.

Without loss of generality, the baseband form will be used for our analysis in this thesis. The complex envelope of an OFDM signal given by the following equation is used as the baseband notation.

$$s_k(t) = \begin{cases} \sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} d_{i+\frac{N}{2}} \exp\left[j2\pi \frac{i}{T}(t-t_s) \right], & t_s \le t < t_s + T \\ 0, & otherwise \end{cases} \tag{2.6}$$

The real and imaginary parts of (2.6) are the in-phase (I) and quadrature (Q) of the baseband OFDM signal. Consequently, they have to be respectively multiplied by a cosine and a sine waveform with the desired carrier frequency to generate the corresponding passband signal. At the receiver, each subcarrier is demodulated by downconverting the signal with a subcarrier of the desired frequency and integrating over the symbol period. For example, the complex value for the *m-th* subcarrier d_m is obtained using (2.7) below, where the entire signal is first multiplied by the frequency of m/T, and then integrated over the symbol period T.

$$\begin{aligned} &\int_{t_s}^{t_s+T} s_k(t) \exp\left[-j2\pi \frac{m}{T}(t-t_s) \right] dt \\ &= \int_{t_s}^{t_s+T} \left[\sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} d_{i+\frac{N}{2}} \exp\left(j2\pi \frac{i}{T}(t-t_s) \right) \right] \exp\left[-j2\pi \frac{m}{T}(t-t_s) \right] dt \\ &= \sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} d_{i+\frac{N}{2}} \int_{t_s}^{t_s+T} \exp\left[j2\pi \frac{i-m}{T}(t-t_s) \right] dt \\ &= T d_{m+\frac{N}{2}} \end{aligned} \tag{2.7}$$

As indicated by (2.7), the integral is zero for all the subcarriers other than the desired one. Then the desired output for the signal demodulation, $d_{m+\frac{N}{2}}$ multiplied with a constant factor T, is exactly the integration for the *m-th* subcarrier.

Since each subcarrier has an exact integer number of cycles within OFDM symbol duration, the orthogonality between subcarriers is guaranteed. In addition, the spacing between adjacent subcarriers has to be $1/T$. That means the number of cycles for subcarriers is incremental within the OFDM symbol period T. When the OFDM symbol is sampled with a sampling period T/N, the mathematical model for discrete time signal is given by:

$$s(n) = s_k(nT/N) = \sum_{i=0}^{N-1} d_i \exp\left(j2\pi \frac{in}{N}\right), \quad n = 0,1,\cdots,N-1 \tag{2.8}$$

which is exactly the representation of an inverse DFT (IDFT) for PSK or QAM symbols.

In practice, the FFT and IFFT algorithms are used to reduce the computational complexity. In general, a direct computation of an N-point DFT requires N^2 complex multiplications and $N(N-1)$ complex additions [3]. By using radix-2 FFT algorithm, the calculation can be reduced to $\frac{N}{2}(\log_2 N - 1)$ complex multiplications and $N\log_2 N$ complex additions. Although the radix-4 algorithm requires the same number of additions, the number of multiplications can be even further reduced to $\frac{3}{8}N(\log_2 N - 2)$. Moreover, it usually does not require full multiplication in FFT, but phase rotation performed by **co**ordinate **r**otation **d**igital **c**omputer (CORDIC) algorithm [27].

Due to the multipath fading effects, the received signal is a sum of a series of transmitted signals with different delays and attenuation, which causes time dispersion and thus ISI. One solution is to insert a guard interval before each OFDM symbol in the time domain. However, if the signal is simply zero padded within the guard interval, two subcarriers are no longer orthogonal to each other, since there is no integer number of cycle difference between them within the FFT interval [2]. The ICI introduced is eliminated by cyclically extending the OFDM symbol during the guard interval. The duration of the guard interval is selected to be larger than the maximum delay spread of the channel.

According to the above analysis, the basic architecture for a baseband OFDM system that contains the essential parts is shown in Figure 2.3.

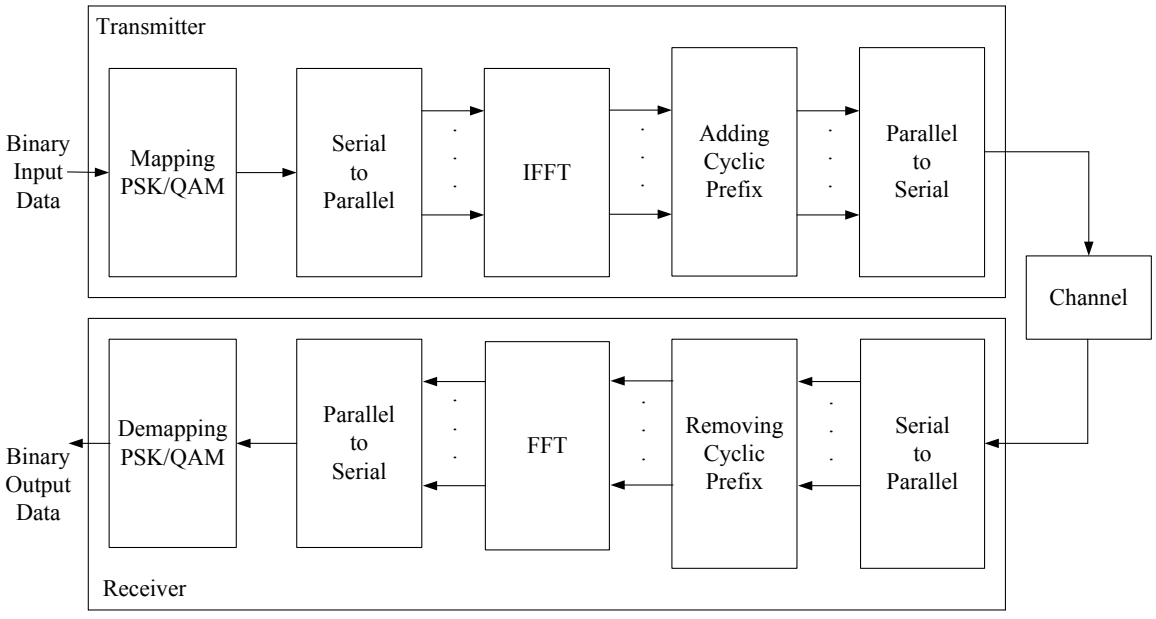


Figure 2.3 Basic architecture of a baseband OFDM system.

To avoid sharp phase transitions at the symbol boundaries, normally another block, which is windowing for pulse shaping, is added in a realistic OFDM system. As a result, spectrum rolls off rapidly and spectrum utilization is greatly improved. After windowing, the baseband OFDM symbol can be rewritten as:

$$s_k(t) = \begin{cases} w(t-t_s) \sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} d_{i+\frac{N}{2}} \exp\left[j2\pi \frac{i}{T}(t-t_s-T_g) \right], & t_s \le t < t_s + (1+\beta)T_{sym} \\ 0, & otherwise \end{cases} \quad (2.9)$$

where T_g is the guard interval duration, $T_{sym} = T + T_g$ is the OFDM symbol period, symbol starting time $t_s = kT_{sym}$, and $w(t-t_s)$ is the pulse shaping window, which is usually a raised cosine filter, and β is the roll-off factor.

At the transmitter, the discrete samples of the signal pass through a digital-to-analog converter and then transmitted. Oversampling is performed to avoid any aliasing that may occur. In a practical communication system, as specified in the wireless standards like 802.11a, some other building blocks, like channel coding, symbol interleaving, synchronization and channel estimation, are introduced.

2.2 OFDM Specifications in IEEE 802.11a Standard

2.2.1 Introduction to IEEE 802.11 standard family

IEEE 802.11 standard family was introduced in 1997 with the development of wireless LAN and its evolution is still ongoing. In September 1999, an important amendment,

IEEE 802.11a standard, was released to meet the high data rate requirement. In this standard, a physical layer that applies OFDM modulation scheme is added. The data rate ranges from 6 Mbps to 54 Mbps and the signal is transmitted over a RF band at 5 GHz [13]. This stimulated a great amount of interest both in the academic and industrial sectors, and has led to very rapid advances. In 2003, a new standard, 802.11g was released, in which the OFDM technique is still the main technology; however, the operating frequency was changed to 2.4 GHz. In 2009, multiple antennas were taken into consideration and MIMO combined with OFDM technique was introduced in 802.11n. The 802.11n version operates at both 2.4 GHz and 5 GHz, and the channel bandwidth is increased from 20 MHz to 40 MHz. The data rate was increased significantly to 600 Mbps in the case of four MIMO streams. Recently, the IEEE standard committee has been working on a new standard called 802.11ac, which operates at 5 GHz and is able to support data rate up to gigabits per second [30].

The 802.11a PHY can be divided into two elements: the physical layer convergence protocol (PLCP) and the physical medium dependent (PMD) sublayers. PLCP maps the medium access control (MAC) frames by adding some specific fields, while the PMD transmits PLCP with OFDM modulation technique. Through the layer separation, the MAC layer is independent of the PHY layer, and therefore, any modifications in the MAC layer will not affect the implementation of the PHY layer.

In the following subsections, the detailed parametric design of an OFDM system will be presented based on an application scenario of indoor wireless communication. All the parameters are defined according to the IEEE 802.11a standard.

2.2.2 System Parametric Design

In contrast to Figure 2.3, Figure 2.4 demonstrates a more realistic architecture of a baseband OFDM system.

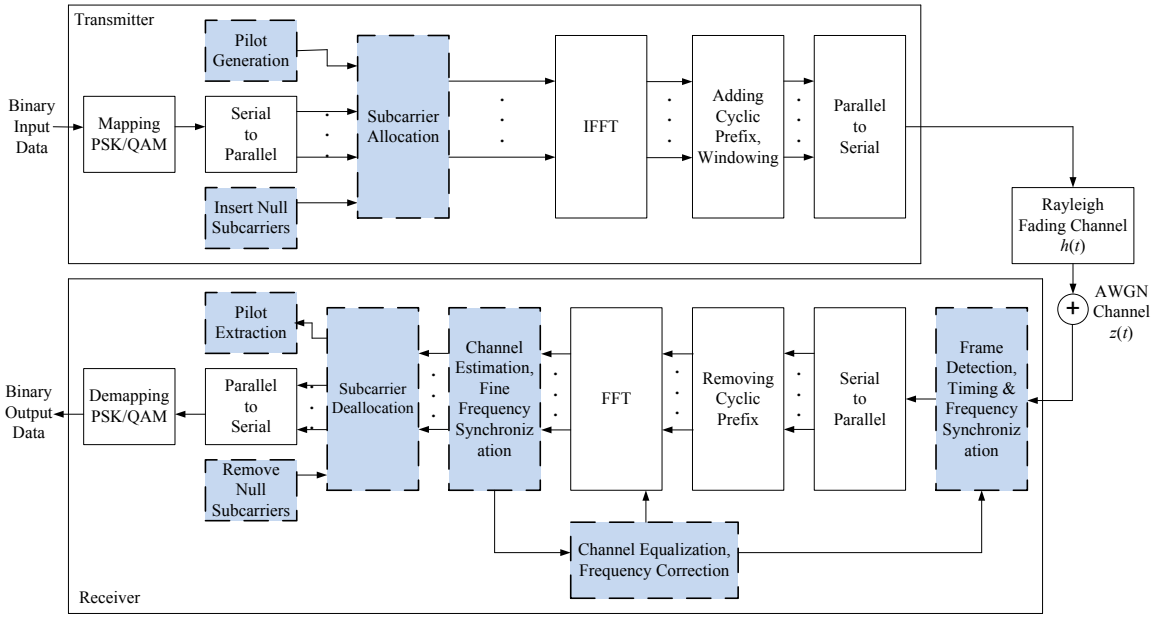


Figure 2.4 Architecture of a practical baseband OFDM system.

Compared to the fundamental architecture shown in Figure 2.3, a number of building blocks are added in Figure 2.4, marked with blue and dashed on line. At the transmitter, several "null" subcarriers are added to the data subcarriers in order to perform oversampling of the transmitted signal. Here, "null" means the symbol carried on this subcarrier has a value of zero. At the same time, pilot subcarriers used for channel estimation are also inserted. The subcarriers are allocated at the input of the IFFT block to generate a phase shift. The windowing for pulse shaping is achieved after CP extension.

At the receiver, the frame detection and synchronization for both timing and frequency is performed in the first place. Channel estimation is realized after the FFT block outputs the preambles in the frequency domain. The result is fed back to the FFT block for the equalization, which eliminates the effects of fading channel, while the fine synchronization for both timing and frequency is also added to further improve the system performance.

The following parameters should be determined ahead of a detailed system design.

- Delay spread of the channel
- Available bandwidth
- Signal data rate

It has been measured from the power delay profile that the channel has a delay spread less than 300 ns for indoor environment, even in some large buildings, shopping centers and laboratories [2]. A key parameter that determines other parameters in the system is the guard duration of 800 ns [2], which effectively protects the signal from ISI in the indoor environment and some of the outdoor wireless communication environments. The symbol duration is chosen to be five times the guard duration for limiting the power and bandwidth loss, and is set to 4.0 μs in our case. Hence, the OFDM symbol rate is 0.25 mega symbol per second (Mbaud).

The useful OFDM symbol duration without the guard interval is 3.2 μs. So the subcarrier spacing, which is the reciprocal of the useful symbol duration, can be

determined as 312.5 kHz. Assuming that there is a bandwidth of 20 MHz available, the number of subcarriers is calculated to be 64. This is exactly the same as the specification defined in IEEE 802.11a standard.

In practical systems, portion of the channel bandwidth is allocated for the following purposes:

- Pilot subcarriers are reserved for channel estimation.
- Some null subcarriers for realizing oversampling are inserted to avoid aliasing.
- Windowing is carried out to reduce the out-of-band spectral energy.

If 48 out of 64 subcarriers are used as data carriers to deliver user information and 4 subcarriers are kept as pilot, the number of occupied subcarriers is 52. In this case, when a raised-cosine window with a roll-off factor $\beta = 0.02$ is utilized, the total occupied bandwidth is

$$(1+0.02)\times(52\times 312.5kHz)\approx 16.6MHz$$

Oversampling is accomplished by appending some zeros before and after the data vector in the frequency domain as shown below.

$$\overbrace{0,\ 0,\ \ldots,\ 0,}^{1/2\ appended\ zeros}\ \underbrace{d_{-\frac{N_d}{2}},\ d_{-\frac{N_d}{2}+1},\ \ldots,\ d_{-1}}_{Negative\ subcarriers},\ \underbrace{d_1,\ d_2,\ \ldots,\ d_{\frac{N_d}{2}},}_{Positive\ subcarriers}\ \overbrace{0,\ 0,\ \ldots,\ 0}^{1/2\ appended\ zeros}$$

The nonzero data values are mapped onto the subcarriers around 0 Hz, and the zeros are mapped onto frequencies around sampling rate [2]. For symmetry, the indice of positive

subcarriers start from 1. In practice, an FFT shift is done, and consequently, null subcarriers are located in the middle of the data vector at the input of the IFFT block. The details will be described in Section 2.2.3.

If the QPSK modulation is applied on each subcarrier, each symbol for an individual subcarrier has two bits. The bit rate achieves,

$$\frac{1}{4.0\,\mu s} \times 48 \times 2 = 24\,Mbps$$

without channel coding. In terms of system design, data rate is always a system specification, based on which the modulation scheme and coding rate can be calculated. In the 802.11a standard, the data rate ranges from 6 Mbps to 54 Mbps with different coding rates and different modulation methods.

Table 2.1 summarizes the design parameters defined for a general OFDM system. These parameters are determined for an indoor wireless communication system. This assumption is well matched with the application scenario of the IEEE 802.11a standard. Therefore, the system parameters calculated are very similar to the specification for OFDM in IEEE 802.11a PHY. The difference is that channel coding is not considered in our design.

Table 2.1 System parameters defined for the proposed OFDM system.

Parameter	Description	Value
B_w	Available channel bandwidth	20 MHz
σ_τ	Delay spread of the channel	< 300 ns
T_g	Guard interval duration	0.8 μs
T_{sym}	OFDM symbol period	4.0 μs
T	Effective symbol duration (FFT period)	3.2 μs (= T_{sym} - T_g)
Δf	Subcarrier spacing	312.5 kHz (= 1 / T)
N_g	Number of guard samples	16
N	FFT size	64 (= B / Δf)
N_d	Number of data subcarriers	48
N_p	Number of pilot subcarriers	4
N_u	Number of used subcarriers	52 (= $N_d + N_p$)
B_u	Signal occupied bandwidth	16.6 MHz
R_b	Data rate without coding	24 Mbps
	Modulation type for each subcarrier	QPSK

2.2.3 IEEE 802.11a Standard Specifications

IEEE 802.11a is applied to packet based communication systems. In an OFDM frame for 802.11a PHY, a preamble is transmitted first, followed by the signal field and transmitted data. As indicated in Figure 2.5, an OFDM frame has the general form:

$$s_{OFDM}(t) = s_{preamble}(t) + s_{SIGNAL}(t - T_{preamble}) + s_{data}(t - T_{preamble} - T_{SIGNAL}) \qquad (2.10)$$

where

$$s_{preamble}(t) = s_{short}(t) + s_{long}(t - T_{short}) \tag{2.11}$$

The preamble starts with 10 short training symbols (STSs) from T_1 to T_{10}, followed by a guard interval ($GI2$) and two long training symbols (LTSs) L_1 and L_2. Both the short and long training sequences have an 8-μs duration and the entire preamble lasts for 16 μs. Then, a 3.2-μs signal symbol, as well as 800 ns guard interval is transmitted. This field bears some information necessary for the data symbols, such as the coding rate and length. Finally the various data symbols that carry user information are transmitted. Each data symbol has a duration of 4.0 μs, within which there is a 800-ns CP, as described in Section 2.2.2.

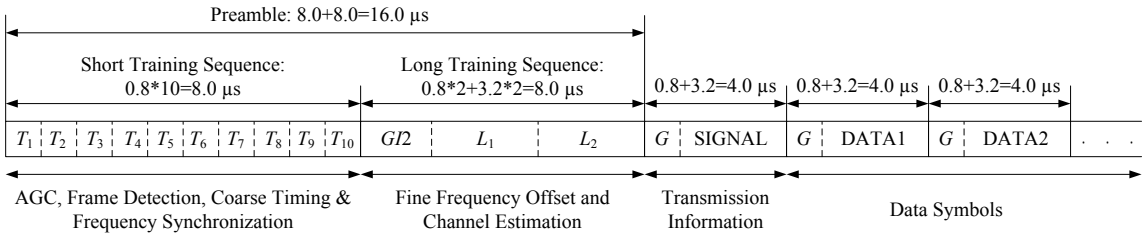


Figure 2.5 OFDM frame structure.

The short training sequence is used for AGC convergence, frame detection, as well as coarse timing and frequency acquisition. Each symbol in this sequence has a duration of 800 ns and contains 16 samples, and is identical to one another.

After the short training sequence is transmitted, a 1.6-μs guard interval that contains 32 samples is introduced. The LTS is cyclically extended within this interval. Then two identical LTSs with the same duration of 4.0 μs are followed. The long training sequence is used for fine frequency offset estimation as well as channel estimation.

From the analysis in the previous subsections, 52 subcarriers are used for an OFDM data symbol. Oversampling is achieved by adding 12 null subcarriers in order to eliminate aliasing which might occur during digital to analog conversion. Because FFT shift is performed, the null subcarriers with a value of zero are located in the middle of the input vector for the IFFT block. The subcarriers with index number from -26 to +26 are allocated as shown in Figure 2.6. Note that dc carrier is not used to transmit data. The short and long training sequences can also be applied to this mapping rule, since they both have a length of 52 samples with frequency index from -26 to +26.

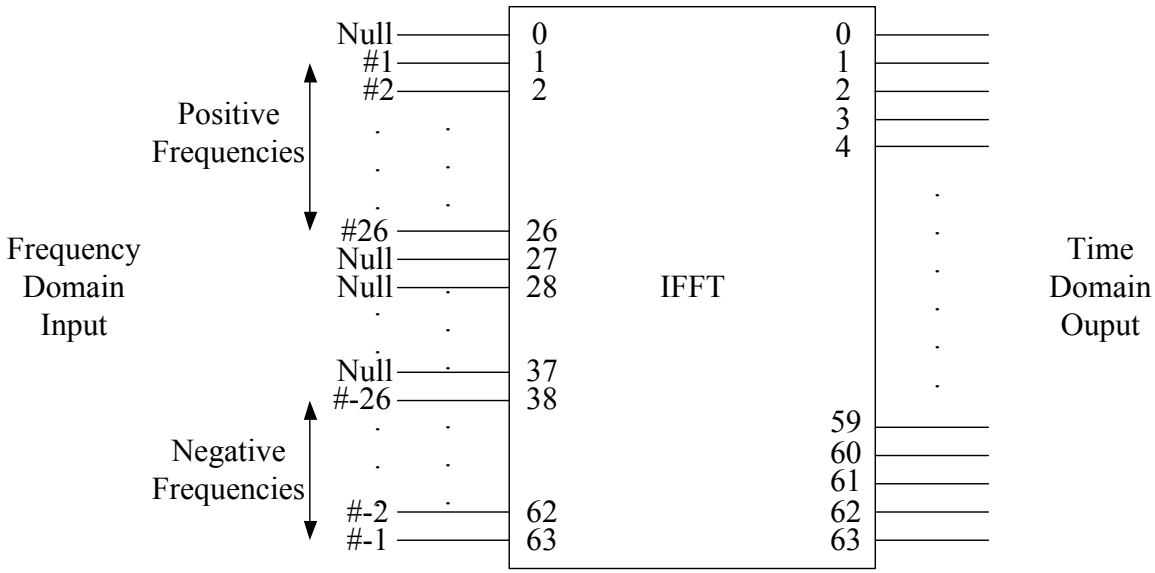


Figure 2.6 Input and output of IFFT block at the transmitter.

Among the 52 subcarriers, 48 are used for carrying data information and 4 are used as pilots for channel estimation purpose. The allocation of the data and pilot subcarriers is shown in Figure 2.7, where the pilots are located at the subcarrier -21, -7, 7, 21, and the remaining 48 subcarriers are used for the modulated data symbols d_0 to d_{47}.

Figure 2.8 shows the power spectrum of an OFDM signal. It is obvious that the power spectrum has peaks at the pilot subcarriers, and has dips at the null subcarriers.

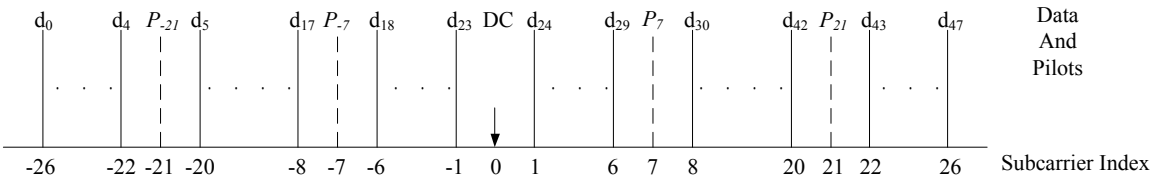


Figure 2.7 Data allocation onto subcarriers.

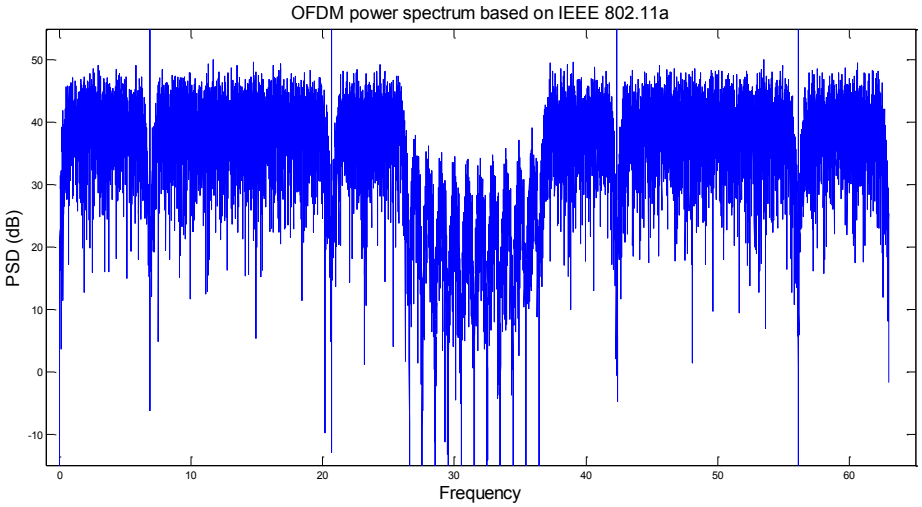


Figure 2.8 Power spectrum of the OFDM signal.

2.3 Design and Performance Analysis of a Basic OFDM System

An OFDM system with system parameters specified in IEEE 802.11a PHY is modeled in MATLAB. BER verses bit energy to noise spectral density (E_b/N_0) curve is plotted and compared to the theoretical one to verify the performance of the designed system.

2.3.1 Floating-point Modeling for OFDM System

According to Figure 2.4, the transmitter, the receiver and the channel are included in the system modeling. First, a random binary sequence, in which "0" and "1" are uniformly distributed, is generated as the input data at a bit rate of 24 Mbps. There is one sample for each bit data.

At the transmitter, the binary bit sequence is mapped to a serial of complex values using PSK or QAM modulation. In the proposed design, the QPSK scheme is selected. The generated QPSK sequence has a symbol rate of 12 Mbaud per second. The subcarrier elements for an individual OFDM symbol, corresponding to 48 QPSK symbols, are grouped to form parallel transmitted data through the "Serial to Parallel" block. So the OFDM symbol rate achieved is 0.25 Mbaud per second. The four pilots are generated and inserted to data subcarriers together with 12 nulls. The entire 64 subcarriers are then allocated as specified in 802.11a to form the input of a 64-point IFFT block. At the output of IFFT, a vector consisting of 64 samples in the time domain is generated. The CP is added by copying the last 16 samples of the vector at the beginning. So an OFDM symbol consisting of 80 samples in its time period is generated. These 80-sample vectors are then converted to a serially delivered sequence and transmitted sequentially. According to the analysis in Section 2.2.1, the guard interval duration, which is chosen to be 16-sample length, is large enough to avoid ISI.

For channel modeling, we separate the effects of fading and additive noise into two parts. First, the transmitted OFDM signal passes through a Rayleigh fading channel

$h(t)$. Then a Gaussian-distributed noise is added to the faded signal before it arrives at the receiver.

The basic idea for the receiver design is to reverse the operation that is performed at the transmitter. However, synchronization and channel estimation blocks are added to detect the signal and equalize the channel effects. After it is detected and synchronized, the signals are converted to 80-sample vectors by serial to parallel conversion. As added prefix, the first 16 samples of each vector are removed. Then FFT block transforms the 64 samples in the time domain to 64 subcarriers in the frequency domain. After the "Subcarrier Deallocation" block, the 48-data vector and the 4-pilot vector are separated. The data vector that consists of 48 QPSK symbols are then converted to serial and demodulated. Finally, a binary sequence is obtained and compared to the original input to the transmitter to calculate the BER.

Channel estimation is performed when the long training sequence is received. The estimated factors are then fed back to equalize the channel effects on each data symbol in an OFDM frame. The synchronization, channel estimation and equalization are the most challenging tasks in the system design, and these will be discussed in Chapters 4 and 5. In this section, we assume perfect timing and focus on the performance of the design under an AWGN channel.

2.3.2 QPSK Modulation

As a frequency multiplexing scheme, the data on each subcarrier of OFDM can be modulated with phase difference or/and amplitude of the carrier (PSK or QAM). QPSK,

also considered as 4 QAM, is selected in this design, since it can be easily modified to implement more complex modulation schemes, like 16 QAM and 64 QAM.

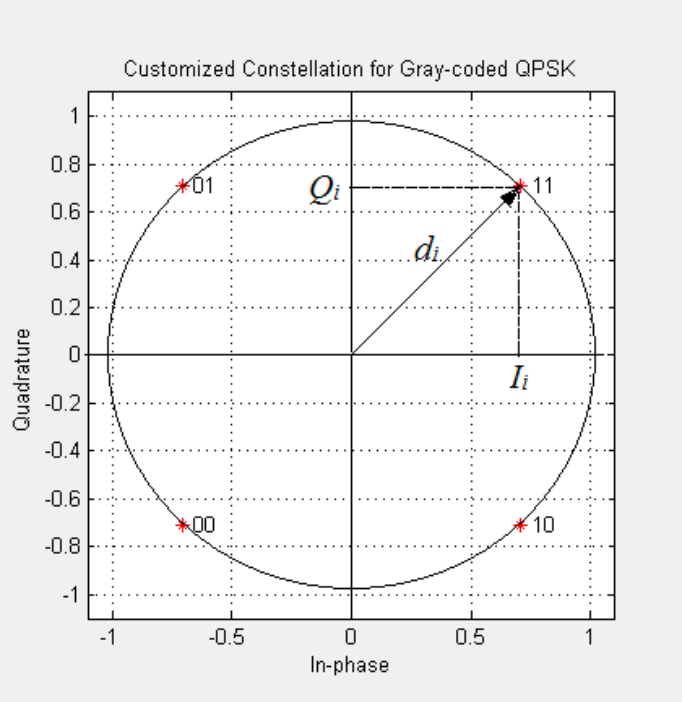


Figure 2.9 QPSK constellation with gray coding.

QPSK modulation scheme has four phase states separated by $\pi/2$. Therefore, each QPSK symbol carries a two-bit data ($log_2 4 = 2$). The constellation diagram according to gray-coded mapping is illustrated in Figure 2.9. The I and Q values are normalized to $1/\sqrt{2}$ in order to achieve the same average power for all mappings. The details are shown in Table 2.2.

Table 2.2 QPSK symbol mapping with gray coding.

Bit Set	*I* Value	*Q* Value	Phase (degree)
11	$1/\sqrt{2}$	$1/\sqrt{2}$	$\pi/4$
01	$-1/\sqrt{2}$	$1/\sqrt{2}$	$3\pi/4$
00	$-1/\sqrt{2}$	$-1/\sqrt{2}$	$-3\pi/4$
10	$1/\sqrt{2}$	$-1/\sqrt{2}$	$-\pi/4$

When coherent detection is applied, the BER expression for QPSK modulation over AWGN channel is given by [29].

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \tag{2.12}$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{+\infty} \exp\left(-\frac{y^2}{2}\right)dy, \quad for\ x > 0 \tag{2.13}$$

2.3.3 Simulation Results of the Proposed OFDM System

Since the subcarriers are orthogonal to one another, which means they are independent, the theoretical BER of the OFDM system is the same as that for a single carrier system [3]. In the proposed design, the BER over an AWGN channel is same as that of a QPSK modulation system given in (2.12). Figure 2.10 gives a comparison of the theoretical

BER curves with that of the MATLAB simulation of the floating-point model. It is observed that the simulation results agree very well with the theoretical ones.

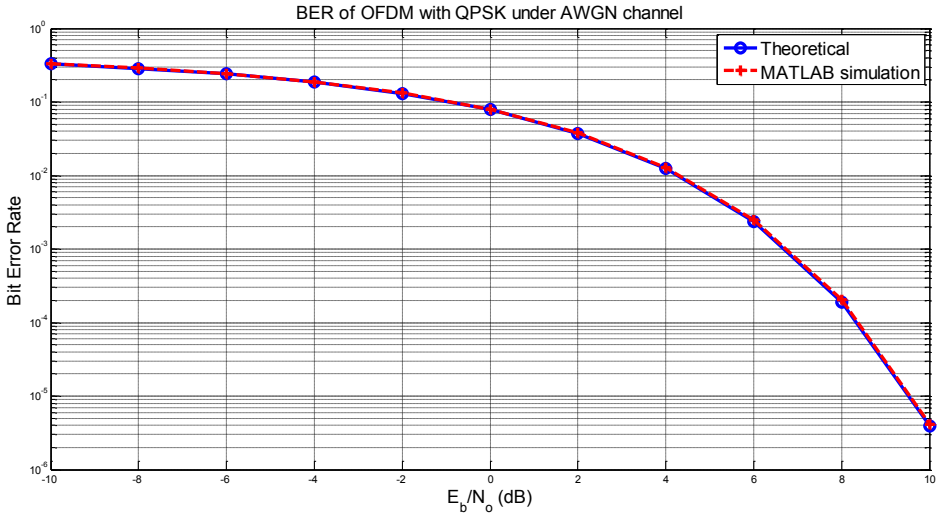


Figure 2.10 BER performance of an OFDM system under AWGN channel.

In this simulation, the pre-defined bit energy to noise spectral density E_b/N_0 is known. To calculate the BER, the transmitted signal should go through an AWGN channel, which is modeled as a zero-mean complex Gaussian process $\mathcal{N}(0, \sigma_z^2)$. The noise variance is obtained by:

$$\sigma_z = \sqrt{\frac{N_0}{2}} = \sqrt{\frac{1}{2}\frac{E_s}{E_s / N_0}} \tag{2.14}$$

in which

$$\frac{E_s}{N_0} = \frac{N_u}{N + N_g} \cdot \log_2 q \cdot \frac{E_b}{N_0} \tag{2.15}$$

where E_s is the normalized OFDM symbol energy. q is the number of states in the modulation scheme, which is four in this design, and N_u, N and N_g are the number of used subcarriers, FFT size, and the duration of guard interval, respectively. They have been defined in Table 2.1.

In summary, this chapter has given the theoretical analysis to system design of an OFDM system based on the IEEE 802.11a PHY. The theoretical system performance has been verified by MATLAB simulation In the next chapter, we will introduce the FPGA implementation of the proposed OFDM system. The BER performance of the fixed-point model will be compared with that of the floating-point one presented in this chapter.

Chapter 3 FPGA Implementation of a Basic OFDM System

This chapter presents the hardware implementation of a fundamental OFDM system demonstrated in Chapter 2. First, the fixed-point modeling of the transmitter and the receiver is designed block by block in XSG. Each functional block is verified by checking the waveforms in scope. Then, the complete OFDM system is integrated together. It is followed by the introduction of hardware co-simulation, which provides a hardware-in-the-loop verification. The BER results are calculated and compared with that of the floating-point model. Finally, the target FPGA device is described and the implementation results in terms of resource consumption are given.

3.1 Implementation of a Baseband OFDM Transmitter

As per Figure 1.1, the hardware implementation starts with the establishment of the fixed-point model in XSG, which is performed with a top-down approach. In this model, numbers are represented in signed two's complement form, denoted as "Fix_16_15". 16 bits are used, in which the most significant one is the sign bit and the remaining 15 bits represent the fractional part. This fixed-point representation is used in most of the computations including IFFT and FFT. A few blocks show different forms due to precision requirements. However, the fractional parts are always truncated to 15 bits.

Figure 3.1 shows the block diagram of the OFDM transmitter. The binary input is first converted to QPSK symbols by the "Mapping" block. At the same time, the "Pilot Generation" block generates the pilot subcarriers. Both of them are fed into the "Subcarrier Allocation" to generate the IFFT input. After the "IFFT & Adding Cyclic Prefix" block, the OFDM data symbols are generated. The "Transmitter Controller" arranges the order of operation of all the above blocks, as well as the "Preamble Generation" block, which stores the short and long training symbols. It also decides whether to send the preamble or the OFDM data symbols by the "Multiplexer" (MUX) block.

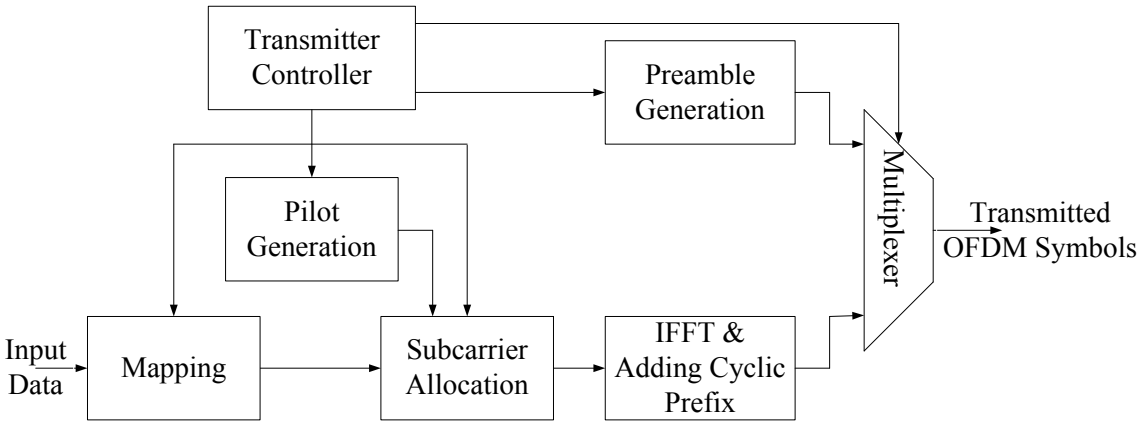


Figure 3.1 Block diagram of the OFDM transmitter.

The binary input is generated at the FPGA main clock speed, which is 40 MHz. Since one QPSK symbol contains two binary bits and the symbol period elongates to two times the bit duration, the controller works at 20 MHz clock frequency, which is implemented by setting the sample period to twice the FPGA clock period. In fact, the clock speed for all the blocks at the transmitter except the "Mapping" block is 20 MHz.

On the other hand, the efficiency of data transmission is 48/80 = 60%. Therefore, the transmission data rate is 60%*40 Mbps = 24 Mbps.

3.1.1 Transmitter Controller

The "Transmitter Controller" controls all the blocks at the transmitter so as to generate valid OFDM symbols. It is implemented as a state machine regulated by a counter counting from 0 to 320. From Figure 3.2, it is seen that the controller is activated by a signal "init" and generates several signals that masters other blocks at the transmitter. "en_sym" enables the formation of OFDM data symbols and selects the MUXs. "en_sts", "en_gi2" and "en_lts" go to "Preamble Generation" block and trigger the generation of STS, *GI*2 and LTS, respectively. "state" indicates the controller states and "cnt" counts the clock cycles so that we can monitor the states of the controller. "start_tx" is the start signal for "Mapping", "Pilot Generation" and "Subcarrier Allocation" blocks.

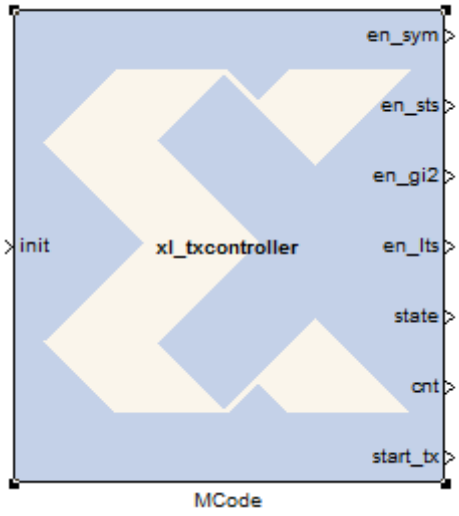


Figure 3.2 Implementation module of Transmitter Controller.

The state diagram is shown in Figure 3.3. At each clock cycle, generation of the transmitted signal is scheduled as following. When the signal "init" goes to high, the counter starts to count clock cycles. The controller changes from state 0, the initial state, to state 1. "en_sts" turns to high, indicating the start of STS. When the counter counts to 160, the controller goes to state 2, the generation of *GI*2. At this state, "en_gi2" goes to high and enables LTS cyclically extended within this guard interval. The controller changes to state 3 when the counter continues to 192. The LTS is periodically generated after "en_lts" turns to high. When the counter counts to 320, the controller goes to the last state, state 4, and indicates the formation of the OFDM data symbols. On the other hand, when the counter counts to 90, the "start_tx" goes to high. This is the start signal for the OFDM transmitter to accept the input data and create the data symbols. The controller is well designed so that the OFDM signal is transmitted sequentially and continuously.

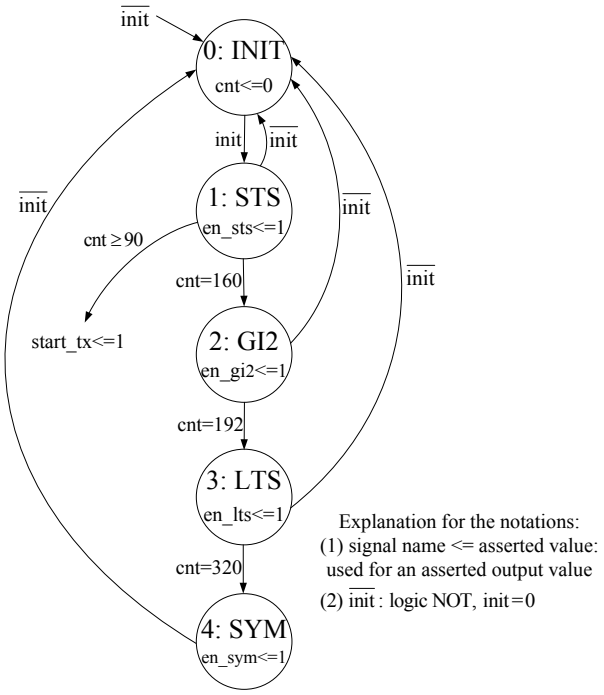


Figure 3.3 State diagram of Transmitter Controller.

3.1.2 Preamble and Pilot Generation

The preamble is transmitted at the beginning of each OFDM frame for synchronization and channel estimation purpose. The details of these preamble symbols are described in Chapters 4 and 5. The "Preamble Generation" block stores these symbols and outputs them according to the controller regulation.

The implementation of "Preamble Generation" is demonstrated in Figure 3.4. The real and imaginary components of STS are stored in two 16 x 16 read-only memories (ROMs). A counter repeatedly counting from 0 to 15 is used to provide the reading address of the two ROMs. The enable signal from the controller, "en_sts", activates the counter and ROMs and lasts for 160 clock cycles.

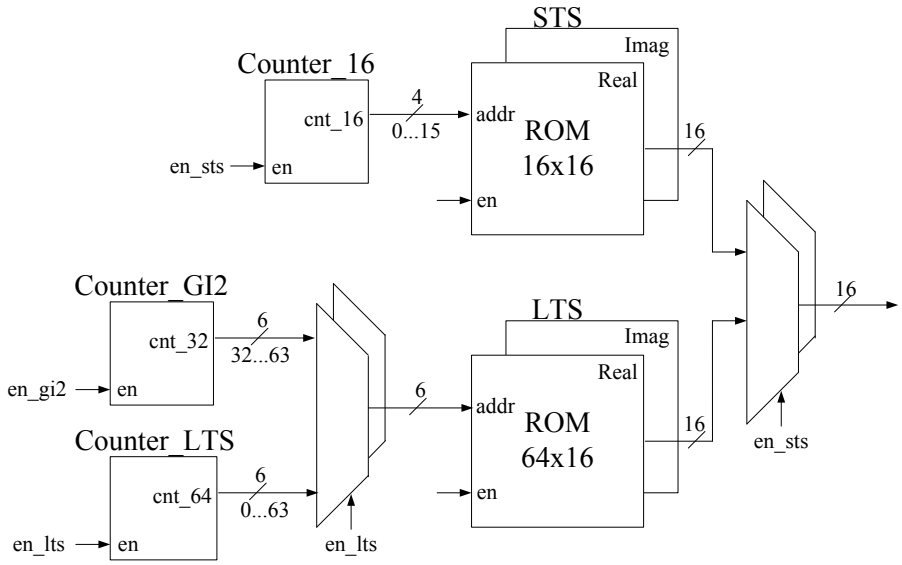


Figure 3.4 Block diagram of Preamble Generation.

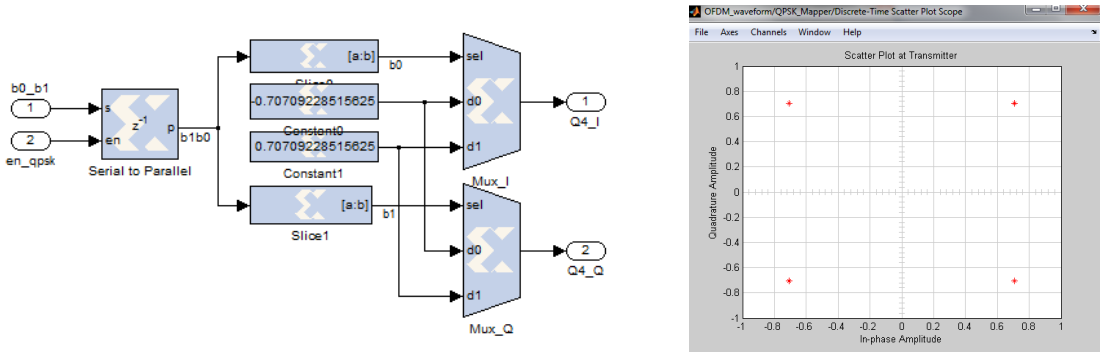
Similarly, the LTS values are stored in two 64 x 16 ROMs. They are activated when either "en_gi2" or "en_lts" goes to high. And the ROMs output either the last 32

samples as *GI*2, or the two 64-sample LTS symbols. Two counters named "Counter_GI2" and "Counter_LTS" are used to provide the reading address for guard and effective symbols, respectively.

The "Pilot Generation" block generates the subcarrier values carried on the pilot tones. Apart from the ROMs that stores the I and Q components, another ROM that stores the pilot subcarrier positions is also required.

3.1.3 QPSK Mapping

To map the binary input to QPSK symbol, the input sequence is first converted to two-bit symbols by serial to parallel conversion. They are then used as the select signal to two multiplexers to decide the I and Q values of the desired complex symbol. The multiplexers can be replaced by ROMs for QAM modulation schemes. The implementation module is illustrated in Figure 3.5(a). The enable signal of the block "en_qpsk" is connected to "start_tx" from the controller. Figure 3.5(b) depicts the constellation diagram for the QPSK gray-coded mapping.



(a) (b)

Figure 3.5 QPSK Mapping (a) implementation module and (b) constellation diagram.

Figure 3.6 illustrates the waveforms in scope of the QPSK mapping. By comparing the clock enable signals, we can conclude that the clock frequency reduces by half after serial to parallel conversion. The binary stream is mapped to the two-bit QPSK symbols. The most significant bit of a QPSK symbol controls the in-phase MUX, while the least significant bit controls the quadrature MUX. For a more complex circuit, such as the 16 QAM and 64 QAM, the I and Q values could be stored in ROMs.

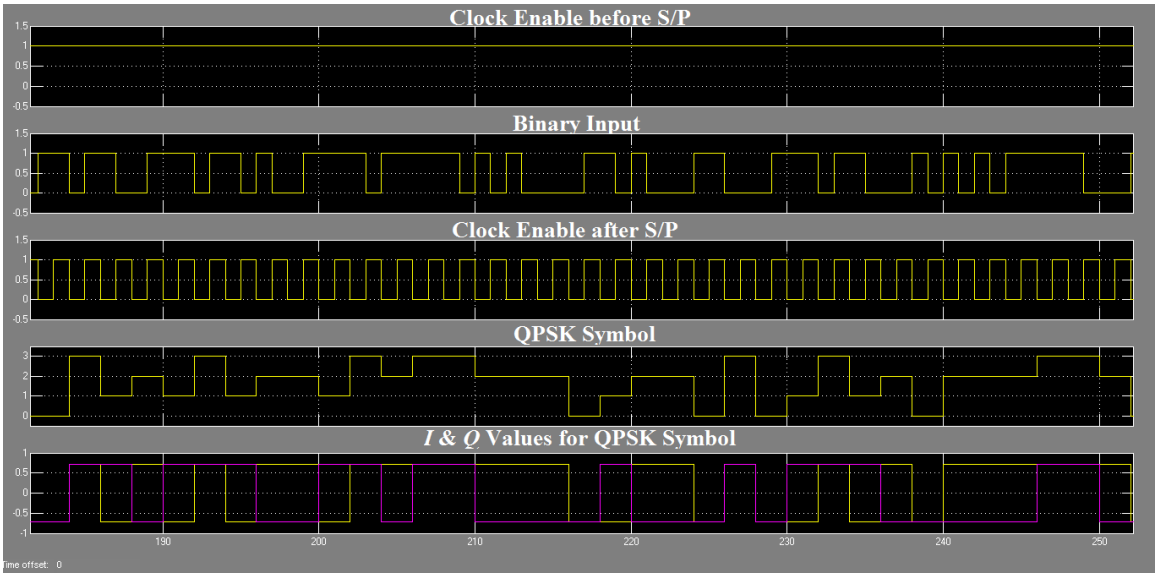


Figure 3.6 Waveforms of the QPSK mapping.

3.1.4 Subcarrier Allocation

There are three purposes for designing the "Subcarrier Allocation" block. The first is to allocate the data and pilot subcarriers, the second is to insert null subcarriers which carries "zero" information, and the third is to implement the FFT shift. After this block, 64 inputs for IFFT are prepared.

The implementation module is depicted in Figure 3.7. Two 64 x 16 dual port random-access memories (RAMs) are used. The null subcarriers are achieved by defining the contents of the two RAMs at relative addresses to be zeros. Ports A of the RAMs are used for data carrier writing, while ports B are dedicated for pilot writing and the 64-subcarrier reading. They are controlled by two counters. “Counter_We_48” block stores the 48 data carrier positions. According to the addresses specified by it, the I and Q values of the input symbols are written into two RAMs from ports A. At the same time, FFT shift is inherently performed. “Counter_Rd_64” uses a counter to generate the reading addresses of ports B. When ports B are enabled and “Counter_Rd_64” counts from 0 to 63, the desired 64 inputs for IFFT are outputted and the “en_ifft” is high, notifying the IFFT block to perform calculations.

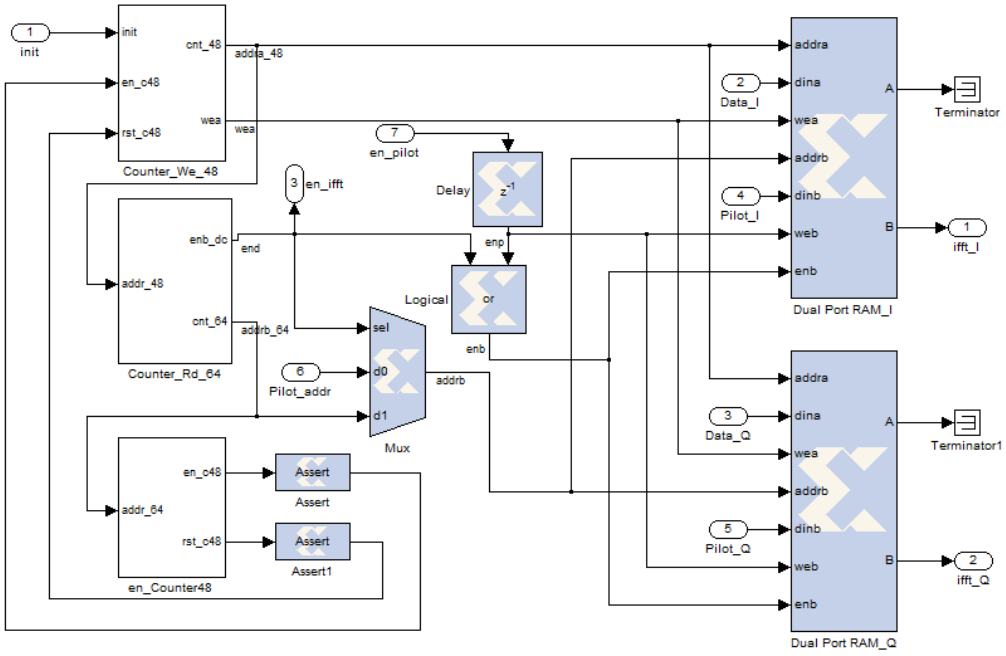


Figure 3.7 Implementation module of Subcarrier Allocation.

The reading should start after the writing is finished. Therefore, the output of "Counter_We_48" is fed into "Counter_Rd_64" to point the start of symbol reading. For continuous signal processing, the output of "Counter_Rd_64" is also fed back to "Counter_We_48" through a combinational logic, indicating the reading is finished and the RAMs are ready for the next symbol.

3.1.5 IFFT and Add Cyclic Prefix

The proposed OFDM system is designed for burst communication environment. Therefore, it requires continuous data processing. A highly-pipelined architecture is required for IFFT/FFT implementation. The IFFT/FFT blocks are imported from Xilinx's IP core library, named Fast Fourier Transform 7.1 (xfft_v7.1). It implements the Cooley-Tukey algorithm and employs the radix-2 decimation-in frequency (DIF) method. In the pipelined architecture, several radix-2 butterflies and banks of memories are used for each engine, so that the data processing is pipelined and the high throughput is achieved. In this design, the core computes a 64-point complex forward or inverse FFT. Figure 3.8 shows the configuration of this IP core while IFFT is being performed.

The real and imaginary components of the input are numbers represented in the form Fix_16_15. The "start" signal indicates the data loading. By defining the "cp_len", the CP is automatically added during IFFT output process. The core implements IFFT when the "fwd_inv" is set to low. "scale_sch" shows the scaling schedule for each radix-2 pair. In this design, it is set to "101010". The output results are scaled to 1/8, which avoids overflow.

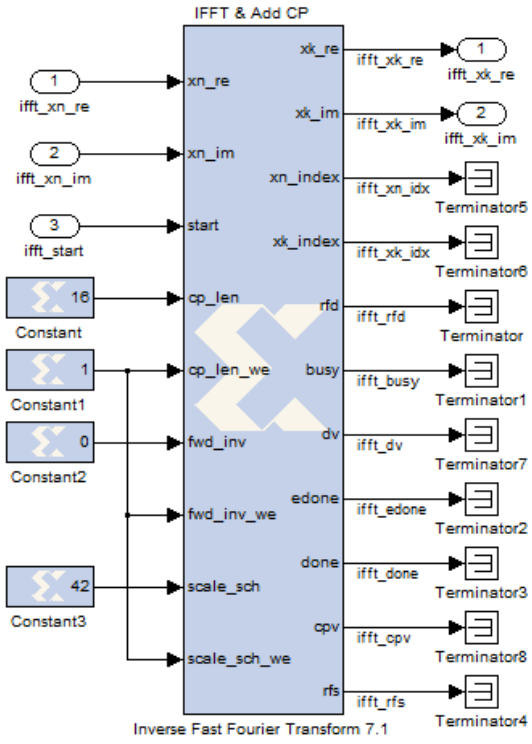


Figure 3.8 Implementation module of IFFT.

Figure 3.9 describes the data loading process of the IFFT block. The default state for “rfs” signal is high. It turns to low at the next clock when the “start” signal is activated. Since the CP is inserted, the “start” signal for next OFDM symbol should wait for 16 clock cycles to reactivate. During the data loading and CP generation process, “rfs” is low. “rfd” signal turns to high at the next clock when “start” goes to high. It indicates the loading of 64 inputs. “xn_index” counts the input index, ranging from 0 to 63. When “busy” goes to high, the core starts the calculation of the FFT algorithm.

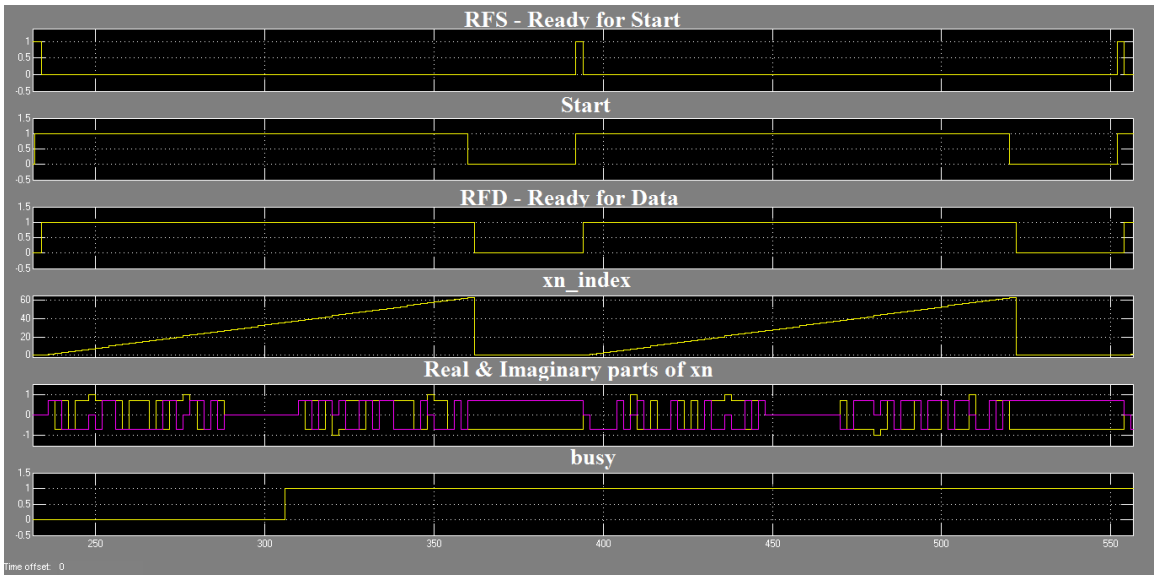


Figure 3.9 Waveforms of data loading for IFFT block.

Figure 3.10 illustrates the IFFT output process. As an early reminder, “edone” goes to high one clock cycle prior to “done” going to high, which reveals the completion of the transform calculation and the starting of the output process. The output is produced in natural order represented by “xk_index”. Within one OFDM symbol period, it first counts from 48 to 64, when the 16-sample CP is being sent, The “cpv” is high during this period of time. Then “xk_index” counts from 0 to 63, showing the efficient OFDM symbol is sent. “dv” is always high when the valid data is sent, irrespective of the CP or the efficient symbol. Because of the scaling, the real and imaginary components of the output “xk” are also fix_16_15 numbers.

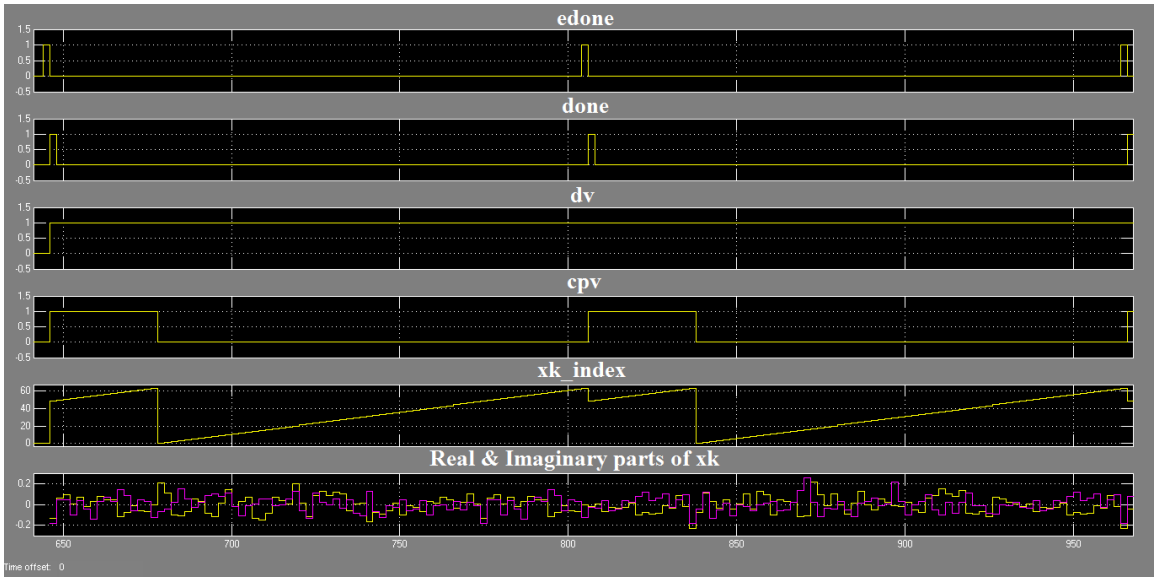


Figure 3.10 Waveforms of data output for IFFT block.

3.2 Implementation of a Baseband OFDM Receiver

Figure 3.11 shows the block diagram of the receiver design. It performs reverse operations to those at the transmitter. The received OFDM signal is first fed to "Removing Cyclic Prefix" to get the 64 inputs of FFT. "Subcarrier Deallocation" separates the pilot and data subcarriers that are used for the demodulation of OFDM symbols. In the proposed design, perfect synchronization is assumed. "Receiver Controller" receives the start signal from frame detection and synchronization circuits and arranges the operation sequence of other blocks. Similar to that at the transmitter, the clock speed for all the blocks at the receiver except "Demapping" is 20 MHz, and the "Demapping" block works at 40 MHz clock.

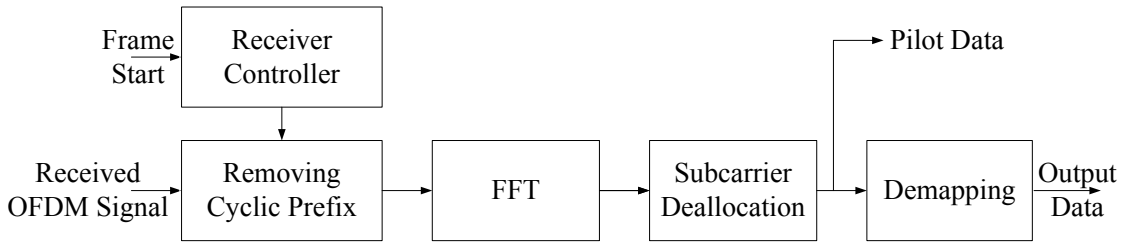


Figure 3.11 Block diagram of the OFDM receiver.

3.2.1 Receiver Controller

The "Receiver Controller" is also implemented as a state machine. The states of the receiver are controlled by two counters. One counts from 0 to 16, denoted as "cnt16", and is used to indicate the CP duration for each OFDM data symbol. Another one counts from 0 to 63, denoted as "cnt64", and is utilized to monitor the receiver state when it receives LTS and efficient OFDM data symbols. The block implementation is shown in Figure 3.12. The controller starts to work when "rx_start" goes to high. "en_lts" and "en_l2" turn to high when the entire long sequence and the second LTS are received, respectively. "en_sym" is high when the effective OFDM symbol is received. During this period, the FFT block is enabled.

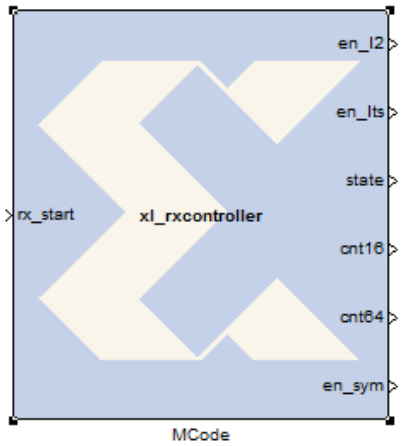


Figure 3.12 Implementation module of Receiver Controller.

The state diagram is shown in Figure 3.13. At each clock cycle, the processing of the received signal is scheduled as following. When the signal “rx_start” goes high, the controller goes to state 1 and the first LTS is received. “Counter_64” starts to count clock cycles. When it counts to 63, the controller jumps to state 2, and the receiver starts to receive the second LTS. “en_l2” turns to high and the block that averages the two LTSs starts to operate. “en_lts” keeps high in states 1 and 2. State 2 is finished when “Counter_64” re-counts from 0 to 63. Then the controller jumps to state 3 and “Counter_16” counts from 0 to 15, indicating the receiving of CP. After that, the controller changes to state 4. The “Counter_64” counts from 0 to 63, indicating the FFT period of an OFDM symbol. “en_sym” is high at this state. If “rx_start” remains high, which means it is in the same OFDM frame, the transformation of state 3 and state 4 are repeated, indicating the continuous transmission of OFDM symbols.

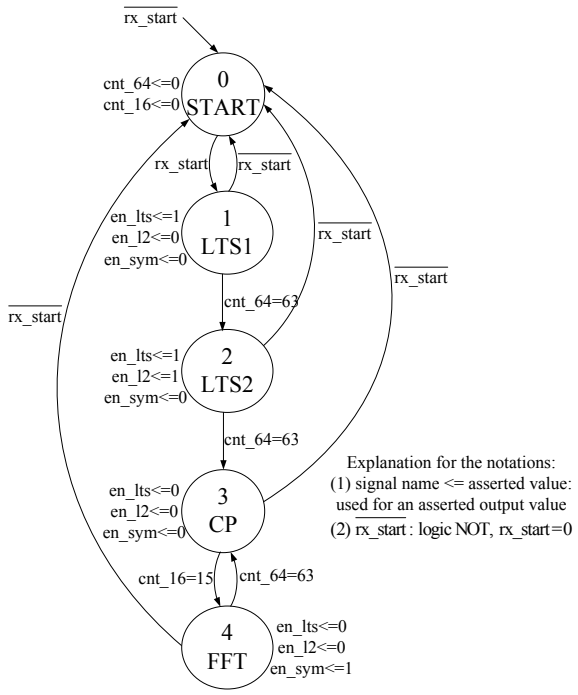


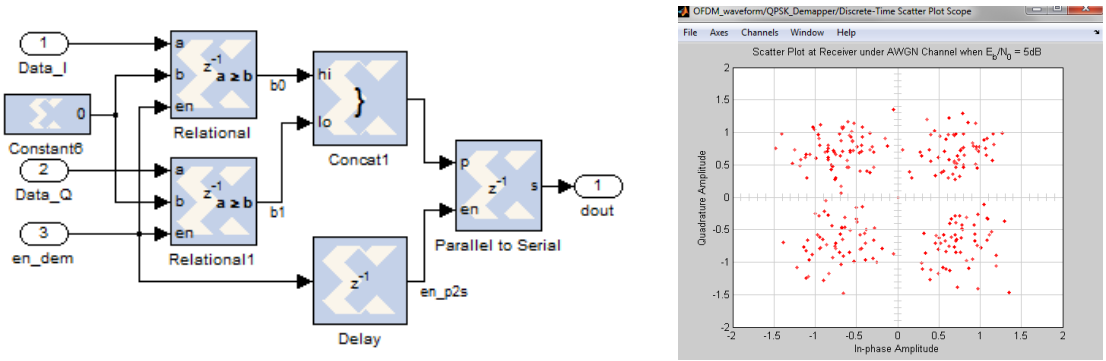
Figure 3.13 State diagram of Receiver Controller.

3.2.2 Removing Cyclic Prefix and FFT

The CP removing block is implemented by two registers. They store the real and imaginary parts of the effective OFDM symbols, respectively. The two registers are only enabled when "en_sym" is high, so that the CP values will not be saved and the desired 64-sample symbols are obtained. These samples are sequentially fed to the FFT block, which is implemented by Xilinx's xfft_v7.1 IP core, where "fwd_inv" is set to high.

3.2.3 QPSK Demapping

The demodulation of the QPSK symbols is performed by the hard decision method. First, I and Q values of the symbols are compared with a threshold 0. Then the decisions are grouped to form a 2-bit QPSK symbol. Next, the QPSK symbol is converted to binary data through "Parallel to Serial" conversion. Figure 3.14(a) shows the implementation module. Figure 3.14(b) illustrates the constellation diagram after demodulation under an AWGN channel at E_b/N_0 = 5 dB.



(a) (b)

Figure 3.14 QPSK Demapping (a) implementation module and (b) constellation diagram.

3.2.4 Subcarrier Deallocation

As the reverse process of “Subcarrier Allocation”, the design of “Subcarrier Deallocation” is also required to implement three kinds of functions: extract the data and pilot subcarriers; remove the null subcarriers; and perform FFT shift. After this block, 48 data subcarriers and 4 pilots are separated.

Figure 3.15 illustrates the implementation module. Two 48 x 16 dual port RAMs are utilized. Ports A are used for data carrier writing, while ports B are dedicated for data carrier reading. “Counter_We_48” enables writing process at ports A only if the data carriers are received. The input I and Q values are written to the RAMs according to the addresses specified by “Counter_We_48”, which stores the positions of the 48 data carriers. At the same time, FFT shift is inherently performed. After the 48 data carriers are saved, it starts to read these values sequentially from ports B. The reading addresses are generated by “Counter_Rd_48” using a counter counting from 0 to 47. The control signals are from the output of the FFT block.

“Pilot Extractor” block is used to separate the pilot subcarriers from the input symbol. It is implemented by two registers that are enabled when “xk_idx” matches the pilot subcarrier positions. Then the registers store the I and Q values of the pilot subcarriers and send them out.

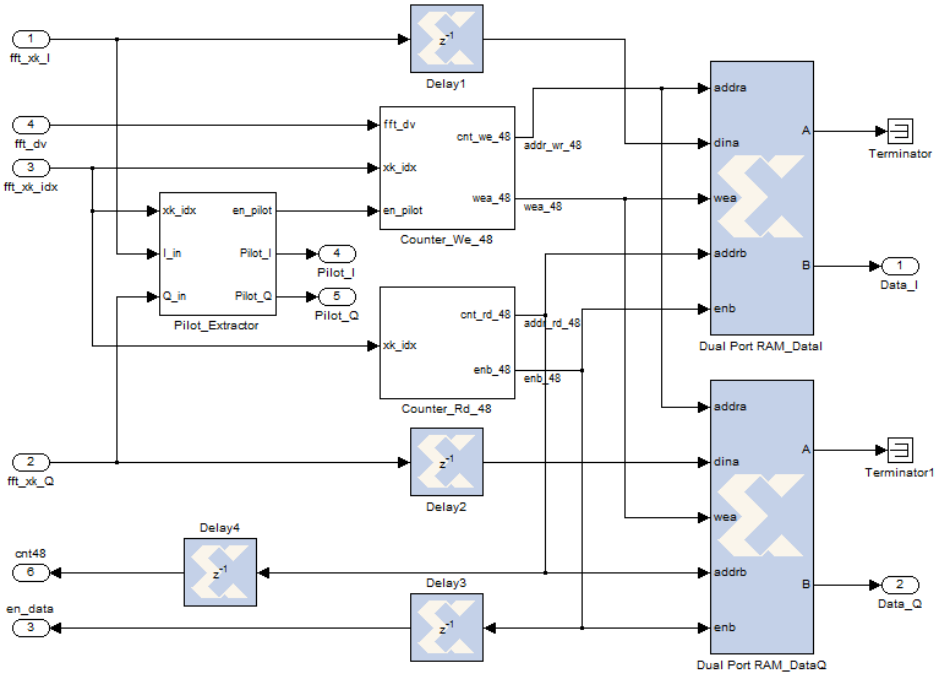


Figure 3.15 Implementation module of Subcarrier Deallocation.

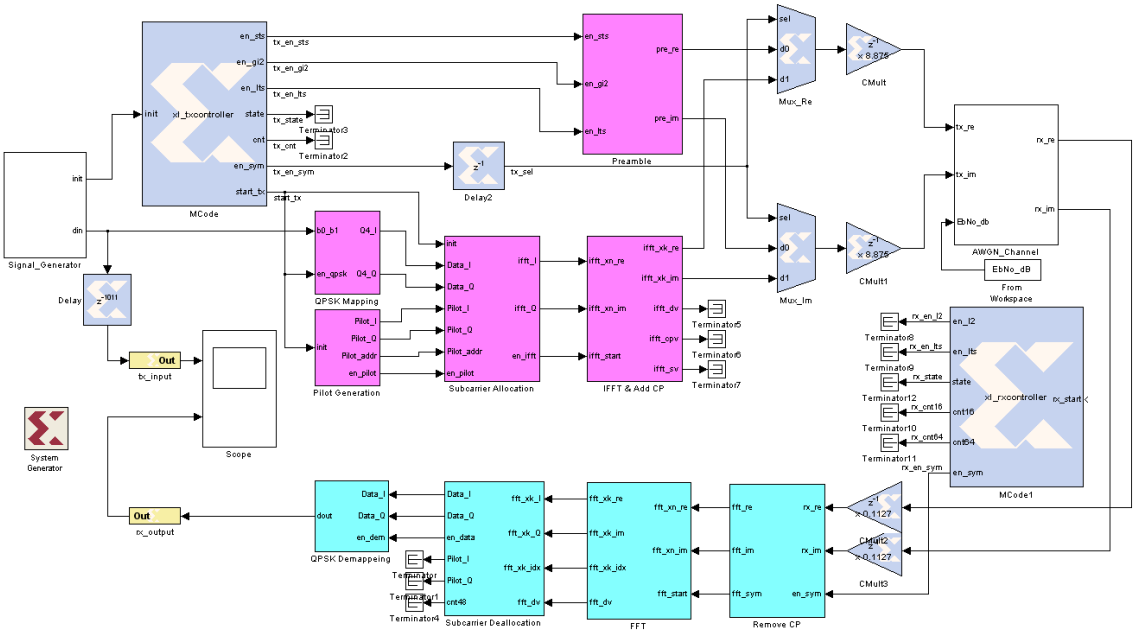


Figure 3.16 Implementation module of the entire baseband OFDM system.

3.3 Hardware Co-simulation and Implementation Results

3.3.1 Integration of the Entire System

According to the analysis in Sections 3.1 and 3.2, the modeling of the complete baseband OFDM system is illustrated in Figure 3.16. At the end of transmitter, the signals are amplified so that the transmitting power of each FFT sample is normalized to unity. At the receiver, the inverse operation is performed to get the actual transmitted symbols.

Figure 3.17 compares the input and output waveforms under an AWGN channel at E_b/N_0 = 5 dB. It is observed that there is one bit demodulated error in each OFDM symbol. So the BER is roughly 0.01, which matches the MATLAB simulation result in Chapter 2. Therefore, the system model is verified to meet the design requirements.

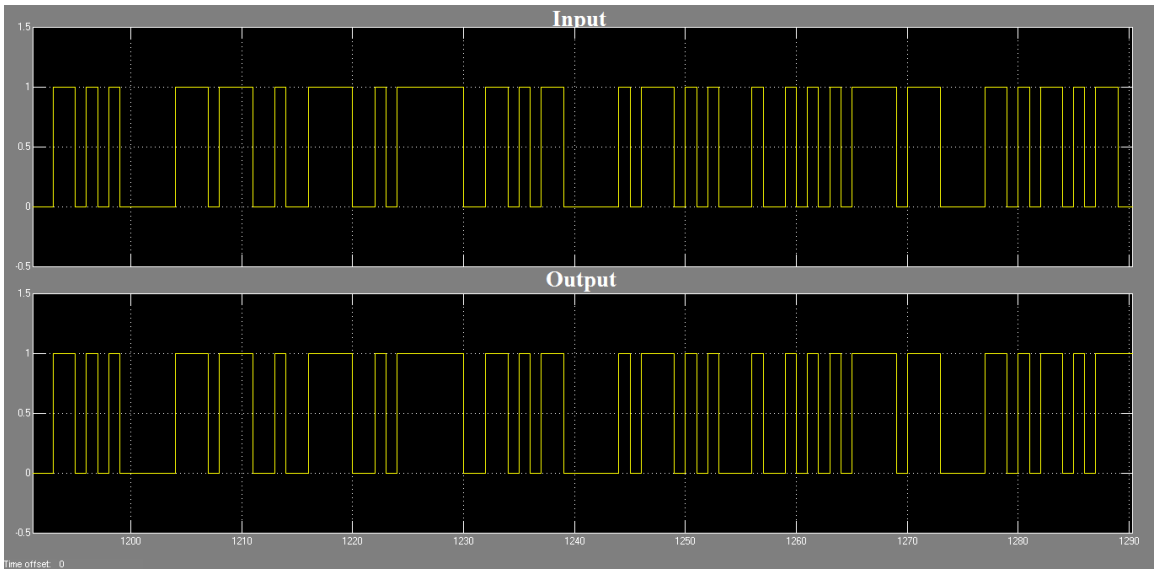


Figure 3.17 Input/Output comparison under an AWGN channel at E_b/N_0 = 5 dB.

3.3.2 Hardware Co-simulation and BER Performance Comparison

XSG is a system developing tool integrated in Simulink [16], and provides a series of packaged IP cores presented in the form of block set. The system designer does not need to perform detailed coding for each block, but focus on the system design and the interfaces between block sets. After system modeling, the VHDL code is generated. XSG will automatically generate a template file that allows you to instantiate the IP cores into the HDL design. It then facilitates the system designers, especially in developing large and complex systems.

In order to simplify hardware verification, the hardware co-simulation, which provides the hardware-in-the-loop verification, is proposed using XSG. The input vectors are generated and the output is analyzed by MATLAB and Simulink. The XSG allows a direct interface between a real-time system and Simulink [39]. Figure 3.18 demonstrates a simplified co-simulation flow. The Xilinx ISE tool is also required since it is actually running in the background while the implementation of the hardware is performed [39].

Figure 3.18 Hardware co-simulation within XSG [51].

Figure 3.19 shows a picture of hardware co-simulation model for the baseband OFDM system and its programming to FPGA board. It uses a JTAG point-to-point connection. The VHDL codes are generated after the co-simulation.

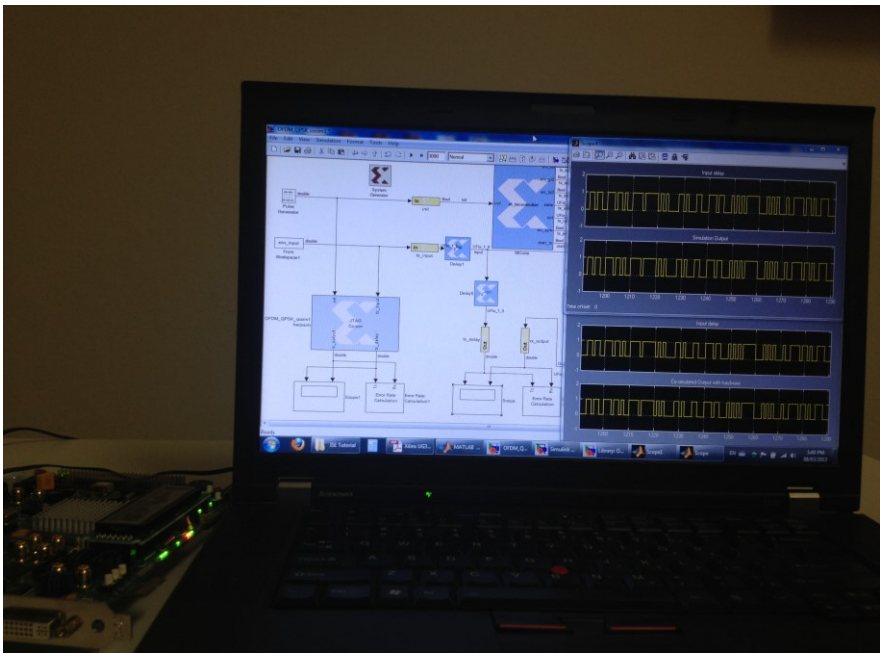


Figure 3.19 Hardware co-simulation for the proposed OFDM system.

The BER results can be calculated through the simulation. Figure 3.20 depicts the BER versus E_b/N_0 curves under an AWGN channel. We can observe that the BER curve of the 16-bit fixed-point model matches well with that of the floating-point model at low SNR, while a slight deviation occurs at high SNR. This is due to the fact that the loss of the numerical precision has more impact on the simulation results at high SNR. In this design, the signal is generated by normalizing the transmission power of each FFT sample to unity and decreasing the noise power for effectively increasing SNR. Therefore, in the case of high SNR, the precision used is not enough to represent the noise and this numerical precision loss has led a slight degradation of BER performance.

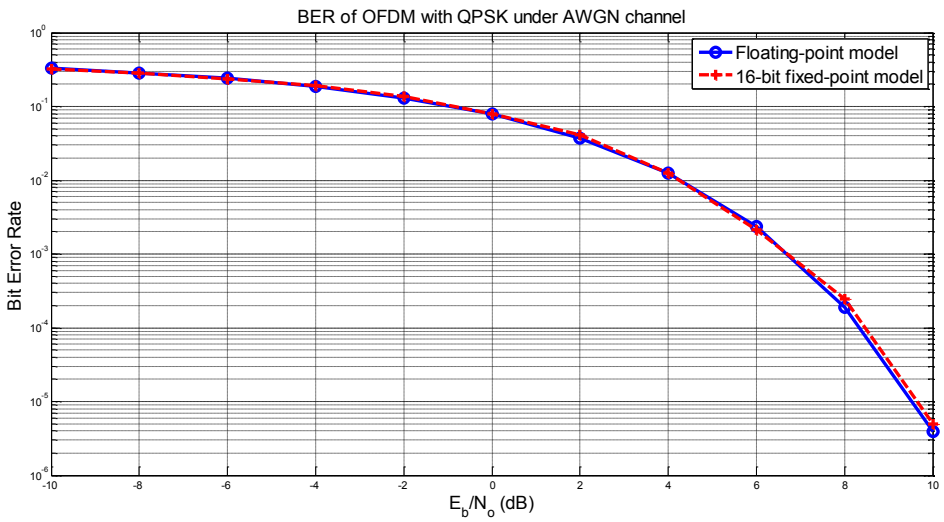


Figure 3.20 BER comparison of an OFDM system under AWGN channel.

3.3.3 Hardware Implementation Results

Through the hardware co-simulation process, the VHDL coding of the system is accomplished. The generated VHDL codes are imported into the ISE tool to make further system verification at the RTL level. Design synthesis is performed with Xilinx's XST and the gate-level netlist is obtained. After that, place and routing are performed. The routing delays are back annotated to the gate-level netlist for timing analysis and design optimization. Finally, a bitstream is generated to program the target FPGA board.

The proposed work is implemented on an Xilinx XUPV5-LX110T evaluation platform and targeted to a XC5VLX110T-1FF1136 device. The CLBs are 160 x 54 arrays. The target device includes 17,280 slices with each containing four LUTs and four flip-flops, 64 DSP48E slices, and 148 block RAMs with 36 Kb size. There are 32 global

clock networks and the board clock could achieve 550 MHz clock speed, which fully satisfies the system requirements.

Table 3.1 and Table 3.2 summarize the area results with regard to resource consumption, and the clock and timing results for the transmitter and receiver, respectively. The timing reports generated from Xilinx Timing Analyzer tool indicates no timing conflict in the design. The maximum frequency is large enough to generate 40 MHz clock and drive the whole system. Moreover, the board could support a system with its maximum frequency up to 550 MHz.

According to the simulation results, the OFDM transmitter takes 15.6 μs or 624 clock cycles to finalize the generation of the first OFDM symbol. It could be performed within the period of preamble transmission. The receiver takes 16 μs or 640 clock cycles to completely demodulate the first OFDM symbol. Although the latency is large, the OFDM symbols are generated continuously at a symbol rate of 20 Mbaud because of the pipelined design of the system. Moreover, the pipelining design makes the delay of the clock net very small, at about 2 ns, which improves the system performance.

Table 3.1 Area results for basic transmitter and receiver.

Transmitter			***Receiver***		
Resources	***Used***	***%***	***Resources***	***Used***	***%***
Number of Slice Registers	2328	3	Number of Slice Registers	2329	3
Number of Slice LUTs	2408	3	Number of Slice LUTs	2222	3
Number of bonded IOBs	43	6	Number of bonded IOBs	43	6
Number of BlockRAM/FIFO	4	2	Number of BlockRAM/FIFO	1	1
Number of BUFG/BUFGCTRLs	1	3	Number of BUFG/BUFGCTRLs	1	3
Number of DSP48Es	6	9	Number of DSP48Es	6	9

Table 3.2 Clock and timing results for basic transmitter and receiver.

Transmitter		***Receiver***	
Parameters	***Time/ Frequency***	***Parameters***	***Time/ Frequency***
Maximum Delay of Clock Net (ns)	2.034	Maximum Delay of Clock Net (ns)	2.053
Minimum Period (ns)	8.125	Minimum Period (ns)	3.822
Maximum Frequency (MHz)	123.077	Maximum Frequency (MHz)	261.643
Maximum path delay from/to any node (ns)	8.125	Maximum path delay from/to any node (ns)	3.806

In this chapter, the implementation of a baseband OFDM system has been proposed from fixed-point modeling to device programming. The BER performance of the system under an AWGN channel has been evaluated using hardware co-simulation. The system functionality has been verified both at the system level and the RTL level.

Chapter 4 Design and Implementation of Channel Estimation for OFDM Systems

Chapters 2 and 3 are focused on the design and implementation of basic OFDM system and its performance under an AWGN channel. This chapter discusses the channel estimation techniques for the proposed OFDM system. First, the modeling and classification of multipath fading are introduced. The statistics for Rayleigh and Rician channels are then discussed. In addition, typical channel estimation techniques are introduced and compared. The LS estimation algorithm is applied to the proposed OFDM system and the BER performance over Rayleigh fading channels are simulated and compared. Finally, the receiver including estimation and equalization is implemented using FPGA.

4.1 Wireless Communication Channel

In wireless communication environments, signals may encounter reflection, refraction, and scattering during its propagation. Therefore, they arrive at the receiver through many different paths. This phenomenon is called multipath transmission as shown in Figure 4.1. As a result, in spite of the ideal AWGN channel, the transmitted signals also go through the multipath fading channel in the wireless communication systems.

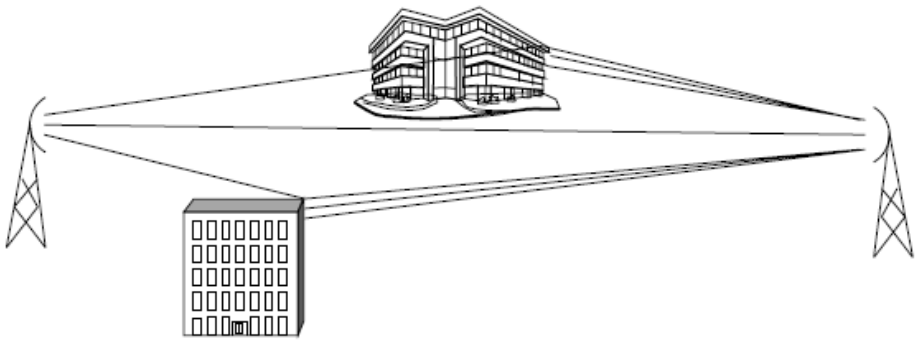
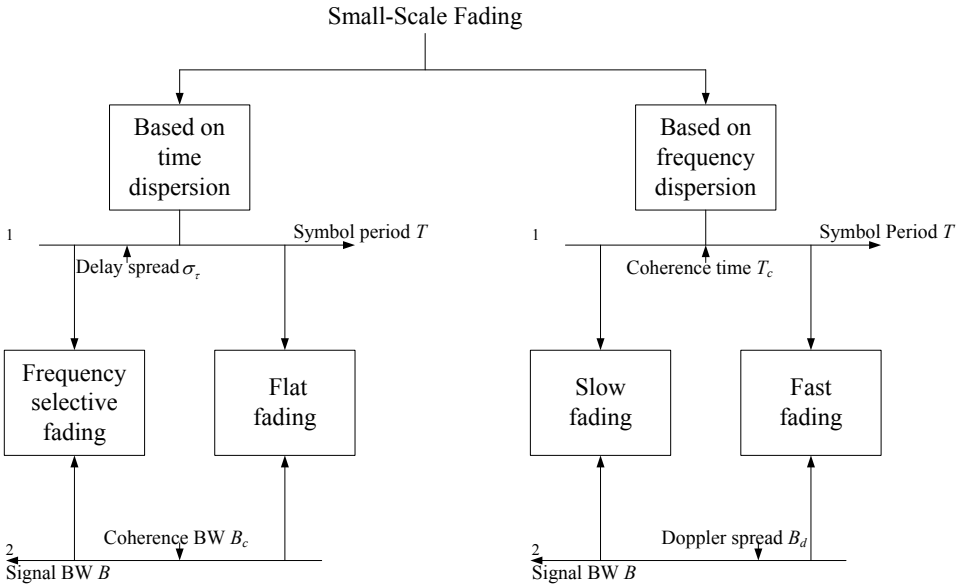


Figure 4.1 Multipath propagation [1].

4.1.1 Classification of Fading

The signal may encounter two kinds of fading during its propagation, which are large-scale fading and small-scale fading. Large-scale fading describes the signal power loss over a long transmit distance. In contrast, small-scale fading, which is caused by multipath propagation, refers to the rapid fluctuation of the signal strength over a short period of time or distance. The fading effects may be influenced by the transmission environment, relative speed of the receiver compared to that of the transmitter and surrounding objects, and the relationship between the signal bandwidth and the channel bandwidth as well.

In this thesis, we concentrate on the small-scale fading, which is also called multipath fading in some literature. In terms of time dispersion and frequency dispersion, the small-scale fading is characterized into four types. The classification is demonstrated in Figure 4.2 and will be described in detail in the following.



[1] Relationship of channel and signal period in time domain

[2] Relationship of channel and signal bandwidth in frequency domain

Figure 4.2 Classification of fading.

1) Slow and Fast Fading

When the receiver is moving during the propagation of the incoming signal, the spectrum of the received signal encounters a dispersion within a frequency range named Doppler spread, denoted by B_d. It is determined by the maximum Doppler shift f_d, which is defined as [28]:

$$f_d = \frac{v}{\lambda} = \frac{v}{c / f_c} = \frac{v}{c} f_c \tag{4.1}$$

where v is the constant velocity of the moving receiver, λ is the wavelength of the signal, c is the speed of light, and f_c is the signal carrier frequency. The time varying nature of a channel is characterized by B_d or coherence time T_c, which are defined as

$$B_d = 2f_d \tag{4.2}$$

$$T_c = \frac{0.423}{f_d} \tag{4.3}$$

Both of them describe the time variation rate of the channel compared to signal variation. Within the coherence time defined in (4.3), the channel is considered to have no variation.

When T_c is very large compared to the symbol period of the transmitted signal T, the signal changes much faster than the channel. The channel is considered as a slow fading channel. In this case, Doppler effects are very small or almost negligible. On the contrary, if T_c is smaller than T, or the channel has a high Doppler spread, the channel varies within a symbol period, then it is considered as a fast fading channel. To summarize, the channel is slow fading if

$$T_c >> T \text{ or } B_d << B,$$

and it undergoes fast fading when

$$T_c < T \text{ or } B_d > B.$$

where $B = 1/T$ is the bandwidth of the transmitted signal.

2) Flat and Frequency Selective Fading

To characterize the time dispersion property of a multipath fading, a pair of parameters which are inversely proportional to each other are used. They are the delay

spread σ_τ and the coherence bandwidth B_c. If σ_τ is far less than T, which means the channel bandwidth is much larger than that of the signal, the channel is called a flat fading channel. And it can be approximately modeled as a single Dirac function with a constant amplitude and linear phase response. In contrast, when σ_τ is larger than T, the channel will have frequency selective fading effects on the transmitted signals. In summary, the channel is flat fading if

$$\sigma_\tau << T \text{ or } B_c >> B,$$

and it undergoes frequency selective fading when

$$\sigma_\tau > T \text{ or } B_c < B.$$

By using the OFDM technique, the signal bandwidth is separated into several slices. Even if the channel is frequency selective within the entire band, it preserves the flat fading characteristics for each subcarrier slice. This effectively improves the system robustness against time dispersion in frequency selective fading channel.

4.1.2 Modeling of Multipath Fading Channel

In the fading channel, each path has an individual delay and attenuation that are varying with time and thus, generates multiple copies of the transmitted signal. All these weighted delayed copies are added at the receiver. Therefore, the baseband equivalent channel can be modeled as [31]

$$h(t,\tau) = \sum_i \tilde{\alpha}_i(t)\,\delta\left(\tau - \tau_i(t)\right) \tag{4.4}$$

where $\delta(\bullet)$ is the Dirac delta function, and $\tilde{\alpha}_i(t)$ and $\tau_i(t)$ are the complex-valued attenuation and excess delay of the *i-th* path component at instant time *t*, respectively. The excess delay is the time difference between the *i-th* arriving component and the first arriving one.

If the transmitted signal $s(t)$ is sampled at sampling period T_s, that is, it is band-limited to a two-sided bandwidth $B_s = 1/T_s$, it can be written in the discrete-time form as [32]

$$s(t-\tau) = \sum_l s(t - lT_s)\sin c\left(B_s(\tau - lT_s)\right) \tag{4.5}$$

Then the received signal at time *t* is

$$\begin{aligned} r(t) &= h(t,\tau) * s(t) \\ &= \int_{-\infty}^{\infty} h(t,\tau)\,s(t-\tau)\,d\tau \\ &= \int_{-\infty}^{\infty} h(t,\tau)\left(\sum_l s(t - lT_s)\sin c\left(B_s(\tau - lT_s)\right)\right)d\tau \\ &= \sum_l s(t - lT_s)\int_{-\infty}^{\infty} h(t,\tau)\sin c\left(B_s(\tau - lT_s)\right)d\tau \end{aligned} \tag{4.6}$$

where * denote convolution operation. Substituting (4.4) into (4.6) and after simplifying, we get

$$r(t) = \sum_l s(t - lT_s)\sum_i \tilde{\alpha}_i(t)\sin c\left(B_s\left(\tau_i(t) - lT_s\right)\right) \tag{4.7}$$

Defining the tap-gain $h_l(t)$ as:

$$h_l(t) = \sum_i \tilde{\alpha}_i(t) \sin c\left(B_s\left(\tau_i(t) - lT_s\right)\right) \tag{4.8}$$

we have

$$r(t) = \sum_l h_l(t) s(t - lT_s) \tag{4.9}$$

Therefore, the channel can be modeled as a tapped-delay-line (TDL) with equal spacing T_s. A practical TDL model with length L is shown in Figure 4.3, where $z(t)$ is the additive noise.

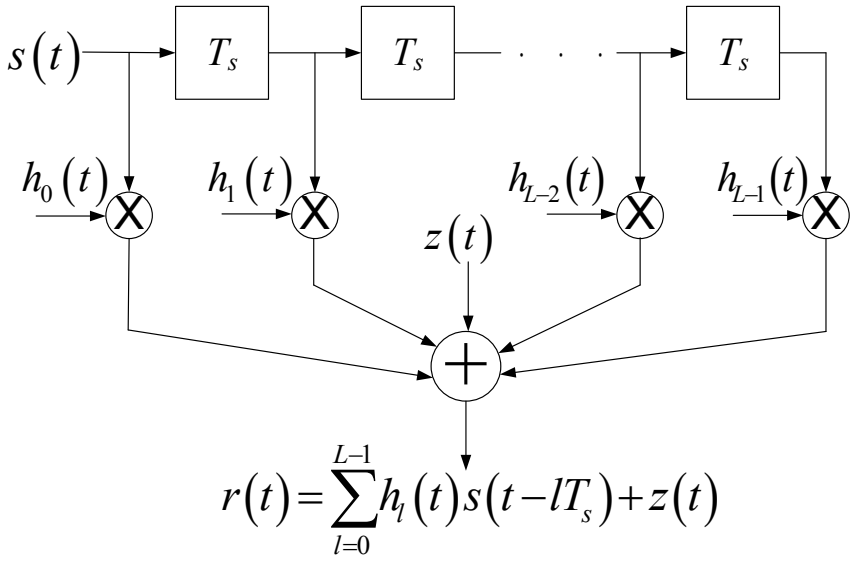


Figure 4.3 Modeling of the multipath channel using equally-spaced TDL.

Sampling at the same sample period of the transmitted signal T_s, the discrete-time representation of the received signal is given by

$$r(n) = \sum_{l=0}^{L-1} h_l(n) s(n - l) + z(n) \tag{4.10}$$

In the special case when the path attenuations $\tilde{\alpha}_i(t)$ and delays $\tau_i(t)$ are constant over a short period of time, that is, $\tilde{\alpha}_i(t) = \tilde{\alpha}_i$ and $\tau_i(t) = \tau_i$, the l_{th} tap gain is simplified to be

$$h_l = \sum_i \tilde{\alpha}_i \sin c\left(\frac{\tau_i}{T_s} - l\right) \tag{4.11}$$

4.1.3 Statistics of Fading Channel

Due to the existence of a large amount of scatters and the resulting independent propagation paths, in general, each multipath channel tap is the sum of a large number of diffuse components and a possible dominant component, namely, a light-of-sight (LOS) component. In general, the channel tap coefficient h_l can be modeled as [33]

$$h_l = \underbrace{\sqrt{\frac{K}{K+1}}\sigma_l e^{j\psi_0}}_{LOS} + \underbrace{\sqrt{\frac{1}{K+1}}\sum_{i=1}^{I}\sigma_l \alpha_i e^{j\psi_i}}_{NLOS} \tag{4.12}$$

where Rician factor K is defined as the power ratio of the LOS component to that of the non-line-of-sight (NLOS) components, σ_l is the average amplitude and independent of time, and α_i denotes the normalized real amplitude of each component satisfying $\sum_{i=1}^{I}\alpha_i^2 = 1$. The phase for each path is

$$\psi_i = 2\pi f_c \tau_i(t) + \varphi_i = 2\pi f_d \cos(\theta_i)t + \varphi_i \quad i = 0,1,\cdots,I \tag{4.13}$$

in which f_d is the maximum Doppler shift, θ_i is the angle between the direction of the receiver motion and the waveform arrival, which is independent and identically distributed (i.i.d.), and φ_i is i.i.d., and follows uniform distribution on $(-\pi, \pi]$. Therefore, the NLOS components can be modeled as circular symmetric complex random variables [31] with the sum of normalized power $E\left[|h_l|^2\right] = \sigma_l^2$.

When $I \to \infty$, according to the central limit theory, each channel tap can be described as a complex-valued Gaussian process. While there is no dominant component, which means the Gaussian process has a zero mean, the baseband signal envelope $|h_l|$ conforms with the Rayleigh distribution. If there is a dominant component, the Gaussian process has a non-zero mean, the envelope follows the Rician distribution.

1) Rician Fading

When there is a stationary dominant component such as an LOS signal, multiple random variables caused by multipath propagation are superimposed on this dominant signal. The channel can be modeled as a non-zero mean complex Gaussian process. Its envelope follows the Rician distribution with the following probability density function (pdf),

$$p(z) = \begin{cases} \frac{z}{\sigma^2} \exp[-\frac{z^2 + \alpha_0^2}{2\sigma^2}] I_0(\frac{\alpha_0 z}{\sigma^2}), & for \quad z \geq 0, \quad \alpha_0 \geq 0 \\ 0 \qquad\qquad , & otherwise \end{cases} \tag{4.14}$$

where α_0^2 and $2\sigma^2$ are the dominant signal power and the total of non-dominant signal power, respectively, $I_0(\bullet)$ is the zeroth-order modified Bessel function of the 1st kind, and the Rician factor $K = \frac{\alpha_0^2}{2\sigma^2}$. From Figure 4.4, it is obvious that when $K \to 0$, the power of the LOS path decreases and can be considered as an NLOS path, the Rician distribution turns into Rayleigh distribution; when $K \to \infty$, the power of the LOS path increases, and the Rician distribution turns into a Gaussian distribution. In the proposed design, a poor channel environment is assumed without an LOS path. Therefore, Rayleigh distribution is applied.

2) Rayleigh Fading Distribution

Rayleigh distribution is usually used for describing the statistics of the envelope for flat fading signal, or that of an individual multipath channel tap [28]. It has a pdf described by

$$p(z) = \begin{cases} \frac{z}{\sigma^2}\exp[-\frac{z^2}{2\sigma^2}], & for \quad z \geq 0 \\ 0 \qquad\qquad , & otherwise \end{cases} \tag{4.15}$$

The squared magnitude $\gamma = z^2$ is exponentially decayed with a pdf of

$$p(\gamma) = \frac{1}{2\sigma^2} e^{-\frac{\gamma}{2\sigma^2}}, \ \gamma \geq 0 \tag{4.16}$$

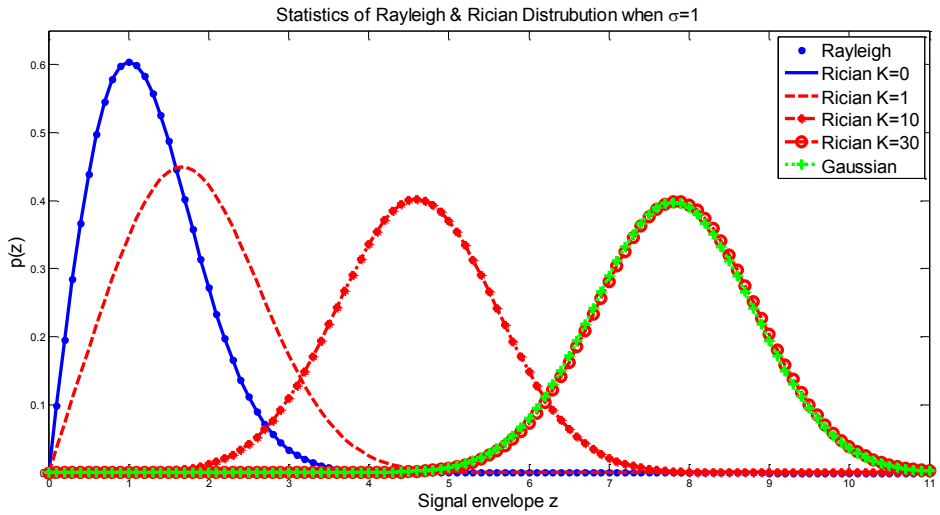


Figure 4.4 The pdfs of Rayleigh and Rician distributions.

4.2 Channel Estimation and Equalization for OFDM System

As discussed before, the transmitted signal will experience degradation in terms of amplitude attenuation and phase variation during its propagation along a multipath fading channel. These impairments will result in a significant degradation in the system performance compared to that in an AWGN channel. Usually, coherent demodulation is adopted at the receiver, as it achieves a performance better than that of a non-coherent demodulation scheme in terms of BER. For coherent demodulation, it is required that we have the knowledge of channel variations so that the channel effect could be compensated at the receiver. This process is called channel estimation and equalization [3].

4.2.1 Channel Estimation Methods for OFDM System

Basically, channel estimation methods can be classified into three classes: pilot-assisted (PA), decision-directed (DD), and blind.

In the blind channel estimation approaches, a large amount of received data is required to analyze the statistics of the received signal that are utilized to estimate the channel. Although the absence of pilot improves the bandwidth efficiency, the application of blind estimation is limited to time-varying channels [34] for it requires complex computation and suffers from slow convergence. Compared to PA channel estimation methods, blind estimation methods usually have worse performance, especially in fast fading channels [35].

The PA and DD channel estimation may apply the same estimation algorithms. The difference is the source of the input to the estimator [3]. In the PA channel estimation, the pilot symbols named preambles or pilot subcarriers are transmitted periodically across time or frequency axis. With the knowledge of the pilots, the receiver can extract the channel information for the dedicated time and subcarriers. After applying various interpolation methods, the channel estimates for specific time and subcarrier can be obtained.

DD channel estimation, however, uses the channel estimates for previous symbols to demodulate the current OFDM symbol. In such methods, all the subcarriers of each OFDM symbol are used to estimate the channel. Since the pilots are not used, the bandwidth and power loss brought by pilots in PA methods are reduced. However, the accumulation of estimation errors makes it not as reliable as the PA methods. If the

channel varies significantly within adjacent symbols, the system performance may degrade greatly.

Even in the DD channel estimation methods, some preambles or pilots are required to perform an initial estimation. The PA estimation techniques are widely adopted in most wireless communications, especially the burst communication system, since it can achieve reliable estimation accuracy in a short time. In the system design, the tradeoff among estimation precision, pilot length and density, and signal throughput and power is usually considered.

4.2.2 Pilot-assisted Estimation for OFDM System

In PA estimation methods, pilots are inserted in time and frequency domains in certain patterns. The two basic kinds of 1D patterns presented in Figure 4.5, are block-type and comb-type. They can be combined to form different 2D patterns as shown in [36].

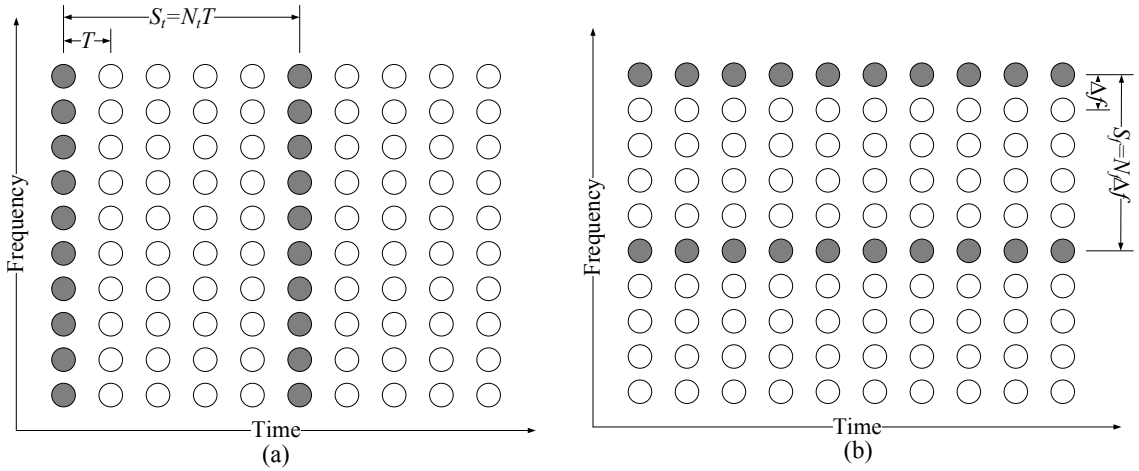


Figure 4.5 Pilot arrangement for (a) block-type and (b) comb-type [45].

For block-type, the pilots are inserted into all the subcarriers in every period of time to form a special class of symbols named preambles or training symbols, marked as solid circles in Figure 4.5(a). This type is suitable for the estimation of a frequency-selective fading channel, but not efficient for a fast fading channel. To improve the system performance over a fast fading channel, certain subcarriers in each symbol should be reserved as pilots. This is the comb-type shown in Figure 4.5 (b).

In order to estimate the time varying characteristics, the channel should remain invariant within the coherence time $T_c \approx 1/B_d$. In other words, the pilot period needs to be smaller than T_c. Define S_t to be the pilot spacing in time [3]:

$$S_t = N_t T \tag{4.17}$$

where N_t is the spacing between adjacent pilot symbols in time domain, and T is the OFDM symbol time. For successful estimation, it should satisfy [2]

$$S_t < \frac{1}{B_d} \tag{4.18}$$

Similarly, for the purpose of characterizing channel frequency variation, the pilot spacing S_f in the frequency domain must be smaller than the coherence bandwidth, which is inversely proportion to the channel delay spread σ_τ.

$$S_f < \frac{1}{\sigma_\tau} \tag{4.19}$$

where

$$S_f = N_f \Delta f \tag{4.20}$$

4.2.3 Least Squares Estimation

The LS estimation is the most fundamental method in pilot-assisted algorithms [37]. As the proposed research is based on the burst communication system described in 802.11a, the channel is assumed to remain the same over the time of a burst [19]. In this case, LS estimation is appropriate to be applied. Although other algorithms such as minimum-mean-square-error (MMSE) perform better than LS, it is adopted in the proposed system for its low implementation complexity.

Suppose **S** is the transmitted signal in the frequency domain, **R** is the frequency response of the received signal, **H** is the frequency impulse response of the fading channel, $\hat{\mathbf{H}}$ is the estimate of **H**, and **Z** is the additive noise. Then,

$$\mathbf{R} = \mathbf{H} \cdot \mathbf{S} + \mathbf{Z} \tag{4.21}$$

in which

$$\mathbf{R} = \left[R_0, R_1, \cdots, R_{N-1} \right]^T$$

$$\mathbf{S} = diag\left[S_0, S_1, \cdots, S_{N-1} \right], \text{ with } E\{S_k\} = 0 \text{ and } Var\{S_k\} = \sigma_s^2$$

$$\mathbf{H} = \left[H_0, H_1, \cdots, H_{N-1} \right]^T$$

$$\mathbf{Z} = \left[Z_0, Z_1, \cdots, Z_{N-1} \right]^T, \text{ with } E\{Z_k\} = 0 \text{ and } Var\{Z_k\} = \sigma_z^2$$

where $diag[\bullet]$ denotes the diagonal matrix and $[\bullet]^T$ represents matrix transpose, and $E[\bullet]$ and $Var[\bullet]$ are the mean and variance, respectively. The basic idea of LS

estimation is to find the estimates of the channel that minimizes the cost function $J\left(\hat{\mathbf{H}}\right)$ as given by

$$J\left(\hat{\mathbf{H}}\right)=\left\|\mathbf{R}-\hat{\mathbf{H}}\mathbf{S}\right\|^2=\left(\mathbf{R}-\hat{\mathbf{H}}\mathbf{S}\right)^H\left(\mathbf{R}-\hat{\mathbf{H}}\mathbf{S}\right) \tag{4.22}$$

where $\left[\bullet\right]^H$ denotes conjugate transpose. By forcing $\frac{\partial J\left(\hat{\mathbf{H}}\right)}{\partial\hat{\mathbf{H}}}=0$, we get the desired $\hat{\mathbf{H}}$:

$$\hat{\mathbf{H}}=\mathbf{S}^{-1}\mathbf{R} \tag{4.23}$$

The mean square error (MSE) is an important parameter that describes the performance of an algorithm. It is defined by

$$MSE=E\left\{\left(\mathbf{H}-\hat{\mathbf{H}}\right)^H\left(\mathbf{H}-\hat{\mathbf{H}}\right)\right\} \tag{4.24}$$

By substituting (4.23) in (4.24), we get

$$MSE=\frac{E\left\{\mathbf{Z}^H\mathbf{Z}\right\}}{E\left\{\mathbf{S}^H\mathbf{S}\right\}}=\frac{\sigma_z^{\,2}}{\sigma_s^{\,2}} \tag{4.25}$$

4.3 MATLAB Simulation Results for an OFDM System

4.3.1 Simulation of IEEE 802.11a Channel

According to analysis in Section 4.1, the multipath fading channel is modeled as a finite impulse response (FIR) filter. For an indoor wireless channel, the channel tap gains

conform to Rayleigh distribution. The average power delay profile follows exponential model [2]

$$p_l = p_0 \cdot e^{-\beta_0 l}, \quad l = 0, \cdots, L-1 \tag{4.26}$$

and satisfies $\sum_{l=0}^{L-1} p_l = 1$, where p_0 is the average power of first tap, β_0 is a parameter determined by the sampling time T_s and the delay spread σ_τ, and L is the channel length. The channel is modeled as

$$h(t) = \sum_{l=0}^{L-1} \sqrt{p_l} h_l \delta(t - lT_s) \tag{4.27}$$

where h_l is an i.i.d. complex Gaussian random variable with $\sigma^2 = \frac{1}{2}$. A typical IEEE 802.11 channel with $\sigma_\tau = 25$ ns and $T_s = 25$ ns is shown in Figure 4.6.

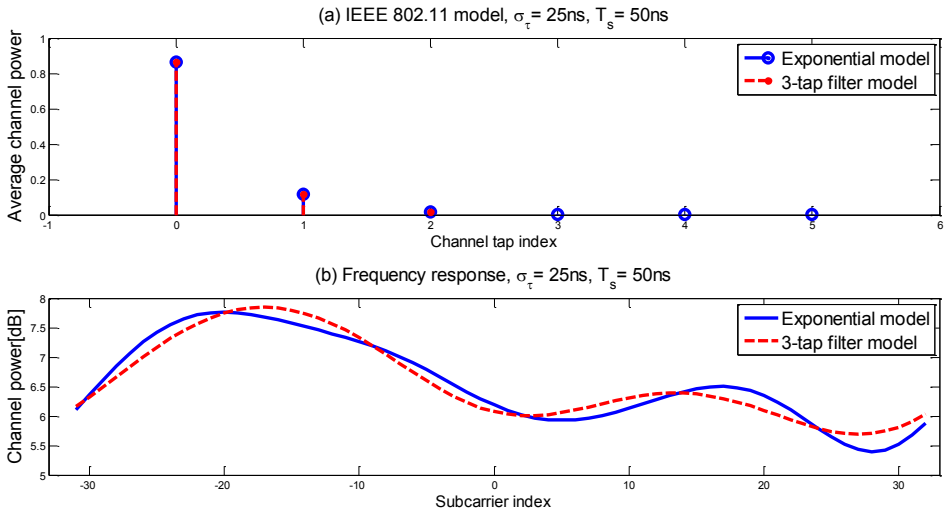


Figure 4.6 Simulation of IEEE 802.11 channel (a) average channel power and (b) frequency response.

The channel is assumed to be invariant in one OFDM frame which consists of 10 OFDM symbols. The channel tap gains changes for each OFDM frame. The frame size is set to be relatively small to verify the system performance in a severe communication environment. In practical indoor communications, it is properly chosen to reduce the overhead. For implementation simplicity, the first three taps are used to generate the proposed channel. From the frequency response shown in Figure 4.6(b), we can observe that the 3-tap FIR filter model keeps the generality of the desired channel.

4.3.2 LS Estimation for OFDM System Based on 802.11a

For the indoor communication system based on IEEE 802.11a, the channel is assumed to be invariant in one frame whose period lasts for several OFDM symbols. Therefore, the LS technique based on block-type arranged pilots is applied. The channel is estimated from the long training sequence illustrated in Figure 2.5. It includes two identical LTSs. Each symbol has 64 samples in the time domain and consists of 53 subcarriers (including dc) as indicated below.

$$L_{-26,26} = \begin{Bmatrix} +1,+1,-1,-1,+1,+1,-1,+1,-1,+1,+1,+1,+1,+1,+1,-1,-1,+1, \\ +1,-1,+1,-1,+1,+1,+1,+1,0,+1,-1,-1,+1,+1,-1,+1,-1,+1, \\ -1,-1,-1,-1,-1,+1,+1,-1,-1,+1,-1,+1,-1,+1,+1,+1,+1 \end{Bmatrix}$$

The estimates of the channel frequency response is rewritten according to (4.23) as

$$\hat{\mathbf{H}} = \mathbf{L}_{\mathbf{X}}^{-1}\mathbf{L}_{\mathbf{Y}} \tag{4.28}$$

where $\mathbf{L_X}$ and $\mathbf{L_Y}$ are the transmitted and received long symbols in the frequency domain, respectively. To further improve the system performance, the average of the two received LTSs is used to calculate the channel estimates.

$$\hat{\mathbf{H}} = \frac{1}{2}\left(\mathbf{L_X}^{-1}\mathbf{L_{Y1}} + \mathbf{L_X}^{-1}\mathbf{L_{Y2}}\right) = \frac{1}{2}\mathbf{L_X}^{-1}\left(\mathbf{L_{Y1}} + \mathbf{L_{Y2}}\right) = \mathbf{H} + \frac{1}{2}\left(\mathbf{Z_1} + \mathbf{Z_2}\right) \tag{4.29}$$

in which $\mathbf{L_{Y1}}$ and $\mathbf{L_{Y2}}$ are the frequency transforms of the first and second received LTS, respectively, and $\mathbf{Z_1}$ and $\mathbf{Z_2}$ are the additive noise of the two LTSs. Figure 4.7 compares the frequency response between the true channel and the estimates at different noise values. The comparison is performed along all the 52 non-null subcarriers. From this figure, we can conclude that the deviation between the two responses is reduced with increasing E_b/N_0. In any case, the estimation with two LTSs yields better results than that with one LTS. However, the improvement is reduced at high E_b/N_0.

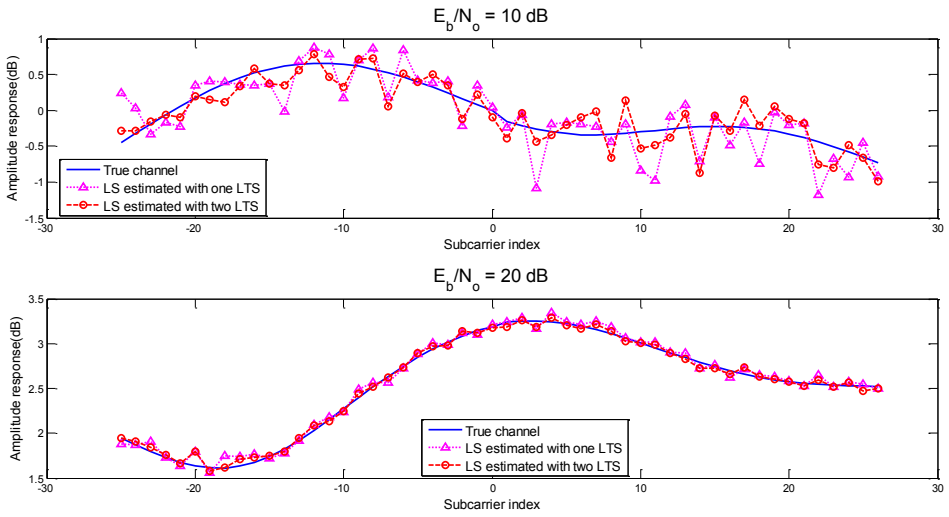


Figure 4.7 Comparison of channel response between estimates and true channel.

In addition, when two LTSs are used, MSE is reduced to one-half of that using only one LTS. According to (4.25),

$$\begin{cases} MSE_{1LTS} = \dfrac{\sigma_z^{\,2}}{\sigma_s^{\,2}}, & when \quad one \quad LTS \quad is \quad used \\ MSE_{2LTS} = \dfrac{\sigma_z^{\,2}}{2\sigma_s^{\,2}}, & when \quad two \quad LTS \quad are \quad used \end{cases} \tag{4.30}$$

From Figure 4.8, it is seen that the simulated MSE matches the theoretical one given by (4.30). When two LTSs are used to estimate the channel, MSE performance improves by about 3 dB compared to that using only one LTS, which is the same as given by the theoretical analysis.

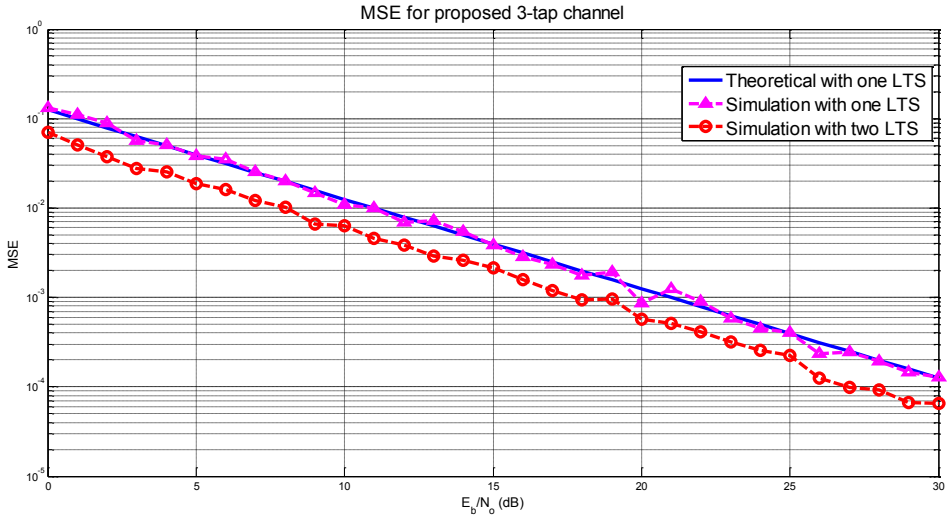


Figure 4.8 MSE comparison.

4.3.3 Simulation of BER Performance

According to the analysis given in Section 2.3.2, the BER of the OFDM system is the same as that for a single carrier system in the absence of ISI and ICI. When the coherent QPSK modulation is performed, the instantaneous BER is considered as a conditional error probability given that the received signal to noise ratio (SNR) γ_b is known [38]. According to (2.12), it is expressed as

$$P\left(E|\gamma_b\right) = Q\left(\sqrt{2\gamma_b}\right) \tag{4.31}$$

where γ_b is also a random variable as we can treat the fading channel as an AWGN channel with a random gain h_l and given by

$$\gamma_b = \frac{|h_l|^2 E_b}{N_0} \tag{4.32}$$

The BER for a fading channel is obtained by averaging $P\left(E|\gamma_b\right)$ over the pdf of γ_b:

$$P_b = \int_0^\infty P\left(E|\gamma_b\right) p\left(\gamma_b\right) d\gamma_b \tag{4.33}$$

When the channel is flat and slowly fading, the fading coefficient h_l follows the Rayleigh distribution with standard deviation $2\sigma^2$. Also, h_l^2 has a chi-square distribution with two degrees of freedom with pdf shown in (4.16). Thus, γ_b is also chi-square distributed. Define

$$\overline{\gamma}_b = E\left(\gamma_b\right) = E\left(h_l^2\right)\frac{E_b}{N_0} = 2\sigma^2\frac{E_b}{N_0} \tag{4.34}$$

Then, the pdf of γ_b can be written as

$$p(\gamma_b) = \frac{1}{\overline{\gamma}_b} e^{-\frac{\gamma_b}{\overline{\gamma}_b}} \tag{4.35}$$

By substituting (4.35) into (4.33) and performing the integration, we get

$$P_b = \frac{1}{2}\left(1 - \sqrt{\frac{\overline{\gamma}_b}{\overline{\gamma}_b + 1}}\right) \tag{4.36}$$

The BER performance of a Rayleigh fading channel is compared to that of an AWGN channel in Figure 4.9. It is observed that for a fading channel, if the channel estimation is not performed, the BER is almost 0.5, which is very high, and cannot get improved when E_b/N_0 is increased. A BER comparison of AWGN and Rayleigh fading channels asserts that the fading effects degrades the system performance especially at high E_b/N_0. Also, when the LS estimation is applied, the BER performance got a significant improvement, and the improvement increases as E_b/N_0 goes higher. On the other hand, the simulation results show a degradation of around 0.5 dB compared to the theoretical values.

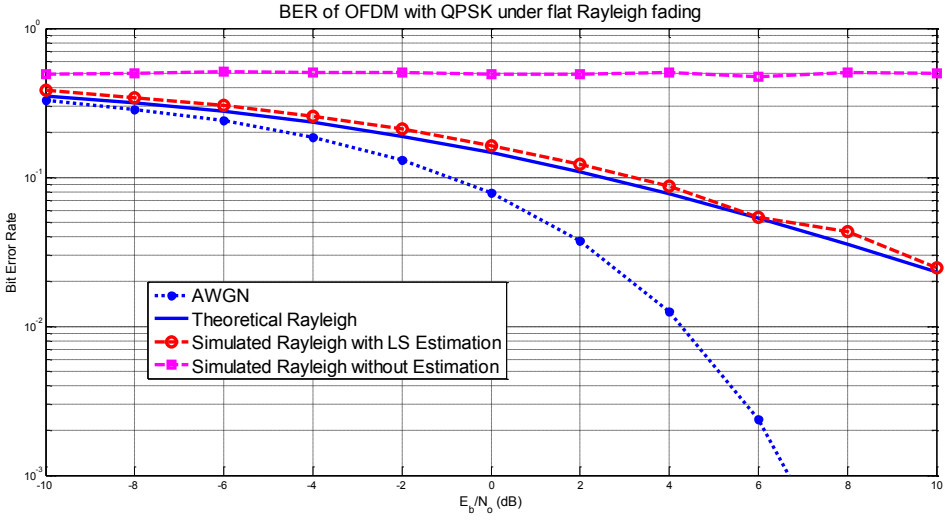


Figure 4.9 BER performance of an OFDM system under Rayleigh fading channel.

To illustrate the BER performance over a frequency selective channel, Table 4.1 lists five kinds of fading channels with various models. For implementation simplicity, the frequency selective channel is modeled as a 3-tap FIR filter. Each channel tap conforms to the Rayleigh distribution. According to Section 4.3.1, the underlying IEEE 802.11 channel is the fourth one in Table 4.1.

The simulation results plotted in Figure 4.10 show that the BER performance remains almost the same for the channels with the same total received power, no matter what kind of channel model is applied. When the total power is doubled, the BER is enhanced by nearly 3 dB.

Table 4.1 Five kinds of fading channels.

Channel Index	Average Power Model	Average Power Proportion	Total of Received Power
1	Flat	1	$\sum_{l=0}^{L-1} p_l = 1$
2	Equally	1:1:1	$\sum_{l=0}^{L-1} p_l = 1$
3	Halfly Decayed	1:0.5:0.25	$\sum_{l=0}^{L-1} p_l = 1$
4	Exponentially Decayed	$1 : e^{-\beta_0} : e^{-2\beta_0}$	$\sum_{l=0}^{L-1} p_l = 1$
5	Exponentially Decayed	$1 : e^{-\beta_0} : e^{-2\beta_0}$	$\sum_{l=0}^{L-1} p_l = 2$

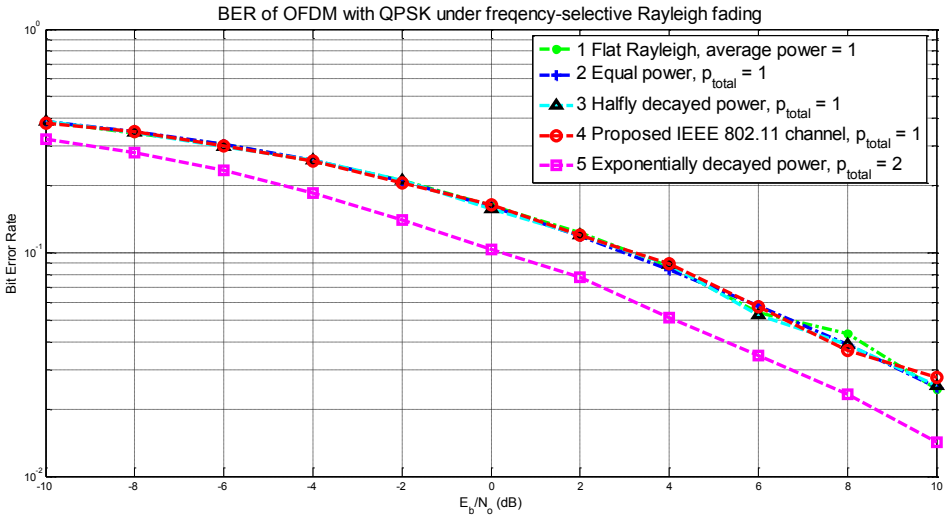


Figure 4.10 BER performance of an OFDM system under frequency-selective channels.

4.4 FPGA Implementation of an LS Estimator

4.4.1 Modeling of the Receiver with LS Estimation

The channel frequency response is estimated using LS algorithm. The long training sequence is used to perform the estimation after the synchronization. In this research, ideally perfect synchronization is assumed.

Using (4.28) and (4.29), the two identical long symbols are averaged to improve the estimation quality. To reduce the calculation complexity, the average is performed in the time domain before FFT processing. The result is not affected because the FFT transform is linear. Using L_{y1} and L_{y2} to represent the two received LTSs in the time domain, we have

$$\mathbf{L_Y} = \frac{1}{2}\left(\mathbf{L_{Y1}} + \mathbf{L_{Y2}}\right) = \frac{1}{2}\left[FFT\left(L_{y1}\right) + FFT\left(L_{y2}\right)\right] = FFT(\frac{L_{y1} + L_{y2}}{2}) \tag{4.37}$$

The channel effects are equalized by applying zero forcing (ZF) method. Recall that **R** is the frequency transformation of the received OFDM symbols. The estimation of transmitted symbol is obtained by dividing the received symbols in frequency domain by estimated channel response. Since the division calculation is complicated and resource-consuming, it is accomplished by complex multiplication without performance reduction. Thus, we obtain the following estimated frequency-domain symbol,

$$\hat{\mathbf{S}} = \hat{\mathbf{H}}^{-1}\mathbf{R} = \frac{\mathbf{L_X}}{\mathbf{L_Y}}\mathbf{R} = \frac{\mathbf{L_X}}{\left|\mathbf{L_Y}\right|^2}\mathbf{L_Y}^{*} \cdot \mathbf{R} \tag{4.38}$$

For QPSK demodulation using hard decision method, $\hat{\mathbf{S}}$ are compared with threshold "0" to determine whether the transmitted data is bit 0 or 1. Therefore, the circuit could be further simplified by avoiding the division of LTS power.

$$\hat{\mathbf{S}} = \mathbf{L_X} \cdot \mathbf{L_Y}^{*} \cdot \mathbf{R} \tag{4.39}$$

The block diagram that implements the LS estimator is demonstrated in Figure 4.11. First, the average of LTS in the time domain is calculated. After FFT, $\mathbf{L_Y}$ is obtained. As stated in Section 4.3, they are fed to the "Subcarrier Deallocation" to get the 48 data subcarriers of $\mathbf{L_Y}$. At the same time, the values at the pilot subcarriers are also separated. These data subcarriers of $\mathbf{L_Y}$ are stored in two 48 x 16 single port RAMs. The result of the complex multiplication of $\mathbf{L_Y}^{*} \cdot \mathbf{R}$ is finally multiplied by $\mathbf{L_X}$, which is -1 or 1, stored in a 48 x 2 ROM.

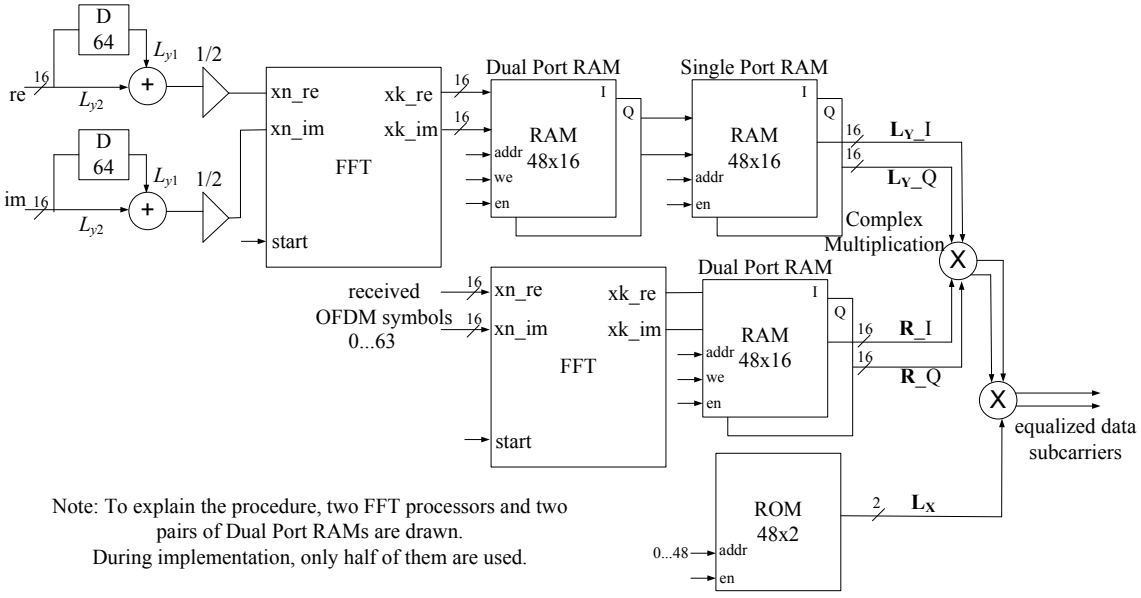


Figure 4.11 Block diagram of the channel estimation and equalization.

Figure 4.12 shows the implementation of the OFDM receiver including channel estimation and equalization. Three blocks are added in addition to the basic OFDM receiver design in this implementation diagram. The "LTS_Average" block calculates the average of LTSs ($\frac{L_{y1}+L_{y2}}{2}$). The "LTS_RAM" uses two single port RAMs to store the complex values for data subcarriers of LTS ($\mathbf{L_Y}$). The "Equalizer" performs complex multiplication of $\mathbf{L_X} \cdot \mathbf{L_Y}^* \cdot \mathbf{R}$ and removes the channel effects for data subcarriers.

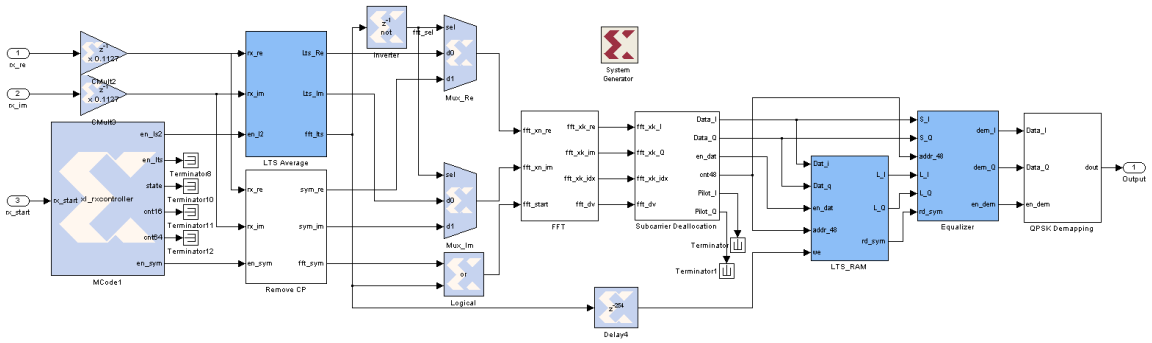


Figure 4.12 Implementation module of the channel estimation and equalization.

The kernel block, which is the equalizer, is implemented as shown in Figure 4.13. The complex multiplication is implemented using four multipliers and two adders. The complete circuit requires six multipliers in total.

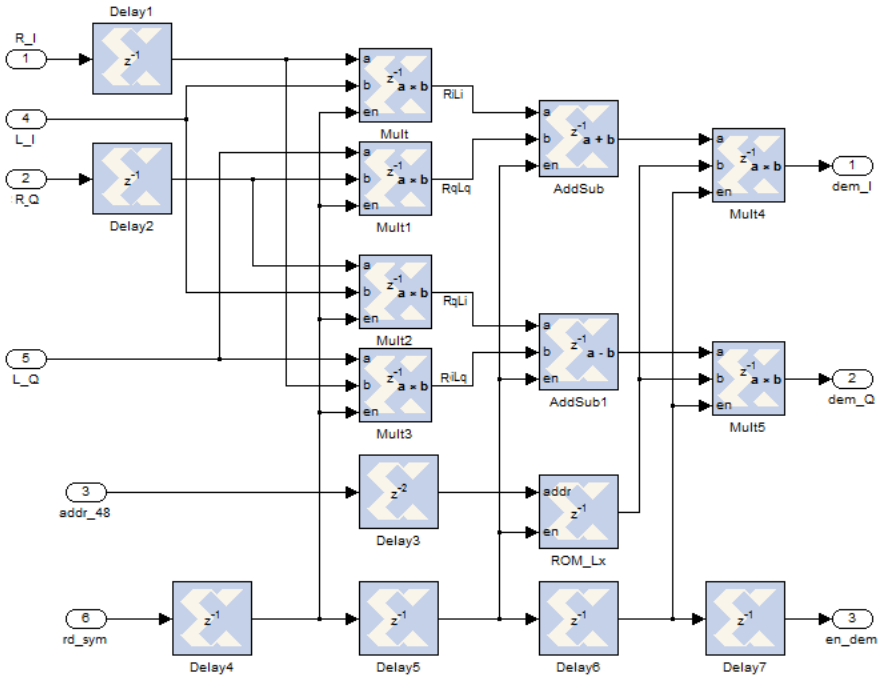
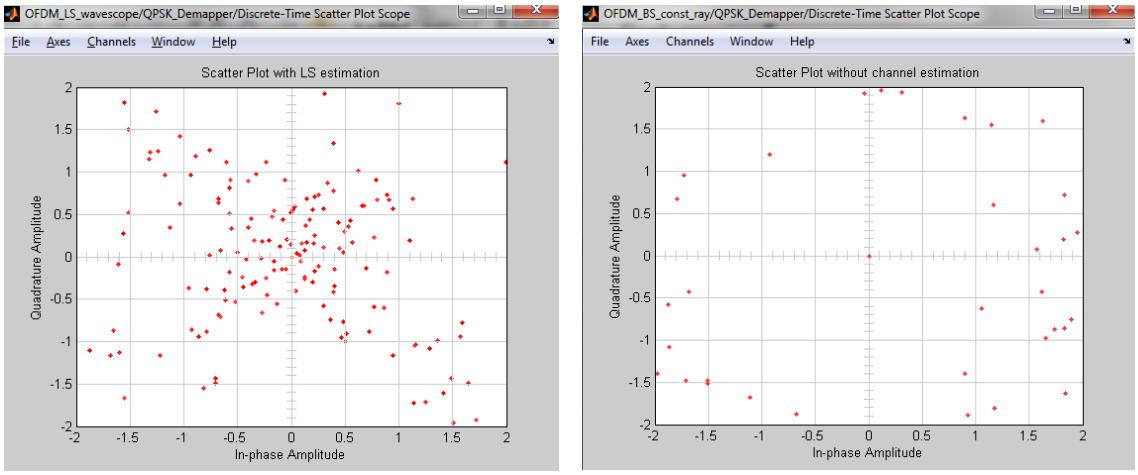


Figure 4.13 Implementation module of the equalizer.

4.4.2 System Performance

The fading channel with index 4 in Table 4.1 is applied. It is modeled according to IEEE 802.11 standards. For implementation simplicity, the channel is built based on a 3-tap FIR filter with each tap gain conforming to the Rayleigh distribution. Figure 4.14(a) and (b) depict the constellation diagrams at the receiver with LS estimation and without estimation, respectively. The value of E_b/N_0 is 5 dB.



(a) (b)

Figure 4.14 Constellation diagram at the receiver (a) with LS estimation and (b) without channel estimation.

The BER versus E_b/N_0 curves based on floating-point and fixed-point systems are illustrated in Figure 4.15. It is observed that the BER curve of the 16-bit fixed-point model matches that of the floating-point model at low SNR, while the degradation occurs at high SNR. The BER degradation reaches 0.9 dB at E_b/N_0 = 10 dB.

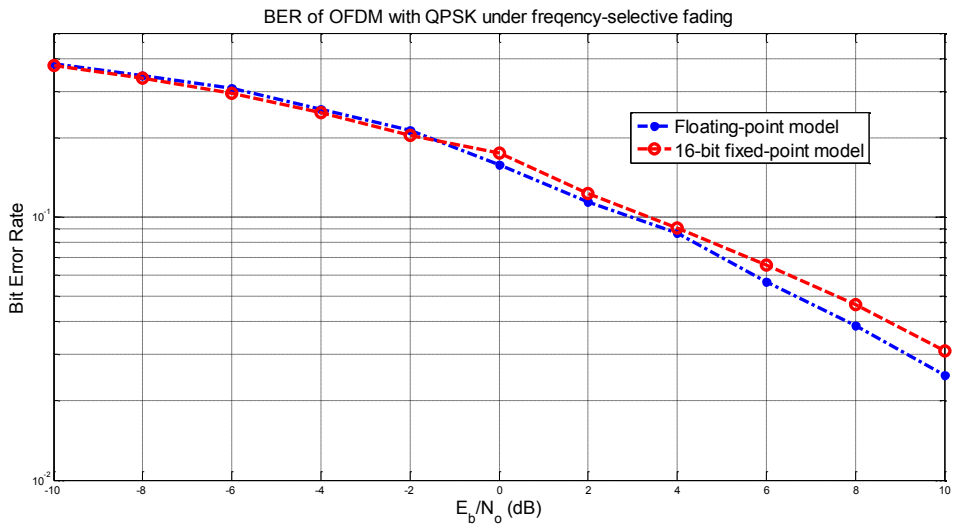


Figure 4.15 BER comparison of an OFDM system under frequency-selective fading channel.

4.4.3 Implementation Results

Table 4.2 and Table 4.3 list the resource consumption and timing results for the receiver with LS estimation, respectively. It is observed that no timing conflict occurs in the design. The maximum frequency is 97.991 MHz, which is enough to generate 40 MHz clock and drive the whole system. By comparing Table 3.1 and Table 4.2, we can conclude that when the channel compensation is implemented at the receiver, the consumption of resources like slice registers, slice LUTs, bonded IOBs and BlockRAMs are almost increased by 1%, while the usage of DSP48Es is increased by three times that used in the basic design. The minimum period is almost doubled and the maximum frequency is reduced by half. The maximum path delay is almost doubled. However, because of high pipelined design, the clock delay remains to be around 2 ns.

Table 4.2 Area results for receiver with LS estimation.

Resources	***Used***	***%***
Number of Slice Registers	2806	4
Number of Slice LUTs	2675	3
Number of bonded IOBs	47	7
Number of BlockRAM/FIFO	3	2
Number of BUFG/BUFGCTRLs	1	3
Number of DSP48Es	18	28

Table 4.3 Clock and timing results for receiver with LS estimation.

Parameters	***Time Frequency***
Maximum Delay of Clock Net (ns)	2.121
Minimum Period (ns)	8.287
Maximum Frequency (MHz)	120.671
Maximum path delay from/to any node (ns)	8.287

4.5 Summary

This chapter has studied the channel estimation and equalization for OFDM systems. It started with an introduction of multipath fading channel, followed by channel modeling according to IEEE 802.11. After a brief comparison of different estimation techniques, the LS algorithm was presented to the proposed OFDM system. The LS estimator was implemented at the receiver by adding the functional blocks to the basic receiver. To verify the system performance, simulations were carried out over Rayleigh fading channels and BER vs E_b/N_0 curves were obtained. The implementation results have shown that receiver with channel estimation and equalization requires a small increase of resource consumption compared to the basic receiver.

Chapter 5 Design and FPGA Implementation of OFDM Synchronization

The channel estimation technique proposed in Chapter 4 is based on the assumption that the coherent reception is perfect. That is, the system synchronization is perfectly implemented. Synchronization is one of the most challenging and important tasks to any receiver using coherent modulation, especially for OFDM systems which are highly sensitive to synchronization errors. This chapter proposes a synchronizer based upon the delay and correlation algorithm and the synchronization system is modeled within XSG. Simulation results have been provided for presenting the system synchronization performance clearly. The implementation results on FPGA are given at the end.

5.1 Synchronization for OFDM

5.1.1 Introduction to OFDM Synchronization

Generally speaking, synchronization is divided into timing and frequency synchronization. In practice, the oscillator does not produce a carrier at exactly one frequency, and the carrier frequency is modulated by random phase jitter [2]. The phase noise introduced results in a mismatch of frequencies between the local oscillators at the transmitter and the receiver. In addition, Doppler effects in fading channel also causes frequency offsets at the receiving carrier. As a result, the numbers of subcarriers cycles

within the FFT period are not integers anymore, and the orthogonality property between subcarriers is not kept, causing ICI for an OFDM system. In contrast, in a single carrier system, the phase noise and frequency offsets only reduce the received SNR. Therefore, OFDM is extremely sensitive to frequency offsets rather than single carrier systems [3]. Any frequency offset inevitably introduces ICI. This is considered as a shortcoming of the OFDM technique. However, with the utilization of frequency synchronization techniques, the loss of orthogonality could be compensated. So the performance degradation caused by frequency errors could be minimized.

Rather than frequency offsets, OFDM is more insensitive to timing errors. From the analysis given in Chapter 2, when the symbol timing offset is less than the guard interval, there is no ISI or ICI introduced. Although the timing offset may produce a time varying phase rotation to each subcarrier, it could be eliminated by means of channel estimation [3]. However, if the timing offset is longer than the guard interval, ISI would be introduced and the orthogonality between subcarriers will no longer be preserved. The task of timing synchronization is to find the symbol boundaries to prevent ISI and ICI. Even if the timing errors are small enough, time synchronization could improve the system robustness to multipath fading.

5.1.2 Classification of Synchronization Schemes

For OFDM systems, it is always required that the time and frequency synchronization be accomplished concurrently [3]. The most popular methods are the correlation techniques, which utilize the correlation between the signal and it replica for synchronization.

There are two kinds of correlation techniques. One is based on the cyclic extension, the other is based on the training symbols, also called preambles, which are known to the receiver in advance. Since the correlation methods that use cyclic extension can only detect symbol timing, but fail in finding as to where a packet starts, it is only appropriate in broadcast systems. For packet transmission with high data rate, the methods based on training symbols are more suitable and reliable, as they are able to track the frame, and synchronize the time and frequency coarsely in very short time.

For a burst OFDM system, the first synchronization task is the estimation of the start point of a frame, which is called frame/packet detection. The following sections show the implementation of a frame detector for OFDM packets.

5.1.3 Introduction of Preambles for IEEE 802.11a

The proposed work is designed under a burst communication environment according to the IEEE 802.11a standard. It is assumed that the channel does not change significantly during one OFDM frame. From the standard introduced in Chapter 2, an OFDM preamble consists of two parts for the purpose of synchronization. The first part consists of 10 STSs used for frame detection and coarse estimation of time and frequency offset; while the second part is the long training sequence employed for fine frequency tuning and channel estimation, which were described in Chapter 4. The details were shown in Figure 2.5.

The short training sequence consists of 12 subcarriers. The subcarrier elements of index number from -26 to 26 in the frequency domain are given by

$$T_{-26,26} = \sqrt{13/6} \times \begin{Bmatrix} 0,0,1+j,0,0,0,-1-j,0,0,0,1+j,0,0,0,-1-j,0,0,0,1+j,0,0,0,1+j,0,0,0, \\ 0,0,0,0,-1-j,0,0,0,-1-j,0,0,0,1+j,0,0,0,1+j,0,0,0,1+j,0,0,0,1+j,0,0 \end{Bmatrix}$$

where $\sqrt{13/6}$ is the factor that normalizes the average power for STS, in which 12 out of 52 subcarriers are used. The 52 subcarriers, as well as a dc sub-channel, are mapped into a 64-point IFFT converter according to the mapping rule shown in Figure 2.6. The 64 outputs in the time domain happen to be 4 repeated symbols with 16 samples each. This 16-sample STS is then repeated 10 times to form a 160-sample short training sequence. Each symbol in this sequence has a duration of 800 ns and contains 16 samples. They are identical to one another.

The structure of IEEE 802.11a preamble enables the receiver to use a very simple and efficient algorithm to detect an OFDM frame. This approach was presented by Schmidl and Cox in [40]. In the following sections, we will first compare it with other OFDM synchronization schemes, then present the simulation results of detection, and finally illustrate the implementation procedure.

5.2 Schmidl and Cox Synchronization Scheme

5.2.1 Delay and Correlation Algorithm

With the utilization of the periodic property of the short symbols, cross-correlation of the received signal is employed for detecting the start of the preamble. This is called delay and correlation algorithm. The received signal is complex correlated with its delayed copy and summed over a sliding window. We can also consider it as the cross-correlation

of two halves of the training sequence. The two halves of this sequence are identical to each other. So a statistical measure of the signal, namely,

$$C(n) = \sum_{d=0}^{W-1} r(n+d) r^*(n+d+D) \tag{5.1}$$

is obtained, where n is the sample number, $r(n)$ is the received signal in time domain, and D and W are the lengths of the delay and sliding window, respectively. For IEEE 802.11a application, D and W are chosen to be integer times the samples number in one STS (16). Hence, if there is no OFDM frame being received, the received signal only consists of noise. The correlation function $C(n)$ is a zero-mean random variable. On the other hand, when the start of an OFDM frame is received, $C(n)$ is a cross-correlation of identical short symbols. It reached the maximum value in a very short time. By monitoring the value change of $C(n)$, the start of an OFDM frame is determined.

Several measuring methods have been introduced. The simplest approach is maximum-correlation (MC) [41] scheme. In this scheme, the start of a frame is detected when $C(n)$ reaches to maximum. Since the measured energy is not normalized within the synchronization window and varies in a wide range, it is hard to decide a threshold as the maximum value. So this method is not suitable for multipath fading communication and non-constant envelope modulation [42].

To deal with the problem for MC approach, Schmidl and Cox proposed a MNC-based method in [40]. In this method, another sliding window is applied to calculate the power of the delayed signal

$$P(n)=\sum_{d=0}^{W-1} r(n+d+D)r^{*}(n+d+D)=\sum_{d=0}^{W-1}\left|r(n+d+D)\right|^{2} \tag{5.2}$$

The decision statistic called timing metric is obtained as

$$M(n)=\frac{\left|C(n)\right|^{2}}{\left|P(n)\right|^{2}} \tag{5.3}$$

where a normalization factor of $\left|P(n)\right|^{2}$ is introduced to narrow the fluctuation range.

Other methods include MMSE [43] and maximum-likelihood (ML) [44] schemes. MMSE scheme calculates the average powers of the received signal and its delayed copy, and compares the result with $C(n)$. ML scheme is an optimization of MMSE taking the SNR into consideration. These two schemes are proved to be efficient in continuous transmission systems, but they increase false alarm probability and implementation complexity in burst OFDM communication systems [42].

5.2.2 MATLAB Simulation of MNC Scheme

The estimation of a frame start is determined to be at the sample index n when the timing metric $M(n)$ is maximized. $M(n)$ is normalized and is independent of the absolute received signal power. A threshold "*Thr*" related to SNR is set. A frame is considered to be detected if the following requirement is satisfied.

$$M(n)>Thr \tag{5.4}$$

Figure 5.1 plots M versus sample index n for the implementation of MNC scheme with $D = W = 64$ in the absence of noise. The timing offset is 100 samples. The overall response is in the range of 0 to 1 and the jump at the frame start is very clear. $M(n)$ is at a very low level before the start of a frame. Once the short training sequence is received, where sample index $n = 101$, $M(n)$ jumps to the maximum value of 1 quickly. This jump gives quite a good estimate of the frame starting time. The plateau of maximum value ends at $n = 133$. The length 32 ($= 160 - D - W$) of the plateau equals to the length of two STS periods. The result is analyzed in Figure 5.2. It is shown that the very first sample of the plateau coincides with the beginning of the short training sequence. The right time start is obtained. Hence, once the beginning of the plateau is detected, the time synchronization is achieved in short time. The second plateau in Figure 5.1 starts at $n = 261$, which is the first sample of the long training sequence. Since the LTS is periodic with period 64 and has a total length of 160, there would be a similar plateau of length 32.

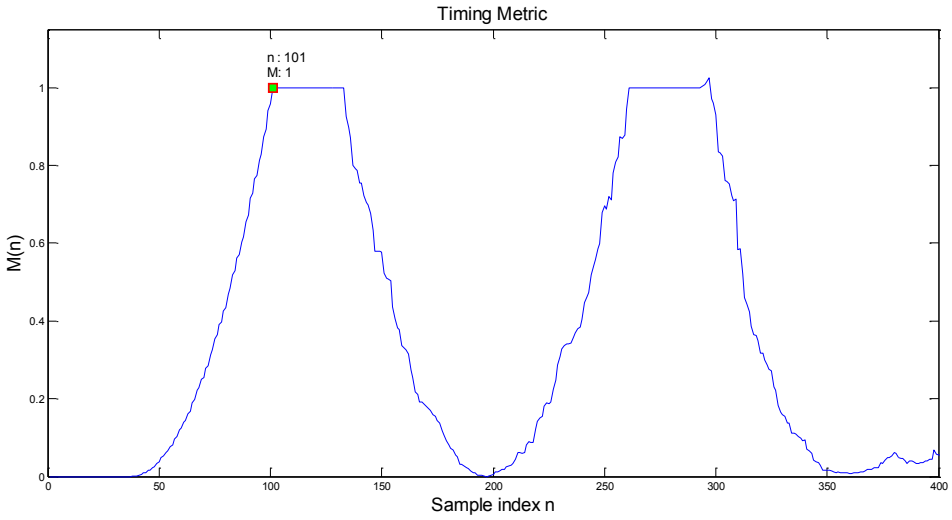


Figure 5.1 Timing metric of MNC scheme in noise-free transmission.

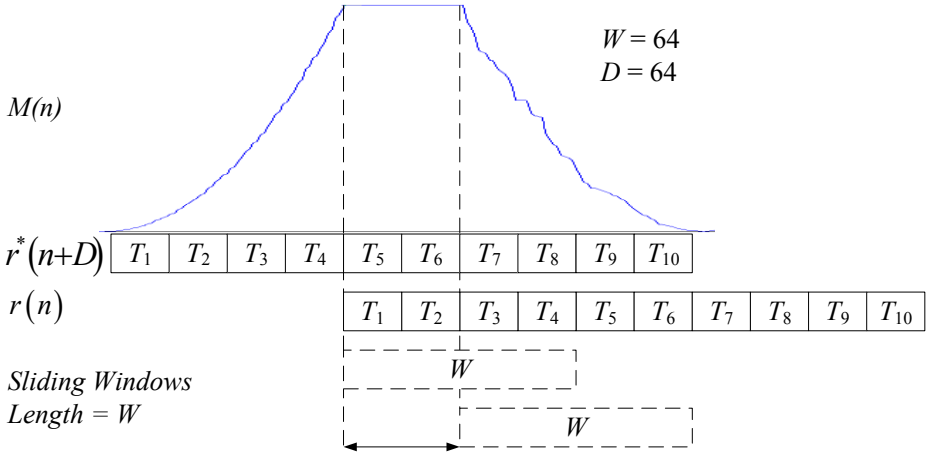


Figure 5.2 Delay and correlation of short training sequence with $W = 64$.

Figure 5.3 illustrates M versus n curves under an AWGN channel for E_b/N_0 = 10 dB and 0 dB. It is observed that the maximum value of M decreases with SNR. This is because with the increase of noise energy, correlation between the two halves of the short training sequence will decrease, but the received power will not. Hence the maximum M and the starting point of frame determined accordingly vary with SNR. By comparing the two curves, it is seen that at low SNR, the jump in the value of M is not as clear as that at high SNR. The side lobe is relatively high, which can easily cause false alarm.

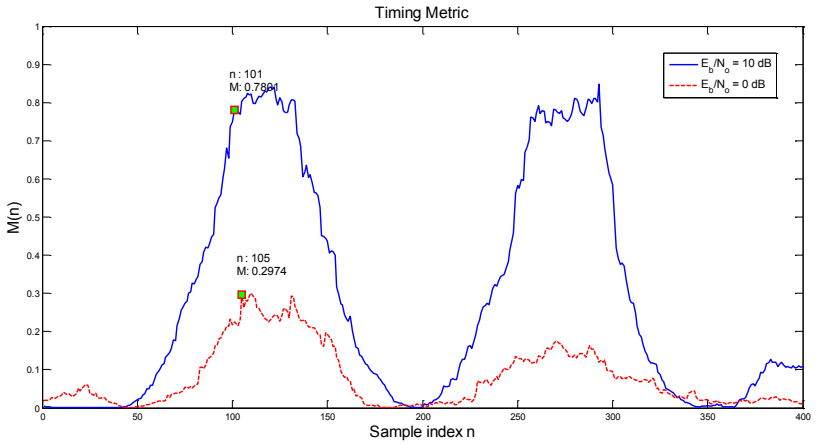


Figure 5.3 Timing metric of MNC scheme under AWGN channel.

Figure 5.4 depicts M versus n curves under a Rayleigh fading channel for E_b/N_0 = 10 dB and 0 dB. Similar to the simulation results of Figure 5.3, M decreases with SNR, and the probability of false detection increases significantly at low SNR. By comparing these two figures, it is observed that under multipath fading, the variation of M during the plateau period is larger than that under an AWGN channel. The detection accuracy is reduced. This performance degradation is extremely severe at low SNR. Therefore, the MNC scheme is not quite robust at low SNR.

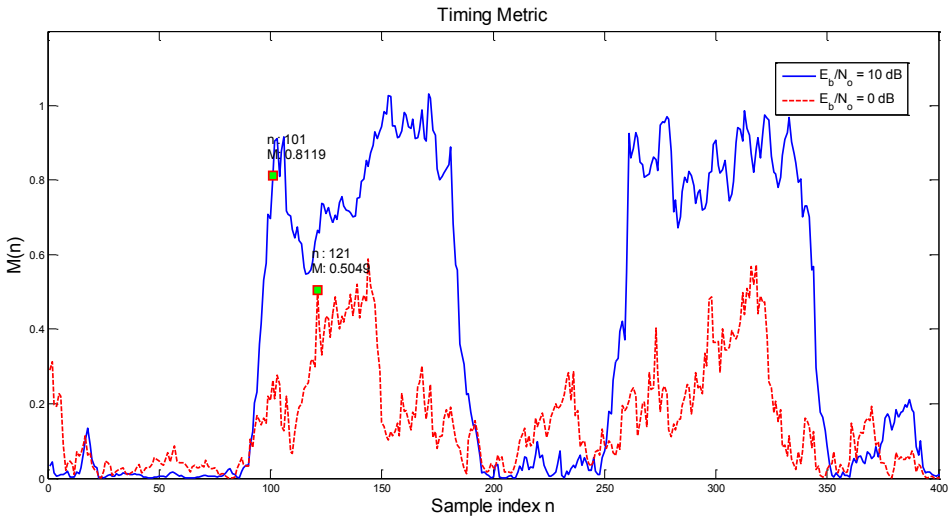


Figure 5.4 Timing metric of MNC scheme under Rayleigh fading channel.

5.3 FPGA Implementation of OFDM Synchronization

5.3.1 Block Diagram of OFDM Synchronization

To reduce the implementation complexity, the correlation function $C(n)$ and $P(n)$ could be calculated with an iterative form in a sliding window

$$C(n+1) = C(n) - r^*(n)r(n+D) + r^*(n+W)r(n+W+D) \tag{5.5}$$

$$P(n+1) = P(n) - \left|r(n+D)\right|^2 + \left|r(n+W+D)\right|^2 \tag{5.6}$$

As shown in Figure 5.5, we use a single stage cascaded integrator-comb (CIC) filter to implement the delay and accumulation operation in (5.5) and (5.6). For example, once the first $C(n)$ is computed, the following $C(n+1)$ could be implemented by adding the next cross-correlation term and subtracting the very first one. This process is performed iteratively.

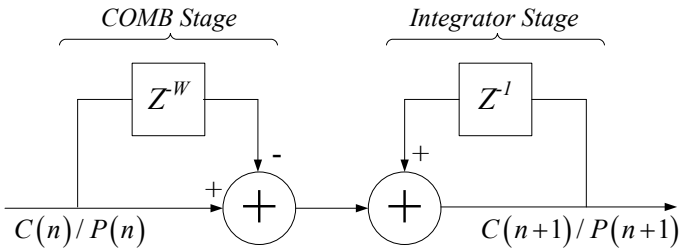


Figure 5.5 Implementation of delay and accumulator with CIC filter.

The implementation of division is very resource consuming. For implementation simplicity, we can convert the division operation to a multiplication and a threshold decision. The frame detection is achieved when

$$C(n)^2 > P(n)^2 \cdot Thr \tag{5.7}$$

where *Thr* is calculated according to the experimental simulation of $M(n)$. The choice of *Thr* is based on SNR and the expected BER. Figure 5.6 shows the block diagram of the OFDM synchronization, in which $D = W = 64$.

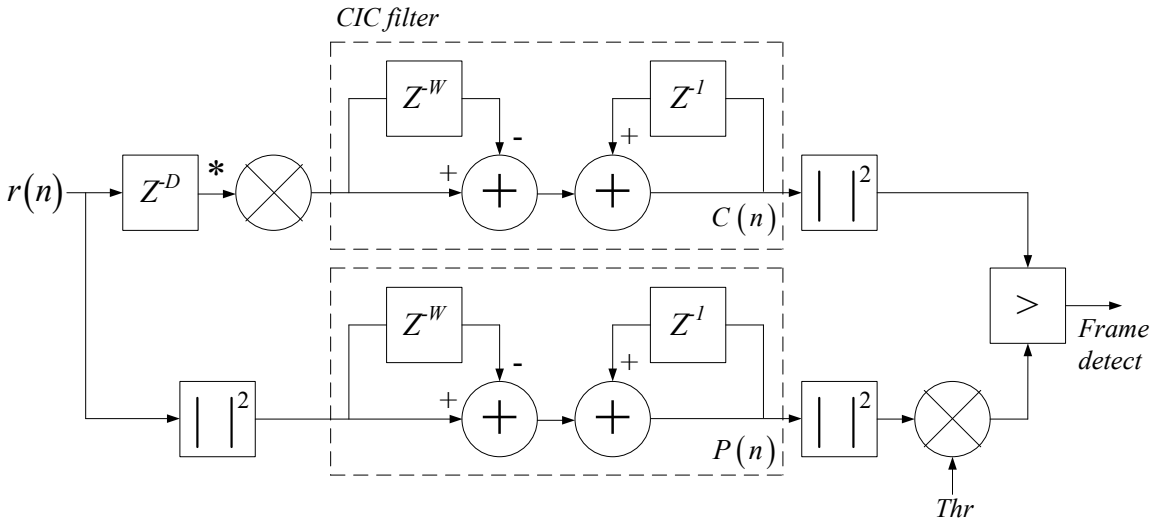


Figure 5.6 Block diagram of the OFDM synchronization.

5.3.2 Modeling of Synchronization Circuit

From the hardware module of CIC shown in Figure 5.7, the accumulator is realized by adding the new coming input to the current CIC output. This is accomplished by an "Assert" block. After accumulation, a subtractor is applied to get the final result. Figure 5.8 illustrates the FPGA implementation for the complete synchronizer. Three CIC filters are employed, in which the real and imaginary part of $C(n)$ are calculated in two branches.

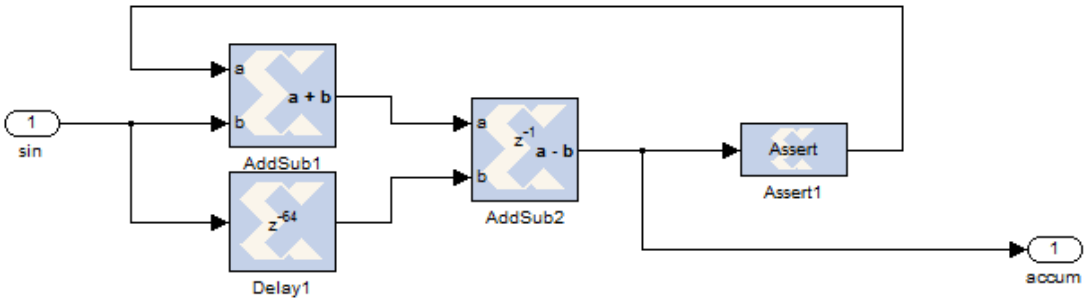


Figure 5.7 Implementation module of CIC filter.

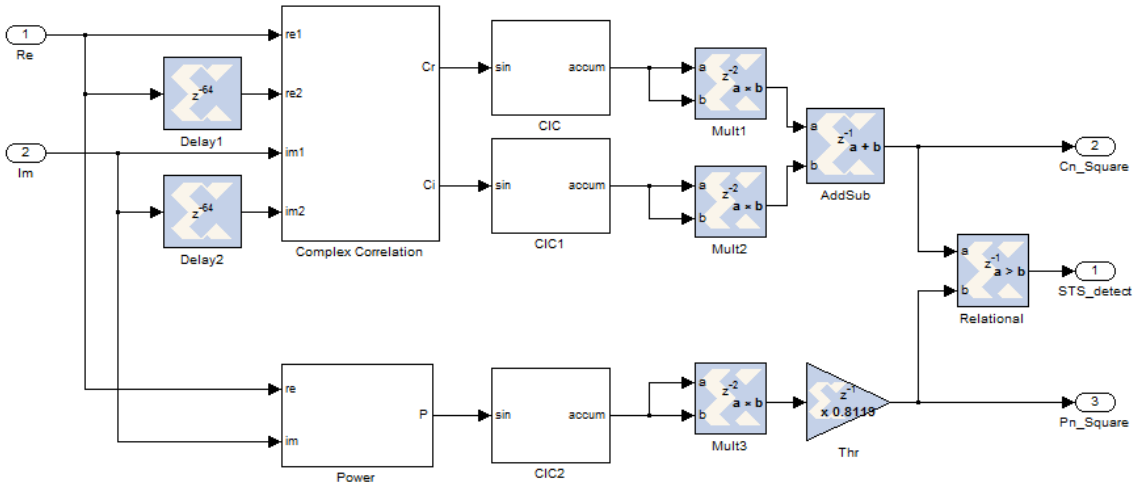


Figure 5.8 FPGA implementation of the OFDM synchronizer.

Figure 5.9 illustrates the output waveform for a noise-free environment. It is seen that $C(n)$ starts to build up from the fifth short symbol, and achieves the maximum value during the ninth symbol. On the other hand, $P(n)$ achieves stable state during the fifth symbol, which matches our theoretical analysis. The frame is detected when sample index n equals to 274, which is the last sample of the eighth short symbol. The first plateau ends at n = 340 and lasts for a period of 33-sample time. The second plateau is from n = 594 to n = 660, which is during the last 33 samples of the second LTS. Here each sample has a period of two clock cycles. There is a length difference of 1 sample for the plateau between MATLAB simulation and FPGA implementation. It is because for the FPGA implementation, the last sample of the eighth symbol has a period of two clock time, while in simulation, it is an instant point. In fact, for the implementation waveform, the sample difference at the plateau is also 32, which is the same as the simulation result.

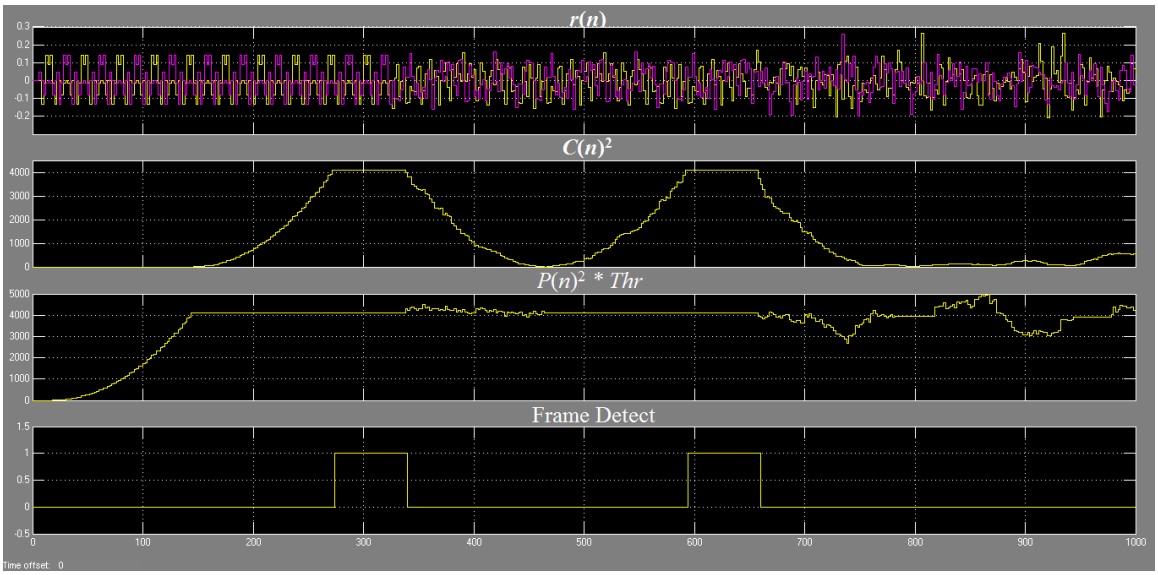


Figure 5.9 Waveforms of the synchronizer in noise-free transmission.

Figure 5.10 and Figure 5.11 show the implementation waveforms under the AWGN and Rayleigh channels, respectively, for E_b/N_0 = 10 dB. When noise occurs, the plateaus will no longer be flat but will fluctuate with time. *Thr* is obtained from the simulation results. It should be dynamically adjusted to meet various SNR environment and different BER requirements. The frame detection point shifts to n = 272 under AWGN channel and to n = 266 under Rayleigh fading. In addition, the plateau will not consist of exactly 33 samples. However, this detecting shift is tolerated. As seen from Figure 5.11, the received signal is affected by multipath fading other than additive noise. For the fading channel, *Thr* is selected relatively lower than that in AWGN channel.

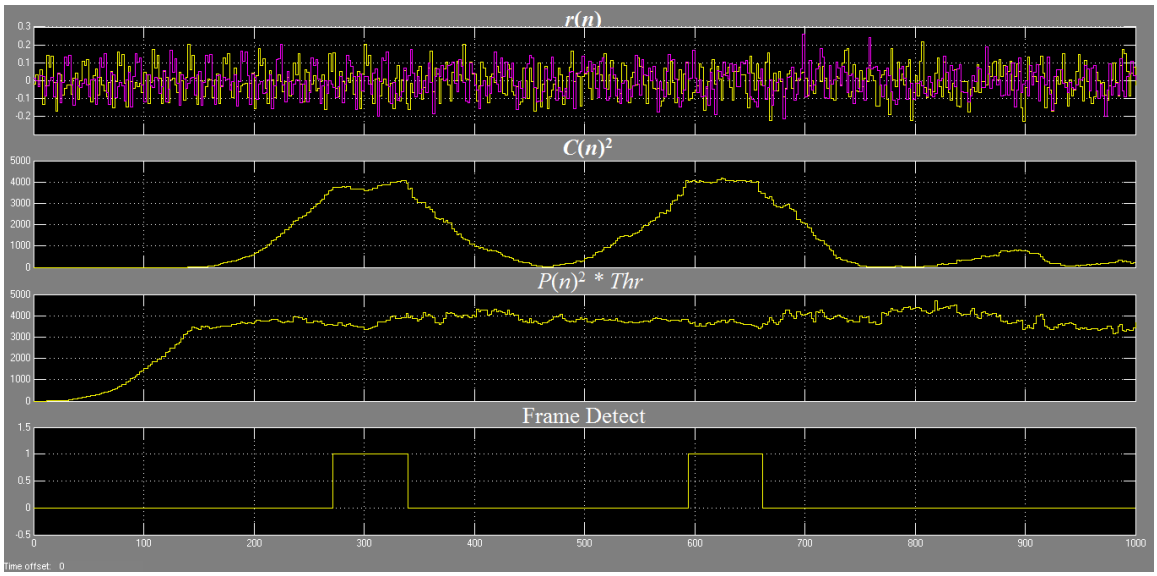


Figure 5.10 Waveforms of the synchronizer under AWGN channel at E_b/N_0 = 10 dB.

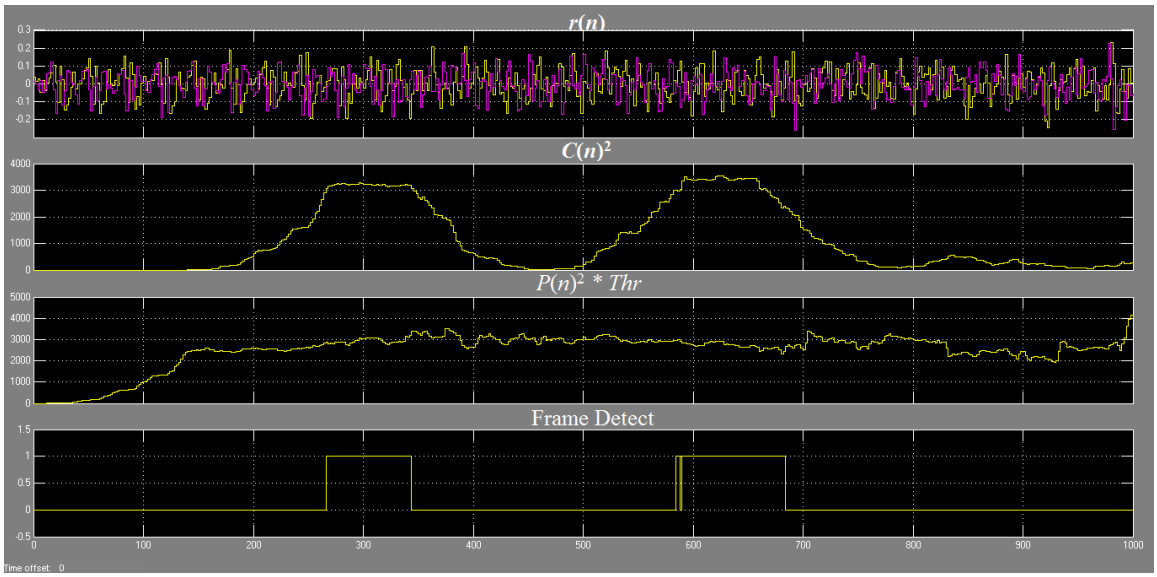


Figure 5.11 Waveforms of the synchronizer under Rayleigh channel at E_b/N_0 = 10 dB.

5.3.3 Implementation Results

After the system functionality is verified under AWGN and Rayleigh fading channels, it is converted to VHDL codes and then the synthesis and programming are performed. The system is programmed to Virtex-5 device. Table 5.1 and Table 5.2 summarize the area and timing results. With high pipelining design, the maximum delay of the clock net is also limited to around 2 ns. The latency is 5-sample time, which equals to 10 clock cycles. Each sample in time domain lasts for two clock cycles. The estimated maximum frequency is 133.905 MHz and is large enough to drive the synchronization circuit. Considering the resource consumption, the usage of I/O and DSP48E device are relatively high, since three accumulators are employed.

Table 5.1 Area results for synchronizer.

Resources	***Used***	***%***
Number of Slice Registers	570	1
Number of Slice LUTs	793	1
Number of bonded IOBs	107	16
Number of BUFG/BUFGCTRLs	1	3
Number of DSP48Es	18	28

Table 5.2 Clock and timing results for synchronizer.

Parameters	***Time Frequency***
Maximum Delay of Clock Net (ns)	2.064
Minimum Period (ns)	7.468
Maximum Frequency (MHz)	133.905
Maximum path delay from/to any node (ns)	7.468

5.4 Summary

In this chapter, the synchronization issue for OFDM system has been addressed. Several correlation-based algorithms have been introduced and the MNC scheme is implemented. The experimental results of MATLAB simulation have been used for circuit modeling. The system has been verified with hardware co-simulation, and has been implemented on Virtex-5 board. The implementation results have shown that the target device is capable of driving the proposed system.

Chapter 6 Conclusion and Future Work

6.1 Conclusion

This thesis has presented the theoretical analysis and simulation as well as FPGA implementation of a baseband OFDM system with channel estimation and timing synchronization. The OFDM system is prototyped based on IEEE 802.11a standard and transmits/receives signals on a 20 MHz bandwidth. With QPSK modulation scheme, the system achieves a throughput of 24 Mbps. Different kinds of transmitting channels have been studied and various estimation and equalization methods are compared. The traditional LS algorithm has been implemented. For the estimation of coarse timing, a modified MNC scheme has been investigated and realized. Starting from the study of OFDM principle, the system has been verified and realized with the help of both MATLAB simulation and hardware implementation. The design of the entire project has been carried out in a top-down approach, from the system design to functional blocks design. The present section gives a summary of the work contained in the thesis.

First, a baseband OFDM system with QPSK modulation is designed according to the system parameters in the IEEE 802.11a standard. The floating-point model is established in MATLAB. The simulation results of the BER performance under AWGN channel match very well with the theoretical waterfall curves. The probability of bit error for the OFDM system is same as that for a single carrier QPSK system.

Then, the fixed-point model with major signals represented by 16-bit number is created in XSG and Simulink. The simulation results with hardware in loop clearly

showed that the BER performance under an AWGN channel matches that in MATLAB simulation, except for a slight degradation in the case of high SNR, which can be attributed to insufficient precision. Then, the transmitter and receiver were separately implemented on Xilinx Virtex-5 board and targeted to XC5VLX110T device using Xilinx ISE 12.1 tool. Detailed analysis of the delay in the transmitter and in the receiver have been studied for understanding the system latency. It is concluded that in the transmitter, it takes 15.6 μs or 624 clock cycles to finalize the generation of the first data symbol and this can be performed within the 16-μs period of preamble transmission. On the other hand, the receiver requires 16 μs or 640 clock cycles to completely demodulate the first OFDM symbol. At the expense of large latency, the system reaches a high throughput. The timing reports generated from Xilinx Timing Analyzer tool indicates no timing conflict in the design.

After that, we present the design and FPGA implementation of the channel estimation and equalization subsystem for indoor wireless communication environment. The LS estimator is presented and realized under the proposed 802.11 channel, which is modeled by 3-tap FIR filter with each tap described by Rayleigh distribution. By comparing the BER results of the fixed-point model with that of the floating-point model, it is observed that there is a BER degradation for large SNR due to the accuracy loss when representing the weak noise in fixed bits. On the other hand, compared to that without estimation, the receiver with LS estimator improves system performance greatly at the expense of increasing the resource consumption of slice registers, slice LUTs, bonded IOBs and BlockRAMs by almost 1%. The usage of DSP48Es is increased by three times that used without channel estimation.

Finally, the synchronization circuitry is implemented. Several correlation-based algorithms are introduced and the MNC scheme which utilizes the cross-correlation of preamble symbols is designed. Experimental results under various communication environments are obtained based on the fixed-point model. The implementation result showed the maximum clock delay is about 2 ns. Since the driving clock of the system is 40 MHz, no conflicts exists in the designed OFDM system and the synthesized circuitry meets the timing constraints.

6.2 Future Work

This thesis offers a detailed design and implementation process of an FPGA-based system, and is an initial work on the implementation part for an industrial project,. It can be used as a basis for the future projects towards applications in advanced wireless communication systems. The work could be extended further to the following aspects:

- Some of the modules that can improve the system performance are not considered in this thesis. As part of further work, we could establish and implement a more practical system. Other modules may consist of pulse shaping, channel coding, interleaving, fine time and frequency synchronization and so on. Since the system is designed with a top-down approach, the implementation of other modules will not affect the modules already implemented in the system or will only cause minor modifications to their interfaces.

- This thesis presents the system design at baseband. With the help of DAC and ADC adapters, it could be extended to RF band and tested on a real wireless channel.

- Algorithm and system optimization could be performed to reduce the hardware resource consumption.

- The designed FPGA system can be realized for other environments according to different standards. Moreover, in order to further improve the system performance under fading channels, other techniques such as MIMO could be combined with OFDM, and more complex and innovative estimation and synchronization schemes could be used. Because of the top-down design approach, the platform can be upgraded with a simple modification to the system. For example, if the system is prototyped towards IEEE 802.11ac, the mapping block could be modified by using ROMs to realize up to 256 QAM. The usage of Xilinx IP core enables an easy upgrade from 64-point FFT to 512-point FFT. Larger bandwidth and in turn higher data rate could be achieved by redefining the sample period during modeling. Some other modules could be added to realize the space timing coding and advanced estimation and synchronization techniques.

Bibliography

[1] L. Hanzo, Y. Akhtman, L. Wang, and M. Jiang, "MIMO-OFDM for LTE, Wi-Fi and WiMAX: Coherent versus Non-coherent and Cooperative Turbo-transceivers," John Wiley & Sons Ltd, United Kingdom, 2011.

[2] R. V. Nee and R. Prasad, "OFDM for Wireless Multimedia Communications," Artech House, Boston, 2000.

[3] F. Xiong, "Digital Modulation Techniques," 2nd Edition, Artech House, Boston, 2006.

[4] R. W. Chang, "Synthesis of band-limited orthogonal signals for multichannel data transmission," Bell System Technical Journal, vol. 45, pp. 1775–1796, December 1966.

[5] B. R. Saltzberg, "Performance of an efficient parallel data transmission system," IEEE Transactions on Communications, vol. 15, pp. 805–811, December 1967.

[6] R. W. Chang and R. A. Gibby, "A theoretical study of performance of an orthogonal multiplexing data transmission scheme," IEEE Transactions on Communications, vol. 16, pp. 529–540, August 1968.

[7] R. W. Chang, "Orthogonal Frequency Division Multiplexing," U.S. Patent 3,488,445, filed November 14, 1966, issued January 6, 1970.

[8] S. B. Weinstein and P. M. Ebert, "Data transmission by frequency-division multiplexing using the discrete Fourier transform," IEEE Transactions on Communications, vol. 19, pp. 628–634, October 1971.

[9] B. Hirosaki, “An orthogonally multiplexed QAM system using the discrete Fourier transform,” IEEE Transactions on Communications, vol. 29, pp. 982–989, July 1981.

[10] B. Hirosaki, S. Hasegawa, and A. Sabato, “Advanced groupband data modem using orthogonally multiplexed QAM technique,” IEEE Transactions on Communications, vol. 34, pp. 587–592, June 1986.

[11] L. J. Cimini Jr, “Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing,” IEEE Transactions on Communications, vol. 33, pp. 665–675, July 1985.

[12] M. Alard and R. Lassalle, “Principles of modulation and channel coding for digital broadcasting for mobile receivers,” EBU Technical Review, pp. 168–190, August 1987.

[13] “IEEE Wireless LAN Medium Access Control (MAC) and Physical Layer Specification,” IEEE Standard 802.11a-1999.

[14] European Telecommunications Standards Institute, “Radio Equipment and Systems (RES); HIgh PErformance Radio Local Area Network (HIPERLAN) Type 1; Functional specification,” ETSI ETS 300 652 ed.1, October 1996.

[15] M. Cummings and S. Haruyama, “FPGA in the Software Radio,” IEEE Communications Magazine, vol. 37, no. 2, pp. 108-112, February 1999.

[16] Xilinx Inc., “System Generator for DSP,” http://www.xilinx.com/support/documentation/dt_sysgendsp_sysgen12-1.htm.

[17] Altera Inc., “DSP Builder,” http://www.altera.com/products/software/products/dsp/dsp-builder.html.

[18] Altera Inc., “FPGA vs. DSP Design Reliability and Maintenance White Paper,” http://www.altera.com/literature/wp/wp-01023.pdf.

[19] C. Dick and F. Harris, “FPGA implementation of an OFDM PHY,” Proceedings of the 37th Asilomar Conference on Signals, Systems and Computers, vol. 1, pp. 905–909, Nov. 2003.

[20] J. Garcia and R. Cumplido, “On the Design of an FPGA-Based OFDM Modulator for IEEE 802.11a,” 2nd International Conference on Electrical and Electronics Engineering (ICEEE) and XI Conference on Electrical Engineering (CIE) Sep. 2005, pp. 114-117.

[21] T. Vladimirova and J. R. Paul, “Implementation of an IEEE802.11a Transmitter Module for a Reconfigurable System-on-a-Chip Design,” 2009 NASA/ESA Conference on Adaptive Hardware and Systems, pp. 305-312, 2009.

[22] M. J. Canet, J. Valls, V. Almenar, J. Marin-Roig, “FPGA implementation of an OFDM-based WLAN receiver,” Microprocessors and Microsystems, vol. 36, issue 3, pp. 232–244, May 2012.

[23] F. Manavi and Y. R. Shayan, “Implementation of OFDM Modem for the Physical Layer of IEEE 802.11a Standard Based on Xilinx Virtex-II FPGA,” in 59th IEEE Vehicular Technology Conference (VTC 2004) 2004, pp. 1768-1772.

[24] K. Sobaihi, A. Hammoudeh, and D. Scammell, “FPGA Implementation of OFDM Transceiver for a 60GHz Wireless Mobile Radio System,” International Conference on Reconfigurable Computing and FPGAs, pp. 185–189, December 2010.

[25] L. Gopal, D. W. S Tze and N. Z. Ishak, “Design of FPGA-Based OFDM Transceiver for DVB-T Standard,” in IEEE Computer Science and Automation Engineering (CSAE), 10-12 June 2011, pp. 193-197.

[26] P. Dong, “Design and FPGA Implementation of a SISO and a MIMO Wireless System for Software Defined Radio,” MS-Thesis, Concordia University, 119 pages, March 2009.

[27] F. Manavi, “Implementation of OFDM Modem for the Physical Layer of IEEE 802.11a Standard Based on Xilinx Virtex-II FPGA,” MS-Thesis, Concordia University, 97 pages, Feburuary 2004.

[28] T. S. Rappaport, “Wireless Communications: Principles and Practice,” 2nd edition, Prentice Hall, Upper Saddle River, NJ, 2002.

[29] B. Sklar, “Digital Communications: Fundamentals and Applications,” 2nd Edition, Prentice Hall, NJ, 2001.

[30] http://en.wikipedia.org/wiki/IEEE_802.11.

[31] D. Tse and P. Viswanath, “Fundamentals of Wireless Communication,” Cambridge University Press, 2005.

[32] C. Iskander, “A MATLAB-based Object-Oriented Approach to Multipath Fading Channel Simulation,” a MATLAB Central submission available in: www.mathworks.com.

[33] C. Tepedelenlioglu, A. Abdi, G. Giannakis, and M. Kaveh, "Estimation of Doppler Spread and Signal Strength in Mobile Communications with Applications to Handoff and Adaptive Transmission," Wireless Communications and Mobile Computing, vol. 1, no. 2, pp. 221-242, 2001.

[34] B. Han et al., “An Iterative Joint Channel Estimation and Symbol Detection Algorithm Applied in OFDM System with High Data To Pilot Power Ratio,” Proc. IEEE Int’l. Conf. Commun., vol. 3, Anchorage, AK, May 2003, pp. 2076–80.

[35] M. K. Ozdemir and H. Arslan, “Channel estimation for wireless OFDM systems,” IEEE Commun. Surveys & Tutorials, vol. 9, no. 2, pp. 18-48, Second Quarter 2007.

[36] Tufvesson, F., and T. Maseng, ‘‘Pilot assisted channel estimation for OFDM in mobile cellular systems,’’ Proc. 47th IEEE Veh. Technol. Conf., Phoenix, Arizona, May 1997, pp. 1639–1643.

[37] H. Yuan, Y. Ling, H. Sun, and W. Chen, “Research on channel estimation for OFDM receiver based on IEEE 802.11a,” 6th IEEE International Conference on Industrial Informatics, INDIN 2008., pp. 35–39, July, 2008.

[38] J. G. Proakis, “Digital Communications,” 4th edition, New York: McGraw-Hill, 2000.

[39] T.Saidani, D.Dia, W.Elhamzi, M.Atri, and R.Tourki, “Hardware Co-simulation for Video Processing Using Xilinx System Generator,” Proceedings of the World Congress on Engineering, vol.1, July 2009.

[40] T. M. Schmidl and D. C. Cox, “Robust Frequency and Timing Synchronization for OFDM”, IEEE Transactions on Communication, vol. 45, no. 12, pp. 1613-1621, December 1997.

[41] T. Keller and L. Hanzo, “Orthogonal Frequency Division Multiplex Synchronization Techniques for Wireless Local Area Networks”, IEEE PIMRC, pp. 963- 967, Taipei, Taiwan, 1996.

[42] L. D. Kabulepa, A. G. Ortiz and M. Glesner, “Design of an Efficient OFDM Burst Synchronization Scheme”, IEEE International Symposium on Circuits and Systems, (ISCAS 2002), vol. 3, Scottsdale, AZ, pp. III-449 - III-452, 26-29 May 2002.

[43] P. R. Chevillat, D. Maiwald, and G. Ungerboek, “Rapid Training of a Voiceband Data-Modem Receiver Employing an Equalizer with Fractional-T Spaced Coefficients”, IEEE Transactions on Communication., vol. 35, no. 9, pp. 869-876,1987.

[44] M. Sandell, J.-J. van de Beek, and P. O. Börjesson, “Timing and Frequency Synchronization in OFDM Systems Using the Cyclic Prefix”, Proc. International Symposium on Synchronization, pp. 16-19, Essen, Germany, 1995.

[45] Y. S. Cho, J. Kim, W. Y. Yang and C. G. Kang, “MIMO-OFDM Wireless Communications with MATLAB”, John Wiley & Sons (Asia) Pte Ltd., Singapore, 2010.

[46] C. H. Dick and H. M. Pedersen, “Design and Implementation of High-Performance FPGA Signal Processing Datapaths for Software Defined Radios”, Embedded Systems Conference Apr. 2001, pp. 1-16.

[47] Sinem Coleri, Mustafa Ergen, Anuj Puri, and Ahmad Bahai, “Channel Estimation Techniques Based on Pilot Arrangement in OFDM Systems,” IEEE TRANSACTIONS ON BROADCASTING, VOL. 48, NO. 3, SEPTEMBER 2002.

[48] Meng-Han Hsieh and Che-Ho Wei, “Channel Estimation for OFDM Systems Based on Comb-Type Pilot Arrangement in Frequency Selective Fading Channels,” January 13, 1998.

[49] Xilinx Inc., “Product Specification of Xilinx DS260 LogiCORE IP Fast Fourier Transform v7.1,” http://www.xilinx.com/support/documentation/ip_documentation/xfft_ds260.pdf.

[50] Xilinx Inc,. ISE 12.1 User Guides, http://www.xilinx.com/support/documentation/dt_ise12-1_userguides.htm.

[51] Xilinx Inc, “Put Hardware in the Loop with Xilinx System Generator for DSP,” http://cdserv1.wbut.ac.in/81-312-0257-7/xilinx/files/Xcell%20Journal%20Articles/xcell_47/xc_sysgen47.pdf.