UNIVERSITY of BRADFORD Library

University of Bradford eThesis

This thesis is hosted in Bradford Scholars – The University of Bradford Open Access repository. Visit the repository for full metadata or to contact the repository team

© University of Bradford. This work is licenced for reuse under a Creative Commons Licence.

THE REALIZATION OF SIGNAL PROCESSING METHODS AND THEIR HARDWARE IMPLEMENTATION OVER MULTI-CARRIER MODULATION USING FPGA TECHNOLOGY

Hassan Saleh Okleh Migdadi

PHD

2015

THE REALIZATION OF SIGNAL PROCESSING METHODS AND THEIR HARDWARE IMPLEMENTATION OVER MULTI-CARRIER MODULATION USING FPGA TECHNOLOGY

Validation and implementation of multi-carrier modulation on FPGA, and signal processing of the channel estimation techniques and filter bank architectures for DWT using HDL coding for mobile and wireless applications

Hassan Saleh Okleh Migdadi
B.Sc., M.Sc.

Submitted for the Degree of

Doctor of Philosophy

Faculty of Engineering and Informatics

University of Bradford

2015

Abstract

THE REALIZATION OF SIGNAL PROCESSING METHODS AND THEIR HARDWARE IMPLEMENTATION OVER MULTI-CARRIER MODULATION USING FPGA TECHNOLOGY

Validation and implementation of multi-carrier modulation on FPGA, and signal processing of the channel estimation techniques and filter bank architectures for DWT using HDL coding for mobile and wireless applications

Keywords

Field-Programmable-Gate-Arrays (FPGAs); Orthogonal Frequency Division Multiplexing (OFDM); Decoded Pseudo Pilot-Data Aided (FDPP-DA); ECG Patient Monitoring System (ECG-PMS).

First part of this thesis presents the design, validation, and implementation of an Orthogonal Frequency Division Multiplexing (OFDM) transmitter and receiver on a Cyclone II FPGA chip using DSP builder and Quartus II high level design tools. The resources in terms of logical elements (LE) including combinational functions and logic registers allocated by the model have been investigated and addressed. The result shows that implementing the basic OFDM transceiver allocates about 14% (equivalent to 6% at transmitter and 8% at receiver) of the available LE resources on an Altera Cyclone II EP2C35F672C6 FPGA chip, largely taken up by the FFT, IFFT and soft decision encoder.

Secondly, a new wavelet-based OFDM system based on FDPP-DA based channel estimation is proposed as a reliable ECG Patient Monitoring System, a Personal Wireless telemedicine application. The system performance for different wavelet mothers has been investigated. The effects of AWGN and multipath Rayleigh fading channels have also been studied in the analysis. The performances of FDPP-DA and HDPP-DA-based channel estimations are compared based on both DFT-based OFDM and wavelet-based OFDM systems. The system model was studied using MATLAB software in which the average BER was addressed for randomized data. The main error differences that reflect the quality of the received ECG signals between the reconstructed and original ECG signals are established.

Finally a DA-based architecture for 1-D iDWT/DWT based on an OFDM model is implemented for an ECG-PMS wireless telemedicine application. In the portable wireless body transmitter unit at the patient site, a fully Serial-DA-based scheme for iDWT is realized to support higher hardware utilization and lower power consumption; whereas a fully Parallel-DA-based scheme for DWT is applied at the base unit of the hospital site to support a higher throughput. It should be noted that the behavioural level of HDL models of the proposed system was developed and implemented to confirm its correctness in simulation. Then, after the simulation process the design models were synthesised and implemented for the target FPGA to confirm their validation.

Acknowledgment

First and foremost, I would like to thank ALLAH (God) for giving me the Opportunity to expand my knowledge and horizons. I am also grateful to ALLAH (God) for the successful completion of my PhD report. I would also like to thank my parents, my wife, and sons and daughters for their inspiration, without them none of these would even have been possible. They have always been my biggest fans and I appreciate that.

Special thanks and gratitude to my academic advisers **Prof R A Abd-Alhameed**and**Dr J M Noras**, for giving me the opportunity to conduct the research work under their supervision. Their scientific wisdom and experience have helped me in achieving my research objectives. I will never forget the many opportunities that they gave me in facilities, publications and teaching. They were like old brothers to me so I pray to ALLAH (God) for their more success and achievements to both of them in this life. Last but not least, my deepest gratitude goes to **Dr S M R Jones** for his valuable advice, help, and support.

My thanks are also extended to my home supervisors **Prof Kasim Al-Aubaidy** from the Faculty of Engineering, Philadelphia University, Jordan; and **Dr Esam A. Qaralleh**, Computer Engineering Department, King Abdullah II Faculty of Engineering - Princess Sumaya University for Technology, Jordan; for their support and all the other people in the research laboratories of Bradford University, especially **Mr HuthifaObidat** and **Dr FauziElmegri**.

Table of Contents

Acronyms

3D-IC	Three Dimensional Integrated Circuit
AOA	Angle of arrival
ASIC	Application Specific Integrated Circuit
BER	Bit Error Rate
CAD	Computer Aided Design
CE	Channel Estimation
CLB	Configurable Logic Block
CP	Cyclic Prefix
CPLD	Complex Programmable Logic Device
CTS	Child Tracking system
DA	DataAided (in chapter 5)
DA	Distributed Arithmetic(in chapter 6)
DFT	Discrete Fourier Transform
DSP	Digital Signal Processing
DWT	Discrete Wavelet Transform
E_b/N_0	BitEnergytoNoiseDensity
ECG-PMS	Electrocardiography Patient Monitoring System
EDIF	Electronic Design Interchange Format
FDPP	FEC (Forward Error Correction coding) Decoded Pseudo Pilot
FDM	Frequency Division Multiplexing

FEC	Forward Error Correction Encoder
FF	FlipFlop
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FPD	Field Programmable Device
FPGA	Field Programmable Gate Array
FPLA	Field Programmable Logic Array
GI	Guard Interval
GPS	Global Positioning System
HCPLD	High Capacity Programmable Logic Device
HDL	Hardware Description Language
HDPP	Hard Decision Pseudo Pilot
IC	Integrated Circuit
ICI	InterCarrier Interference
IDFT	Inverse Discrete Fourier Transform
iDWT	Inverse Discrete Wavelet Transform
IFFT	Inverse Fast Fourier Transform
IIR	Infinite Impulse Response
IP	Intellectual Property
ISI	InterSymbol Interference
LB	Logic Block
LE	Logic Element
LLR	Logarithm of Likelihood Ratio

LSI	Large Scale Integration
LUT	LookUp Table
MCM	Multicarrier Modulation
MIMO	Multiple Input Multiple Output
MSI	Medium Scale Integration
NRE	Non Recurring Engineering
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	OFDM Access
OTP	One Time Programmed
PAL	Programmable Array Logic
PDA	Parallel Distributed Arithmetic
PLA	Programmable Logic Array
PLD	Programmable Logic Device
PLTS	Patient Location Tracking System
PROM	Programmable Read Only Memory
RP	ReProgrammed
RSS	Received signal strength
SDA	Serial Distributed Arithmetic
SISO	Single Input Single Output
SNR	Signal to Noise Ratio
SoC or SOC	System Ona Chip
SPLD	Simple Programmable Logic Device
SSI	Small Scale Integration

TDOA	Time Different Of Arrival
ULSI	Ultra Large Scale Integration
VLSI	Very Large Scale Integration
WBAN	Wireless Body Area Network
WPT	Wavelet Packet Transform
WSI	Wafer Scale Integration
WSN	Wireless Sensors Network

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Nowadays, highbitrate and high quality service are highly required for many applications in wireless communication area. The Orthogonal Frequency Division Multiplexing (OFDM) scheme has been proposed to tackle these issues and it has been applied widely in wireless communication systems. The OFDM transmission scheme provides robust protection against multipath fading and reduces interference caused by delay spread in wireless mobile environment whilemaintaining low-complexity receiver. OFDM also offershigh spectrum efficiency through the provisionof high date rate transmission capacity and high bandwidth efficiency [1][2][3].

Employing Fast Fourier Transform (FFT) based OFDM in a communications system inevitably yields a reduction in its spectral efficiency caused by guard interval insertion. To remedy this, a Discrete Wavelet Transform (DWT) based OFDM has been proposed. Further, DWT allows signals to overlap in both time and frequency domains, thereby mitigating the detrimental effects of Inter Carrier Interference (ICI) and Inter Symbol Interferences (ISI)[4][5]. Nonetheless, channel estimation is still essentially required in wireless environment due to a numberof

factors such as multi-path fading, noise, timing offset, and frequency offset. A channel estimation can be achieved by non-data aided (blind) and data-aided (training)channel estimators. In data-aided (DA), a dynamic channel estimation is realised through sending pilots periodically,yielding reductionin the transmission efficiency. To overcome this, an improved DA algorithm was developed by generating a reference data from the received OFDM symbols[2][5].

Wireless telemedicine hasbecome a hot research field in recent years. Due to the aforementioned wireless channel limitations, an OFDM system has been proposed to be used in such applications to maintain a reliable transmission and reception of biomedical signals[6][7].

There is a large amount of reported work and published papers in the field of OFDMbased information theory and algorithms. However, most of this work has been carried out on using the theoretical approaches and /or simulator and there has been a limited amount of work that focused the attention on the actual practical hardware implementation side of these algorithms in real time systems [8].

Microprocessors, Application Specific Integrated Circuits (ASIC), and Field-Programmable-Gate-Arrays (FPGA) are the three main methodsthat are used to implement digital domain communication systems [9]. Over the last decade, the pre-fabricated and re-programmable silicon devices commonly known as FPGA

have evolved to become one of the key digital circuit implementation as they resolved various past challenges in hardware implementation. FPGA is increasingly used nowadays in a wide range of markets including consumer electronics and communications where these markets need different requirements in terms of power efficiency, performance, and cost [10][11][12].

1.2 Aims and Objectives

Currently, a high quality service and a high bit rate are essential for many modern applications of wireless communication and wireless telemedicine applications are not exception. OFDM system has been used to meet these requirements. However, OFDM counteracts multipath fadingand mitigatesits effects, though it does not eliminatesuch effects completely. Thus, channel estimation is essential.For areal time system, more attention must be exerted on the hardware implementation aspect. Consequently, this study is aimed at proposing and investigating a new OFDM system that can efficiently support one of the wireless telemedicine applications requirements, namely, the Electrocardiography Patient Monitoring System (ECG-PMS)presented. To this end, the following objectives are defined:

I. Extensive review of the FPGA technology.
II. DevelopingHDL (VHDL, Verilog) coding skills for FPGA development.
III. Understanding the MATLAB environment and how to programme in MATLAB.
IV. A Review of the signal processing of OFDM system.

V. Simulatingthe convolutional OFDM system using MATLAB software, then implementand configure the OFDM system onto the FPGA chip.

VI. Compiling an overview of the ECG Patient Monitoring System (ECG-PMS) for wireless telemedicine application, and its requirements.

VII. A review on the channel estimation algorithms.

VIII. Designing an OFDM system that meets the ECG-PMS requirements.

IX. Simulatingthe proposed OFDM system using MALTAB,evaluating its viability in different channel scenarios, comparing it with the convolutional OFDM system and evaluating the proposed system performance for different wavelet families

X. Designing an iDWT/DWT system that supports the ECG-PMS requirements, Implementing and configurationit on the target FPGA chip.

1.3 Thesis Contribution

This main contributions of this research work can be summarised as follows:

I. A fully functionalOFDM transmitter and receiver implemented using target FPGA chip has been developed.

II. A DWT-based OFDM with FDPP-DA channel estimation for ECG-PMS system having a significantly high performance quality at hospital site and withstanding a low transmit signal power at patient site (power efficiency enhancement) has been developed.

III. Distributed Arithmetic-based (DA) architecturefor 1-D iDWT/DWT system implemented by means of a FPGA chip for effectively supporting the ECG-PMS wireless telemedicine application requirements has been developed.

1.4 Organization of the Thesis

Chapter 2: Digital hardware using FPGAs is introduced in this chapter. After programmable logic devices (PLD) evolution is viewed, the architecture of FPGA is described and then FPGA are compared with Application-Specific-Integrated-Circuit (ASIC). The process from design idea to hardware implementation is described in FPGA's Computer Aided Design (CAD). Chapter is ended with an overview of the available commercial FPGA.

Chapter 3: An introduction to OFDM is presented in this chapter. After the advantages of multi-carrier over the single-carrier are explained, the main components of OFDM system using FFT algorithm is described and the idea behind using guard-interval with cyclic-prefix is defined and explained how it mitigate the influence of Inter-Symbol Interference (ISI) and Inter-Carrier Interference (ICI).

Chapter4: an FPGA based OFDM implementation is presented in this chapter. Each component of OFDM system is designed using a high-level design software tools providing by Altera and Math-Work, validated, and implemented on Cyclone II FPGA chip. The allocated resources of the available on the target chip are

addressed and investigated. The process is repeated for OFDM transmitter and receiver.

Chapter5: in this chapter an ECG Patient Monitoring System (ECG-PMS) for wireless telemedicine applications is presented. An OFDM system is proposed to meet the requirements for the ECG-PMS system, where the requirements at patient site are differ than at hospital site. The OFDM system is designs based on wavelet transform and using FEC (Forward Error Correction coding) Decoded Pseudo Pilot (FDPP) for channel estimation. The proposed system was evaluated using MATLAB software in terms of BER performance and the quality of reconstructed ECG signal.

Chapter6: in this chapter, the proposed implementation for iDWT/DWT system used in OFDM-DWT system for using in the ECG-PMS system is designed using Verilog HDL using ALTERA Quartus II Software (V.13) Environment. Then it validated and implemented on ALTERA EP4CE115F29C7 Cyclone IV FPGA chip on DE2-115 board. The allocated resources on the selected FPGA chip, propagation delay, and the data rate are addressed and compared among wavelet families.

Chapter 7: in this chapter, a conclusions of our research have been summarized and a future works have been suggestions.

Chapter 2

Digital Hardware Implementation(FPGA Considerations)

2.1 Introduction

Over the last decade, Field Programmable Gate Arrays (FPGAs) have become one of the key digital hardware circuit implementation media and has been used in several consumer markets. FPGAs has been used in a wide range of devices targeted at achieving different performance, power efficiency and cost requirements. In addition, FPGAs are pre-fabricated silicon chips that can be electrically programmed to become any designed digital system, using built-in logical elements and programmable connections and can be re-configured ondemand. The exclusion of physical fabrication make marketing designs using FPGA easier than other hardware implementation methods, like ASIC [10][12].

This chapter presents an introduction to digital hardware using FPGAs. It defines the relevant terminology in this field and presentson overview of the evolution of programmable logic devices (PLD). Furthermore, the key elements of the internal structure and architecture of FPGA are discussed. Then a comparison between Application-Specific-Integrated-Circuit (ASIC) and FPGA is performed. The

dynamics of a Computer Aided Design (CAD) for FPGA is described, which incorporates the process needed to go from design idea to actual hardware configuration. Finally, commercially available FPGAs are overviewed.

2.2 Terminology Definitions

This section describes the terminology used in this chapter [13]:

Field-Programmable-Device (FPD): describes an integrated circuit (IC) used in an implementation digital hardware, which allows the end user to realize the desired design by configuring the chip. FPD is also referred to as Programmable-logic-devices (PLDs).

Programmable-Logic-Array (PLA): is a small FPD that consists of two logical levels; an AND-plane array followed by an OR-plane array, where both are programmable.

Programmable-Array-Logic (PAL): is a special type of PLA, where the programmable AND-plane array is followed by a non-programmable (fixed) OR-plane array.

Simple-Programmable-Logic-Device (SPLD): is a simple PLD, which is usually either PLA or PAL.

Complex-Programmable-Logic-Device (CPLD): is a more complex PLD, where a multiple SPLD-like blocks are arranged on a single chip. Enhanced-PLD (EPLD), Super-PAL, Mega-PAL are all alternative names for CPLD.

Field-Programmable-Gate-Array (FPGA): is a FPD that allows a very high logic capacity. A FPGA offers narrow logic resources with a high ratio of flip-flops.

High-Capacity-Programmable-Logic-Devices (HCPLDs): is types CPLDs and FPGAs in a single acronym.

Inter-connect: refers to the wiring resources of an FPD.

Programmable-Switch: is a switch used to connect an inter-connect wire to a logic element or to another inter-connect wire, where the switch programming is done by the end user.

Logic-Block (LB): a small circuit block, which is replicated and put in an array form, then the array is put inside an FPD. To implement a desired circuit in an FPD, it is decomposed into simple-sub-circuits, where each one can be mapped into one logic-block in an FPD.

Logic-Capacity: refers to “the amount of digital logic that can be mapped into a single FPD”.

Logic-Density: is the number of logic per unit area in a single FPD.

Speed-Performance: refers to “the maximum operable speed of the circuit when implemented in an FPD”, where it is set by the longest delay through any path and by the maximum clock frequency for the combinational and sequential circuits needed to function the circuit correctly respectively.

2.3 Programmable Logic Device’s Evolution

Programmable-Read-Only-Memory (PROM) was the first Programmable-Logic-Device (PLD) that was introduced in 1970. The address lines and data lines for PROM is used as logic circuit inputs and outputs respectively [13][14]. PROM consists of a fixed AND-plane level followed by a programmable OR-plane level,

represented in Figure 2.1. Figure 2.1 shows a 3X3 PROM that consists of eight AND gates and three OR gates. In general, for an N inputs PROM, a 2^N AND gates are needed for 2^N possible minterms, while logic functions mostly require a few product terms. Therefore, PROMs represents an inefficient architecture for realizing logic circuits and are rarely used in practice [13][14].

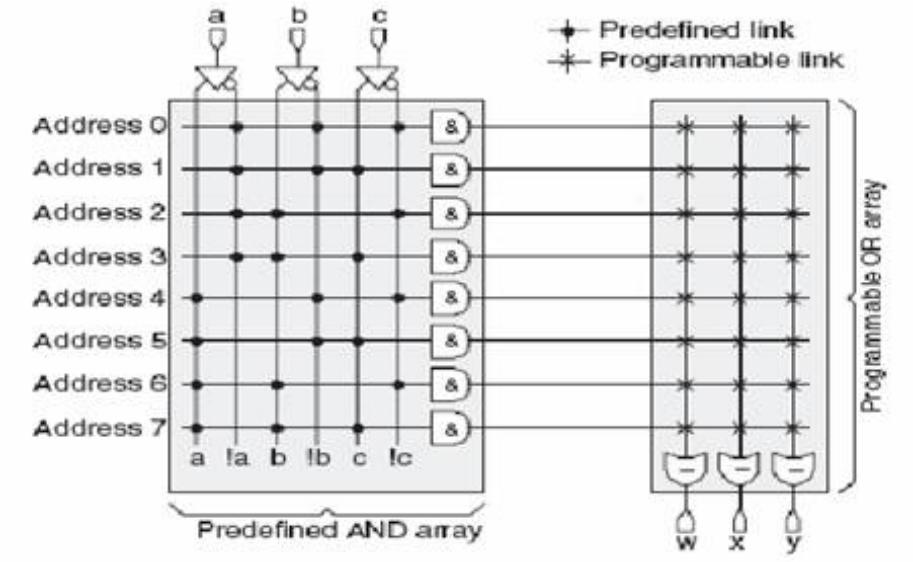


Figure 2-1 3×3 Programmable-Read-Only-Memory PROM

The Field-Programmable-Logic-Array (FPLA) device was introduced in 1995 and aimed at addressing PROM limitations by making both AND-plane and OR-plane arrays programmable. FPLA is also called PLA for short purpose [13][15] As shown in Figure 2.2, inputs or their complements of PLA are fed into the AND-plane array to produce the logical-product-term of any combination of these inputs. Then, the outputs of AND-plane array are fed into OR-plane array to produce the logical-sum-term of any combination of the outputs of AND-plane array. Hence logical functions in sum-of-products form are implemented by PLA [13].

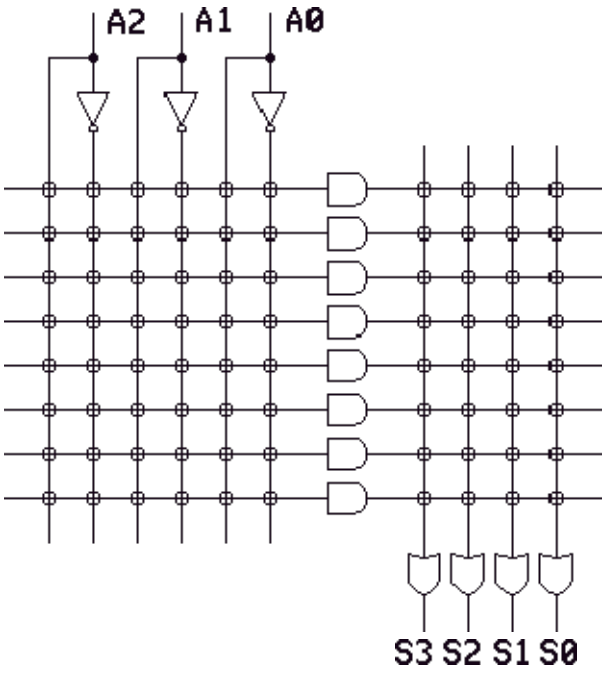


Figure 2-2 Programmable Logic Array (PLA)

From the example in [16], it is evident that the number of AND gates is independent of the number of PLA inputs and the number of OR gates is independent of both the number of PLA inputs and the number of AND-plane array outputs [15]. The two levels of configurable/programmable logic in PLA increase the manufacturing difficulty and caused propagation delays, which introduced an increase in manufacturing cost and a decrease in speed-performance respectively. The high cost and low speed are considered the main drawbacks of the PLA [13].

In late 1970s, Programmable-Array-Logic (PAL) devices were developed to overcome the disadvantages of PLA [13][17]. As illustrated in Figure 2.3, the structure of a PAL is opposite to the structure of a PROM. A PAL consists of a programmable AND-plane array followed by a fixed OR-plane array, while PROM

consists of a fixed AND-plane array followed by a programmable OR-plane array[17]. Because of this featureonly a single level of programmability feature is available in PAL, hence making PAL's devices faster than PLA's [13][17]. Additionally, because ofthe programmability in the AND-plane array, the number of AND gates is independent of the number of PAL's inputs. The shortcoming of generality caused by fixed-OR-plane array has been addressed by producing numerous variants of PALs with various numbers of inputs and outputs, and varies sizes of OR-gate [13]. To realize sequential circuits, Flip-Flops (FFs) are connected to the outputs of OR-gate as shown in Figure2.4 [13].

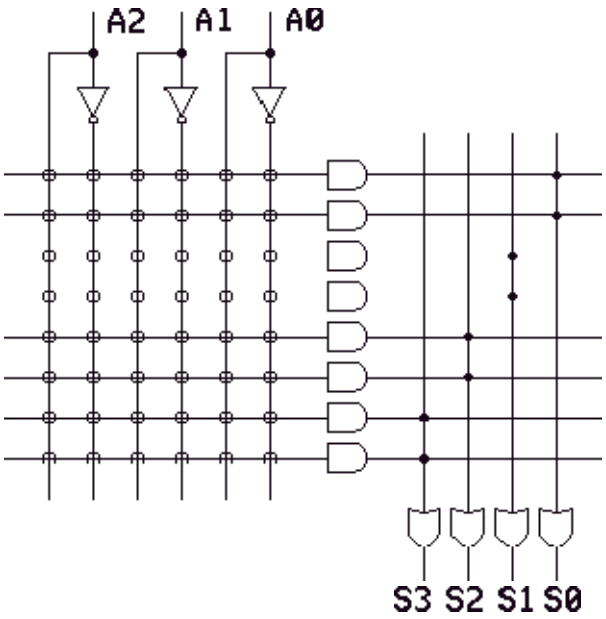


Figure 2-3 Programmable-Array-Logic (PAL)

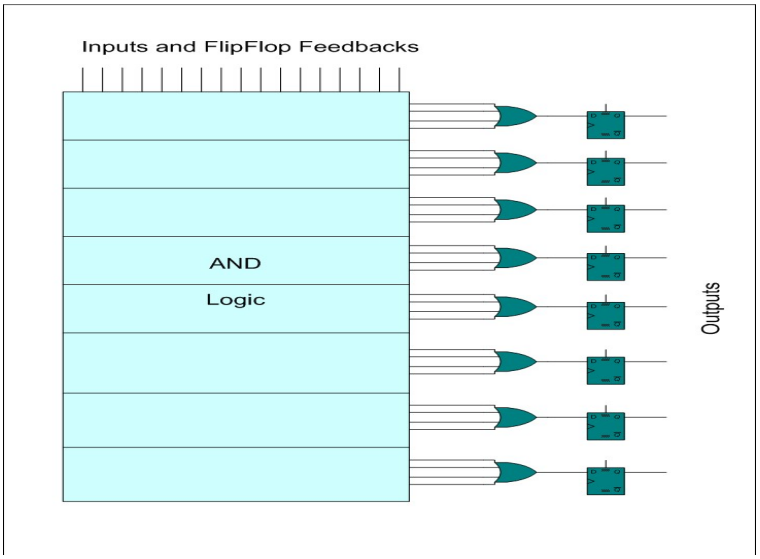


Figure 2-4 Programmable-Array-Logic (PAL) with Flip-Flops (FFs)

PLAs, PALs, and other small PLDs devices are grouped into a category called Simple-Programmable-Logic-Devices (SPLDs), which are considered to have high speed-performance and low cost [13][18].

As the number of SPLD's inputs is increased, the size of the programmable-logic-plane-arrays is grown quickly. This strict SPLD architecture led to consider SPLD as an inefficient architecture for high capacity devices [13].

Complex- Programmable-Logic-Devices (CPLDs) were proposed to address SPLDs limitations by integrating multiple SPLD blocks inside one chip with programmable interconnections between them [13][19].

The delay through one CPLD is usually short and predictable because it has little flexibility in its internal architecture. Predictable timing characteristics feature make

CPLD ideal for applications require critical control, high-speed-performance control, and high-performance-logic [18]. CPLD are inexpensive and require small amount of power, which make CPLD ideal for battery-operated portable and cost-sensitive applications [18].

A different approach called Field-Programmable-Gate-Array (FPGA) can be used to build FPDs with high logic capacity. FPGAs consists of a two dimensional matrix of uncommitted programmable-logic-blocks (depending upon vendor, these blocks are called different things), a programmable routing matrix of interconnections between the programmable-logical-blocks and Input/output blocks [13][18][20]. Figure 2.5 below shows a Typical FPGA structure [18].

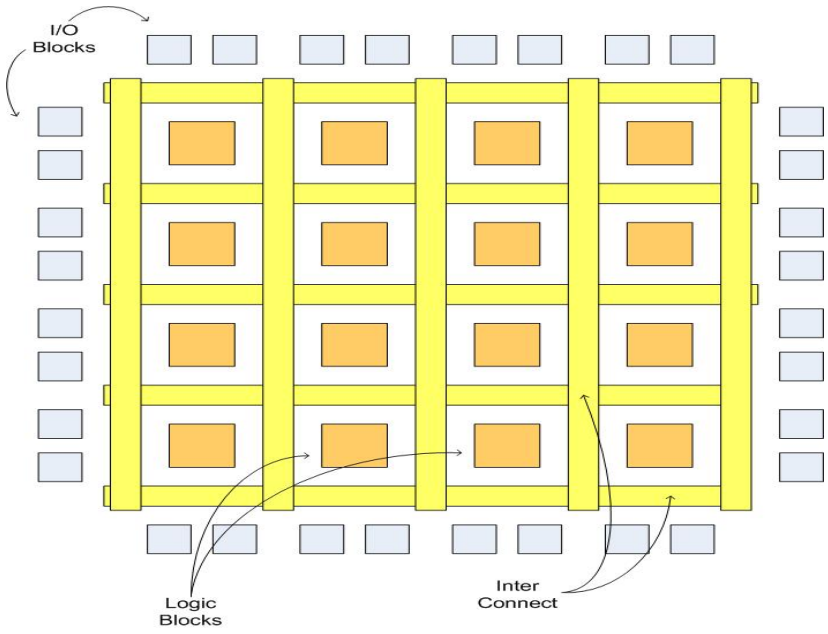


Figure 2-5 FPGA Structure

In addition, FPGA is the only type of FPD that support very high-logic-capacity and only in FPGA the end user performs the FPGA configuration through programming.

These main distinction features between FGPA and other types of FPD result in significant shift in the way of designing digital circuits [13].

2.4 FPGA Architecture

Because FPGAs combine the programmability idea of PLDs and uncommitted gate arrays in ASIC architecture, FPGAs are considered the most powerful programmable-logic-devices [20].

The basic structure of FPGAs is array-based. The most common FPGA architecture consists of a two dimensional matrix of programmable-logic-blocks (called Configurable Logic Block (CLB) by Xilinx, and Logic Element (LE) by Altera for example), a two dimensional matrix of programmable-routing-channels and I/O blocks as shown in Figure 2.5 above. The programmable-logic-blocks can be interconnected via the horizontal and vertical programmable-routing-channels[21].

2.4.1 Programmable-Logic-Blocks

Commonly, the programmable-logic-block consist of function generators, storage elements, and dedicated arithmetic logic circuits. A function generator can be configured as a Look-Up-Table (LUT), while a storage element as edge-triggered Flip-Flop (FF) or as level-triggered latches[21][22].

An n-by-m LUT is a (2^n X m) bit wide memory array where the n-address lines fed into the memory are the LUT input, and m-data lines output from the memory are the LUT output. Thus, a LUT with n inputs can encode and realize any Boolean function of its n inputs by modelling the Boolean function as a truth table and then programing that truth table directly into the memory. For example, one LUT with 4 inputs can implement any Boolean function of up to 4 inputs. In the modern FPGA devices, LUTs with 4-6 input bits are considered as the significant element in the device[13][22][23].

After 1985, Xilinx offered a new FPGA family called XC4000 series. Figure 2.6 below shows the basic CLB used in XC4000 series. CLB consists of three LUTs, two D-FFs, and other logic components for arithmetic circuit. The two LUTs to the right of the figure are fed by the CLB inputs, while the third one are used to combine the other two LUTs when needed. This structure provides the ability to implement any Boolean functions of up to eight inputs in normal mode or two different Boolean functions each one up to four inputs in arithmetic mode where the selection of the mode is programmed into the other logic components [13][21].

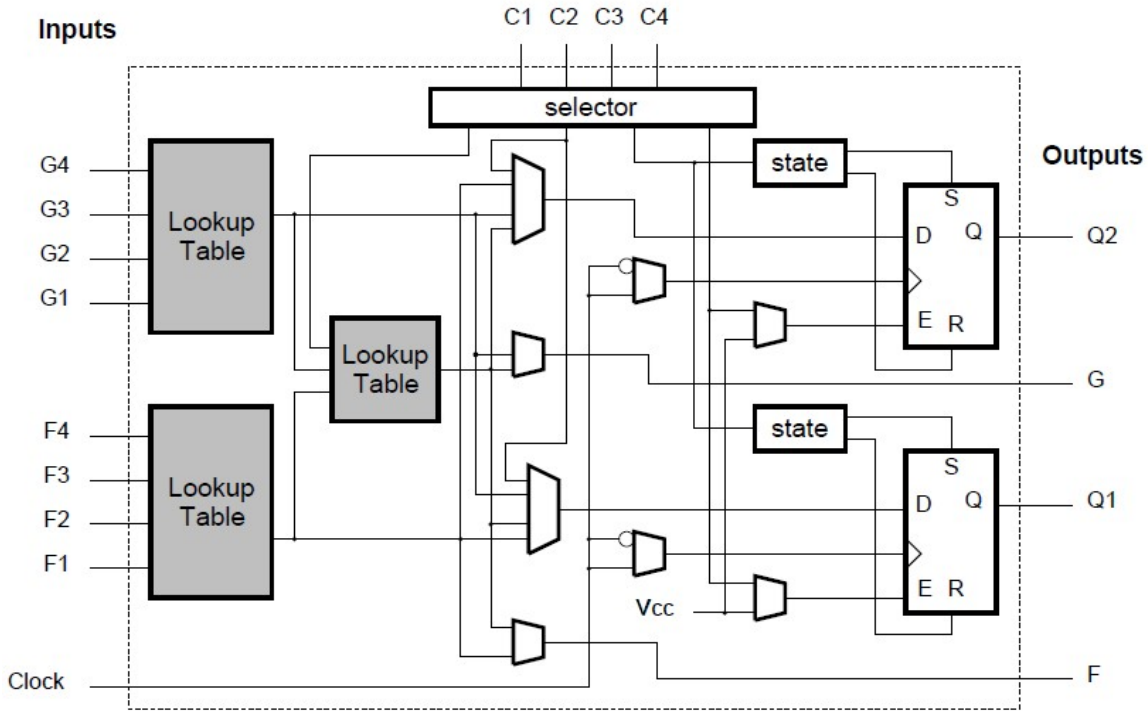


Figure 2-6 Xilinx XC4000 Configurable Logic Block (CLB)

Altera offers another FPGA family called FLEX 8000 series. Figure 2.7 below shows the basic LE used in FLEX 8000 series. LE consists of one 4-input LUT, one SR-FF, and other logic components for arithmetic circuit. The cascade circuit are used to implement a wide AND functions efficiently [13].

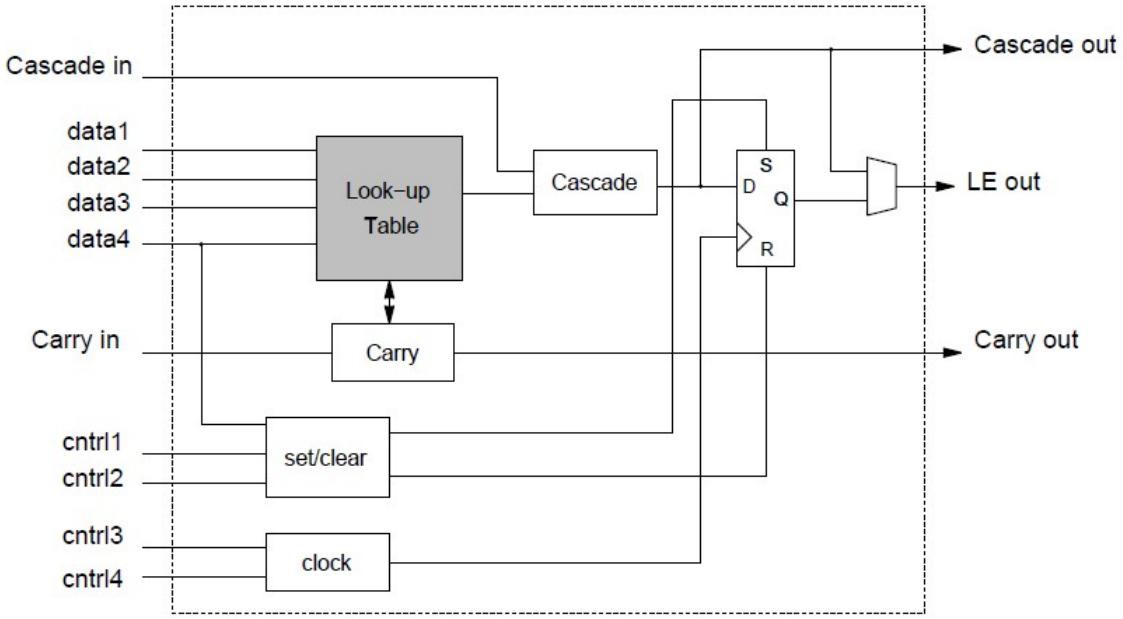


Figure 2-7 Altera FLEX 8000 Logic Element (LE)

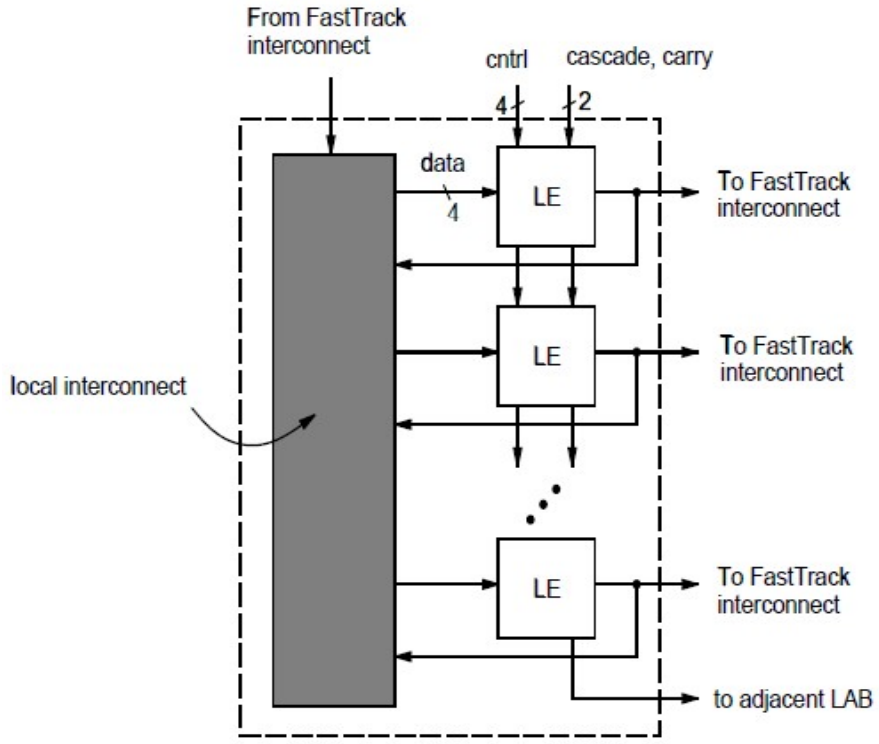


Figure 2-8 Altera FLEX 8000 Logic Array Block (LAB)

In FLEX 8000, Altera grouped each eight LEs in one set called Logic-Array-Block (LAB) as shown in Figure 2.8. Inside each LAB, there is a local interconnect used to connect any LE to any other LE within the same LAB [13].

2.4.2 Programmable-Routing-Channels

To connect logical blocks with each other, a two dimensional matrix of programmable routing channels is used where the structure of this interconnect is considered a significant feature in the FPGA device. Each channel consists of a number of wire-segments, and each wire-segment consists of a number of connection lines. Different types of connection lines are used in FPGAs, for example, to connect distant logical blocks a long connection lines is used, while a short connection lines is used in the case of neighbouring blocks, where for reset all FFs inside the FPGA a reset connection lines is used. Then one of the user-programmable switch technologies is used to connect the wire-segments to the inputs/outputs of the Logical-Blocks, or the wire-segments together [13][24].

Figure 2.9 below shows one of the horizontal-routing-channels, where the vertical-routing-channels, Inputs/Outputs of CLBs, and routing switches are not shown in this figure. The speed-performance of the configured circuit depends on how the CAD tools allocate the wire-segments for each signal. This is because when the signal travels from one CLB to another it needs to pass through a number

programmable-switches, where the particulate setting of wire-segments used determine how many switches crossed [13].

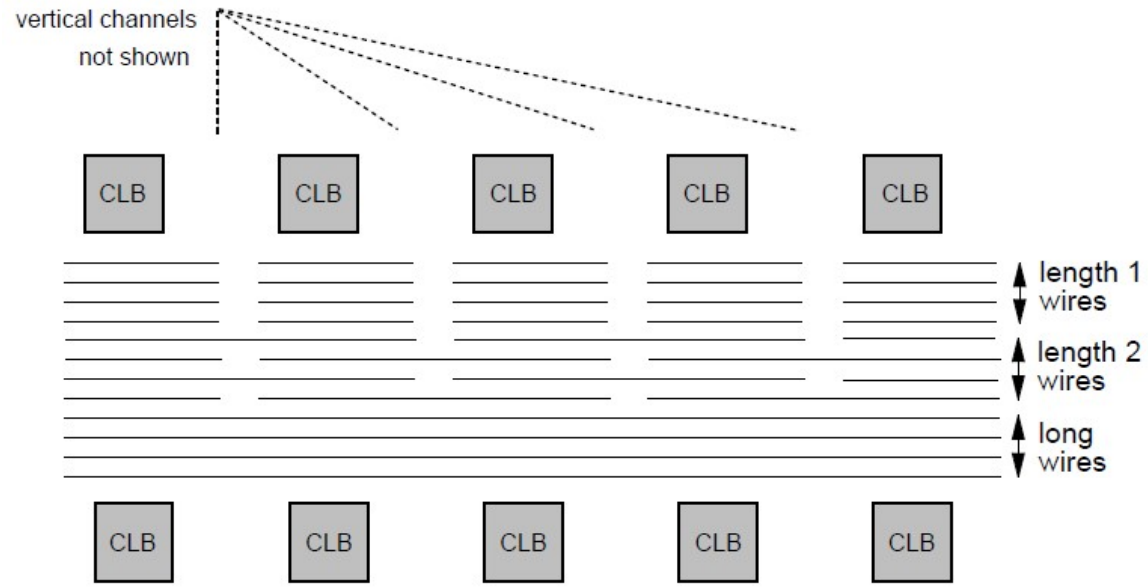


Figure 2-9 Xilinx XC4000 Wire Segments

In the Altera FLEX 8000 series, the short connection lines are replaced by using local interconnect inside the LABs, thus only long connection lines extending the total height and width of the FPGA device are needed. These lines are placed in horizontal and vertical tracks called Fast-Tracks as shown in Figure 2.10. Fast-Tracks contains only long connection lines have two advantages. First one, number of user-programmable switches are fewer than in the FPGAs that use a number of short wire-segments and therefore the interconnect delays are more Predictable in the FLEX 8000 series. Second advantage, CAD tools can easily implement the desired circuit [13].

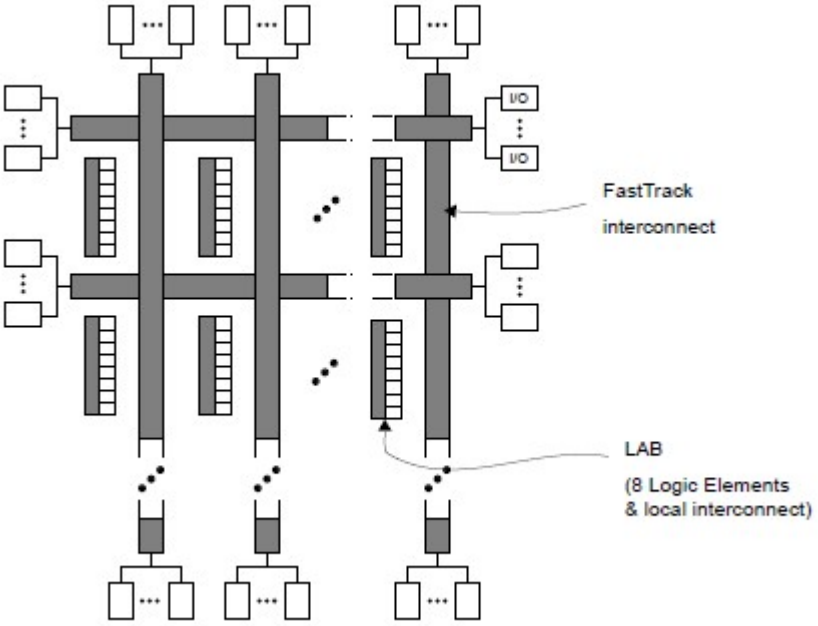


Figure 2-10Architecture of Altera FLEX 8000 FPGAs

There are many approaches used to implement a user-programmable switch. For example, the user-programmable switch used for CPLDs is the EPROM or EEPROM, while for FPGAs is the antifuse or SRAM. However, the inability of using EPROM/EEPROM in FPGAs could not be explained technically [13].

2.4.3 Other FPGA resources

Contemporary FPGAs contain of other resources in addition to Programmable-Logic-Blocks and Programmable-Routing-Channels such as RAM blocks to implement in-chip data storage, clock management blocks to synthesize numerous clock signals, DSP modules to implement blocks used in digital signal processing algorithms such as digital filters, high-speed serial transceivers to support a high-speed I/O, and a hardware-CPU or hardware-PowerPC cores [25].

2.4.4 FPGA configuration memory

The common programming technologies are shown in Table 2.1 below. From most left, first column names the programing technology, second column describes if the programmable device is a One-Time-Programmed (OTP) or a Re-Programmed (RP), third column indicates the need to configure the device after each power-up(volatile) or not (non-volatile), and the last column lists the name of the transistor technology used in the relative programming technology [13][26][27][28].

Table 2-1 summary of Programming Technologies

Name	OTP or RP	Volatile	Technology
Fuse	OTP	No	Bipolar
EPROM	RP, out of circuit	No	UVCOMS
EEPROM	RP, in circuit	No	EECOMS
Antifuse	OTP	No	COMS
SRAM	RP, in circuit	Yes	COMS+
Flash	RP	No	COMS+

Upon configuration memory, FPGAs can be categorised into the following:

a) antifuse-based FPGAs where Actel is one of the leading manufactures for this type. However, the main weakness of these FPGAs is that it can be programmed only once, they are the least susceptible to radiation effect [13][26].

b) SRAM-based FPGAs. For SRAM-based products, Altera and Xilinx are the leading manufactures and offers challenging products. This type of FPGAs can be

re-programmed, but because It is a volatile device, it needs to re-configure after each power-up [13][26].

c) SRAM-based FPGAs with integrated Flash. The configuration data are stored in the internal Flash memory inside the FPGA which is used to configure the SRAM during power-up. Therefore, there is no need for external memory [13][26].

d) Flash-based FPGAs. The main manufacture for this type of FPGA is Actel. Power consumption of this type is much less than of the normal SRAM-based FPGAs [13][26].

In the last two decades, these devices have been changed and improved dramatically and Altera and Xilinx offer many new FPGA families with better performance, power efficiency and cost. The architecture of the Cyclone II FPGA chip used in this project is explained in [29].

2.5 FPGA, Microprocessor, and ASIC

2.5.1 Introduction of IC, ASIC, and FPGA

“An integrated circuit (IC) or monolithic integrated circuit (also referred to as IC, chip, or microchip) is an electronic circuit manufactured by the patterned diffusion of trace elements into the surface of a thin substrate of semiconductor material.

Additional materials are deposited and patterned to form interconnections between semiconductor devices." [30].

IC can be classified into three generations according to some factors such as the number of transistors. In first generation small-scale-integration (SSI) was developed then medium-scale-integration (MSI), and finally large-scale-integration (LSI). Then very-large-scale-integration (VLSI) was developed in the second generation. In the last generation ultra-large-scale-integration (ULSI), wafer-scale - integration (WSI), system-on-a-chip (SoC or SOC), and three-dimensional-integrated-circuit (3D-IC) was developed [30].

A special kind of IC are FPGA and Application-Specific-Integrated-Circuit (ASIC) where IC have been developed for general-propose use, while FPGA and ASIC made-to-order for a particular use. An ASIC contains huge number of logical gates just as a FPGA does. Also modern ASICs include other large hard building blocks such as processor blocks, memory blocks (like flash, ROM, and RAM) and other blocks just like FPGAs. Hardware-Description-Language (HDL), like Verilog or VHDL is used to configure the desired circuit on ASIC by engineer designer the same as on FPGAs [21][31].

Both ASIC and FPGA are ICs designed to be configured by the engineer designer after manufacturing using HDL. However, FPGA contains re-configurable logical blocks and interconnects that provide the ability to all/partial re-configuration and

update the functionality of the all/portion of the design circuit on the FPGA device after shipping unlike ASIC. The re-configuration feature is considered the main different feature between FPGA and ASIC [21][31].

2.5.2 FPGA and ASIC comparison

This section presents the differences between FPGAs and ASICs in terms of logic density, power consumption, circuit speed, non-recurring engineering (NRE), time to market, cost (i.e. Area), and re-configuration to enable engineer designers to make better informed choices between FPGA and ASIC media [21][32].

Historically, compared to FPGAs, ASICs offer many advantages including reduced in silicon area, reduced in power consumption, and increased in performance. However, these advantages come at the cost of an increase in non-recurring engineering, an increase in time to market, and inability to re-programme in the field for debugging when designs are implemented on ASICs [21][32].

On the other hand, modern FPGAs contain large hard building blocks like multiplier/accumulator and memory blocks which significantly increase the energy efficiency, decrees the critical path delay and the area. Thus the gap between FPGA and ASIC in term of logic density, power consumption, circuit speed is reduced [21][32].

Also, when the ASIC designed functions become part of the manufacture's Intellectual Property (IP) and can re-used, they are incorporated into their FPGAs to reduce the overall cost of development, manufacturing, and test of the FPGAs and hence the board space, power, and money are saved which leads to a more reduction in the gap between FPGA and ASIC [33].

To conclude, FPGA makers succeed in their seeking to improve FPGAs which is achieved specially by designing a hardware of mixed heterogeneous blocks such as multiplexers, accumulators, multipliers, and memories which are less programmable but more efficient where most modern FPGAs use them [32].

2.6 Computer-Aided-Design (CAD) flow for FPGA

It is essential to make use of Computer-Aided-Design (CAD) programs and tools when implementing desired circuits in FPDs such as SPLDs, CPLDs, and FPGAs [13]. A general typical CAD flow for implementing the designed circuits in FPDs is shown in the Figure 2.11 below [34].

Certain algorithms are used to optimize the desired circuit by the end user manually without using CAD system. This is because mostly the initial logic entry is not in an optimization form. Because this task is done manually by the end user without using CAD system, it is not included in the Figure 2.11 above [13].

A general typical CAD system includes software for the following tasks:

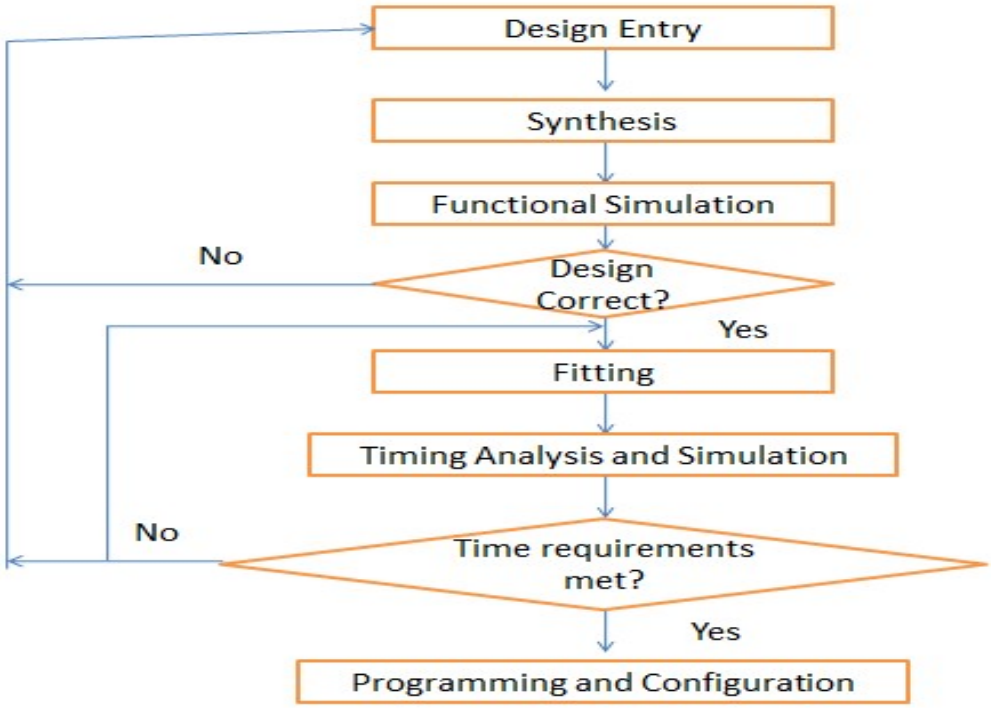


Figure 2-11 CAD flow

2.6.1 Design Entry

The optimized desired circuits are carried out by more than one method. It can be carried out by creating a schematic diagram using graphical CAD programme when we have simple designs. Or it can be carried out by describing the optimized desired circuits in hardware description languages like VHDL and Verilog using text based system for more complex designs [13][34][35].

2.6.2 Synthesis

After design entry, the HDL (VHDL/Verilog) code is converted by the synthesiser to produce the intermediate representation of the hardware design called netlist, where the contents of netlist –which is independent on the specifics of the FPGA- are stored in a standard format named Electronic Design Interchange Format (EDIF). Then the design logic is implemented in devices resources such as logic blocks provided in a FPGA chip by technology mapping [34][35][36][37].

2.6.3 Functional Simulation

Two stages of simulation are used for verifying a design: functional simulation and timing simulation. Functional simulation is performed before fitting task, while timing simulation is performed after fitting task. After synthesis task, the functional simulation is performed, where the functional correctness of the synthesised circuit as it is being designed is verified in this stage. Functional simulation simulates the behaviour of the desired circuit without timing information, it does not take into account the timing properties of the logical blocks and inter-connection-wires inside the FPGA chip, they are assumed ideal and therefore the signal propagates through the circuit without propagation delay. Because of that, functional simulation stage takes much less time than timing simulation stage [34][35][38][39].

2.6.4 Fitting

In fitting, the EDIF netlists produced in synthesis stage is translated into the placed and routed FPGA design. The fitting task consists of four phases: translation,

mapping, placement, and routing. In translation phase, the EDIF netlist is converted to a FPGA's manufacturer netlist format and produce its design file. Using a map program, the logical structures of the design file described in the FPGA's manufacturer netlist are mapped into available resources in the actual selected FPGA chip such as logical-blocks, flip-flops, RAMs, and LUTs and the output of this phase are stored in another netlists. Then, the placement of the resources like logical-blocks defined in the incoming netlists from mapping phase is determined into the resources in the actual chosen FPGA chip. After that, the essential routing wires in the actual chip required to establish a connection between these particular resources are selected. Placement and routing phases - which mostly are performed by one program, defines how the resources of FPGA chip are placed and interconnected inside the actual chosen FPGA chip. These two phases are considered the most important and time consuming phases in fitting task. However placement phase is more important than routing phase because good placements result in possible good routing and bad placement may result in unattainable good routing. At the end of this task, a generation of a programming file is created and stored in a bitstream format, where this binary data are used in the next task. Some company merges more than one phase together, for example translation and mapping phases are combined into one phase and executed by one programme called quartus_map[34][36][40].

2.6.5 Timing Analysis and Simulation

In timing analysis, the timing properties of the logical blocks and inter-connection-wires inside the FPGA chip are taken into account. Therefore, an indication of the expected performance of the desired circuit is provided by analyzing all propagation delays the length of the different paths inside the fitted circuit. Then the time simulation is performed where both functional correctness and timing of the fitted circuit as it is being designed is verified. This task is considered as the closet emulation to actually implementation of the desired circuit into the selected FPGA device by providing the behaviour of the desired circuit when it is actually downloaded onto the chosen chip. Thus, it allows the designer to verify that behaves, functional requirements, and timing requirements of the implemented circuits meets the expected one and solve the problems when it happened before the actually implementation. Time simulation and analysis takes more time than functional simulation, but it is more accurate [34][35][39][41].

2.6.6 Programming and Configuration

The bitstream file is loaded into the chosen FPGA device. Thus, the configuration switches inside the chosen FPGA -which is responsible for the configuration logical-blocks and establishing the required inter-connect wiring connections- are programed. Therefore, the desired circuit is implemented into a physical chosen FPGA chip. Because of that, the execution and testing of a desired hardware design is provided by the FPGA chip at the end of this task [34][35].

2.7 Overview of Commercially Available FPGAs

2.7.1 Market Overview

By looking at the current PLD/FPGA market figures, we found that the competition is bitter and Ironic. Figure 2.12 show that Xilinx controls over 50% of the PLD market and while Altera control about 35%. Also in FPGA market share, both Xilinx and Altera control over 80% of the FPGA market, where Altera alone controls about 30% as shown in Figure 2.13 [42]. Therefore, to provide a more focused discussion, only the main FPGA manufactures (i.e Xilinx and Altera) whose products are currently widespread are mentioned in this section [13].

Both Xilinx and Altera work on the following two main approaches to expand the FPGAs' market:

a) Low-End-Market by reducing the per-FPGA-unit-production-cost.

b) High-End-Market by increasing the per-FPGA-unit-production-capacity.

The First approach; per-FPGA-unit-production-cost; can be achieved through many methods, such as by using 90 nm technology, 300 mm wafers, and by designing a less processing, less memory FPGAs. However, the second approach; per-FPGA-unit-production-capacity; is needed in applications, where high performance DSP are required, an embedded processing and a high-speed I/O [43].

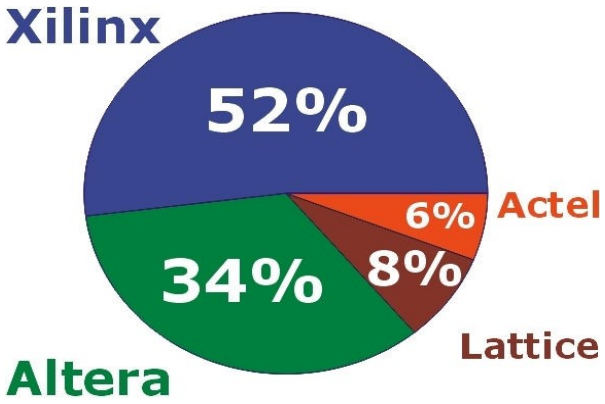


Figure 2-12 PLD market share

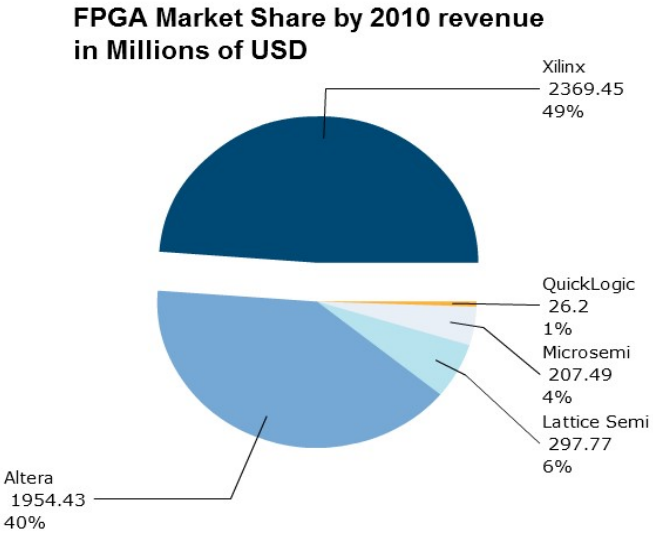


Figure 2-13 FPGA market share

2.7.2 Recent FPGA devices design timeline

The manufacturers of FPGAs provide a wide range of device series. Altera offers Cyclone series at the Low-End to provide lowest cost and power FPGAs, and offers Stratix series at the High-End to provide High-End FPGAs. For a Mid-Range FPGAs, Altera offers Arria series to provide a balanced cost, power, and

performance FPGAs. While Xilinx offers Artix, Virtex, and Kintex at the Low-End, the High-End, and the Mid-Range FPGAs respectively [43][44][45].

Table 2-2 FPGA design timeline

	Altera	Xilinx
High-End FPGA series	Stratix	Virtex
Mid-Range FPGA series	Arria	Kintex
Low-End FPGA series	Cyclone	Artix

Table 2-3 Altera FPGA families Introduction

Altera	Device Family	Year of introduction	Process technology	Recommended for new designs?
Stratix	Stratix	2002	130 nm	No
	Stratix GX	2003	130 nm	No
	Stratix II	2004	90 nm	Yes
	Stratix IIGX	2005	90 nm	Yes
	Stratix III	2006	65 nm	Yes
	Stratix IV	2008	40 nm	Yes
	Stratix V	2010	28 nm	Yes
Arria	Arria GX	2007	90 nm	Yes
	Arria II GX	2009	40 nm	Yes
	Arria II GZ	2010	40 nm	Yes
	Arria V	2011	28 nm	Yes
Cyclone	Cyclone	2002	130 nm	Yes
	Cyclone II	2004	90 nm	Yes
	Cyclone III	2007	65 nm	Yes
	Cyclone IV	2009	60 nm	Yes
	Cyclone V	2011	28 nm	Yes

Table 2.3 shows that Cyclone and Artix series are direct competitors, same as Stratix and Virtex series, and as Arria and Kintex[43][44][45].

Table 2.4 shows the year of introduction, process technology, and if it is recommended for new designs for each family for the three series for Altera. More details such as common features for each family can be found in [46][47][48].

2.7.3 Overview of Development kits

In many cases there is no better way to create, implement, and evaluate a design than with a pre-existing platform from FPGA's manufacturers or one of their development kit partners. FPGA manufacturerdevelopment kits come with everything needed to easily develop and test a system design in a wide range of applications, devices and technologies [49].

Altera and their partners provide a wide variety of development kits that help engineer to simplify the design process and reduce time-to-market [50]. Reference [50] gives us a complete list of the development kits available today and a detail list of the content we will receive on the kit. The documentation that comes with the developments kits make it easy and fast to get the kit working [49].

End market solutions are designed to target an inter-product platform. In reference [51] we can see a complete list of end-markets which Altera provides for specific

solution. Some example end markets include wireless, wireline, broadcast and many others. This reference [51] can help us to decide which Altera device is best situated for our system, which reference designs and IP use, and which literature is available [49].

2.7.4 Educational Board

Altera provides a variety of development and education boards designed to meet educational and researchable needs that help students (undergraduate and researcher) to simplify the design process from simple tasks that clarify essential concepts to complex designs needed to prove a novel proposal [52][53][54][55].

Table 2.5 shows the available educational and researchable boards provided by Altera [52].

Table 2-4 Table Development and Education boards

	Name	Usage				FPGA device	New/Old	I/O feature set	Notes
		Home study	Undergraduate courses	graduate projects	Research				
Teaching purpose	DE0	Yes	Yes			Cyclone III	old	Reduced	A smaller version of DE1
	DE0-		Yes	Yes		Cyclon	new	*	A/D converter

	Nano					e IV			
	DE1		Yes			Cyclone II	new	Reduced	A smaller version of DE2
	DE2		Yes	Yes		Cyclone II	old	Rich	
	DE2-70		Yes	Yes		Cyclone II	old	Rich	
	DE2-115		Yes	Yes		Cyclone IV	new	Rich	
Research purpose	DE3			Yes	Yes	Stratix III	old	High speed	
	DE4			Yes	Yes	Stratix IV	new	High speed	

* Suitable for mobile projects.

Altera DE2 development and Education board shown in Figure 2.14 was designed by professors for university and college laboratory use and Altera DE4 development and education board shown in Figure 2.15 was designed to be an excellent research platform [53][55].

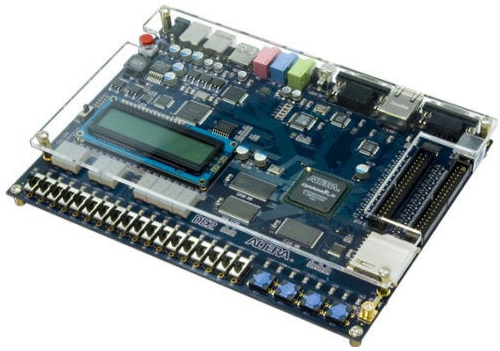


Figure 2-14 DE2 Development and Education Board

Figure 2-15DE4 Development and Education Board

Table 2.6 below shows the main features for DE2, DE3, and DE4 education board.

Table 2-5 “Board Information”

	DE2	DE3	DE4
Feature			
FPGA device	Cyclone II with EPCS16 16-Mbit serial configuration device	Stratix III with EPCS128 128-Mbit serial configuration device	Stratix IV with EPCS128 128-Mbit serial configuration device
Clock	50 MHz clock 27 MHz clock	50 MHz clock	50 MHz clock 100 MHz clock
	External SAM clock input	External SAM clock input External SAM clock output	External SAM clock input External SAM clock output
Memory	8 MB SDRAM 512 SRAM 4 MB Flash	 2 MB SSRAM 64 MB Flash	
	SD memory card slot	SD memory card slot DDR2 memory slot	SD memory card slot DDR2 memory slot
I/O Interfaces			

Built-in USB-Blaster for FPGA configuration	Yes	Yes	Yes
Line In/Out, Microphone In (24-bit Audio CODEC)	Yes		
Video Out (VGA 10-bit DAC)	Yes		
Video In (NTSC/PAL/ Multi-format)	Yes		
RS232	Yes		
Infrared port	Yes		
PS/2 mouse or keyboard port	Yes		
Ethernet Port	Yes (10/100)		Yes (Gigabit)
USB 2.0 (type A and type B)	Yes	Yes	Yes
Expansion headers (two 40-pin headers)	Yes	Yes	Yes
HSMC high-speed headers		Yes	Yes
Serial ATA Port			Yes
PCI Express x8 Edge Connector			Yes

2.8 Conclusion

This chapter has explored many issues in the world of FPGA architecture, CAD, and commercial availability, which has helped the completion of the rest of this

report. Comparison between FPGA and other hardware outcome that are strongly recommended to use FPGA for high-speed digital applications was also performed.

Chapter 3

Multi Carrier Orthogonal Frequency Division Multiplexing

3.1 Introduction

High-bit-rate with high-quality-service is expected from current wireless communication systems. However, in such systems, transmitting data at a high-bit-rate leads to delayed waves, which in turn causes interference problems. Interestingly, Orthogonal Frequency Division Multiplexing (OFDM) transmission scheme was proposed in a bid to overcome this interference problem. OFDM converts a high-bit-rate data streams into low-bit-rate data stream, which are then multiplexed using Frequency Division Multiplexing (FDM). The influence of Inter-Symbol Interference (ISI) and Inter-Carrier Interference (ICI) can be mitigated but not eliminated completely, by inserting a guard interval with cyclic prefix. Hence, the OFDM transmission scheme is robust against multi-path fading[2][3][56].

In this chapter OFDM will be discussed in detail. Investigation will begin from simple single carrier system and then multicarrier systems. Thereafter, different

key components of OFDM system will be discussed and at last its robustness against inter symbol and intra symbol interference will be discussed.

3.2 Single Carrier and Multicarrier Systems

In single carrier systems, there is only one carrier which is used for the purpose of delivering the information of interest. These systems use quite high bandwidth and hence the bit duration is smaller [57]. Impulse noise and signal reflection also cause some problems in these type of systems [57]. Therefore, these problems leads to propose multicarrier system.

The total bandwidth is divided into many bands and data is sent in a parallel manner, using these parallel bands. FDM (Frequency Division Multiplexing) is an example of a multicarrier system, with many advantages over ordinary single carrier systems. In FDM, the available channel is divided into many channels with lower information rate, and as a result each symbol period is long enough to give some immunity to impulse noise and signal reflections[2][57]. However, a problem with Frequency Division Multiplexing that adjacent channels can interfere with each other. Hence a guard band is normally used between the adjacent channels to mitigate this effect as shown in Figure 3.1[2][57]. This guard interval is useful in reducing interference, but some portion of the bandwidth is wasted due to these guard intervals and of course bandwidth is one of the key elements as far as communication systems are concerned.

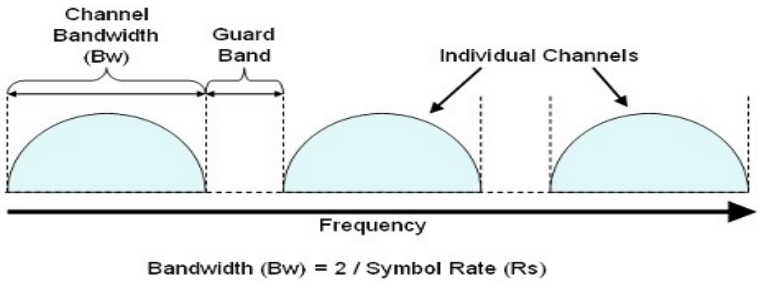


Figure 3-1 FDM System Bandwidth

One can think about eliminating these guard intervals from the FDM system if it can be maintained orthogonally between the adjacent sub-carriers. These orthogonal sub carriers usually overlap in frequency domain but still allow retrieval of the information unless orthogonally is strictly maintained.

3.3 Orthogonal Frequency Division Multiplexing (OFDM)

The spectrum of OFDM is shown in Figure 3.2. It can be seen that there is no guard band between two adjacent subcarriers where Figure 3.2 shows only one channel. If overall bandwidth of OFDM systems is observed, there will be different channels in available bandwidth and in every channel there will be orthogonal sub carriers as shown in Figure 3.3 [2].

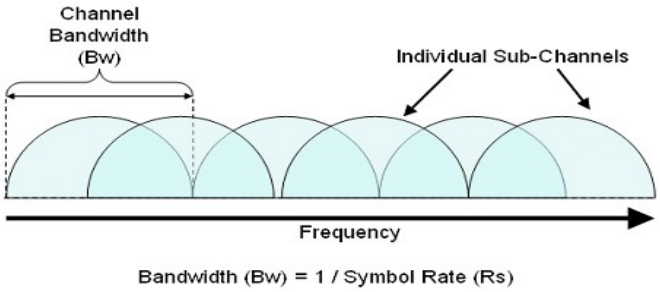


Figure 3-2 OFDM System Spectrum

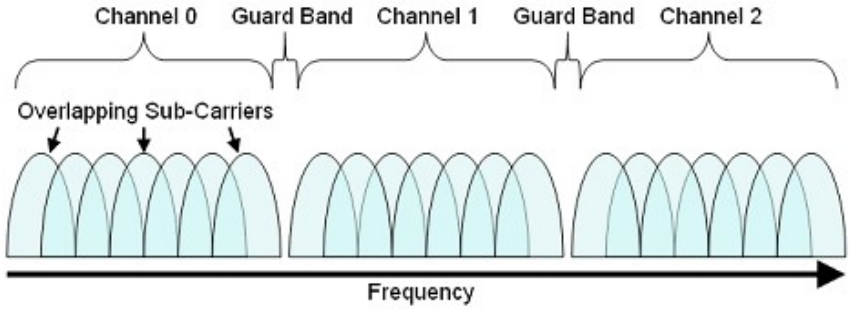


Figure 3-3 OFDM System Spectrum Overview

By comparing the spectrum of FDM system and OFDM system it can be seen that OFDM is subset of FDM as both uses the concept of multi channels but in OFDM there are many orthogonal subcarriers as well.

3.4 OFDM History

The first commercial OFDM based wireless system was introduce in 1990 after the advancement in Digital Fourier Transform (DFT) [3]. Historical developments can be summed up as,

- Multi-carrier high frequency system (Kineplex) that was introduced in 1958 [58].
- First patent of OFDM that was granted in 1970 [59].
- In 1999, IEEE 802.11a standard was introduced.
- An OFDM based cellular network developed under IEEE 802.16e and IEEE 802.20 was developed in 2005.

3.5 OFDM Implementation

The Discrete Fourier Transform (DFT) and The Inverse Discrete Fourier Transform (IDFT) are considered the basics of the OFDM implementation. This can easily give one ideas of how data can be mapped on the orthogonal subcarriers. As a default the IDFT is used on the transmitter side and conventionally IDFT is used to convert frequency domain data to time domain data. This conversion method correlates with the frequency domain data with its sinusoidal basis function, which are obviously orthogonal at some particular frequencies[2][3][57].

In practice, FFT (Fast Fourier Transform) and IFFT (Inverse Fast Fourier Transform) are used in place of DFT and IDFT. These FFT and IFFT are mathematically equivalent to DFT and IDFT respectively. On the transmitter side IFFT takes the 'N' symbols at a time and as a result the output of IFFT will be summation of 'N' symbols and if each symbol has individually period 'T' seconds then OFDM symbol will have length of 'NT'[2][3][57].

3.6 OFDM Transmission Technique

The idea that forms the basics of OFDM transmitter consists of serial to parallel converter and modulator (normally IFFT transmitter). During this parallel conversion process 'n' bits are mapped on 'n' low speed data bits [57]. Initially

'n' bits are held in holding register for 'n' symbol period, hence the period of OFDM symbol will be 'nT'. For essential orthogonally the carrier frequencies are separated by,

$$\Delta f = {}^{1}/_{n \times Tb}$$

3.7 OFDM Transmitter Diagram

OFDM transmitter consists of five blocks: Forward Error Correction (FEC) Encoder, Interleaver, modulator (Mapper), iFFT, and Cyclic Prefix (CP) as shown in Figure 3.4 below[2].

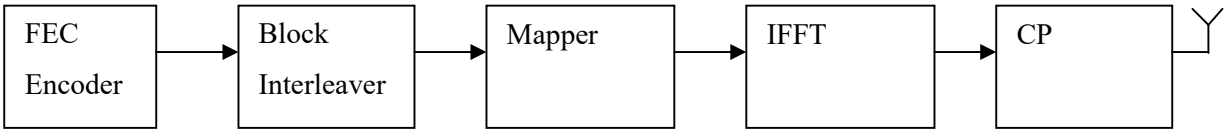


Figure 3-4 OFDM Transmitter Block Diagram

3.7.1 Forward Error Correction (FEC) Encoder

Forward error correction or channel coding is a technique used to control errors in noisy channels. The sender normally performs a redundant encoding using error correcting-code. There are two main classes of forward error correction coding :Block codes and convolutional codes[2][3]. According to

standards,convolutional codes are used in OFDM technique. As a result, only convolutional codes are discussed here.

3.7.1.1 Convolutional Codes:

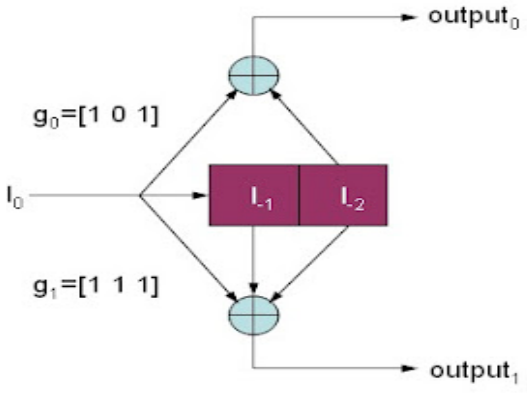
Convolutional codes are normally specified by the number of input bits, the number of output bits, the number of memory registers and constraint length. Constraint length is normally calculated as,

Constraint length=l=x (N-1)

Where x= number of incoming bits

N= number of memory register.

For example, if the calculated constraint length is '3', this means that total number of possible states will be '8' i.e. 2^l[2][3]. A Simple example is shown below in Figure 3.5, with all the important parameters.



Encoder structure for (2,1,3) Convolutional Code with g_0=[1 0 1] and g_1=[1 1 1]

Figure 3-5 FEC Example

There are three ways for explaining the convolutional encoder.

State diagram

Figure 3.6 below shows a state diagram of convolutional encoder with one bit input (shown against every transition) one bit output (shown against every transition) and four possible states. This means that the encoder has a constraint length of '2'. So in this way, how state transition occurs for any input (1/0) and the output of this very encoder (1/0) can easily be understood. Therefore, this state diagram gives us idea of how convolutional encoder works [60].

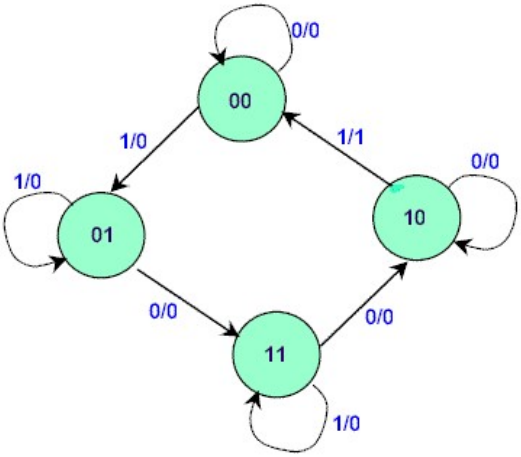


Figure 3-6 State Diagram

Tree Diagrams

Tree diagram is also a good method of explaining how our convolutional encoder will work. Normally have 02 portions of tree and according to the parameters of convolutional encoder it can be seen know how a transition will

occur for any input and what will be the possible output. In tree diagrams if '1' is received we will move towards the upper portion of tree and for zero (0) transition will move towards the lower portion.

Trellis Diagrams

Trellis diagrams are commonly preferred to in convolutional encoding. Simple trellis diagram is shown in Figure 3.7 below. On vertical axis, possible states are shown. Since the way transitions occurs is known, then for every incoming bit output is recorded for every transition and in this way every incoming bit is encoded and sent to next block [61].

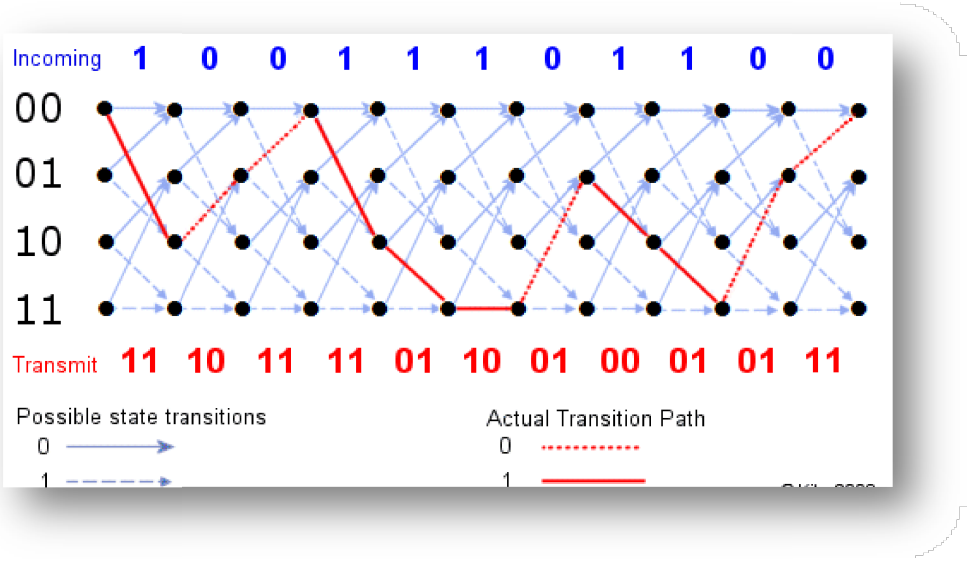


Figure 3-7 Trellis Diagrams

3.7.2 Block Interleaver

The interleaver is very important part of all OFDM transmitter. Adjacent source bits are taken and separated to mitigate the effect of burst error[2][3][57]. There are two different approaches to performing interleaving:

- Block Interleaving
- Convolutional Encoding

According to standard 802.11a, a block interleaver is used and represents the only interleaving approach discussed on this section.

In this type of interleaver, a matrix (block) of order 'M×N' is made. It is very important to be certain of how bits are filled in the matrix and how bits are taken out of the matrix. On transmitter side consecutive bits are filled in the columns and then removed from rows one at a time[2][3]. Same operation is reversed on the receiver side. Interleaving and de-interleaving also have a disadvantage because it adds up to $2MN$ delay.

3.7.3 Mapping

After interleaving is performed, different mapping techniques are availablelike QAMwhere there are two axies for in phase and quadrature component of the interleaved bits. For example if QAM-4 is used then this means that there will be four points on the constellation and every point will represent two bits. Out of these two bits one bit will represent the real part of complex vector and other bit will represent the imaginary part (Quadrature phase component) of the complex vector [3]. The constellation map of QAM-4 shown in Figure 3.8 below.

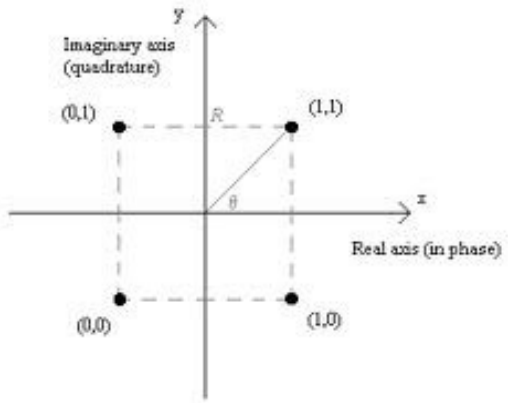


Figure 3-8 QAM-4 Constellation Map

According to IEEE 802.11a standard different mapping (modulation) techniques can be used. The physical layer of OFDM system uses 52 (sub) carriers to achieve different data rates of 6.9…up to 54Mb/s. Out of these 52 sub carriers, four (4) are pilot subcarriers which are used to discard any frequency or phase shift. Actually only 48 sub carriers are used to transmit the actual data without any interference by maintaining the orthogonally. Different mapping techniques recommended in IEEE 802.11a are shown in Table 3.1 below.

Table 3-1 OFDM Mapping Techniques

Data Rate (Mbps)	Modulation	Coding Rate	Coded bits per subcarrier	Coded bits per OFDM symbol	Data bits per OFDM symbol
6	BPSK	1/2	1	48	24
9	BPSK	3/4	1	48	36
12	QPSK	1/2	2	96	48
18	QPSK	3/4	2	96	72
24	16-QAM	1/2	4	192	96
36	16-QAM	3/4	4	192	144
48	16-QAM	2/3	6	288	192
54	64-QAM	3/4	6	288	216

3.7.4 Inverse Fast Fourier Transform (IFFT)

Depending on the number of Subcarriers ‘N’, the IFFT takes the ‘N’ symbols for computation. The IFFT’s basic functions are ‘N’ orthogonal sinusoids with different frequency of 0Hz i.e. DC minimum frequency [57]. So with this IFFT technique data can easily be modulated on ‘N’ orthogonal subcarriers.

FFT is just the inverse of iFFT. Because of that, the basic of FFT and how it works is explained in detailin the section instead of iFFT.

DFT equation is shown below

$$x[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}$$

$$x[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk}, \qquad k = 0,1,2,3 \ldots . N-1$$

$$W_N^{nk} = e^{-j2\pi nk/N}$$

Where W is called twiddle factor.

Firstly, a two- point FFT is, then four-point FFT. After that, 2 four point FFT is used to implement 8 point FFT.

Two- point FFT

$$x[k] = \sum_{n=0}^{1} x[n] W_2^{nk}, \qquad k = 0,1$$

Because $W_2^{0k} = e^{-j0} = 1$

$$and\ W_2^{1k} = e^{-j\pi k} = (-1)^k,$$

We can write

$$x[0] = x[0] + x[1];$$

$$x[1] = x[0] - x[1];$$

The general form of two point FFT is

$$x[k] = x[0] + (-1)^k x[1]$$

Four-point FFT:

$$x[k] = \sum_{n=0}^{3} x[n] W_4^{nk}$$

$$= x[0]W_4^{0k} + x[1]W_4^{1k} + x[2]W_4^{2k} + x[3]W_4^{3k}$$

$$W_N^{nk} = e^{-j(2\pi nk)/N}$$

$$W_4^{0k} = 1$$

$$W_4^{1k} = e^{-j(\pi/2)k} = (-j)^k$$

$$W_4^{2k} = e^{-j\pi k} = (-j)^k$$

$$W_4^{3k} = W_4^{2k} W_4^{1k} = (-1)^k W_4^{1k}$$

Using these results we can write

$$x[k] = x[0] + x[1]W_4^{1k} + x[2](-1)^k + x[3](-1)^k W_4^{1k}$$

$$x[k] = (x[0] + x[2](-1)^k) + ([x[1] + x[3](-1)^k W_4^{1k})$$

Xe[n] =x[2n], n=0,1;

Xo[n] =x[2n+1], n=0,1

Then

$$x[k] = [Xe[0] + Xe[1](-1)^k + Xo[0] + Xo[1](-1)^k] W_4^{1k}$$

The fraction in brackets is in above equation can be recognized as two-points DFT.

$$\mathrm{Xe}[\mathrm{m}] = \mathrm{Xe}[0] + \mathrm{Xe}[1](-1)^{\mathrm{m}}, \quad 0,1;$$
$$Xo[m] = Xo[0] + Xo[1](-1)^{m}, \quad 0,1$$

Since Xe[k] and Xo[k] are periodic

For example

$$Xe[2] = Xe[0] + Xe[1](-1)^{k} = Xe[0]$$

And

$$Xe[3] = Xe[0] + Xe[1](-1)^{k} = Xe[1]$$

The four-point DFT then is

$X[0] = Xe[0] + Xo[0]W_4^{10} = Xe[0] + Xo[0]W_4^{1}$;

$$X[1] = Xe[1] + Xo[1]W_4^{10} = Xe[1] + Xo[1]W_4^{1};$$

$X[2] = Xe[0] - Xo[0]W_4^{12} = Xe[0] - Xo[0]$;

$$X[3] = Xe[1] - Xo[1]W_4^{13} = Xe[1] - Xo[1]W_4^{1};$$

From the above equations, it is obvious that four point DFT can be computed by the generation of two two-point DFTs and by re-composition of terms.

[4 point DFT of X[n]]=[2 point DFT of Xe[n]]+ [2 point DFT of Xo[n]]W_4^{1k}

The above equation is called the re-composition equation of the four point DFT. Thus, 8-points FFT can be generated by combination two of 4- points FFT[2][3].

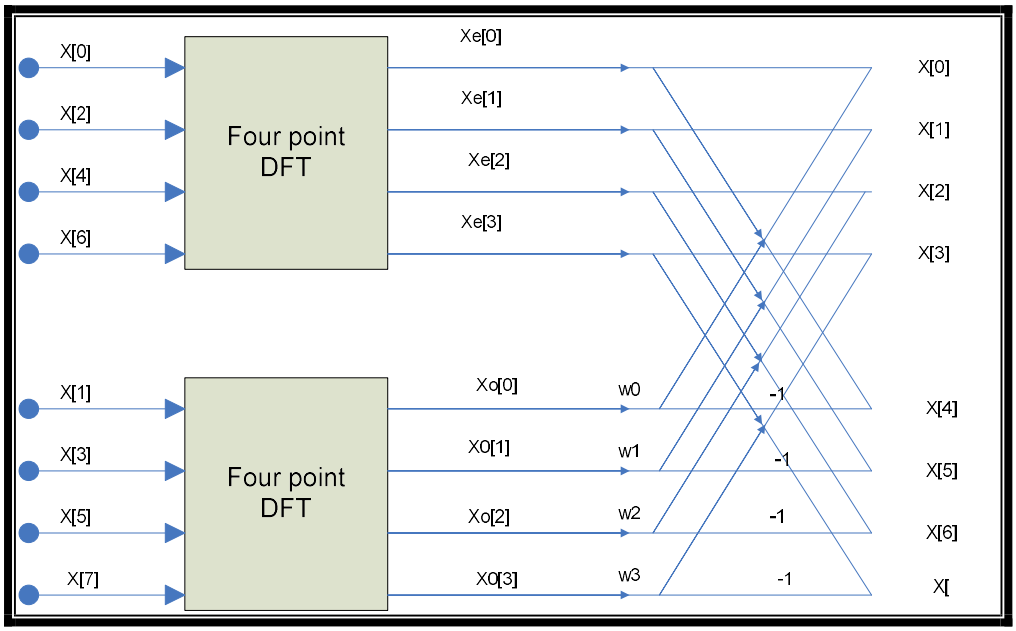


Figure 3-9 8 point FFT block Diagram

3.8 Multipath and Cyclic Prefix

Multipath reception is an important factor to be considered modelling wireless communication systems. There are two main effects on wireless communication systems due to multipath reception. For a fixed point system, the spread of delay is so short that the frequency response of the channel is almost constant over the entire bandwidth and this is normally referred as flat fading as all the components experience same fading. If there is more than one available path for propagation then this is referred to as Rayleigh statistics and spread is obviously longer in this case. The frequency response of the system will change quickly in this case and this can cause severe damage to amplitude and phase of signal of interest. This type of fading is also known as frequency selective fading [3][62].

It is a good practice to increase the transmitter power, though this technique eliminates the effect of flat fading, it is worthless as far as frequency selective fading is concerned and to mitigate this convolutional encoder and decoder are used in transmitter and receiver respectively. In microwave links a fade margin is designed depending on the availability of the link[2][3].

3.8.1 Diversity Receptions

Diversity reception technique is employed to reduce the fade margin. There are three types of diversities, including: Space Diversity, Frequency Diversity, and Polarization Diversity.

Space Diversity

In space diversity, the signal of same frequency is transmitted on different paths and it is probable that both paths will not experience the same amount of fading. There is a DSP based mechanism to check the strength of each signal.

Frequency Diversity

Frequency diversity have same path but transmit signal on two different frequencies. So there will be different attenuations at different frequencies and allows the selection of the most powerful signal.

Polarization Diversity

Polarization diversity have same path and same transmitting frequency but different polarizations like vertical or horizontal polarizations (for Microwave links).

3.8.2 Multipath and OFDM

There are two effects on an OFDM system due to multipath. First one is known as inter-symbol interference and in this type of interference newly received symbols is affected by previously delayed symbol[5][57]. The second type of interference is intra-symbol interference and is present only in multicarrier systems. In this case, the symbol is distorted by its own subcarriers [57].

Inter-Symbol Interference

As in an OFDM system, the data rate is low for individual subcarrier as compared to single carrier system so it is certainly possible to choose the number of subcarriers 'N' such that overall OFDM symbol period will be greater than channel's time span [57]. This can give indication that only the first few symbols will be affected because of inter-symbol interference. Now if some sort of guard interval can be placed in front of every OFDM symbol with all the zeros in this guarding space then the inter-symbol interference will only have effect on this guard space and not on real symbol. This guard interval can easily be removed from the symbol with prior knowledge by the receiver side. As such,

this technique can mitigate the effect of inter-symbol interference, but it is not overly effective as far as intra symbol inference is concerned[5][57].

Intra-Symbol Interference

Interference that occurs amongst the subcarriers of same symbol is called intra-symbol interference [57]. It is a well-known fact from signal and system theory that convolution in time domain is actually equal to multiplication in frequency domain. This property is true for discrete time only if signal is of infinite length or if signal is periodic over entire range. Obviously a signal cannot be infinite in practical terms but it is quite possible to make a signal periodic over the range of intended convolution. This gives an idea of cyclic prefix, which involves the idea of placing last few samples in front of every symbol where time span of replicated samples must be greater than the time span of an OFDM symbol. So, OFDM symbols are now presented as periodic and now the effect of channel becomes multiplicative and both inter-symbol and intra-symbol interference can be handled without any loss of information. Additionally, cyclic prefix can be removed from actual symbol on receiver side with prior knowledge[2][3].

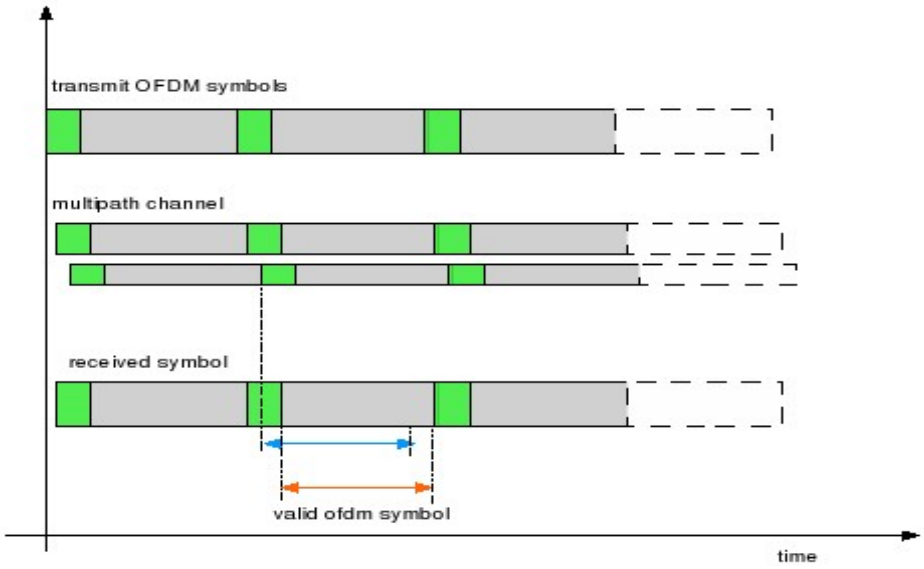


Figure 3-10 Cyclic Prefix in OFDM

3.9 OFDM in Future

Worldwide, leading researchers in the field are taking OFDM as key component in many of the future wireless communication systems. From specifications of IEEE 802.11a/g/ac/ad, one can imagine that it will play important role in WLAN [63]. Edward Rerist, Senior Analyst from Allied Business intelligence suggests that the world should deploy OFDM for 3G solution instead of any other technique [63]. Similarly some researchers also stated that “Japanese are trying efficiently to standardize OFDM for current 4G technology” [63]. Currently, many researchers working on a new OFDM for future 5G technology[64]. From this discussion one can easily determine the importance of OFDM as far as future technological systems are concerned and probably the reason people are doing more research on this topic.

3.10 Conclusion

OFDM is an efficient way to mitigate all the issue associated with conventional single carrier systems and multicarrier system. It can be implemented easily by FFT/IFFT processors and in this way one can achieve higher spectral efficiency. One of the main advantages of OFDM system is that it can easily eliminate the inter symbol and intra symbol interference by using a simple technique of cyclic prefix. Due to these reasons OFDM is strongly recommended for modern wireless communication standards.

Chapter 4

Design of a basic FPGA-Based OFDM Transmitter and Receiver using FFT Algorithm

4.1 Introduction

The outcome of the comparison between FPGA and other hardware in chapter two strongly recommendto use FPGA for high-speed digital applications. Chapter three shows that OFDM system meets the requirements of modern wireless communication system.Because of the needed for hardware implementation in real time system, this chapter presents the encoder/decoder, interleaver/de-interleaver, QAM modulator/de-modulator, and iFFT/FFT modules for a basic OFDM transceiver implemented on FPGA, using Altera EP2C35F672C6 Cyclone II chip as a point of reference. A number of high-level design software tools; Altera Quartus II, Altera's DSP Builder and SOPC Builder, and MathWork's Simulink; are used to implement the OFDM system.

This project divides the design into functional subsystems at both the transmitter and receiver sides. Specifically, at the transmitter side: a convolutional encoder

of rates (2, 1, and 4),2×2 interleaver, 16-QAM mapper, and 64-point iFFT are used. While at the receiver side: 64-point FFT, 16-QAM de-mapper, 2×2 de-interleaver, and soft-decision Viterbi decoder are used.

Each of the above subsystem passes through the following sequence stages: design of the model, simulation using MATALB Simulink, compiling the model with Quartus II software and finally configuration of the FPGA board to implement the desired circuit. The designed circuit in each subsystem is simulated using MATLAB Simulink to visualize the result and to verify the functional correctness of it. The compiling of the desired circuit is achieved with Quartus II software. The desired circuit is analysed and synthesised into a circuit consisting of LEs supported by Cyclone II FPGA chip. Following the completion of the above stages, HDL files are created that are used later in the configuring of the FPGA chip to implement the desired circuit into a physical chosen FPGA chip. Thus, execution and testing of the desired hardware design is provided at the end.

4.2 Why OFDM on FPGA

Because OFDM is carried out on digital domain, it can be implemented into three different methods: ASIC, Microprocessor or Microcontroller, and FPGA. Using a universal purpose microprocessor or microcontroller to implement OFDM system is more important in terms of peripherals such as memory, higher power, and

larger memory space. Also its operation speed is lower compared to ASIC and FPGA. Implementing OFDM system on ASIC hardware is the most power efficient, smallest, and fastest device. However, it is inflexible and needs more time for marketing the designed chip compared to FPGA [9][65][66].

Using a programmable FPGA to implement OFDM system is appealing because it combines the attributes of the ASIC system, including: density, power and speed with the attributes of the microprocessor or microcontroller (general purpose programmability). In addition to these interesting features stated above, reconfiguration attribute of FPGA does not need any physical fabrication facility on the IC. The exclusion of physical fabrication provides the engineer designer with a full control over the actual OFDM design implementation [9][67][68].

4.3 Convolutional Encoder Implementation

4.3.1 Design the Convolutional Encoder Model

A convolutional encoder of rates (2,1,4) – i.e. one bit input, two bits output and four bits register, is implemented using the Graphical state machine editor block in DSP Builder Standard Blockset, where the Quartus State diagram editor is used to parameterize all transitions. The implemented model is shown in the Figure 4.1 and Figure 4.2 below. Table 4.1 shows the look-up table for polynomial generator explained in [69]. Constrains length is calculated using the

formula L = K(m-1), where k is the number of input bits and m is the number of memory register, thus L = 3 and the number of possible state are 2^3 which is equal to 8.

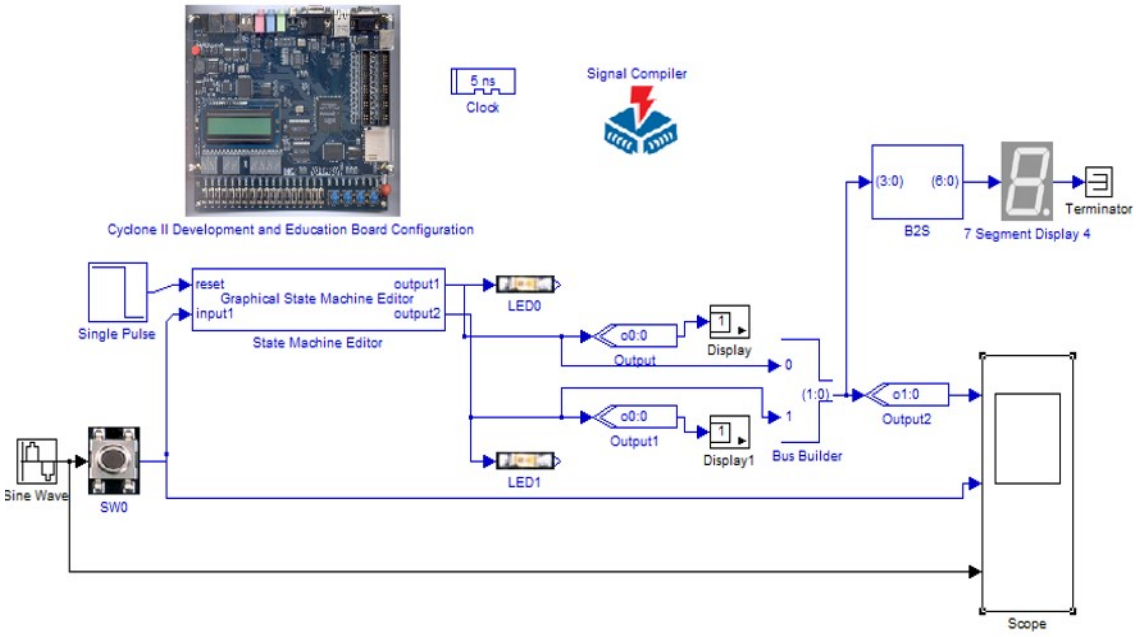


Figure 4-1 Convolutional Encoder of rate (2, 1, 4) Design

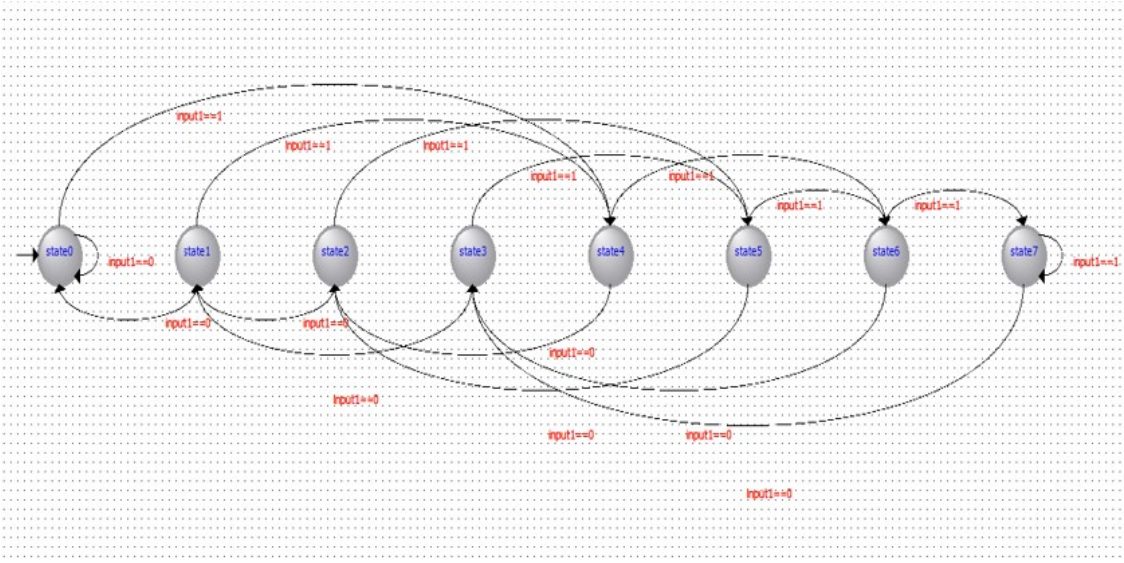


Figure 4-2 State Machine Editor

Table 4-1 Look-up table for encoder of rate (2, 1, 4)

	Input bit		Input state				Output bit			Output state		
1	0		0	0	0		0	0		0	0	0
2	1		0	0	0		1	1		1	0	0
3	0		0	0	1		1	1		0	0	0
4	1		0	0	1		0	0		1	0	0
5	0		0	1	0		1	0		0	0	1
6	1		0	1	0		0	1		1	0	1
7	0		0	1	1		0	1		0	0	1
8	1		0	1	1		1	0		1	0	1
9	0		1	0	0		1	1		0	1	0
10	1		1	0	0		0	0		1	1	0
11	0		1	0	1		0	0		0	1	0
12	1		1	0	1		1	1		1	1	0
13	0		1	1	0		0	1		0	1	1
14	1		1	1	0		1	0		1	1	1
15	0		1	1	1		1	0		0	1	1
16	1		1	1	1		0	1		1	1	1

4.3.2 Simulating the Convolutional Encoder in MATLAB Simulink

The convolutional encoder circuit is simulated using MATLAB Simulink to visualize the result and to verify the functional correctness of the designed circuit. A result in decimal format for one clock cycle is shown in Figure 4.3 below. The figure shows that each single input bithas a two output bits compatible to the

look-up table for polynomial generator mentioned above. Note that the output in scope is the summation of the two output bits in decimal format. The two output bits are viewed separately in binary format using Display block.

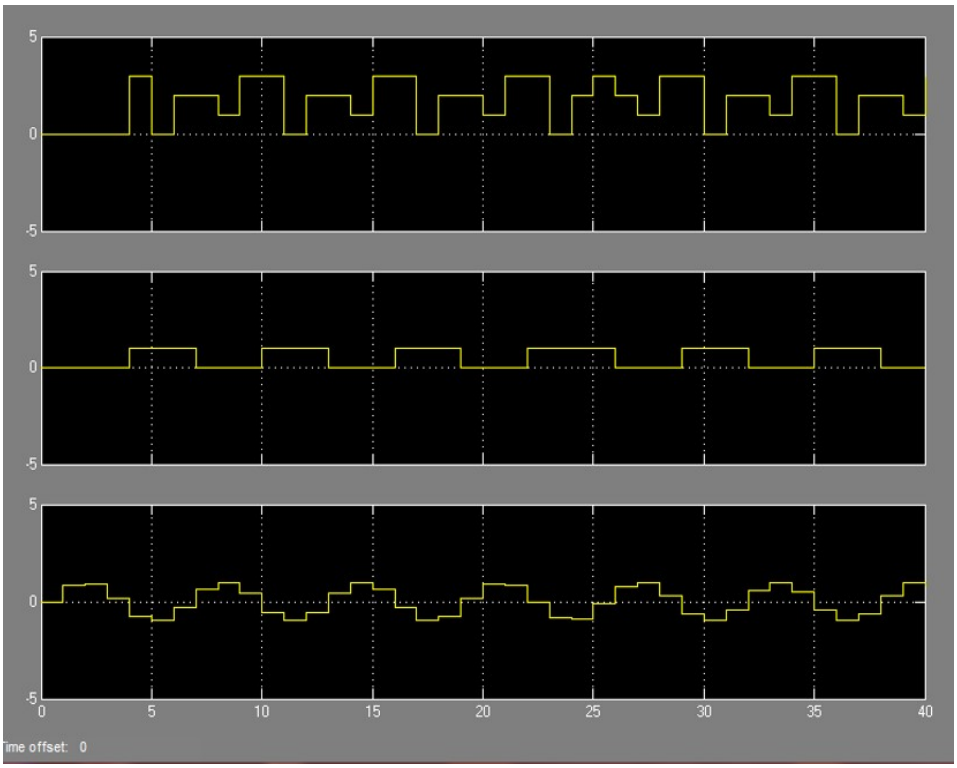


Figure 4-3 Convolutional Encoder of a rate (2, 1, 4) MATLAB Simulink Result

4.3.3 Compiling the Convolutional Encoder with Quartus II Software

The convolutional encoder of rates (2, 1, 4) circuit is analysed, synthesised, and fitted successfully using Quartus II software and HDL files are created. Regarding the resources, the convolutional encoder of rates (2,1,4) implemented using the above design has allocated 18 combinational functions, which are less

than 1% of the functions available on the Cyclone II FPGA chip, and only 9 logical registers but totally 18 (less than 1%) LEs have been used as shown in Figure 4.4 below.

Flow Summary	
Flow Status	Successful - Thu May 24 11:02:11 2012
Quartus II 64-Bit Version	11.0 Build 157 04/27/2011 SJ Full Version
Revision Name	Conv_Encoder_Tx
Top-level Entity Name	Conv_Encoder_Tx
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	18 / 33,216 (< 1 %)
Total combinational functions	18 / 33,216 (< 1 %)
Dedicated logic registers	9 / 33,216 (< 1 %)
Total registers	9
Total pins	16 / 475 (3 %)
Total virtual pins	0
Total memory bits	0 / 483,840 (0 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 4-4 Allocated Resources for Convolutional encoder of rate (2,1,4)

4.3.4 Configuration the FPGA board to implement the convolutional encoder

The convolutional encoder is successfully implemented into the physical Cyclone II FPGA chip using the Quartus II software and USB cable as shown in Figure 4.5 below.

Then the circuit is executed and tested by using the switch0 as the input bit, LED0 LED1 to display the coded bits in digital format, and 7-segment display to display the coded bits in decimal format. All the combination in the Table 4.1 above have been tested and we observed that the result are same as in

simulation one. For example when input is 1 and the current state is 000, the output is 11, which is 3 in decimal as shown in Figure 4.6 below.

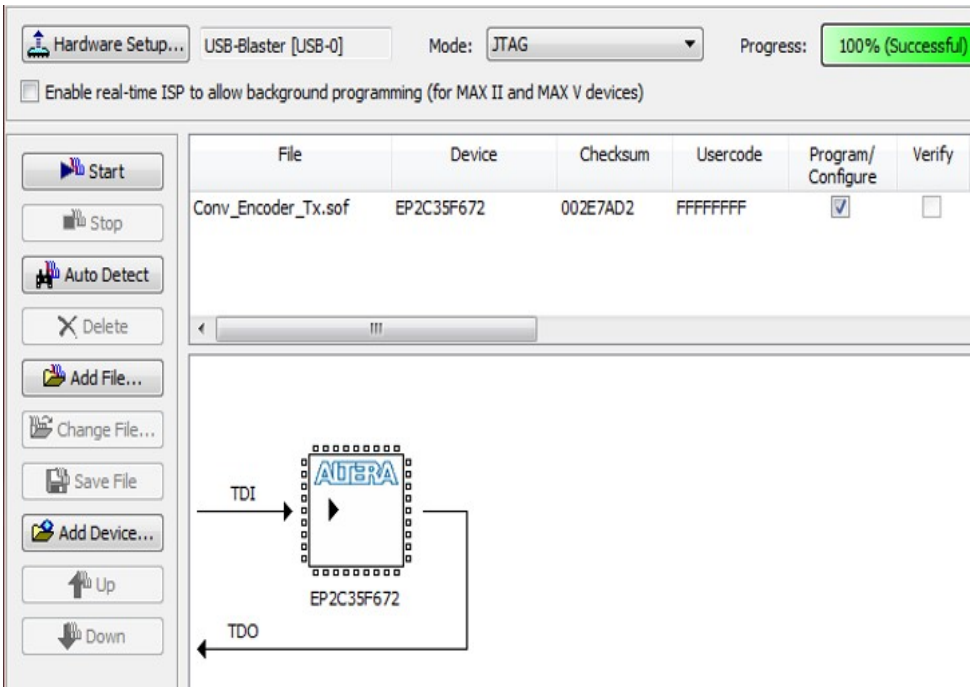


Figure 4-5 Configuration the Convolutional Encoder on Cyclone II FPGA board

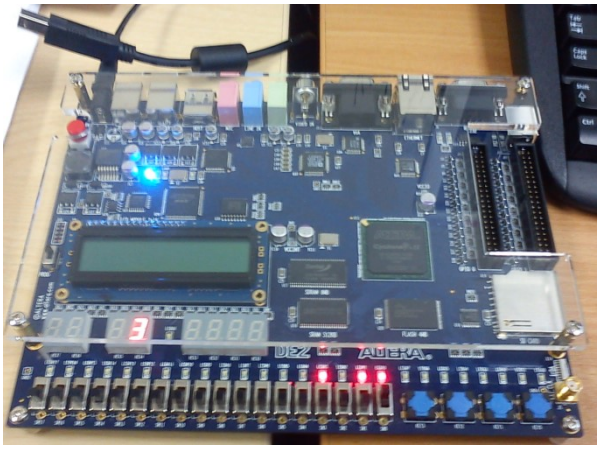


Figure 4-6 Configuration the Convolutional Encoder on Cyclone II FPGA board/example

4.4 Block Interleaver Implementation

4.4.1 Design the Block Interleaver Model

The output of convolutional encoder is fed into a 2×2 interleaver block, where the consecutive bits are fed into the block in column wise and read out into separate consecutive bits in row wise. Interleaving does not add any overhead, but adds up a delay equal to 2×C×R, where C is the number of column and R is the number of row of the interleaver matrix.

Using blocks in the DSP Builder Standard Blockset, a block interleaver model is implemented. As shown in Figure 4.7 below, a two parallel-coded bits are converted to serial format and fed into a FIFO buffer, which is then converted to parallel format and swapped. Thus interleaved bits can work strongly against burst error.

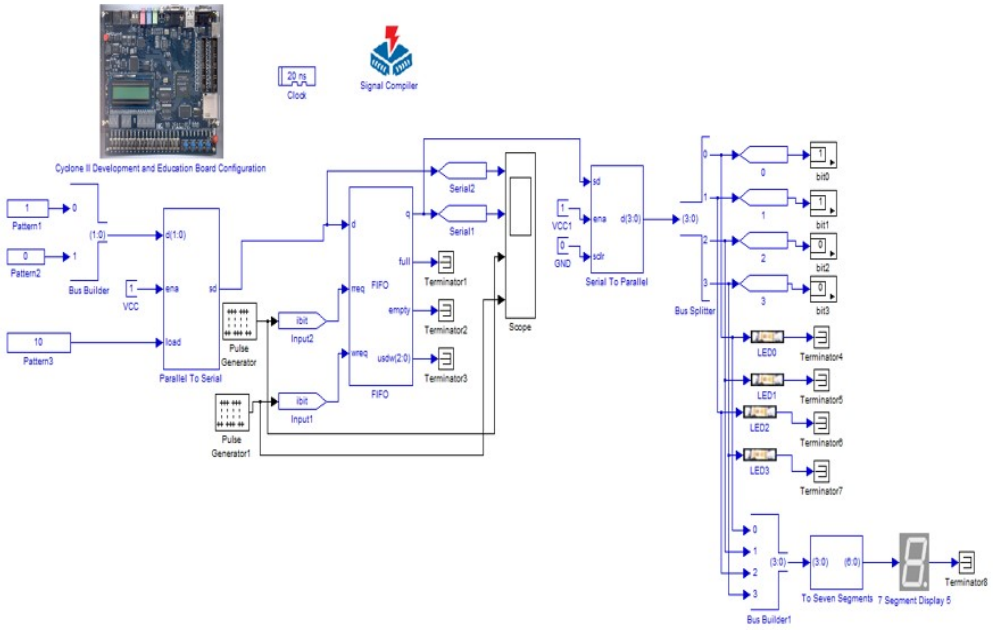


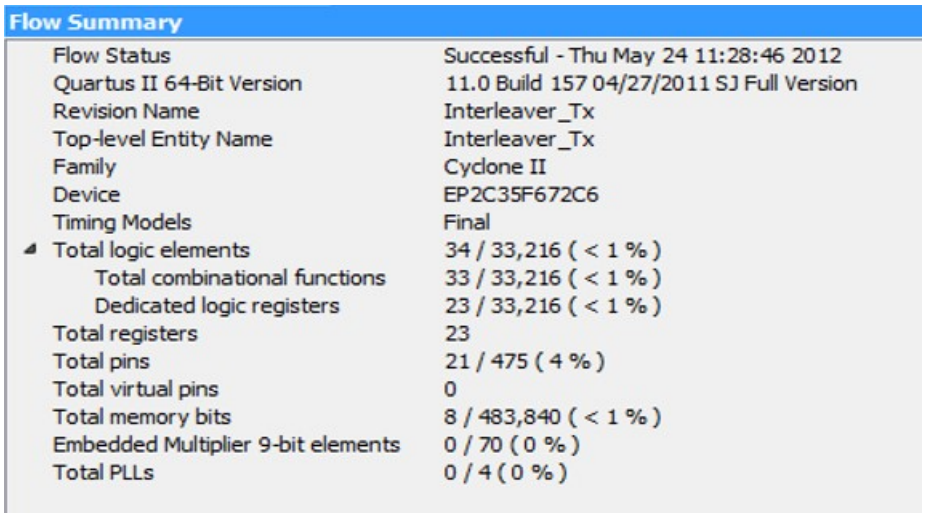
Figure 4-7 A 2×2 Block Interleaver Design

4.4.2 Simulating the Block Interleaver in MATLAB Simulink

The block interleaver circuit is simulated using MATLAB Simulink to visualize the result and to verify the functional correctness of the designed circuit. In this design, ‘10’ sequences are generated every clock cycle, and a block interleaver generated every two-clock cycle, therefore, the design consists of four bits in sequence ‘1010’. Output of the interleave block consists of sequence ‘1100’ as displayed on the display block which is compatible to a 2×2 block interleaver.

4.4.3 Compiling the Block Interleaver with Quartus II Software

HDL files are created, analysed, synthesised, and the 2by2 block interleaver circuit fitted with the Quartus II software. The block interleaver with the above design has allocated these resources on the target device: 33 combinational functions (less than 1%) and 23 (less than 1%) registers out of the available on the Cyclone II FPGA chip as shown in Figure 4.8 below.



Flow Summary	
Flow Status	Successful - Thu May 24 11:28:46 2012
Quartus II 64-Bit Version	11.0 Build 157 04/27/2011 SJ Full Version
Revision Name	Interleaver_Tx
Top-level Entity Name	Interleaver_Tx
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	34 / 33,216 (< 1 %)
Total combinational functions	33 / 33,216 (< 1 %)
Dedicated logic registers	23 / 33,216 (< 1 %)
Total registers	23
Total pins	21 / 475 (4 %)
Total virtual pins	0
Total memory bits	8 / 483,840 (< 1 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 4-8 Allocated Resources for 2by2 Block Interleaver

4.4.4 Configuration the FPGA board to implement the block interleaver

The block interleaver is successfully implemented into the physical Cyclone II FPGA chip using the Quartus II software and USB cable as shown in Figure 4.9 below.

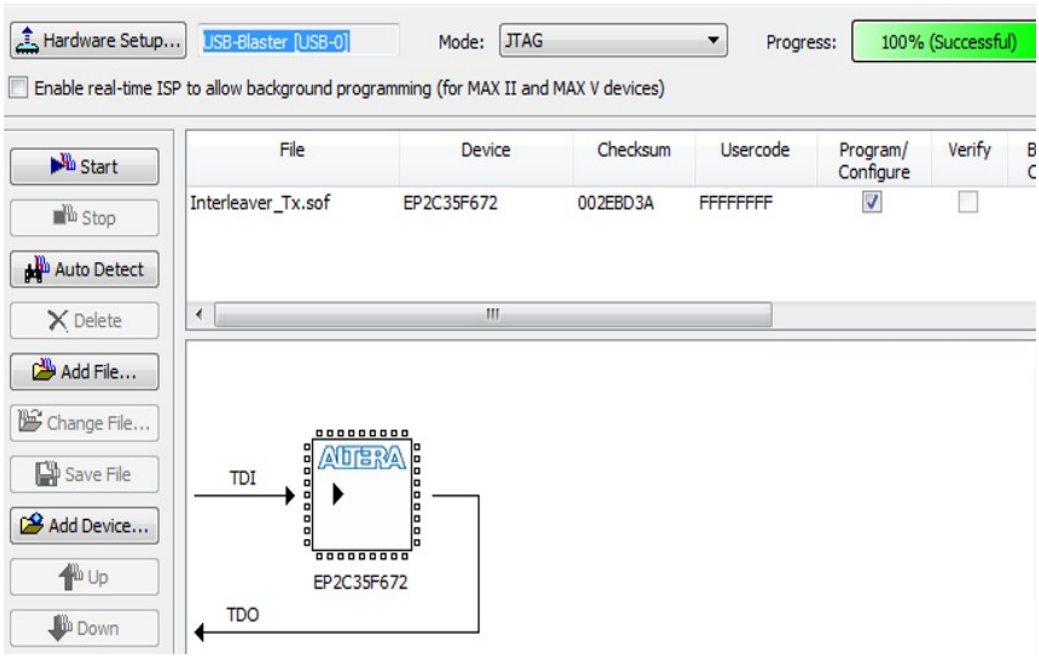


Figure 4-9 Configuration the Block Interleaver on Cyclone II FPGA board

4.5 16-QAM Mapper Implementation

4.5.1 Design the 16-QAM Mapper Model

Outputs of interleaver are fed into a 16-QAM mapper block. The mapper model is implemented based on the gray code mapping shown in Figure 4.10 below, where b0b1 represents real axis and b2b3 represents the imaginary one.

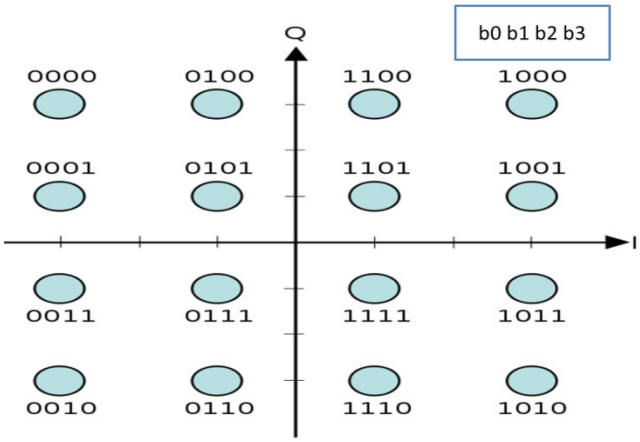


Figure 4-10 16 QAM constellation with Gray code mapping

The four interleaved bits are fed into 16-QAM model, where the first two bits represent the real part and the second two the imaginary part. Each part is handled separately in similar process to achieve an efficient FPGA implementation. As show in Figure 4.11 below, the real bits are fed into a 4×2 ROM, where each combination of the real bits represents one address in the ROM. Then they are fed into a two LUTs, where the upper one gives a valid output for ‘00’ and ‘01’ input combination, and the lower one gives ouput for ‘11’ and ‘10’ combinations. This is because, to achieve gray coded map, LUT does not contain the bits in the normal order, for example, on the real axis the combination ‘10’ represent 3 instead of representing 1 and ‘00’ represent -3 instead of 3, therefore two LUTs are used instead of one. Finally, one of the two-output combinations is selected using a multiplexer based on one of the two input bits. For example the output from the upper LUT is selected if the input bits are ‘00’ or ‘01’.

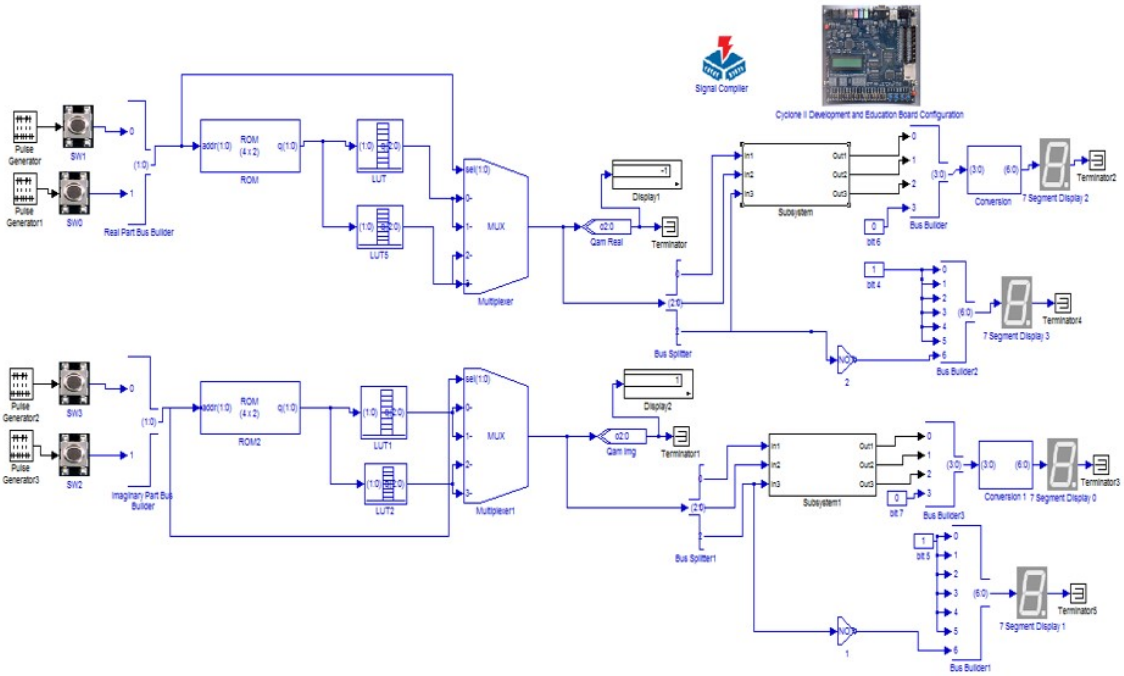


Figure 4-11 16-QAM mapper design

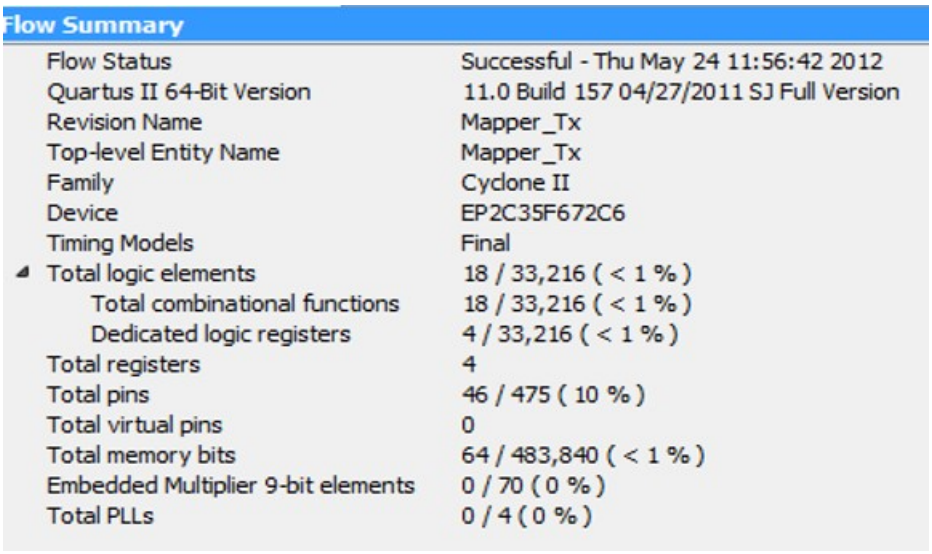
4.5.2 Simulating the 16-QAM Mapper in MATLAB Simulink

The 16-QAM mapper is simulated using MATLAB Simulink to visualize the result and to verify the functional correctness of the designed circuit. All the 16 input combinations are checked and the result is displayed on the Display block and the block output shows that the result is compatible to gray code mapping explained in the previous section.

4.5.3 Compiling the 16-QAM Mapper with Quartus II Software

After the 16-QAM mapper with the design explained above has been successfully analysed, synthesised, and fitted and the HDL files created using

the Quartus II software, it was compiled to investigate the allocated resources of the used FPGA chip. Figure 4.12 below shows that in all a total of 18 LEs has been allocated (18 combinational functions and only 4 logical registers was allocated out of the available on the chosen FPGA device.



Flow Summary	
Flow Status	Successful - Thu May 24 11:56:42 2012
Quartus II 64-Bit Version	11.0 Build 157 04/27/2011 SJ Full Version
Revision Name	Mapper_Tx
Top-level Entity Name	Mapper_Tx
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	18 / 33,216 (< 1 %)
Total combinational functions	18 / 33,216 (< 1 %)
Dedicated logic registers	4 / 33,216 (< 1 %)
Total registers	4
Total pins	46 / 475 (10 %)
Total virtual pins	0
Total memory bits	64 / 483,840 (< 1 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 4-12 Allocated Resources for 16-QAM mapper

4.5.4 Configuration the FPGA board to implement the 16-QAM mapper

The implementation of the 16-QAm mapper into the physical Cyclone II FPGA chip using the Quartus II software was successful as shown in Figure 4.13 below.

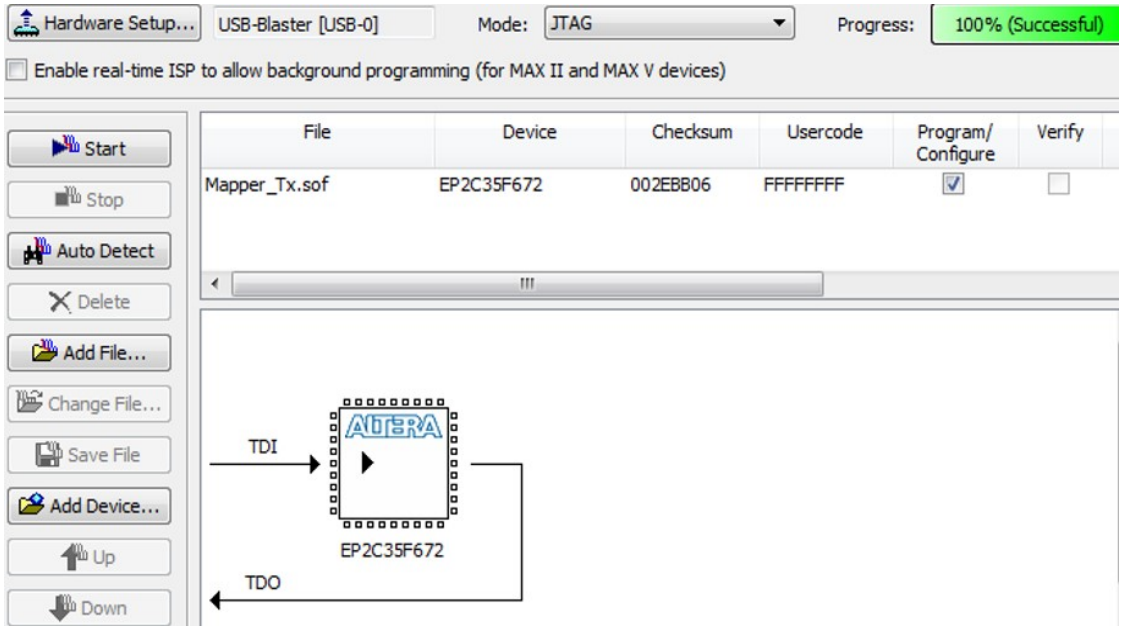


Figure 4-13 Configuration the QAM mapper on Cyclone II FPGA board

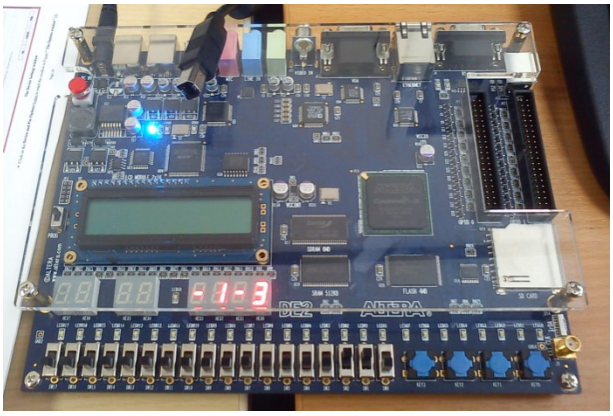


Figure 4-14 Configuration the 16 QAM mapper on Cyclone II FPGA board/example

Then the QAM Mapper circuit is executed and tested by using switches to feed the input bits, and 7-segments to display the mapped data on decimal format. All the combination of input bits have been tested and we observed that the result are same as simulation one. As an example when real input bits are ’01’ selected

by switch 0 and switch 1 respectively, and imaginary input bits are ‘10’ selected by switch 2 and switch 3 respectively to construct the input ‘0110’, we get the real mapped data ‘-1’ displayed on 7-segment 2 and 7-segment 3, and the imaginary mapped date ‘-3’ displayed on 7-segment 0 and 7-segment 1 as shown in Figure 4.14 below.

4.6 64-Point IFFT Implementation

4.6.1 Design the 64-Point IFFT Model

One of the Mega Core Functions blocks is called FFT Mega Core function. By using it, a complex FFT/IFFT circuit can be implemented and for a high performance applications a higher level of parameterization is used [70].

Altera provides two architectures for FFT Mega Core Function: i) Fixed Transform Size Architecture, and ii) Variable Streaming Architecture [70].

Fixed Transform Size Architecture achieves a best trade-off between minimum size and maximum SNR requirements by using a block-floating point representation. Input of this architecture is a complex data vector of size N in two complementally format and the output is a complex vector but in a natural order where N is the transformation size [70].

Variable Streaming Architecture has two types of architecture: I) a radix-2^2 single delay feedback architecture and II) a mixed radix-4/2 architecture. In the first type, a fixed point representation is used and thus the input data widths are grown naturally through to output to keep a high output SNR. While in second type, a single precision floating point representation is used to represent a large dynamic range of values and keep a high output SNR [70].

Figure 4.15 below shows a 64-point IFFT Mega Core design.

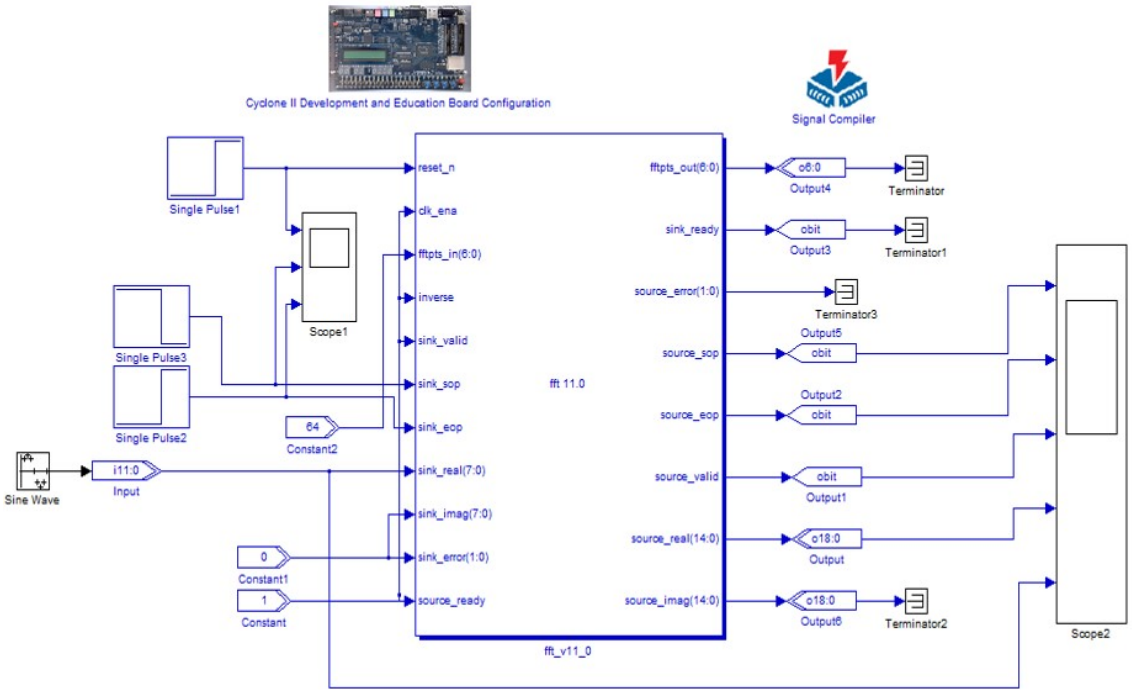


Figure 4-15 64-IFFT Mega Core Design

4.6.2 Simulating the 64-Point IFFT in MATLAB Simulink

Transform length is setting up to 64 points and a variable streaming architecture with Natural Order and Fixed point data representation is selected using the Parameterize block inside FFT Mega Core Function. Then the 64-point IFFT circuit is simulated using MATLAB Simulink to visualize the result and to verify the functional correctness of the designed circuit.

The length of input data is setting up making the input to valid and selecting a start of packet (sop) and an end of packet (eop), and thus the IFFT is calculated on the sample points only between sop and eop. This makes the design more flexible to provide an effective result in any environment.

The result of running the MATALB Simulink simulation for the designed circuit is shown in Figure 4.16 below. The first two graphs – from the top- are the start of packet and end of packet respectively, while the third one shows when the data is valid. Manually, 12 bits are used to represent each input sample, where, automatically using a parameterized, 19 bits are used to represent each output sample. By doing this, it is guaranteed that any output can be represented within the provided bits regardless of any change in length of words, and in number of addition multiplication in IFFT during processing of this model. Note that the IFFT is only calculated when input data is valid.

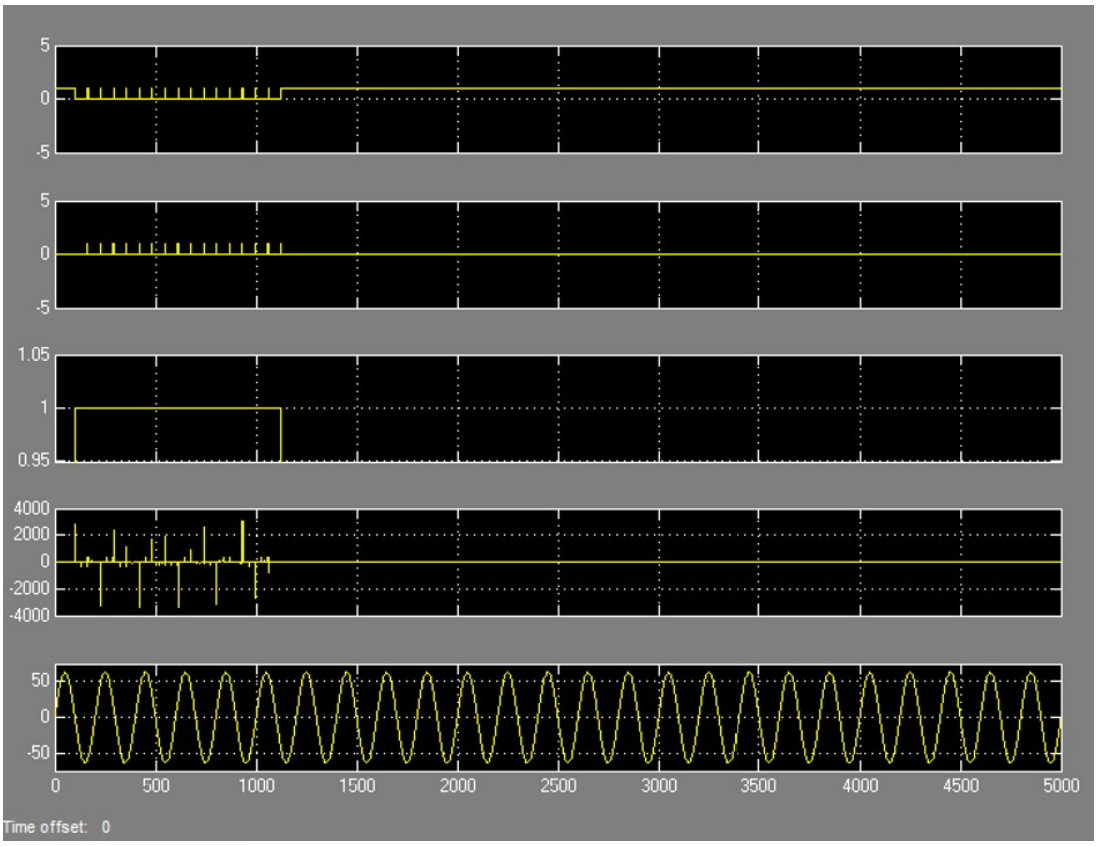


Figure 4-16 64-point IFFT MATLAB Simulink Result

4.6.3 Compiling the 64-Point IFFT with Quartus II Software

The 64-point IFFT block from Mega Core Functions block with the settings up as mentioned above has been analysed, synthesised, and frittered and consequently the HDL files has been created smoothly using the Quartus II software. By investigation of the allocated resources of Cyclone II FPGA device used through compiling of the circuit as shown in Figure 4.17 below, it is founded that 5% (1754) of the LEs inside the selected FPGA device have been allocated,

which is greater than the allocated resources by all the previously described circuits of OFDM system.

Flow Summary	
Flow Status	Successful - Thu May 24 14:55:16 2012
Quartus II 64-Bit Version	11.0 Build 157 04/27/2011 SJ Full Version
Revision Name	iFFT_Tx
Top-level Entity Name	iFFT_Tx
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	1,754 / 33,216 (5 %)
Total combinational functions	1,287 / 33,216 (4 %)
Dedicated logic registers	1,451 / 33,216 (4 %)
Total registers	1451
Total pins	63 / 475 (13 %)
Total virtual pins	0
Total memory bits	1,326 / 483,840 (< 1 %)
Embedded Multiplier 9-bit elements	16 / 70 (23 %)
Total PLLs	0 / 4 (0 %)

Figure 4-17 Allocated Resources for 64-point IFFT

4.6.4 Configuration the FPGA board to implement the 64-point IFFT model

The 64-point IFFT model is successfully implemented into the physical Cyclone II FPGA chip using the Quartus II software and USB cable as shown in Figure 4.18 below.

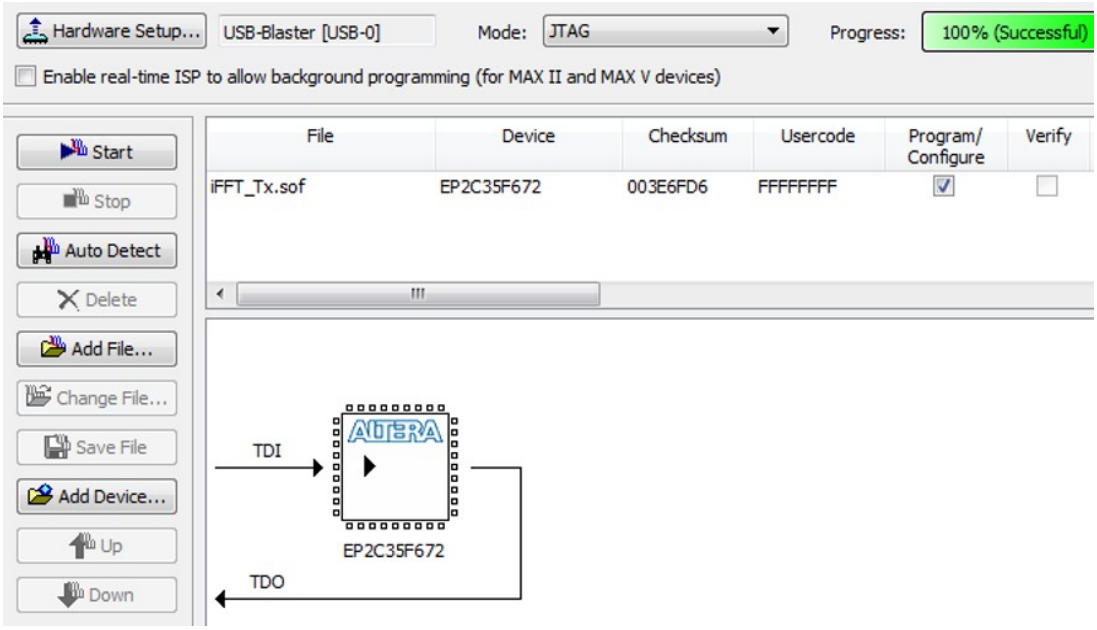


Figure 4-18 Configuration the IFFT circuit on Cyclone II FPGA board

4.7 OFDM Transmitter using FFT Algorithm Implementation

4.7.1 Design the OFDM Transmitter using FFT Algorithm

A full OFDM transmitter using FFT algorithm has been designed by integration of the previous circuits: convolutional encoder of rates (2,1,4), 2×2 block interleaver , 16-QAM mapper, and 64-point IFFT , as shown in Figure 4.19 below.

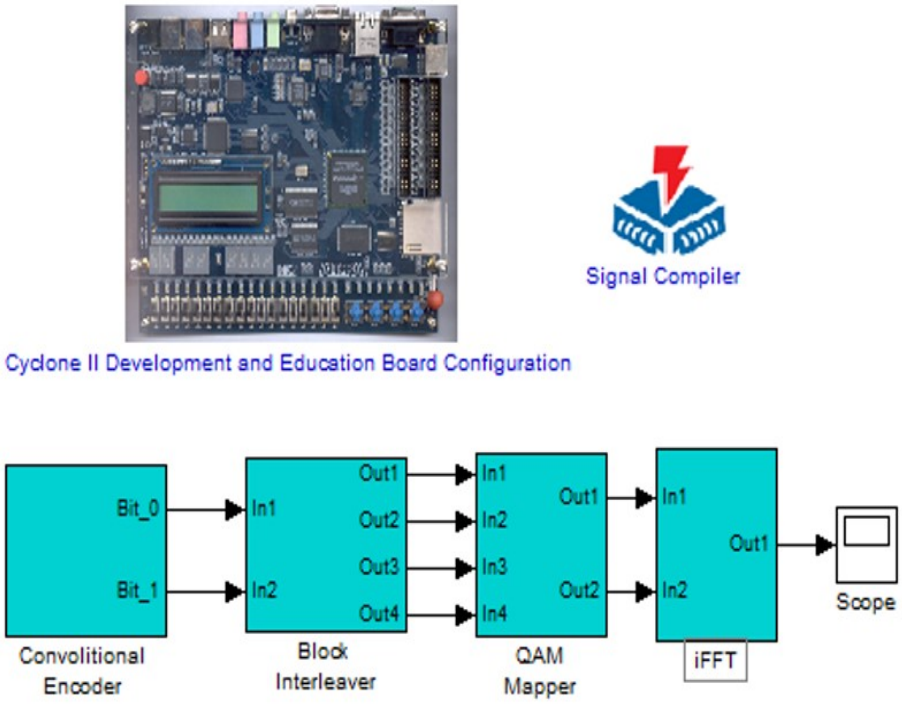


Figure 4-19 OFDM Transmitter -using FFT algorithm- design

4.7.2 Simulating the OFDM using 64-Point IFFT in MATLAB Simulink

By using MATLAB Simulink, output of the OFDM Transmitter is visualized and the functional correctness is verified as shown in Figure 4.20 below, which is the output of the 64-point IFFT block.

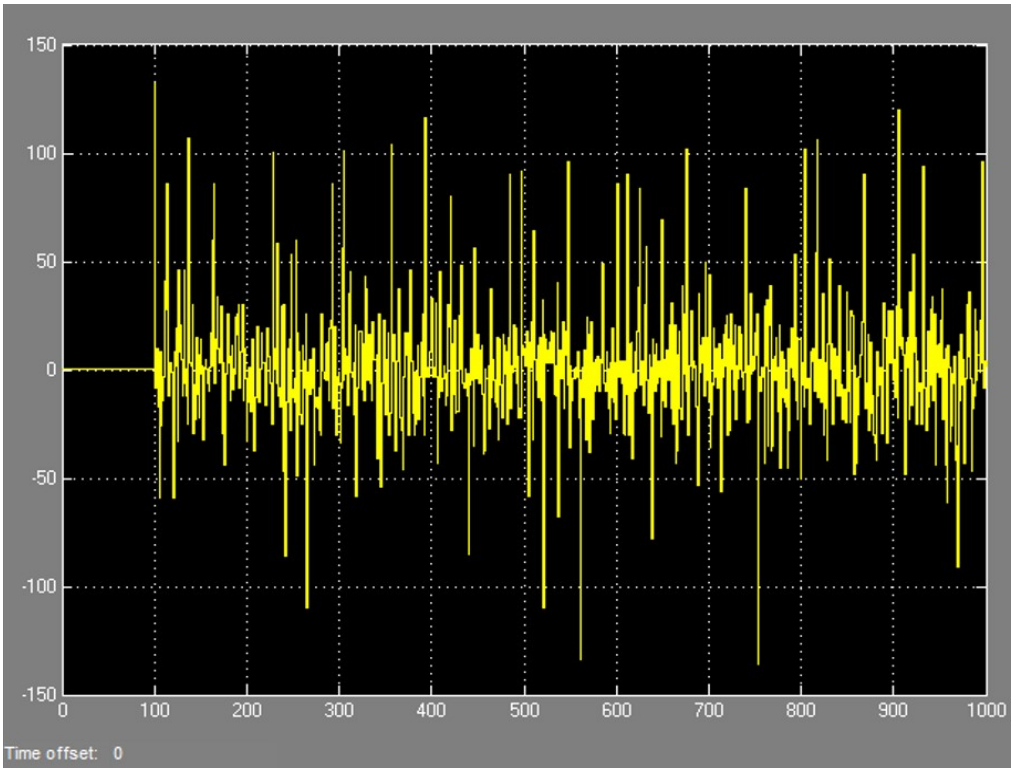
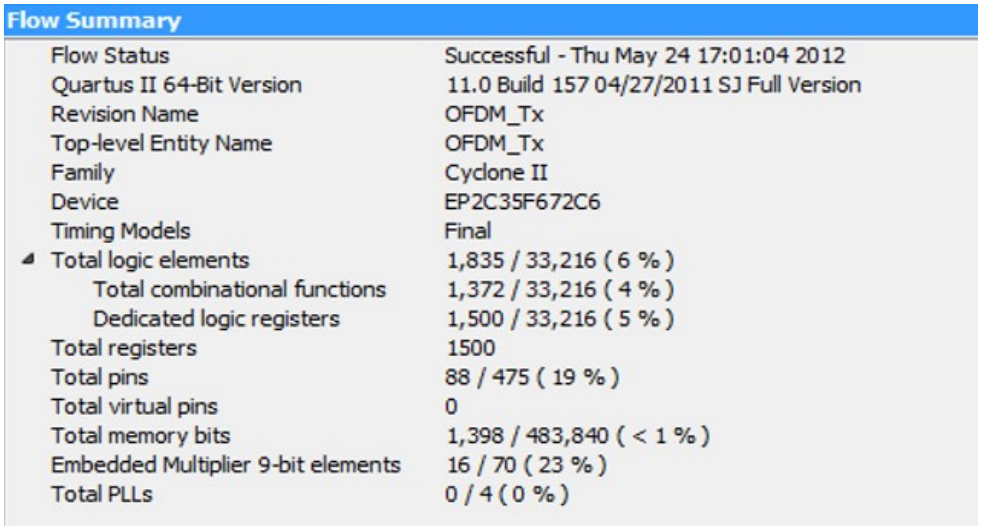


Figure 4-20 OFDM Transmitter -using FFT algorithm- Result

4.7.3 Compiling the OFDM Transmitter using FFT Algorithm with Quartus II Software

The OFDM Transmitter –using FFT algorithm- designed above has been analysed, synthesised, and frittered successfully and thus the HDL files has been created correctly, using the Quartus II software. It can be seen from the Figure 4.21 below that about 1372 combinational functions (4%) and 1500 (5%) registers and a total of 1835 LEs (6%) out of the available resources has been allocated on the Cyclone II FPGA chip.



Flow Summary	
Flow Status	Successful - Thu May 24 17:01:04 2012
Quartus II 64-Bit Version	11.0 Build 157 04/27/2011 SJ Full Version
Revision Name	OFDM_Tx
Top-level Entity Name	OFDM_Tx
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	1,835 / 33,216 (6 %)
Total combinational functions	1,372 / 33,216 (4 %)
Dedicated logic registers	1,500 / 33,216 (5 %)
Total registers	1500
Total pins	88 / 475 (19 %)
Total virtual pins	0
Total memory bits	1,398 / 483,840 (< 1 %)
Embedded Multiplier 9-bit elements	16 / 70 (23 %)
Total PLLs	0 / 4 (0 %)

Figure 4-21 Allocated Resources for OFDM Transmitter using FFT algorithm

4.7.4 Configuration the FPGA board to Implement the OFDM Transmitter using FFT Algorithm

The OFDM transmitter using FFT algorithm has been implemented successfully into the physical Cyclone II FPGA chip using the Quartus II software and USB cable as shown in Figure 4.22 below.

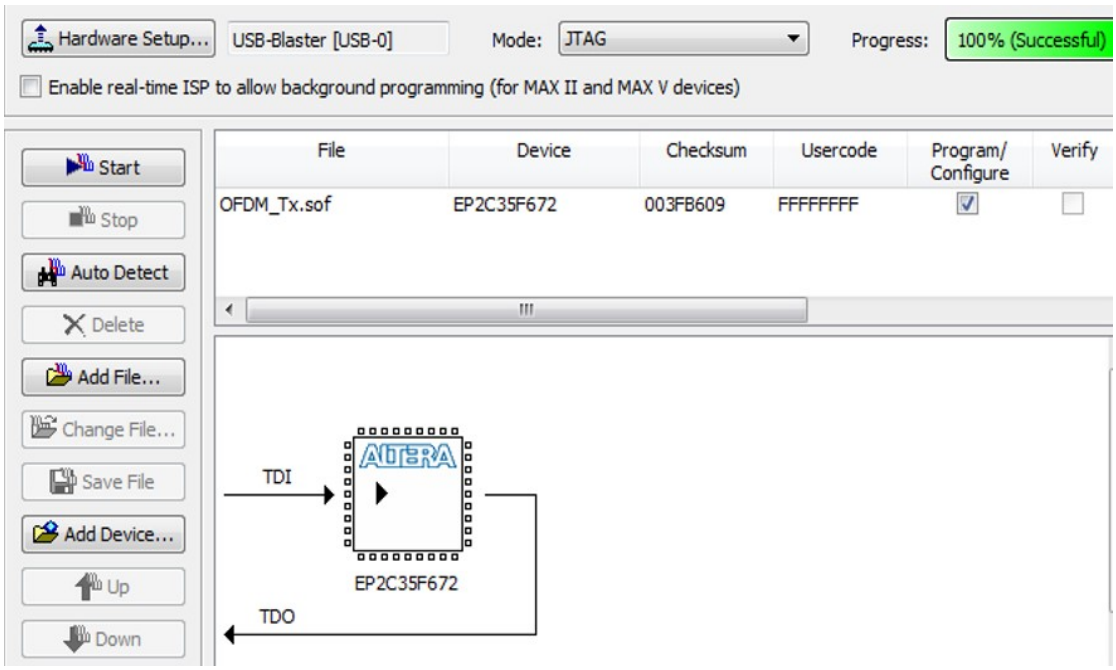


Figure 4-22 Configuration the OFDM Transmitter circuit on Cyclone II FPGA board

4.8 OFDM Receiver using FFT Algorithm Implementation

On receiver side, FFT, De-Block-Interleaver, and De-QAM mapper models are implemented with similar techniques that used IFFT, Block-Interleaver, and QAM mapper models in transmitter side but in reverse order. However, for channel coding block, a Viterbi Decoder is implemented in receiver side instead of convolutional encoder in transmitter side as explained in the next section.

4.8.1 Design of Viterbi Decoder

To provide error correction over a noisy communication channel, a convolutional encoder and Viterbi decoder at transmitter and receiver respectively are used [71].

By using a Viterbi Compiler provided by Altera, which consists of a soft-decision Viterbi decoder, area optimization, and high-performance Mega Core functions, a wide range of Viterbi decoders can be implemented. In addition to that, Altera provides two architectures for this purpose: parallel architecture and hybrid architecture
[71].

At receiver, a soft-decision Viterbi decoder with compatible parameters to convolutional encoder at transmitter is implemented as shown in Figure 4.23 below:

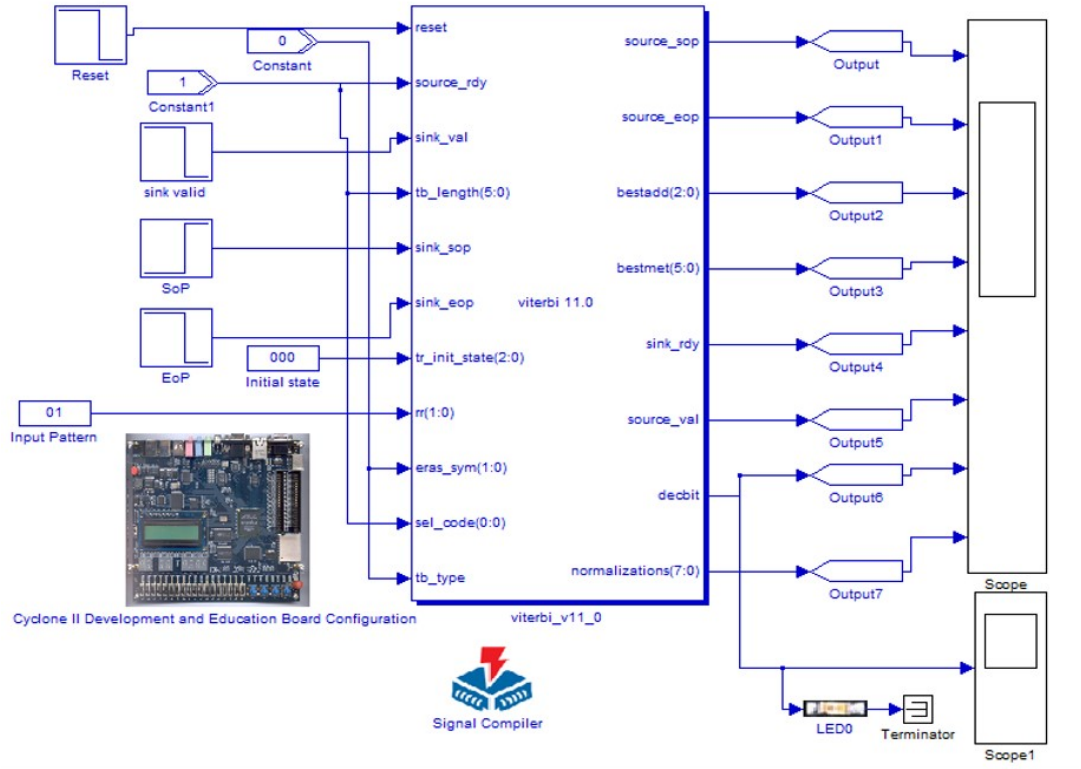


Figure 4-23 Soft-Decision Viterbi Decoder Design

4.8.2 Simulating the Soft-Decision Viterbi Decoder in MATLAB Simulink

Using the Parameterize block inside Viterbi Compiler a parallel Architecture of Viterbi Compiler is chosen, a Viterbi mode is used with coded bits equal to 2 bits and constraint length to 4, a 24 is assigned to the trace-back where the soft-bits are setting up to 1, and a 8dB is assigned to SNR.

A Viterbi decoder computes the branch matrix and path matrix for each incoming pair of bits and keeps on moving. The path of the highest matrix is selected by the decoder in the last stage and then it traces back.

The soft-decision Viterbi decoder circuit is simulated using MATLAB Simulink to visualize the result and to verify the functional correctness of the designed circuit as shown in Figure 4.23 below.

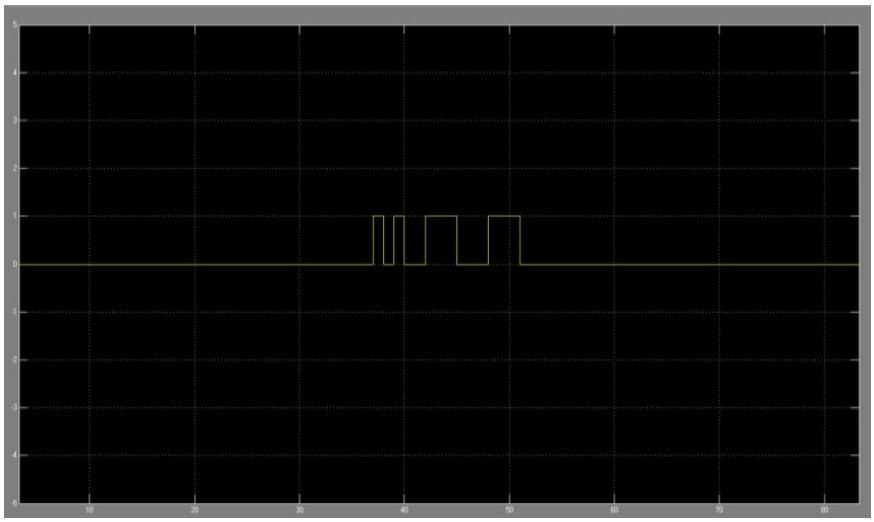
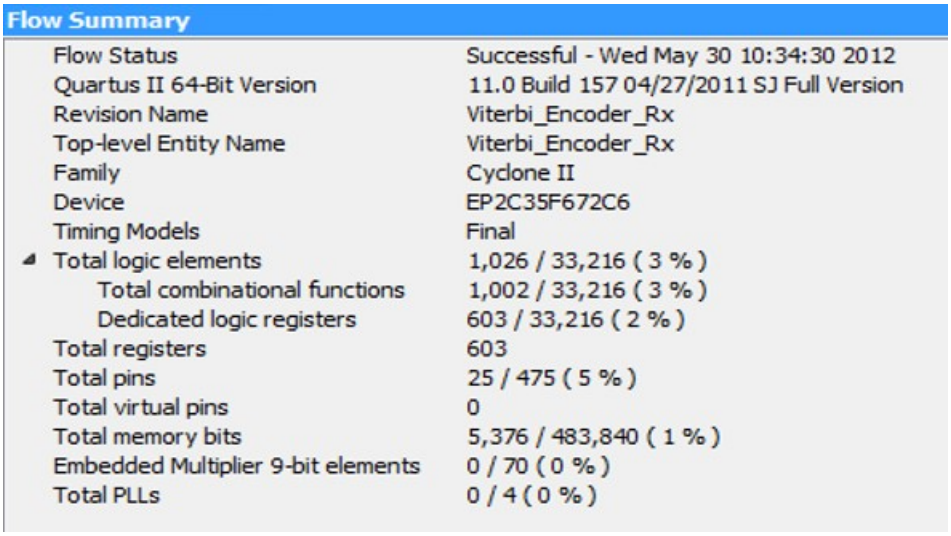


Figure 4-24 Soft-Decision Viterbi Decoder MATLAB Simulink Result

4.8.3 Compiling the Soft-Decision Viterbi Decoder with Quartus II Software

By using the Quartus II software, the HDL files for Viterbi Compiler model containing a soft-decision Viterbi decoder has been created smoothly after it has been analysed, synthesised, and fitted successfully. As shows in Figure 4.25 below, 3 %(1002) of the combinational functions and 2 %(603) of the logic registers inside the Cyclone II FPGA device have been allocated for the soft-decision Viterbi decoder designed in previous sections.



Flow Summary	
Flow Status	Successful - Wed May 30 10:34:30 2012
Quartus II 64-Bit Version	11.0 Build 157 04/27/2011 SJ Full Version
Revision Name	Viterbi_Encoder_Rx
Top-level Entity Name	Viterbi_Encoder_Rx
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	1,026 / 33,216 (3 %)
Total combinational functions	1,002 / 33,216 (3 %)
Dedicated logic registers	603 / 33,216 (2 %)
Total registers	603
Total pins	25 / 475 (5 %)
Total virtual pins	0
Total memory bits	5,376 / 483,840 (1 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 4-25 Allocated Resources for Soft-Decision Viterbi Decoder

4.8.4 Configuration the FPGA board to Implement the Soft-Decision Viterbi Decoder

The Viterbi Compiler using the soft-decision Viterbi decoder has been implemented successfully into the physical Cyclone II FPGA chip using the Quartus II software and USB cable as shown in Figure 4.26 below.

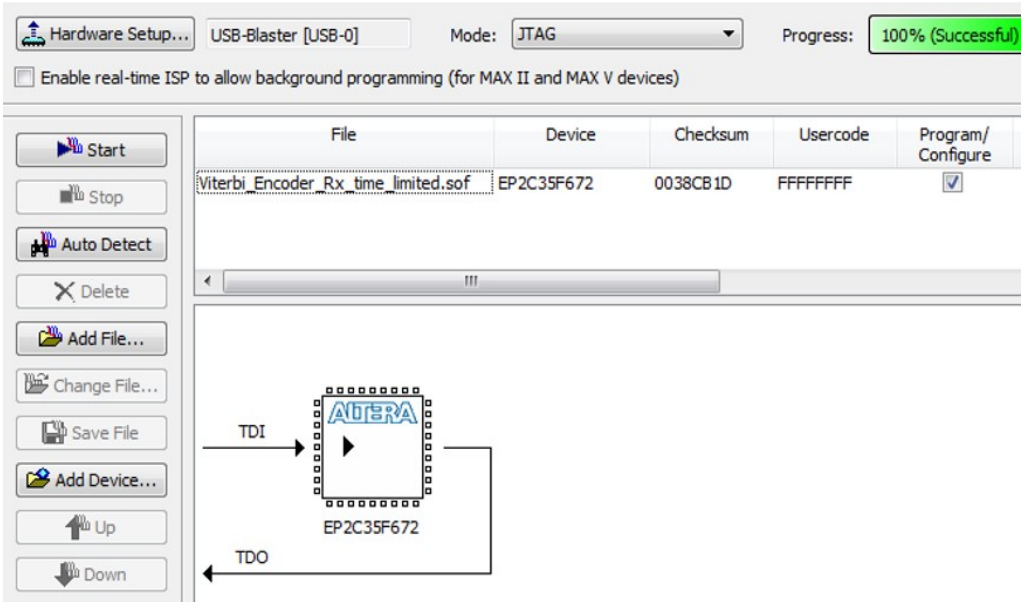


Figure 4-26 Configuration the Soft-Decision Viterbi Decoder on Cyclone II FPGA board

4.9 Full OFDM receiver

A full OFDM receiver using FFT algorithm was designed by integration of the following parts: 64-FFT, 16-QAM De-mapper, 2×2 block de-interleaver, and soft-decision Viterbi decoder of rates (2, 1, 4). Then output of the designed circuit is visualized and the functional correctness of the model is verified by using MATLAB Simulink. After that, the circuit was analysed, synthesised, and frittered, and the HDL files was created successfully by using the Quartus II software. Result of compiling the desired circuit using Quartus II software shows that 8% (2754) LEs out of the available resources was allocated on the selected FPGA device as shown in Figure 2.27 below.

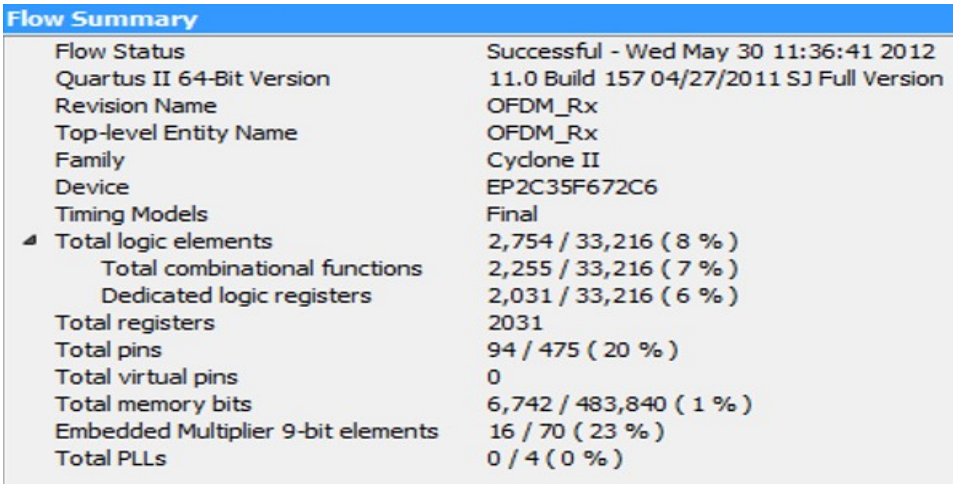


Figure 4-27 Allocated Resources for OFDM Receiver using FFT algorithm

And finally, the OFDM receiver using FFT algorithm was implemented successfully into the physical Cyclone II FPGA chip using the Quartus II software and USB cable as shown in Figure 4.28 below.

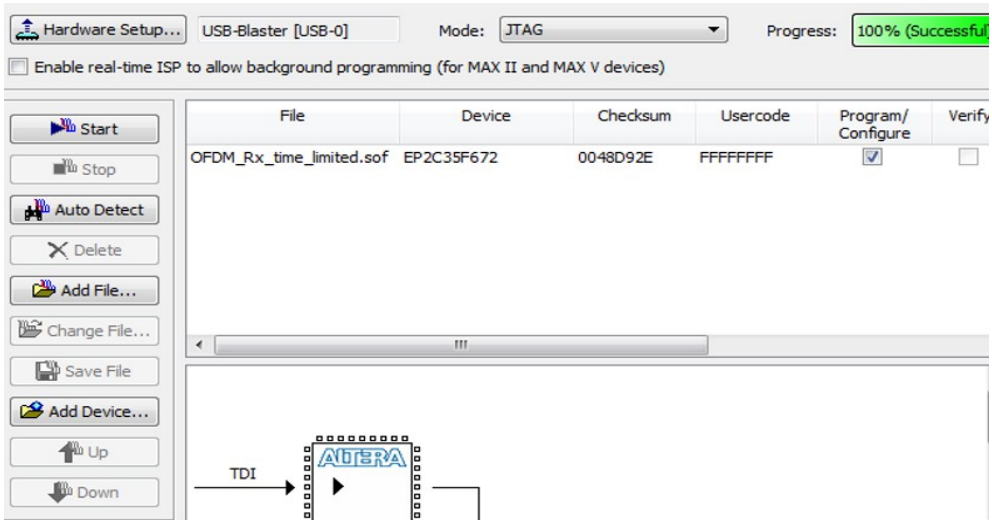


Figure 4-28 Configuration the OFDM Receiver using FFT algorithm on Cyclone II FPGA board

4.10 Conclusion

The basic design of the OFDM transmitter and receiver were validated and implemented onto FPGA successfully using Quartus II high level design tool. Altera IP Mega Core functions were used in this work such as FFT Mega Core function in IFFT/FFT modules and Viterbi Compiler in soft decision Viterbi decoder modules. The result shows that implementing the basic OFDM transceiver allocates about 14% (6% at transmitter and 8% at receiver) of the available logical elements (LE) resources on Altera Cyclone II FPGA chip, in which they mostly cover the FFT, IFFT and soft decision encoder.

Chapter 5

Channel Estimation for OFDM FFT/DWT in Multi-Carrier Modulation used in Wireless Telemedicine Applications

5.1 Introduction

This chapter proposes an OFDM system to be used in an ECG Patient Monitoring System (ECG-PMS) for wireless telemedicine applications. The proposed system is designed to support a low transmitted signal power at the patient site in the ECG-PMS, and a high BER performance at the hospital site in the ECG-PMS system. This is achieved by implementing a wavelet-based OFDM system that uses channel estimation based on Data-Aided (DA) technique, called FEC (Forward Error Correction coding) Decoded Pseudo Pilot (FDPP). Also in this chapter, the performance of wavelet-based OFDM system and Discrete Fourier Transform (DFT)-based OFDM system using FDPP-based as well as HDPP-based channel estimation techniques are compared. The system model was studied using MATLAB software in which the average BER was addressed for randomized data, and the difference between the transmitted ECG signal and the reconstructed one is used to measure the quality of ECG signal.

This chapter is organized as follows; Section 5.2 gives a literature review and proposed work. Section 5.3 explains the ECG-PMS system. Section 5.4 outlines the configuration of the basic OFDM transmission scheme. Section 5.5 evaluates the effects of AWGN and fading channel on the OFDM-(FFT/DWT) transmitted signal. Section 5.6 describes and investigates how the FEC and interleaving increase the BER performance of the OFDM-(FFT/DWT) transmission system. Section 5.7 describes types of pilot insertion. Further, it depictswhy the basic channel estimation technique is ineffective in the case of time-varying fading.Section 5.8 explains how the periodic channel estimation technique that is obtained by the improved DA technique overcomes this problem, without affecting the transmission efficiency. FDPP-DA based channel estimation is discuss in details in Section 5.9. The LLR calculation method usedin Soft Decision FEC is illustrated in Section 5.10. Evaluation of the proposed OFDM transmission system has been presented in Section 5.11. Finally, Section 5.12 concludes the outcomes of this chapter.

5.2 Literature Review

5.2.1 OFDM

The basic concept of OFDM is to divide high data rate streams into parallel, lower rate streams transmitted over a number of orthogonal subcarriers. Each subcarrier is orthogonal to the others and carries a portion of the transmitted

information. Hence, OFDM is different from the commonly used Frequency Division Multiplexing (FDM). OFDM is a special case of multicarrier modulation (MCM) that divides a communication channel spectrum into a number of equally spaced frequency bands, a modulation or multiplexing technique. The main reason for using OFDM is to increase robustness against frequency selective channels and narrowband interference. In single carrier systems, one fade or interferer can cause an entire link to fail, but in multicarrier systems only a small percentage of the subcarriers will be affected[4][62].

5.2.2 OFDM-FFT versus OFDM-DWT and OFDM-WPT

OFDM design was proposed to overcome Inter Carrier Interference (ICI), with the Discrete Fourier Transform (DFT) ensuring subcarrier orthogonality, using Fast Fourier Transform (FFT) algorithm. However, a high level of side lobes of FFT rectangular window leads to a high level of interferences and lower level of performance in under multipath fading conditions. A guard interval was proposed to overcome the delay spread of the channel through inserting a cyclic prefix (CP), which requires bandwidth and power and so reduces the spectral efficiency of the transmitted signal[4][62].

A Wavelet OFDM design was proposed to overcome the limitations of the conventional OFDM design, where in the OFDM system the FFT is replaced by

the Discrete Wavelet Transform (DWT), which provides a higher suppression of sides lobes compared to the FFT. Thus, in Wavelet OFDM there is no need for the CP, unlike conventional OFDM [4][72]. In conventional OFDM systems, signals overlap in the frequency domain only, where as in Wavelet OFDM systems, signals overlap in both the frequency and time domains. The use of wavelet in OFDM systems reduces both ICI and Inter Symbol Interference (ISI) [4].

5.2.3 OFDM FFT/DWT performance evaluation Review

Much research effort has been focused to assess OFDM-FFT and OFDM-DWT system performance, and to compare both systems employing Zero-Forcing Equalization. The impact of channel models over DFT-OFDM and DWT-OFDM systems using Binary Phase Shift Keying (BPSK) modulation is given by [73]. The results show that the BER performance of both OFDM systems differ over different transmission media; AWGN, Rayleigh fading, and multipath fading channel. the use of DWT with the OFDM is extended to a new form called Wavelet Packet Transform (WPT)[4].The performance of FFT-OFDM and WPT-OFDM systems over AWGN channel is compared in [74]for various QAM constellation points. The results show that the best BER performance gain of WPT-OFDM system over FFT-OFDM system when the number of QAM modulation is 8. The Performance of Wavelet OFDM system using the Haar wavelet has been investigated in [4] over AWGN channel. The results shows that

the DWT based OFDM lost its multi-carrier characteristic and the WPTis used to overcome this problem. In[75]it has been reported that Wavelet OFDM has better BER performance over FFT-OFDM in flat Rayleigh fading channel. Furthermore , the impact of the mothers' wavelets and the number of iterations used in the computation of Wavelet-OFDM are investigated in [75]. The influence of the wavelets mother on the BER performance of a Wavelet OFDM in frequency selective fading channel is evaluated in [76]. The results reveal that the wavelet having best frequency localizationperforms better.

5.2.4 Channel Estimation

Channel estimation is the evaluation of the effect of the wireless channel on the received signal. In order to mitigate hostile channel effects on the received signal, precise channel estimation is required to provide information for further processing of the received signal. Channel estimation can be categorized as non-data-aided and data-aided. Non-data-aided or blind channel estimators obtain the channel response from the statistics of the received signals. No specialized reference (or training signal) is needed and the transmission efficiency is retained. However, without precise knowledge of the transmitted signals, a large number of data must be collected in order to obtain reliable estimation. Data-aided channel estimators require known reference (training) signals to be transmitted, thus occupying bandwidth. Rapid and accurate channel estimation can be achieved by comparing the received signal with reference

signals. A sufficient number of such reference signals must be inserted according to the degree of channel variation, namely coherence time and coherence bandwidth of the channel under estimation [5].

5.2.5 Channel Estimation for OFDM-FFT/DWT e Review

Other research has been carried out to assess and compare the training-based channel estimation algorithms performance for OFDM systems using either FFT or DWT, where pilot symbols are sent periodically. A Walsh coded training signals was proposed in [77] to be orthogonal in time domain, so the Channel Impulse Response (CIR) in MIMO-OFDM system is calculated without calculating the FFT/iFFT. As a result, the computational complexity of the used channel estimator is reduced. At their paper, the performance of the OFDM system is measured by the Least-Squares (LS) channel estimator's mean square error (MSE) and BER. A performance evaluation of wavelet-based OFDM and conventional-based OFDM systems using Linear Minimum Mean Square Error (LMMSE) based channel estimation algorithm as well as Least Square (LS) based channel estimation algorithm has been conducted in [5].

The periodically training sequence necessitates wasteful transmission, which reduces the spectral efficiency. To overcome this problem, improved Data-Aided (DA) based channel estimation has been proposed[1][78][79] which is explained later.

Essentially, there are two improved data-aided based channel estimation algorithms: FDPP and Hard Decision Pseudo Pilot (HDPP). The author's previous research proposed FDPP-DA channel estimation and focused on the performance comparison of FDPP-DA channel estimation and HDPP-DA channel estimation in DFT-based OFDM systems under AWGN and Rayleigh fading conditions [1].

5.2.6 Telemedicine

Monitoring and checking one's health has become very important with many people going for regular checks to ensure a good health status. In an effort to provide more effective health monitoring systems, e-health is being developed to monitor health constantly without the need for physical attendance at a health care centre. E-health involves the exchange of information or services from patient to a health care centre or hospital for logging and possible further investigation of the received data. In view of this, a reliable communication link is required to transmit the data to the health care centre in near real-time [80].

Telemedicine is a term that involves remotely providing treatments to a patient by a specialist who is not physically present in the patient's location, a particular aspect of e-health. It uses telecommunication and computer network technology to increase and aid better health services in medicine. Performed wirelessly

through e-conference or e-video calls, it involves the use of both mobile and multimedia data/images [81].

Due to the broadband limitations in our wireless communication systems, effective communication of such services is usually limited by inter symbol interference (ISI) and selective frequency fading. OFDM tends to minimize or eliminate these challenges, and this seems very apt for use in wireless telemedicine applications[6][7][81].

The OFDM is used in sharing biomedical data over hybrid wireless networks with little limitation in bandwidth and power requirements. Conditions such as path loss, fading, co channel interference and noise disturbances which all have very severe impacts on the wireless systems contribute to the capacity of the signals been much lower than wired channels and higher BER. The signals are primarily transmitted using Bluetooth, ZigBee and satellite communications systems amongst others in telemedicine.

OFDM technique tends to reduce the challenges of the previous systems used in the links for an effective channel of communication. OFDM is regarded as the best suitable method because high data can be achieved this way and is best suited for obtaining reliable reception of signals affected by ISI and selective frequency fading as previously mentioned [81].

5.2.7 Using of OFDM in Telemedicine Review

A coded OFDM based FFT (COFDM-FFT) that can be adapted for wireless telemedicine has been proposed in [81]. The performance of the proposed COFDM-FFT system is compared with un-coded OFDM based FFT system for various modulation schemes over AWGN and Rayleigh channel. An OFDM-FFT based baseband transceiver has been proposed in [7] for Ubiquitous healthcare monitoring applications to achieve an energy-efficient Wireless Body Area Network (WBAN) solution. The system evaluation has done by establishing an ECG transmission platform in the proposed design. To reduce the design complexity, a zero forcing equalization is chosen. An OFDM-FFT Based Cognitive Radio (OFDM-FFT-CR) system has been proposed in [6] for wireless telemedicine application to achieve the reliable transmission and reception for ECG signals in the presence of interference, where the cognitive radio has been chosen to utilize the available spectrum efficiently. The telemedicine system consists of a mobile telemedicine unit at emergency site and base station at the expert/hospital site.

5.2.8 Proposed Work

Telemedicine technology used at the patient site is desirable, whereby critical parameters such as ECG signals are transmitted for remote monitoring of the patient. It provides medical experts with continuous ECG information about the

patient, thereby enabling doctors to detect emergencies, especially valuable for elderly and dependent patients, then sending an alarm signal to the Patient Location Tracking System (PLTS) proposed in [82]. Reducing the transmission energy of the portable wireless body transmitter at patient site is desirable to achieve a long duration monitoring. Increasing the quality of the received critical ECG signal at the hospital site is also desirable to achieve a more accurate picture of patients' situations[6][7].

There do not appear to be reports of wireless communication systems used outside hospitals which satisfy the above two requirements. In order to respond to this situation, a framework for the real time remote monitoring of patient heart health based on a platform that consist of a portable wireless body transmitter unit at the patient site and a base unit at the medical-experts/hospital site would be presented in this paper. The primary aim is to establish a low power transmitted signal from the remote ECG monitor, with high BER at the receiving end, with reliable and stable transmission and reception of biomedical signals over wireless channels with little or no adverse effect from interference and noise. The system uses wavelet-based OFDM and FDPP-based channel estimation. This proposed OFDM system can be integrated into the ECG Patient Monitoring System (ECG-PMS).

System performance has been investigated, with the effects of AWGN and multipath Rayleigh fading channels included in the analysis. The performances of

FDPP-DA-based channel estimation and HDPP-DA-based channel estimation for DFT-based OFDM system as well as Wavelet-based OFDM systems are compared.

5.3 ECG-PMS Model

The main components of the ECG Patient Monitoring System (ECG-PMS) used in Personal Wireless telemedicine systems are:

1) Portable wireless transmitter at the patient site
    a) biomedical ECG sensor, memory, battery;
    b) sample and hold wavelet based OFDM transmitter;
    c) RF transmitter.
2) Base unit at the central site:
    a) RF Rx;
    b) wavelet based OFDM with FDPP-DA channel estimation receiver,
    c) host PC with telemedicine software to analyse the reconstructed ECG signal.
    d) medical experts.

The basic model of the proposed system is shown in Figure **5-1**, the patient site, with the hospital/ medical experts' site is shown in Figure **5-2**. The biomedical sensor acquires the ECG signal from the patient, the data is digitized and fed to the OFDM-DWT transmitter.

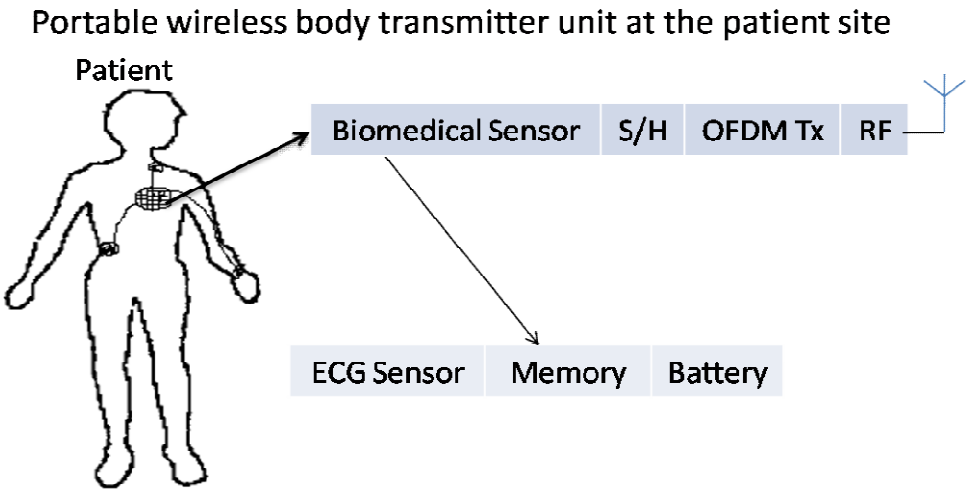


Figure 5-1 ECG Patient Monitoring System (ECG-PMS) - patient site

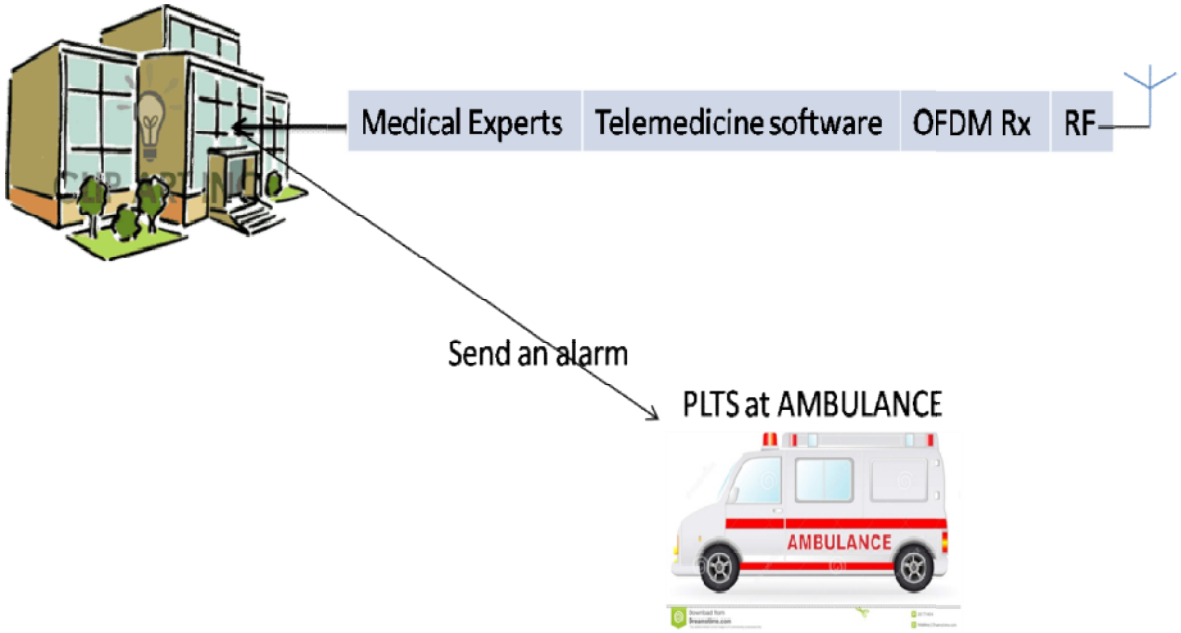


Figure 5-2 ECG Patient Monitoring System (ECG-PMS)-hospital/ medical experts' site

5.3.1 Patient site

At the patient site, the main criterion is power consumption because the device is portable. For longer operation time, an energy-efficient of the proposed system is needed. This requirement can be achieved by:

1) Reducing the active duration by increasing the date rate. The burst mode is used to transmit the body signals, because the transmission rate is usually higher than the rate at which body signals are acquired [7]. Figure **5-3** shows the system time line. To reduce the active period, the proposed system adopts OFDM to get both a megabit per second (Mbps) data rate and reliable transmission with narrow bandwidth.

2) Reducing the transmitted power without affecting the BER performance. Thus, the proposed system adopts OFDM-DWT using FDPP-based channel estimation to get both a high BER performance with low SNR.

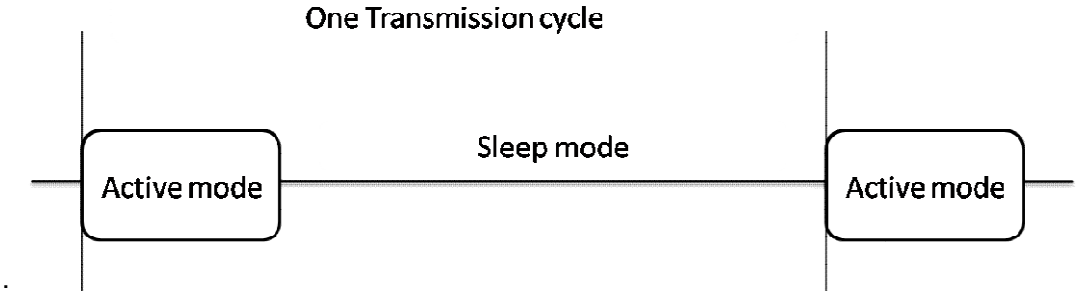


Figure 5-3Behavior Time Line

5.3.2 Hospital site

At the hospital site, the main criterion is the quality of the reconstructed ECG signal, to ensure correct decision making, especially if an emergency arises and

an alarm signal is required to alert the PLTS at the ambulance site. Thus, the proposed system adopts OFDM-DWT using FDPP-based channel estimation to get higher BER performance.

The details of the proposed OFDM system is explained in details in next sections.

5.4 Configuration of the Basic OFDM Transmission Scheme

5.4.1 Transmitter

A high speed serial data stream is serial to parallel converted into M sequences in the nthinterval. Then the parallel data is mapped onto a constellation diagram of any digital modulation technique, such as BPSK, QPSK, 8PSK, 16- QAM and 64- QAM. Next, the modulated signals are frequency division multiplexed by either Inverse Discrete Fourier Transform (iDFT) circuitor by Inverse Discrete Wavelet Transform (iDWT). iDFT can be computed efficiently in practice using an inverse Fast Fourier Transform (iFFT) algorithm. Also by using iFFT or iDWT, orthogonally between subcarriers in each OFDM symbol is allowed. Then, the iFFT/iDWT output samples form an OFDM signal [2][72]. " In practice , these samples are not enough to make a real OFDM signal .The reason is that there is no oversampling present , which would introduce intolerable aliasing if one would pass these samples through a digital to analog converter" [3] . A solution is adding a number of zeros to the modulated input data [3]. Input of iFFT/iDWT block is (m,n) , for $m = 1,2,\dots,M+K$, and $n = 1,2,\dots.,N$. Where M is the

number of subcarriers data in each OFDM Symbol, K is the number of zero subcarriers (null data) in each OFDM symbol and N is the number of OFDM symbols in data field. To present an oversampling, the K zeros data in each OFDM symbol must be added in the middle of that symbol. This is to ensure that the K zeros data are mapped onto frequencies close to half the sampling rate, and the M non-zero data are mapped onto the subcarriers around $0\ Hz$[2],[3]. To conclude, $X(m.n)$, for $m = 1,2,\ldots..,M+K$, and $n = 1,2,\ldots.,N$ are fed into allocated subcarrier-channels, then multiplexed with N_f samples in an iFFT or with L decomposition stages in iDWT circuit to produce an OFDM signal [2][4][79],, where N_f= M +K, L= $\log_2$(M+K) . Finally, the OFDM signal is in time domain in iFFT case and wavelet domain in iDWT case, so it just needs to convert to serial to be ready to transmit to air [2][56][72].A basic OFDM system is shown in Figure **5-4**.

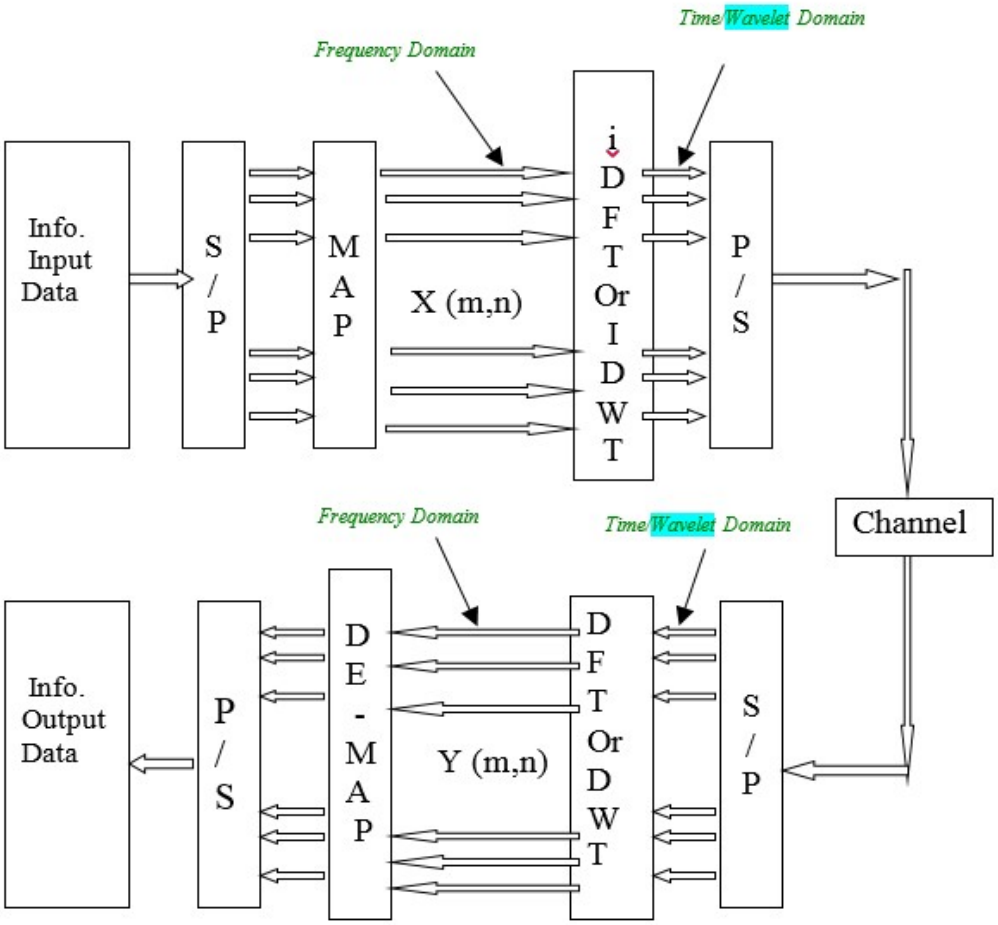


Figure 5-4 Basic OFDM transmission scheme

5.4.2 Receiver

The serial signal is converted to parallel, then fed into FFT/DWT circuit and de-multiplexed as a subcarrier–channel data to recover the received subcarrier data $Y(m,n)$for $m = 1,2,\ldots..,M+K$, and $n = 1,2,\ldots.,N$. At both transmitter and receiver, iFFT and FFT must have the same $N_f$sample point, iDWT and DWT have same L reconstruction/decomposition stages. Now the data in frequency

domain, so K zeros data that is added at transmitter is removed.After that the received subcarriers data are demodulated by the same modulation technique used at transmitter. Finally, it is converted from parallel to serial to recover the data stream[2][62].

5.5 Source Data Channel

Source data that is generated at transmitter travel to receiver through a mobile radio communication channel. During this trip, data is affected adversely by noise and fading as would follow see in subsequent subsections.

5.5.1 AWGN Channel Effect

The above configuration assumes that the data transfers through an ideal channel, (5.1) is true for $m = 1,2,\ldots..,M$, and $n = 1,2,\ldots..,N$. [2].

$$Y(m,n) = X(m,n) \tag{5.1}$$

However, in practice there is no ideal channel, and various noises such as AWGN influence the channel. Hence,$Y(m,n)$can be represented by (5.2)[2][79]:

$$Y(m,n) = X(m,n) + W(m,n) \tag{5.2}$$

Where $W(m,n)$ is the AWGN variable with independent In-phase (Ich) and Quadrature-phase (Qch) components, so $W(m,n) = Wi(m,n) + jWq(m,n)$ .The OFDM system with AWGN is shown in Figure **5-5**.

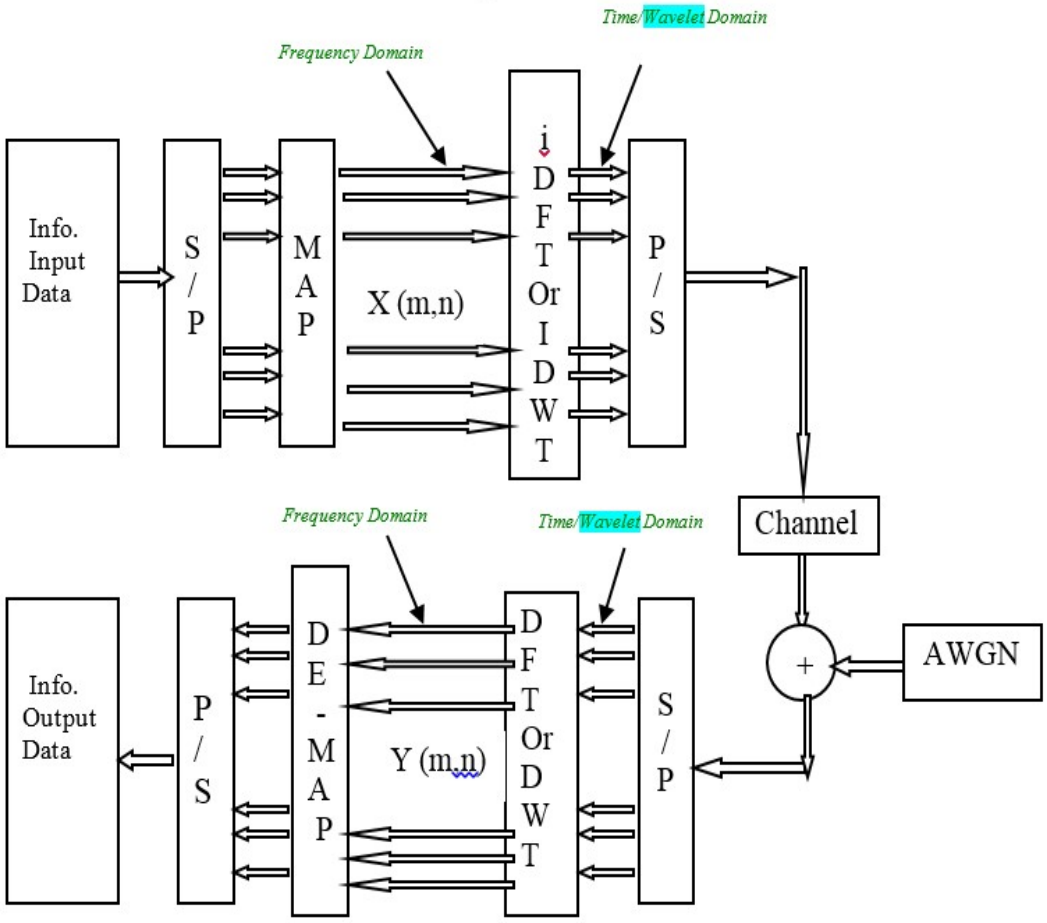


Figure 5-5 Basic OFDM transmission scheme under AWGN

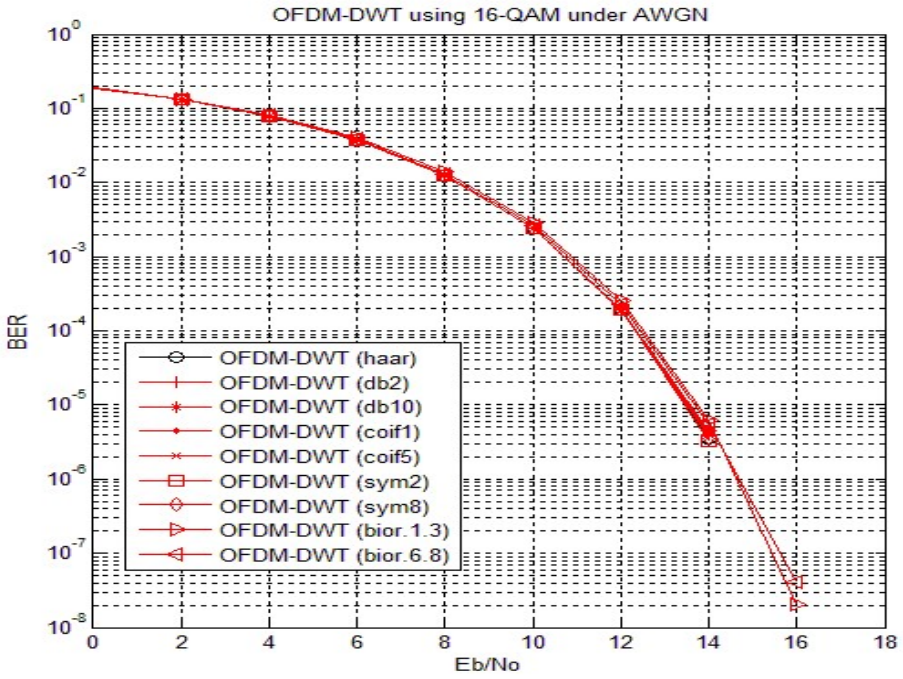
Experiment 5-1

A computer simulation is used to evaluate the transmission scheme described in Figure **5-5** in terms of average BER. Table **5-1** shows the simulation parameters,

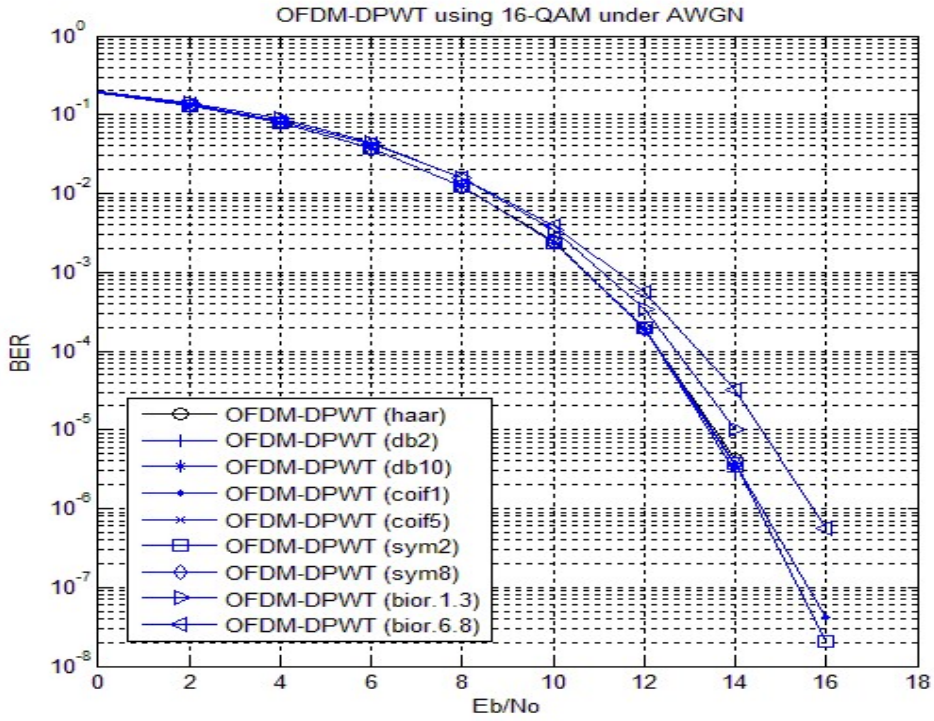
where for each value of (E_b/N_0), the system runs loop times. The average BER as a function of E_b/N_0 under AWGN conditions is shown in Figure **5-6**

Table 5-1 Simulation parametersof OFDM system under AWGN effect

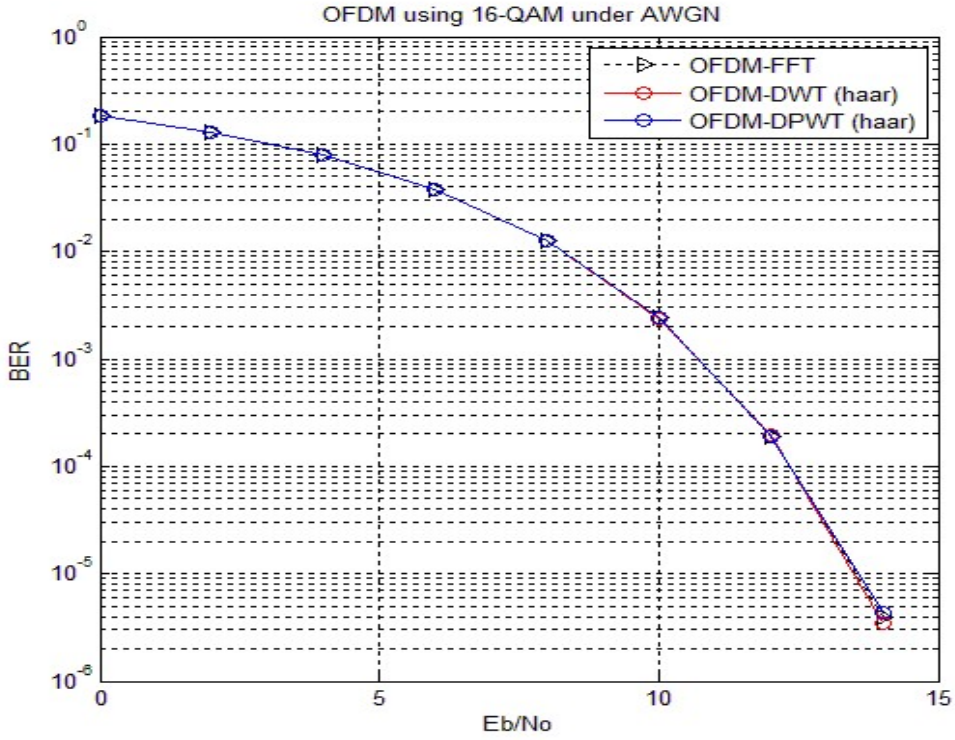
Modulation	16-QAM
Number of data subcarrier	48
Number of null subcarrier	16
Discrete Transform	FFT where Nf =64 DWT where decomposition stages = log2(64)= 6
E_b/N_o	0,2,4,…, and 20
Number of OFDM symbol	1000
Loop	500
Propagation Model	AWGN



(a)


OFDM-DPWT using 16-QAM under AWGN
BER
Eb/No
OFDM-DPWT (haar)
OFDM-DPWT (db2)
OFDM-DPWT (db10)
OFDM-DPWT (coif1)
OFDM-DPWT (coif5)
OFDM-DPWT (sym2)
OFDM-DPWT (sym8)
OFDM-DPWT (bior.1.3)
OFDM-DPWT (bior.6.8)

(b)


OFDM using 16-QAM under AWGN
BER
Eb/No
OFDM-FFT
OFDM-DWT (haar)
OFDM-DPWT (haar)

(c)

Figure 5-6 Average BER versus E_b/N_0 for OFDM-FFT/DWT/DPWT

It can be seen form Figure **5-6** (a) that BER performance is slightly similar for all DWT families in the OFDM-DWT case with 16-QAM under AWGN. However, in the case of DPWT, the DPWT (haar) has a slightly better BER performance among other families as shown in Figure **5-6** (b). Figure **5-6** (c) shows that 16-QAM-OFDM performance in terms of BER for FFT/DWT/DPWT in AWGN channel performed much in the same way.

5.5.2 Fading Channel Effect

Wireless channel experiences different detrimental effects such as ISI and ICI in addition to other noise sources of [5]. In this scenario a fading channel exists, characterized by a Rician or Rayleigh distribution.Figure **5-7** shows the OFDM transmission scheme under AWGN and fading.

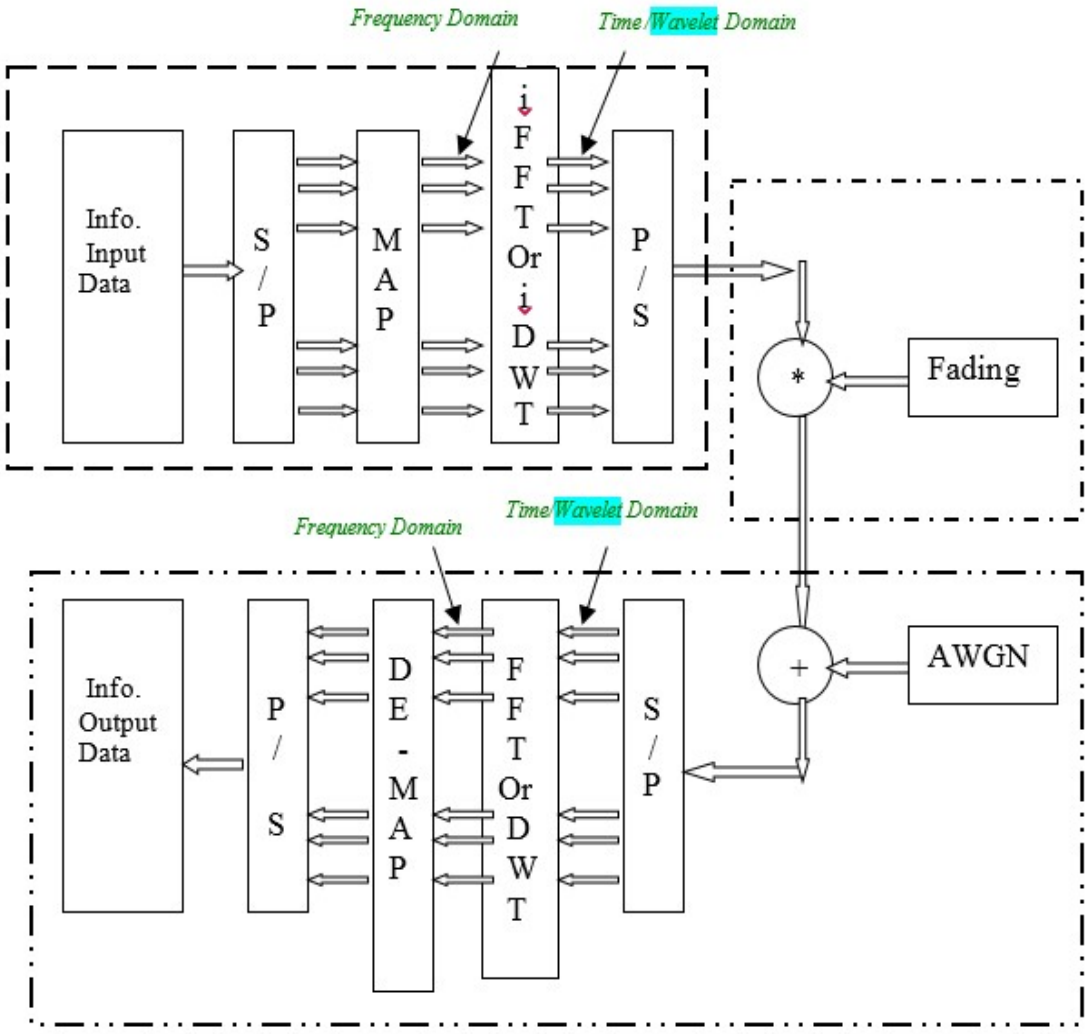


Figure 5-7 Basic OFDM transmission scheme under AWGN and Fading channel

The $Y(m,n))$, with AWGN in a fading channel can be represented by

(5.3)[2][79]:

$$Y(m,n) = X(m,n).H_r(m,n)$$

$$+\sum_{g=1}^{M} X(g,n-1).H_S(g,n)$$

$$+\sum_{g=1}^{M} X(g,n).H_C(m,g,n)$$

$$+W(m,n) \tag{5.3}$$

Where: $H_r(m,n)$ denotes the real channel frequency response, $H_S(g,n)$ denotes the component of ISI, $H_C(m,g,n)$ denotes the component of ICI, and $W(m,n)$ is the AWGN variable.

Experiment 5-2

A computer simulation was used to evaluate the transmission scheme described in Figure **5-7** in terms of average BER.Table **5-2** shows the simulation parameters, where for each value of ($\mathrm{E_b/N_0}$), the system runs loop times. The average BER as a function of $\mathrm{E_b/N_0}$ under AWGN and one-path Rayleigh fading conditions is shown in Figure **5-8**.

Table 5-2 Simulation parametersof OFDM system under AWGN and fading effects

Modulation	16-QAM
No.of data subcarrier	48
No. of null subcarrier	16
Discrete Transform	FFT where Nf =64 DWT where decomposition stages L = log2(64)= 6

E_b/N_o	0,2 , 4,…, and 20
Number of OFDM symbol	1000
OFDM symbol duration	4.8 μ Sec in case OFDM-FFT 4.0 μ Sec in case OFDM-DWT
Propagation Model	AWGN , one path Rayleigh fading
Max. Doppler frequency (f_D)	0 and 50 Hz
Loop	500

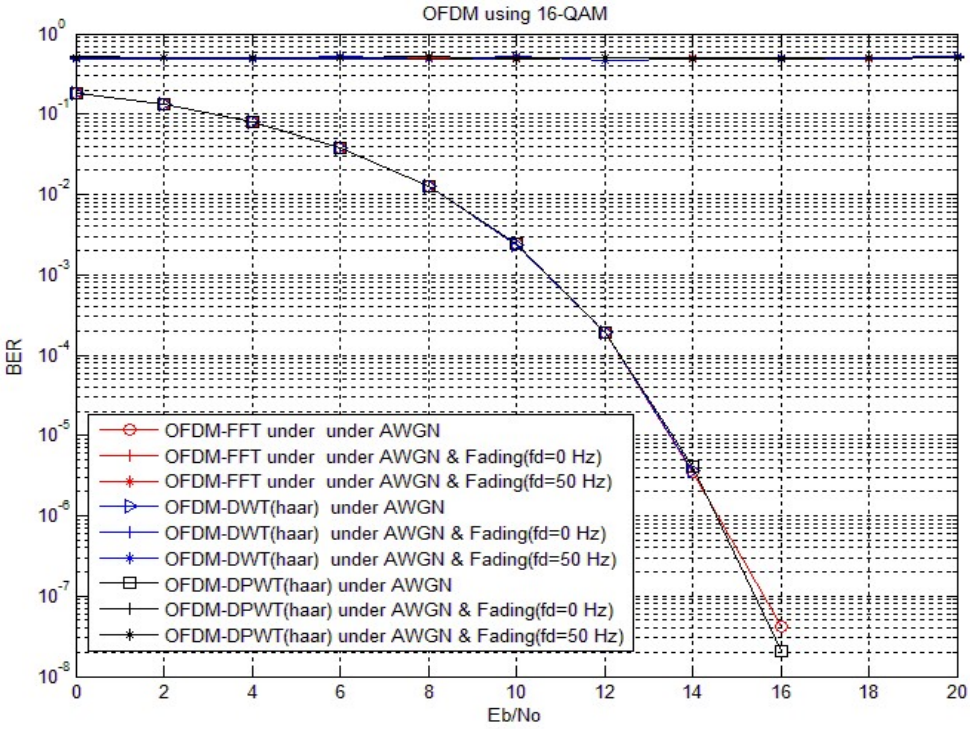


Figure 5-8 Average BER performance of Basic OFDM transmission scheme versus E_b/N_0

Figure **5-8**demonstrates that the average BER is close to 0.5 for both OFDM-FFT and OFDM-DWT when subjected to a fading channel conditions,and apparantly deep fades occur in the fading channel casegiving rise to the probability of losing

few subcarriers completely, and as such dominates the overall BER largely, leading to having a bit-error probability close to 0.5[2][3]. f_D =0 signifies that there is a fading, but no mobility.

5.6 FEC and Interleaving

5.6.1 Effect of FEC on OFDM System

The signal strength is varied from its normal value due to various impairments in radio channel such as noises interference, channel coding is essential. Channel coding inevitably adds redundant bits to an information sequence to facilitate error detection/correction due to channel distortion. Among others, Forward-error-correction coding (FEC) techniques such as block codes, convolutional codes, and turbo codes is used for channel coding. Atthe transmitter, the information data is encoded by one of FEC methods and then decoded at the receiver. Decoding can be done by hard-decision or soft-decision [2][3][83].Figure **5-9**shows the OFDM transmission scheme employing decoding block. The implementation of a soft-decision decoding is explained in the upcomingsections.

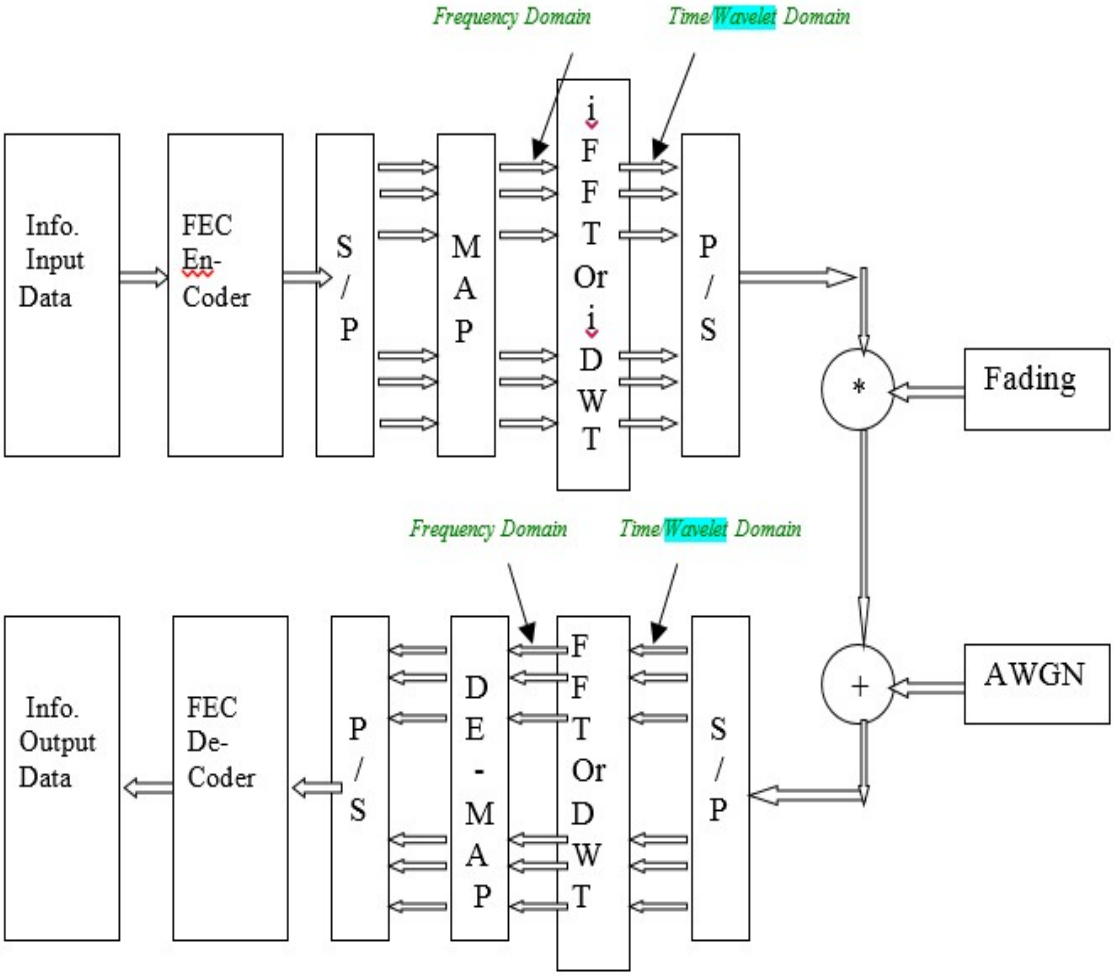


Figure 5-9 Basic OFDM transmission scheme employing FEC under Fading and AWGN

Experiment 5-3

A computer simulation has been used to gauge the quality of the system described in Figure **5-9** in terms of BER. Table **5-3**outlines the simulation parameter values, where for each value of (E_b/N_0), the system runs loop times. The average BER as a function of E_b/N_0 under AWGN is shown in Figure **5-10**.

Table 5-3 Simulation parametersof OFDM system employing FEC under AWGN and Doppler frequency effect

Modulation	16-QAM
FEC	convolutional codes: (a) un-coded (b) (c) (d) a constrain length 7 convolutional code with code rate =1/2, 1/3, and =1/4 respectively
No.of data subcarrier	48
No. of null subcarrier	16
Discrete Transform	FFT where Nf =64 DWT where decomposition stages L = log2(64)= 6
E_b/N_o	0,2 , 4,…, and 20
Number of OFDM symbol	1000
OFDM symbol duration	4.8 μ Sec in case OFDM-FFT 4.0 μ Sec in case OFDM-DWT
Propagation Model	AWGN
Loop	500

A coding gain is defined as the decrease in E_b/N_0 to obtain a certain BER performance when applying channel coding. Figure **5-10** shows that for OFDM-DPWT, the coded link with code rate ¼ have 3 dB code gain compared with that of the un-coded link for the same BER criterion of 10^{-5}, and about 1 dB code gain compared with the coded one having a code rate of ½. The BER performance is the same for OFDM-FFT/DWT/DPWT for a code rate of ¼. A hard decision Viterbi decoder has been used at the receiver side in this experiment.

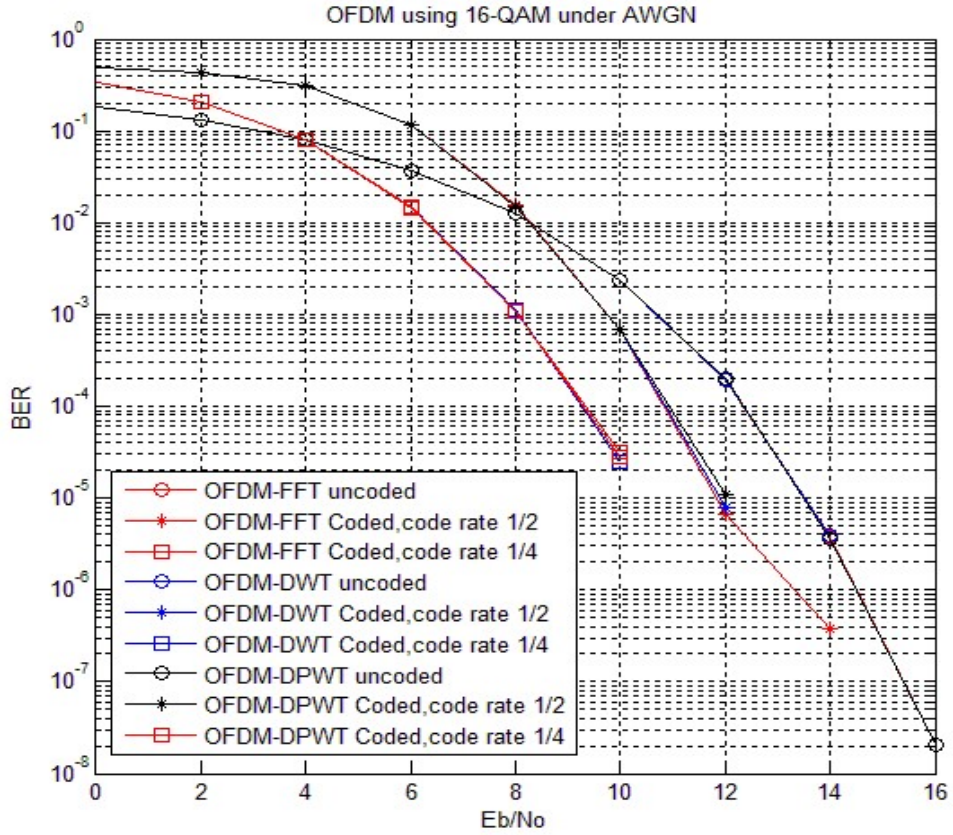


Figure 5-10 Average BER of OFDM employing FEC versus E_b/N_0 under AWGN

5.6.2 Interleaving Effect on OFDM System

Channel errors occur due to fading. These errors could be random or burst errors. EmployingFEC codingwould be too complex to counteract burst errors. Hence, FEC codes are not designed to deal with error burst, and there is a certain number of bits that can be corrected by FEC code. An interleaving is applied to transform burst errors into random errors before it is fed into the decoding circuit at the receiver. There exists a number of interleavingtechniques,such as block interleaving, (Pseudo) random interleaving,

and convolutional interleaving.At the transmitter, the encoded data is interleaved by one of Interleaving methods and then de-interleaved at the receiver before decoding circuit [2][3]. Figure **5-11** shows the OFDM transmission scheme employing interleaving circuit.

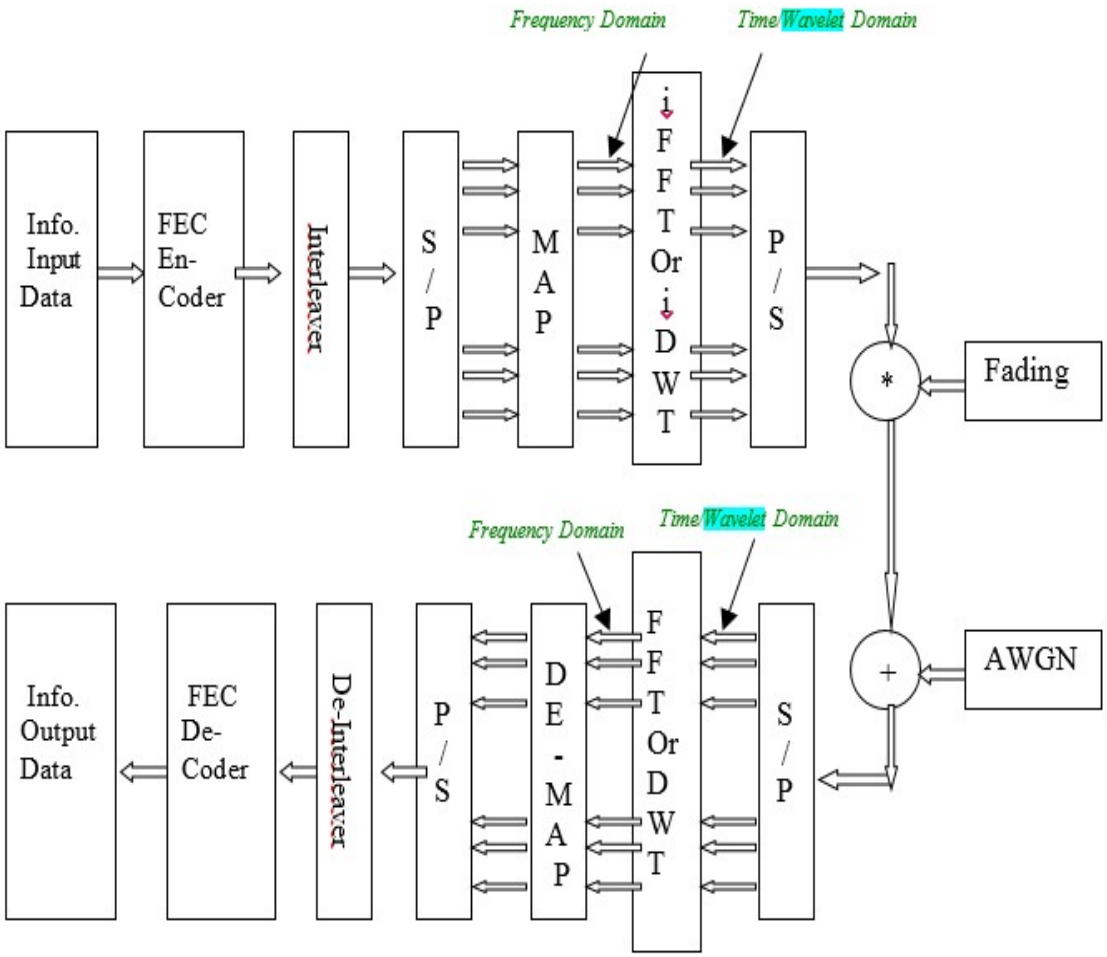


Figure 5-11 Basic OFDM transmission scheme employing interleaving under AWGN and Fading channel

Experiment 5-4

To assess the BER performance of the system depicted in Figure **5-11**,it has been simulated by means of a computer program.Table **5-4**lists the simulation parameter value, where for each value of (E_b/N_0), the system runs loop times. The average BER as a function of E_b/N_0 under AWGN is shown in Figure **5-12**.

Table 5-4 Simulation parametersof OFDM system employing Interleaving under AWGN and Doppler frequency effect

Modulation	16-QAM
FEC	convolutional codes: a constrain length 7 convolutional code with code rate =1/2
Interleaving type	No interleaving Interleaving with Block interleaving
No.of data subcarrier	48
No. of null subcarrier	16
Discrete Transform	FFT where Nf =64 DWT where decomposition stages L = log2(64)= 6
E_b/N_o	0,2 , 4,…, and 20
Number of OFDM symbol	1000
OFDM symbol duration	4.8 μ Sec in case OFDM-FFT 4.0 μ Sec in case OFDM-DWT
Propagation Model	AWGN
Loop	500

In Figure **5-12** it is apparent that at BER = 10^{-5}, a degradation of about 1 dB in E_b/N_0 has been noticed needed for when using the coded link with interleaving compared with the coded link without interleaving. However, the BER performances is the same for OFDM-FFT/DWT/DPWT when employing interleaving.

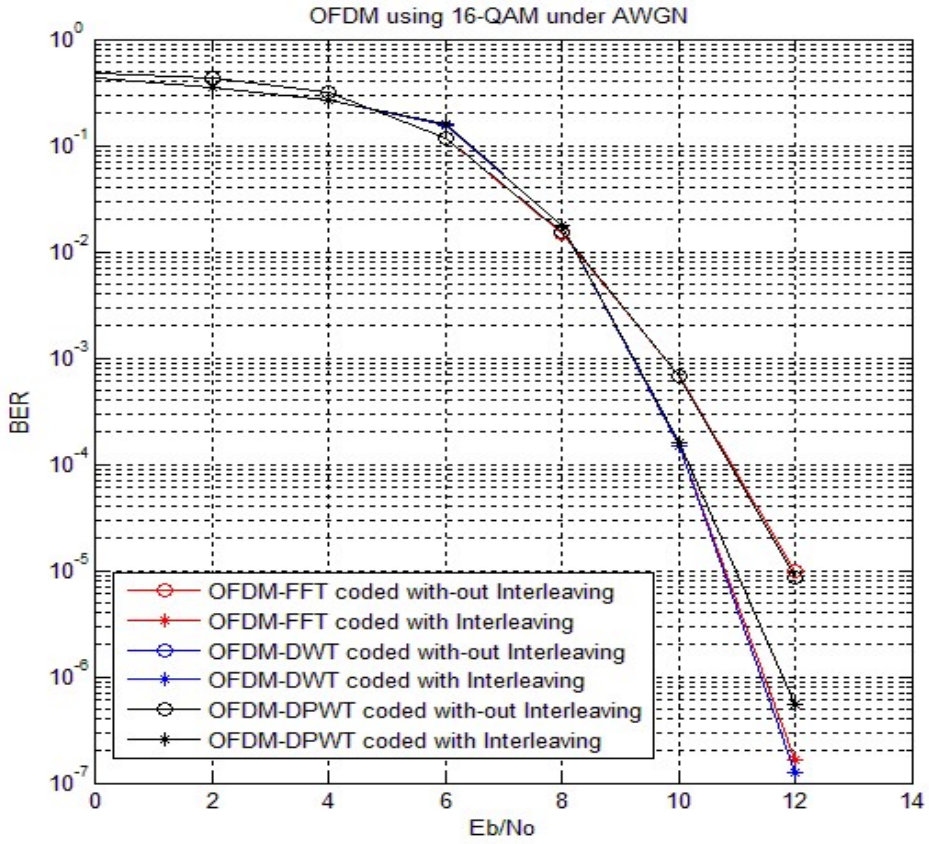


Figure 5-12 Average BER of OFDM employing interleaving versus E_b/N_0 under AWGN

5.7 Channel Estimation and Pilot Insertion

In Experiment 5-2, at the transmitter, the data bits are modulated on the subcarriers by 16-QAM. At the receiver, due to the carrier frequency offset, timing offset, and frequency selective fading, the constellation for each subcarrier has a random phase shift and amplitude fluctuations as shown in Figure **5-13**. inFigure **5-13**, $QAM_i = A_i e^{j\phi i}$ is the transmitted QAM subcarrier i, $(QAM_i)^{\wedge} = A_i e^{j\phi \wedge i}$ is the received one. To estimate the data bits at the receiver, knowledge of these unknown phase and amplitude variations is required. In coherent detection approach, the task of channel estimation circuit is to find these unknown values, and to use these values to determine the best possible decision boundaries for the constellation of each QAM subcarrier. Thereafter, the QAM symbols are ready to convert to binary soft decision[2][82].

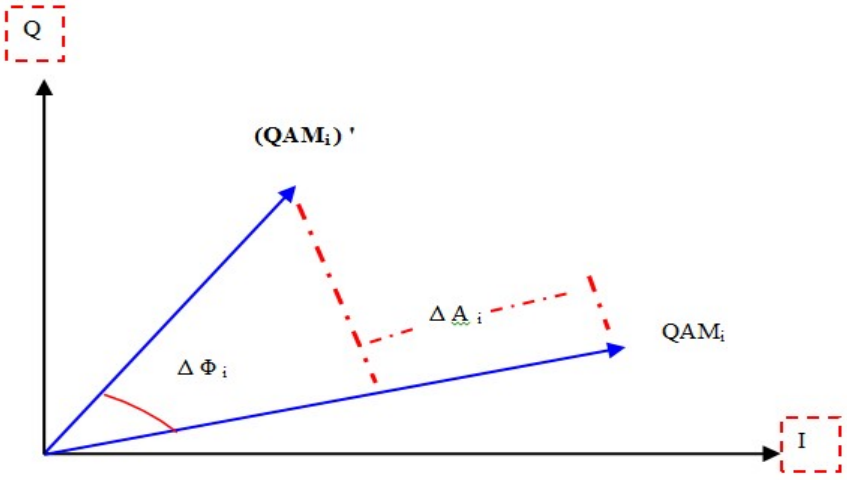


Figure 5-13Transmitted and Received QAM subcarrier

5.7.1 Pilot Insertion Types

There are several channel estimation techniques, some of which are based on the pilot-data-insertion. There exitseveral methods that can be used to insert pilot-data at transmitter. The basic two types of inserting pilots to perform channel estimation are block type and comb type. As shown in Figure **5-14**, in block type, all subcarriers are used as pilots in a specific period, whereas in comb type, part of subcarriers are always reserved as pilot for each symbol [2][3][56].

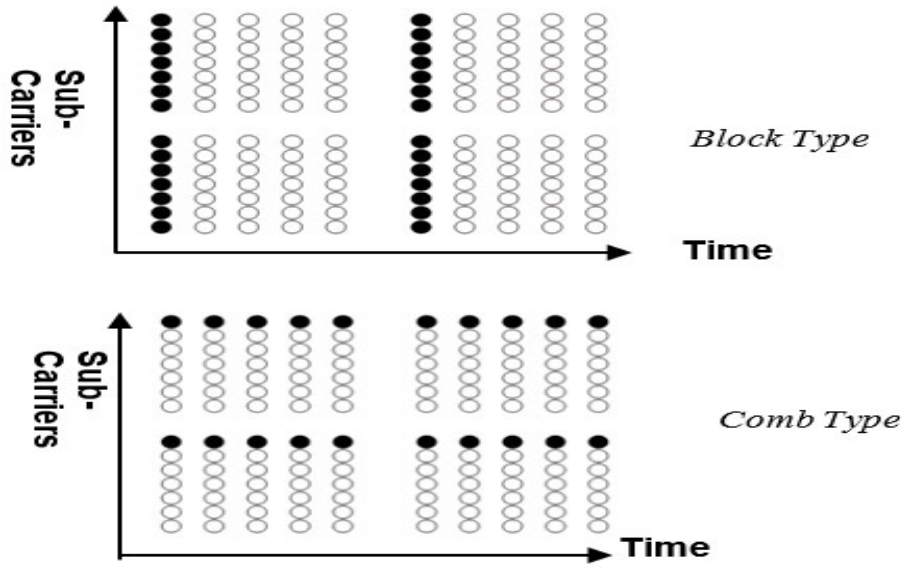


Figure 5-14 Types of pilot Arrangement for channel estimation

In our research block type pilot insertion is used. Channel estimation is performed by inserting pilots into all of subcarriers of OFDM symbols with specific period. Accordingly our calculation will be in frequency domain. Thus, the configuration of the OFDM transmission scheme will be as in Figure **5-15** and the

OFDM packet structure is as shown in Figure **5-16**. In Figure **5-16**, data source has N OFDM symbols, with each OFDM symbol having M subcarrier channels. This packet structure is placed after removing zeros subcarrier – null subcarriers.

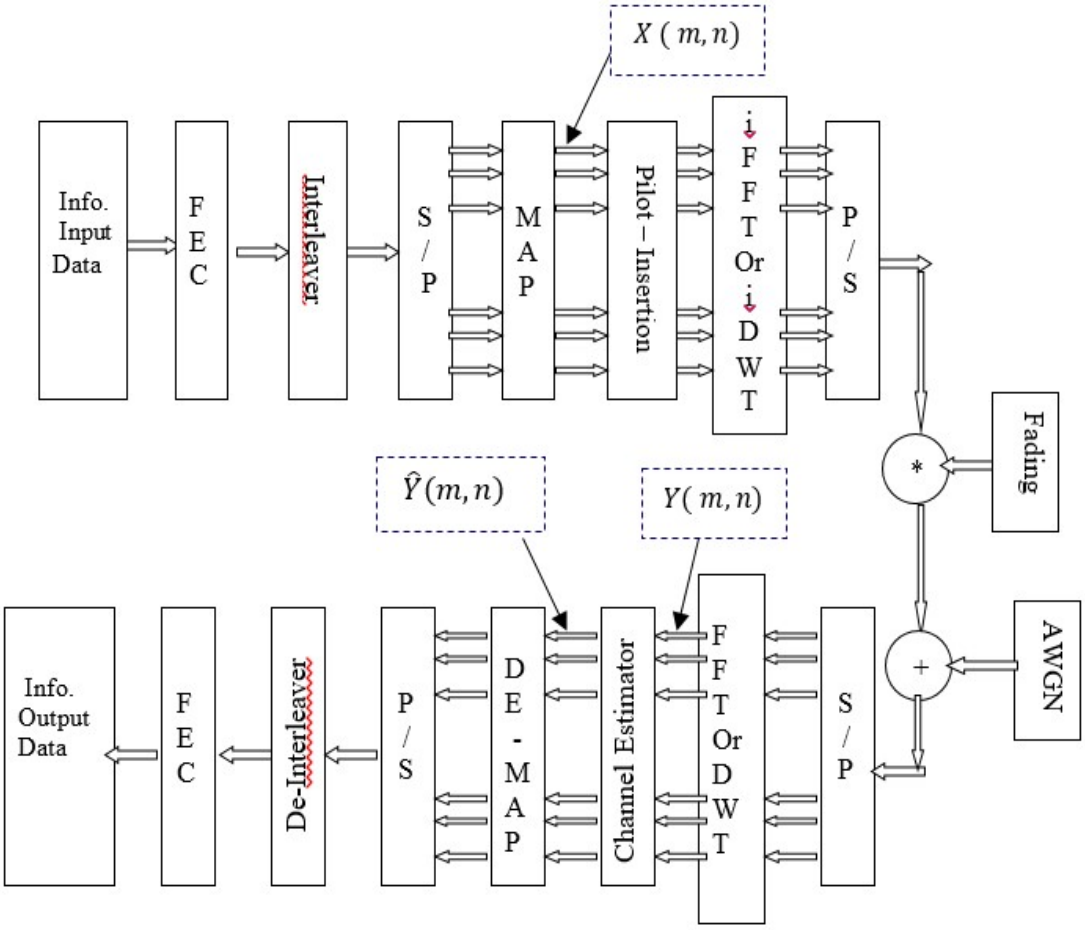


Figure 5-15 OFDM transmission scheme employing basic channel estimation method

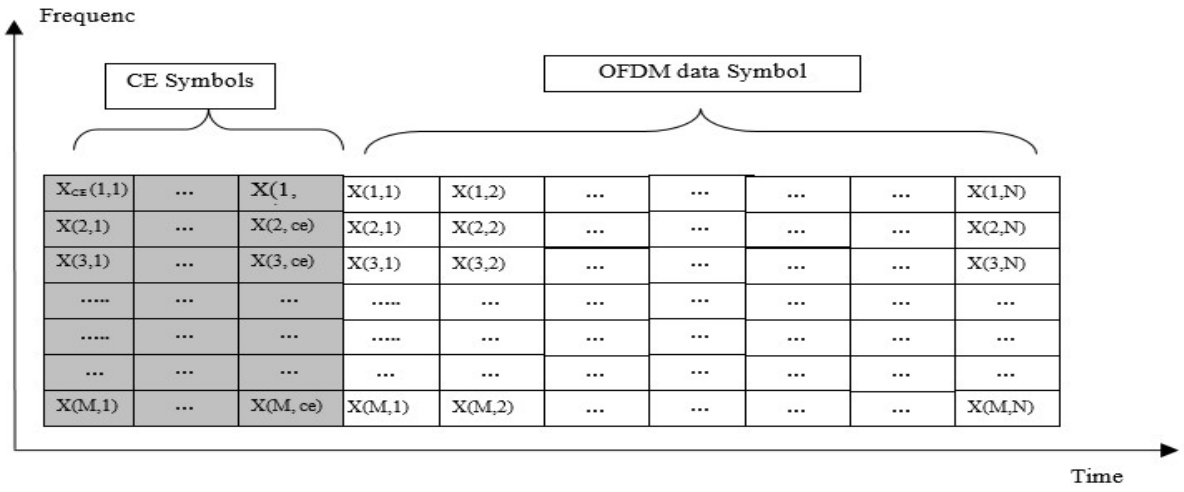


Figure 5-16 OFDM packet structure

5.7.2 Basic Channel Estimation Technique

A Channel frequency response can be obtained from the following equation:

$$H(m,n) = \frac{Y(m,n)}{X(m,n)} \tag{5.4}$$

Using (5.3)

$$H(m,n) = H_r(m,n) + \frac{1}{X(m,n)} \sum_{g=1}^{M} X(g,n-1).Hs(g,n)$$

$$+ \frac{1}{X(m,n)} \sum_{g=1}^{M} X(g,n).Hc(m,g,n) + \frac{W(m,n)}{X(m,n)}$$

$$= H_r(m,n) + \Phi(m,n) + \Gamma(m,n) \tag{5.5}$$

where : $H_r(m,n)$ denotes the real channel frequency response , $\Phi(m,n)$ denotes the error caused by ISI and ICI ,and $\Gamma(m,n)$ denotes the error caused by AWGN [79].

The pilot symbols that are inserted along the frequency axis before iFFT/iDWT circuit at the transmitter are called Channel Estimation (CE) Symbols. Not only the position but also the data inside these CE symbols are known at the receiver. As a result, the channel frequency response can be estimated on CE symbols by applying (5.4). Moreover, by averaging the estimated values of channel frequency response overall CE symbols for each subcarrier-channel, the accuracy of channel estimation is improved [2][79].The average channel frequency response on CE symbols can be represented as:

$$\hat{H}(m,0) = \frac{1}{N_{CE}} \sum_{ce=1}^{Nce} \frac{Y_{CE}(m,ce)}{X_{CE}(m,ce)} \tag{5.6}$$

Where $Y_{CE}(m,ce)$ and $X_{CE}(m,ce)$ are the received and sent pilot subcarriers respectively, and N_{CE} is the number of CE symbols in the OFDM packet. The average channel frequency response on CE symbols for subcarrier $m$is used to recover subcarrier m in all OFDM data symbols in the packet as shown in (5.7).

$$\hat{Y}(m,n) = \frac{Y(m,n)}{\hat{X}(m,0)} \quad for\, m = 1,2,...,M \quad and \quad n = 1,2,...,N \tag{5.7}$$

BER can be determined by comparing recovered subcarrier data $\hat{Y}(m,n)$, and sent subcarrier data $X(m,n)$ for $m = 1,2,\ldots..,M$, and $n = 1,2,\ldots.,N$,.

In the above channel estimation technique, the channel estimates obtained from CE symbols is used to recover all the received OFDM data symbols in one

packet. It implies, under the assumption that the fading channel is time in-variant within the packet duration, this method is effective[2].

5.8 Modified Channel Estimation Techniques

In a time-varying fading channel induced by Doppler shifts, the channel estimated value at time t1 is different from the estimated value at t2 as shown in Figure **5-17**, so that the channel frequency response at the start of a packet differs from that at the tail of the transmission packet as shown in Figure **5-17**. The resulting channel estimation error would adversely affect the channel estimation accuracy.

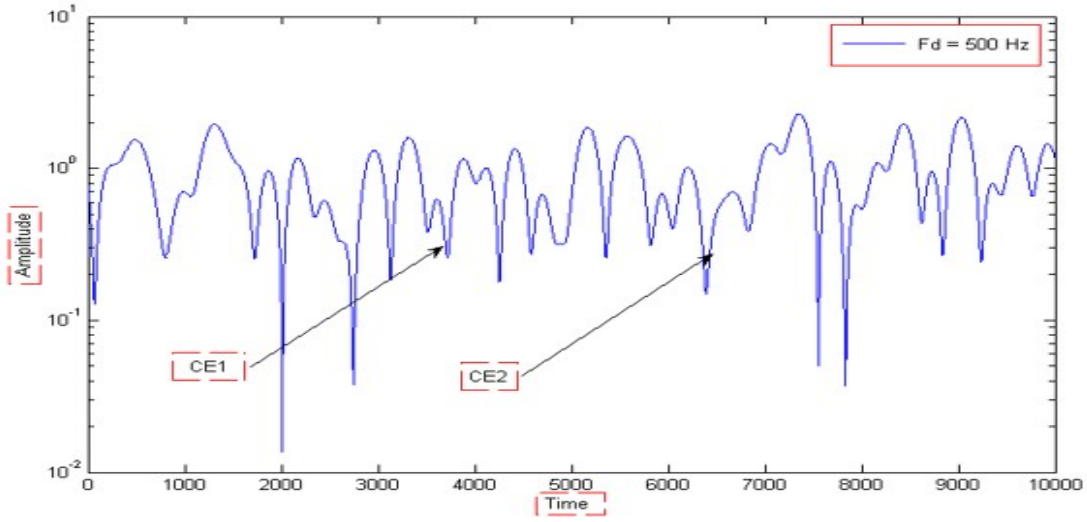


Figure 5-17 Maximum Doppler frequency Fd = 500 Hz versus time

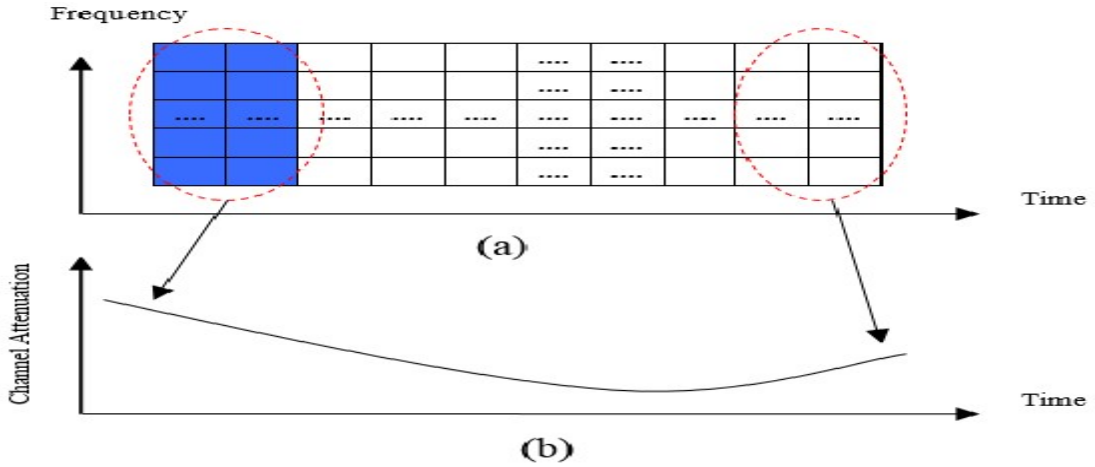


Figure 5-18 (a) Packet structure (b) time variance of channel frequency response at a certain subcarrier

5.8.1 Periodic Pilot Insertion Technique

By increasing the number of CE symbols, the accuracy of channel estimation goes up more slowly than the bit rate goes down, which is not a satisfactory method. A periodic estimation of channel frequency response has been tried to reduce the channel estimation error, whereby pilot data is inserted periodically in all frequency intervals and at a certain time intervals before the iFFT/iDWT circuit at the transmitter [56]. The packet structure in Figure **5-19** shows that the data packet is divided into sub-packets and in the preamble of each sub-packet CE symbols are inserted.

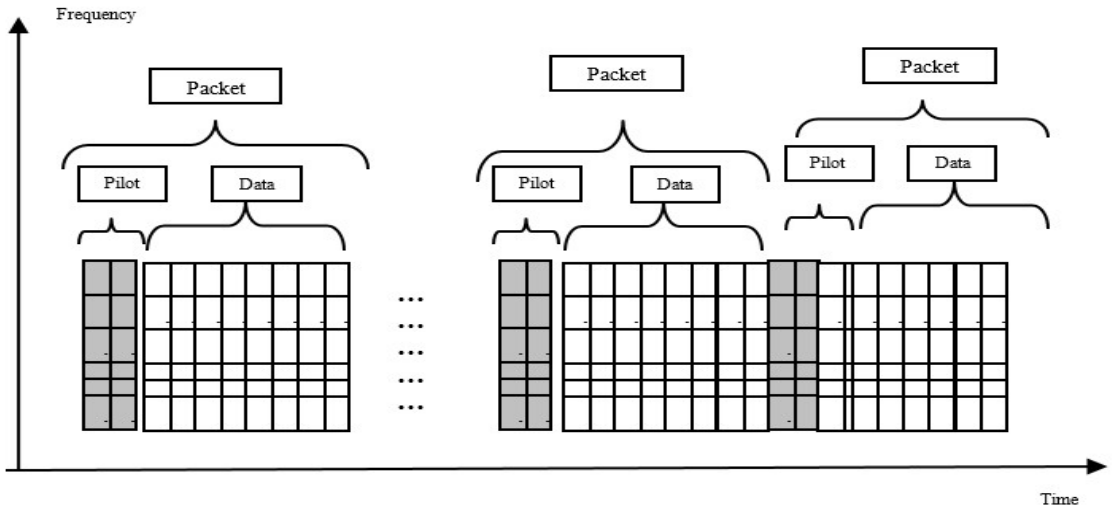


Figure 5-19 Packet structure using periodic pilot insertion for channel estimation

The accuracy of channel frequency is increased using this method, howeverthe transmission efficiency is decreased. To improve the transmission efficiency, the number of pilot data in one OFDM packet must be reduced [2].

5.8.2 Improved Data-Aided (DA) Technique

Another method to achieve periodic channel estimation is an improved Data-Aided (DA) channel estimation technique. A periodic channel estimation of channel frequency response can be achieved by generating a reference data from the received OFDM symbols.

5.8.2.1 DA Technique

The CE symbols are inserted for all frequencies period at the preamble of the packet at the transmitter as shown in Figure **5-17**. At the receiver, the output of

FFT/DWT circuit is split into two parts; the CE symbols and data symbols. From CE symbols, the average channel frequency response [$\widehat{X}(m,0)$for $= 1,2,\ldots,M$] is calculated by(5.6). Furthermore, it will be used to recover the first OFDM data symbol [$\hat{Y}(m,n)$for $m = 1,2,\ldots\ldots,M$, $n = 1$] as in (5.7). The recovered data is fed into the new reference signal generator that is composed of a serial to parallel converter, and remapping onto a constellation diagram of digital QAM.The new reference data [$\hat{X}(m,n)$ for $m = 1,2,\ldots,M$, $n = 1$] is used to calculate the new channel frequency response $\hat{G}(m,n)$ as in(5.8). To increase the channel estimation accuracy, the estimated values of channel frequency response are averaged for each subcarrier channel to calculate the $\hat{H}_{NEW}(m,n)$ as in(5.9). Then, $\hat{H}_{NEW}(m,n)$ is used to recover the data source $Y(m,1)$ onceagainThe process applied on the first OFDM data symbol is repeated on the second OFDM data symbol. However, the average channel frequency response can be obtained by means of (5.10). This process is recursively performed until the last OFDM symbol data in order. Consequently, the periodic channel estimation can be performed. Figure **5-20** shows the OFDM transmission scheme employing a periodic DA channel estimation method [2].

$$\hat{G}(m,n) = \frac{Y(m,n)}{\hat{X}(m,n)} \quad \text{for m} = 1,2,...,\text{M and n} = 1, \tag{5.8}$$

$$\hat{H}_{NEW}(m,n) = \frac{1}{Nce+n} \sum_{p=n-Nce+1}^{n} \hat{G}(m,p) \text{ ,for } m = 1,2,\ldots\ldots,M \text{ and } n = 1 \tag{5.9}$$

$$\hat{H}(m,0) = \hat{H}_{NEW}(m,0) \tag{5.10}$$

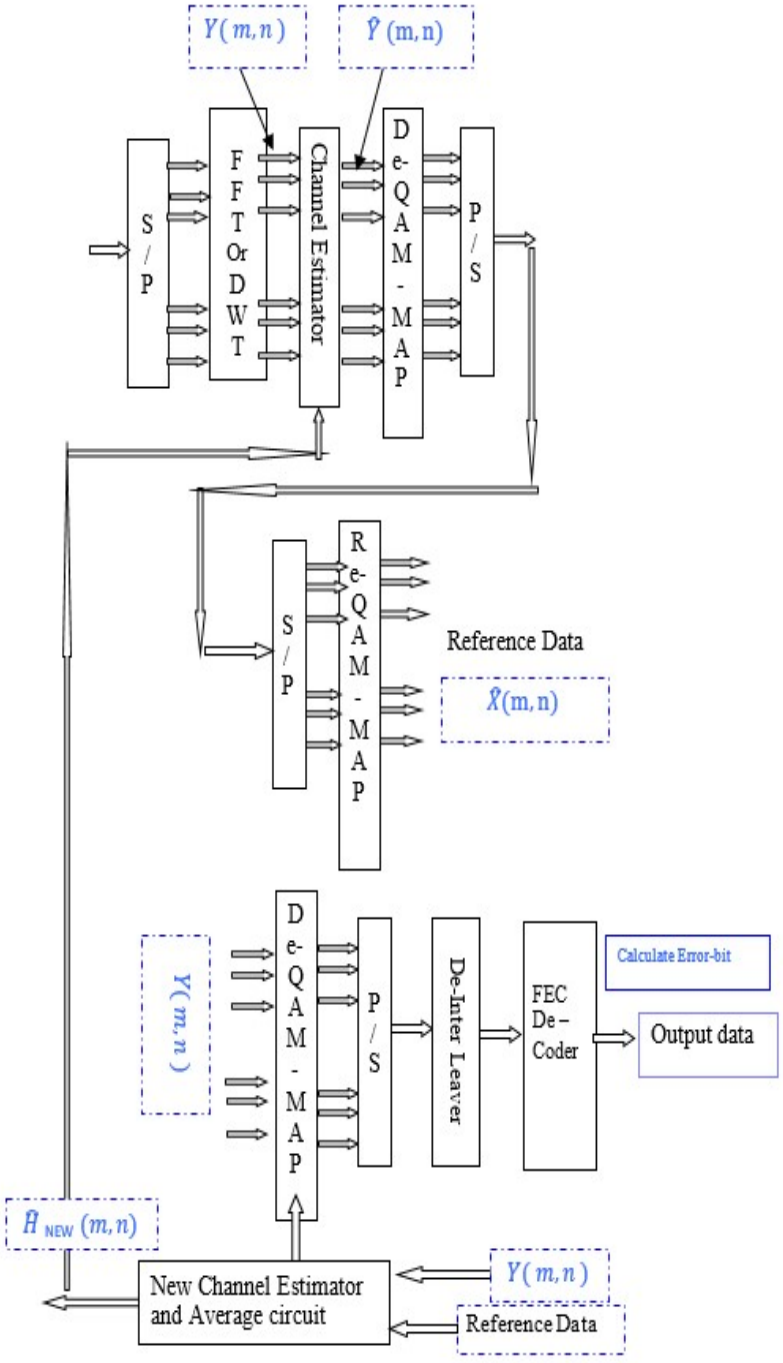


Figure 5-20 Receiver of OFDM transmission scheme employing a basic periodic DA channel estimation method

5.8.2.2 Conventional DA Technique

To improve the accuracy of channel frequency response calculated by DA channel estimation technique, Ref [78]has proposed a forgetting factor method to calculate $\hat{H}_{NEW}(m,n)$ as in (5.11).

$$\hat{H}_{NEW}(m,n) = \alpha.\hat{H}(m,n-1) + (1-\alpha).\hat{G}(m,n) \quad (5.11)$$

Where:

$\hat{H}(m,n)$ is the average of estimated values of channel frequency response untiln$_{th}$ OFDM

$\hat{H}(m,n-1)$ is the average of estimated values of channel frequency response untilOFDM $(n-1)$

$\hat{G}(m,n)$ is the estimated value of channel frequency response in current OFDM symbol.

α is the forgetting factor, 0 ≤ α ≤ 1.

5.8.2.3 Hard Decision Pseudo Pilot (HDPP) technique

This technique also called Threshold DA technique. The advent of Threshold DA technique came about as a result of many improvement trials of conventional DA

method, and Ref [79] has proposed a threshold DA method to calculate $\hat{H}_{NEW}(m,n)$ using(5.12) and (5.13):

$$\hat{\zeta}_k = \left| \frac{\hat{G}(m,n)}{\hat{H}(m,n-1)} \right| \tag{5.12}$$

$$\hat{H}(m,n) = \begin{cases} \hat{H}(m,n-1) if \hat{\zeta}_k < \lambda \\ \hat{G}(m,n) otherwise \end{cases} \tag{5.13}$$

Hard Decision Pseudo Pilot (HDPP) technique is based on DA channel estimation. In the following sub-section, a detailed treatment ofa channel estimation technique called: FEC (Forward Error Correction coding) Decoded Pseudo Pilot (FDPP) technique, which is also based on DA channel estimation, is provided.

5.9 FEC (Forward Error Correction coding) Decoded Pseudo Pilot (FDPP) Channel Estimation Technique

Further along the development process,the threshold DA method had undergone several improvements led to the development of Forward Error Correction coding (FEC) Decoded Pseudo Pilot (FDPP) channel estimation method proposed in Ref [1]based on DA channel estimation that permitting a periodic channel estimation of channel frequency response by generating a reference data from the received OFDM symbols.However, FDPP method differs from

previous methods in terms of the generation of reference data and the use of new channel frequency response. It is worthy of notethat all of the aforementioned channel estimation techniques are employed in OFDM-FFT system so farand none of them has been used in the OFDM-DWT system .

Figure **5-21** and Figure **5-22** illustrate the configuration of the 16-QAM-OFDM transmission scheme. In the transmitter as shown in Figure **5-21** a high speed binary serial data is fed into the FEC encoder and redundant tail bits are added.Then, the coded serial data are scrambled by bit interleaver. Furthermore, the interleaved high speed serial data is mapped by signal mapping circuit for gray-coded 16-QAM. This mapped high speed serial data is then converted to parallel low speed data with M sequences at the n^{th} interval, then digitally modulated by 16-QAM at the QAM mapping circuit. The modulated data is then fed into allocated subcarrier-channels, which are denoted as $X(m,n)$for $m = 1,2,\ldots\ldots,M$, and $n = 1,2,\ldots\ldots,N$ .where M is the number of subcarrier channels in each OFDM symbol, and N is the number of OFDM data symbol in the data field.A fixed pattern pilot symbol is inserted at fixed time interval in the preamble of the transmission data of each parallel subcarrier channel. The time-domain pilot symbols are called channel estimation (CE) symbols, and the number of these symbols is N_{CE}.The subcarrier data and pilots are either multiplexed with $N_f$samples in iFFT circuit or decomposed with L decomposition stages in iDWT to generate an OFDM symbols. The output of iFFT/iDWT circuit is an OFDM

symbol in time/wavelet domain, and all subcarriers in each OFDM symbol are orthogonal. In the OFDM-FFT system, a guard interval (GI) is inserted before each OFDM symbol to avoid ISI that iscaused by multipath fading channel whose time delay is longer than the transmission time of an OFDM signal. In order to maintain orthogonality between subcarriers, GIhas to be in a cyclically extended signal. However, in OFDM-DWT, GI is not required. Finally the information symbols converted to serial form and transmitted over the channel[2].

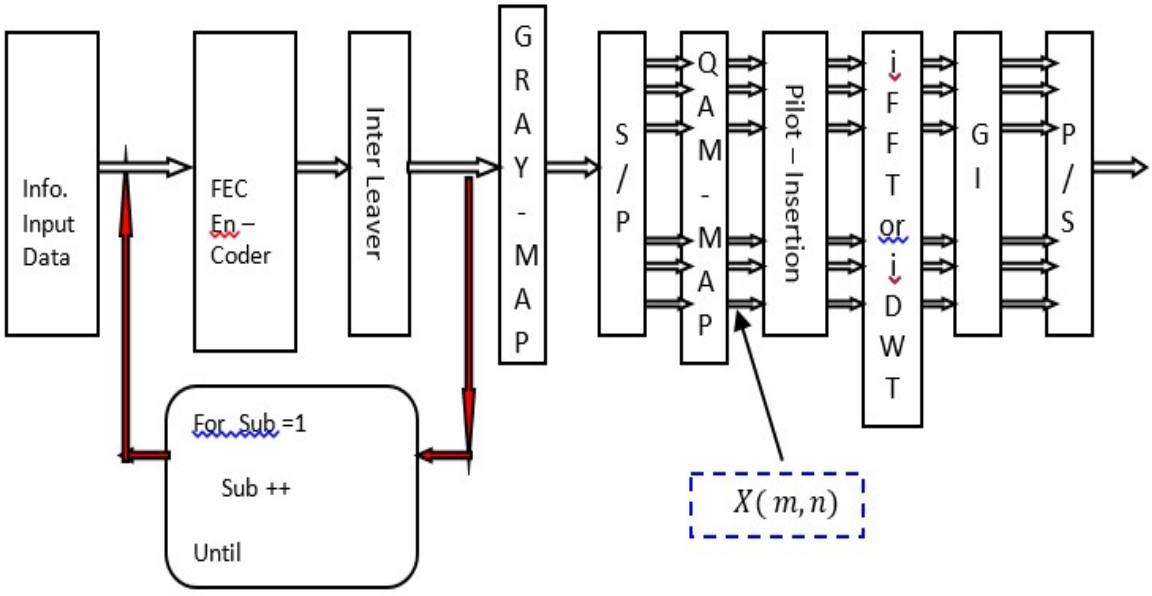


Figure 5-21 Transmitter of OFDM transmission scheme employing FDPP-based channel estimation method

In the transmission scheme above, the high speed serial data is divided into sub-packets. Each sub-packet is then driven into the FEC encoder and redundant bits are added, then scrambled by bit interleaver. When all sub-packets are passed through these two circuits separately, they are reassembled again in the same

order before input into gray mapping circuit. This modification allows the receiver to decode each sub-packet separately as would be demonstrated next[2].

At the receiver, as shown in Figure **5-22**, having the received OFDM signal converted to parallel, GI is removed (in the iFFT at Tx case), and then the signal is then fed to FFT/DWT circuit and de-multiplexed as subcarrier channel data. The average channel frequency response on CE symbols [$\hat{H}(m,0)$] is then calculated by(5.6). Now the received subcarrier data $Y(m,n)$ for $= 1,2,\ldots\ldots,M$, and $n = 1,2,\ldots\ldots,N$, is arranged back to the original form and size at transmitter[2].

The received subcarriers data on the first OFDM data sub-packet is $Y(m,n)$for $m = 1,2,\ldots\ldots,M$, and n = 1, 2, …., N-sub , where N-sub is the number of OFDM symbols in each sub-packet. These received subcarrier data are compensated with $\widehat{H}(m,0)$ .Then the compensation subcarriers date are de-modulated by 16-QAM de-modulator, then it is de-mapped by the de-mapping circuit for de-gray coded 16-QAM.This signal is then scrambled by the bit de-interleaver. Thendecoded by a FEC circuit to obtain binary output data1. Finally, output data1 is compared with i input data to have a measure of is carried out BER1[2].

The recovered data sub-packet is driven into a new reference signal generator in which almost the same operations as in the transmitter are carried out. Hence it

generates a replica of the transmitted OFDM data symbol on each subcarrierchannel, which will be used as new reference signal $\hat{X}(m,n)$ for $m = 1, 2, \ldots., M$, and n = 1, 2, …., N-Sub. $\hat{X}(m,n)$denotes the reference sub-carrier data on the nth OFDM data symbol. Based on (5.4) and (5.6), the channel frequency response of the transmitted OFDM data symbol is re-estimated[2] and can be represented as

$$\hat{G}(m,n) = \frac{Y(m,n)}{\hat{X}(m,n)} \text{,for } m = 1, 2, \ldots., M \text{ , and n = 1, 2, …., N-Sub} \tag{5.14}$$

Where $\hat{G}(m,n)$ is the new channel frequency response.To improve the new channel frequency response accuracy, the new estimated values are averaged on each subcarrier-channel over whole first sub-packet only. Based on (5.6), the new average estimated value can be represented as:

$$\hat{H}_{NEW}(m,0) = \frac{1}{N-Sub} \sum_{n=1}^{N-Sub} \hat{G}(m,n) \text{, for } = 1, 2, \ldots., M \tag{5.15}$$

The received subcarrier data of the first OFDM data sub-packet $Y(m,n)$, [for m =1,2,….., M , and n = 1, 2, …., N-Sub], is fed into a new recovery circuit in which almost the same operations as in the receiver (in the first stage) are carried out. However, the new average estimated value $\hat{H}_{NEW}(m,0)$ is used here instead of $\hat{H}(m,0)$. So, $Y(m,n)$for m =1, 2,…,M , and n =1,2,….,N-Sub, are re-compensated with $\hat{H}_{NEW}(m,0)$. These re-compensation subcarriers date are re-demodulated by

16-QAM de-modulator, then re-de-mapping by signal de-mapping circuit for de-gray coded 16-QAM, and then re-scrambled by the bit de-interleaver. Finally, this signal is re-decoded by the FEC circuit to obtain output binary data. This output data signal is then compared with the input data to calculate the BER[2].

The whole operations op performed onthe first OFDM data sub-packet is repeated on the second OFDM data sub-packet. The average channel frequency response can be obtained from (5.16).

$$\hat{H}(m,0) = \hat{H}_{NEW}(m,0) \tag{5.16}$$

This process is recursively performed until the last data sub-packet gets in order, and, consequently channel estimation can be performed. InFigure **5-22**, BER1 is the BER using channel estimates obtained from previous sub-packet,while BER is the BER using channel estimates obtained from current sub-packet [2].

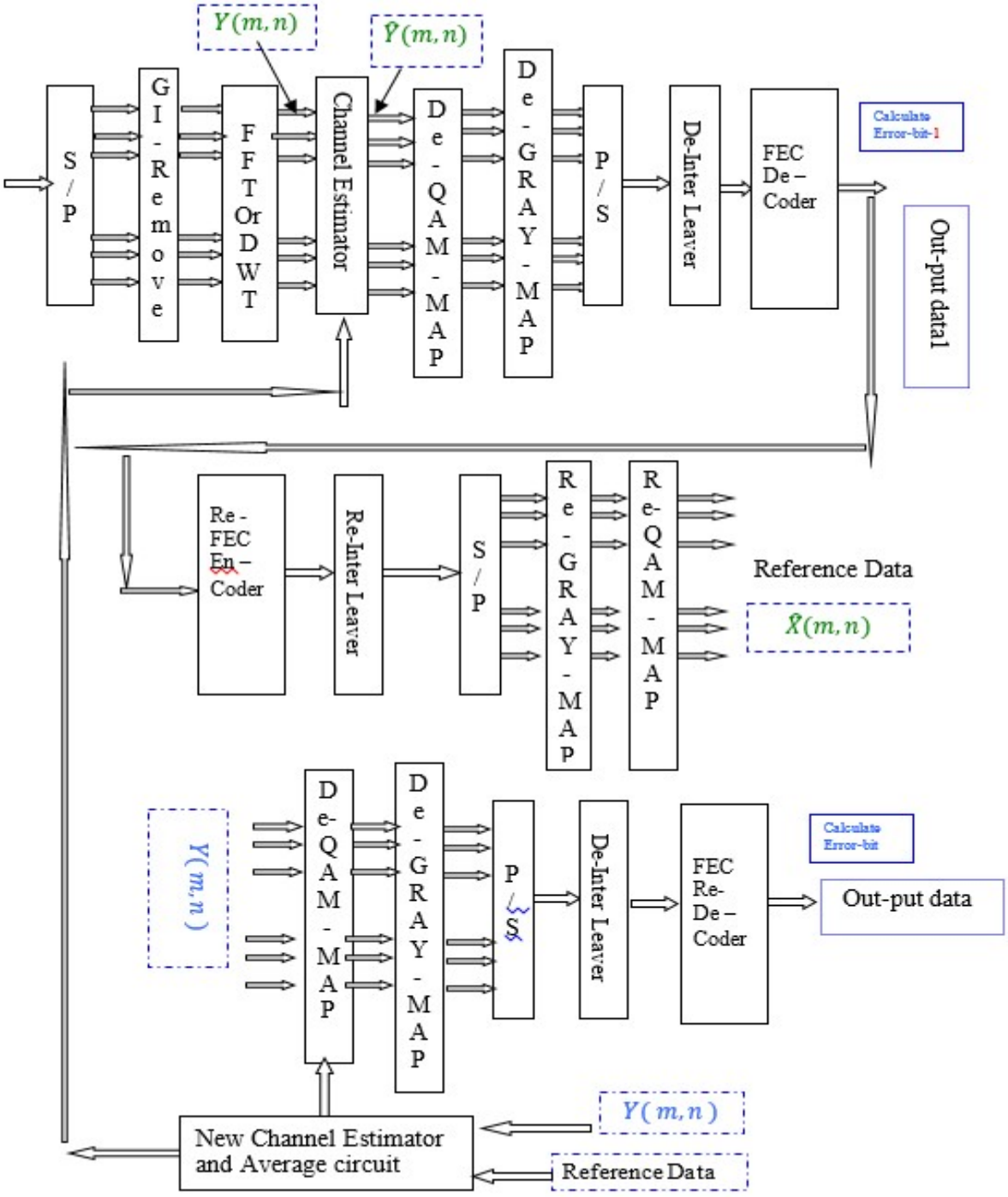


Figure 5-22 Receiver of OFDM transmission scheme employing a FDPP-based channel estimation method

5.10 Logarithm of Likelihood Ratio (LLR) Calculation in Soft Decision FEC

FEC encoder and FEC decoder have been used in our OFDM transmitter and OFDM receiver respectively. The FEC decoder can be either hard-decision or soft-decision, where a soft-decision FEC decoder has a better coding gain performance than the hard-decision one [3]. There is a difficulty in Likelihood calculations for each received bitwhen using the soft-decision Viterbi decoding in 16-QAM-OFDM transmission scheme. A likelihood calculation method in[83] that is based on approximated Logarithm of Likelihood Ratio (LLR) in [84] is used in our simulator. The method is also well explained in [2] as follows:

Our source data is fed into the encoding circuit,then the coded bits are modulated by 16-QAM modulator to produce the subcarriers data $X(m,n) = QAM_i = A_i.e^{j\Phi i} = I_i + jQ_i$. The QAM modulation is based on gray coding. Each subcarrier has a random phase shift and amplitude change that can be shown in their constellation due to fading. The amplitude and angle of the phase rotation caused by fading can be denoted by R_i and $\phi_i$respectively, and the received subcarrier as $Y(m,n) = (QAM_i) = \hat{A}_i e^{(j\hat{\Phi}i)} = \hat{I}_i + j\hat{Q}_i$. The angle and amplitude of channel frequency response that can be estimated from received pilot subcarriers $\hat{R}_i$ and $\hat{\phi}_i$ is used to re-cover $Y(m,n)$. The recovered

subcarrier $\hat{Y}(m,n) = (QAM_i\hat{\hat{}}) = \hat{\hat{A}}_i e^{(j\hat{\hat{\Phi}}i)} = \hat{\hat{I}}_i + j\hat{\hat{Q}}_i$ is converted into four symbol data using (5.17)[84].

$$U_{i1} = \left|\hat{\hat{I}}_i\right| - 2$$

$$U_{i2} = \hat{\hat{I}}_i$$

$$U_{i3} = \left|\hat{\hat{Q}}_i\right| - 2$$

$$U_{i4} = \hat{\hat{Q}}_i \qquad (5.17)$$

These four symbol data are multiplied by $\hat{R}_i$ to improve the accuracy of the estimated data symbols as shown in (5.18)[83], and are finallydriven into the soft-decision Viterbi decoder to recover the transmission data.

$$U_{i1} = (\left|\hat{\hat{I}}_i\right| - 2).\hat{R}_i$$

$$U_{i2} = \hat{\hat{I}}_i.\hat{R}_i$$

$$U_{i3} = (\left|\hat{\hat{Q}}_i\right| - 2).\hat{R}_i$$

$$U_{i4} = \hat{\hat{Q}}_i.\hat{R}_i \qquad (5.18)$$

5.11 Simulation Results

The OFDM-FFT and OFDM-DWT transmission schemes using HDPP-based and FDPP-based channel estimation algorithms described in the previous sections were evaluated with packet structure shown in Figure **5-23** in terms of average BER, where the bit rate for OFDM-DWT and OFDM-FFT are 24 and 20 M bit per second respectively. The simulation parameter values are discussed in next sub-section.

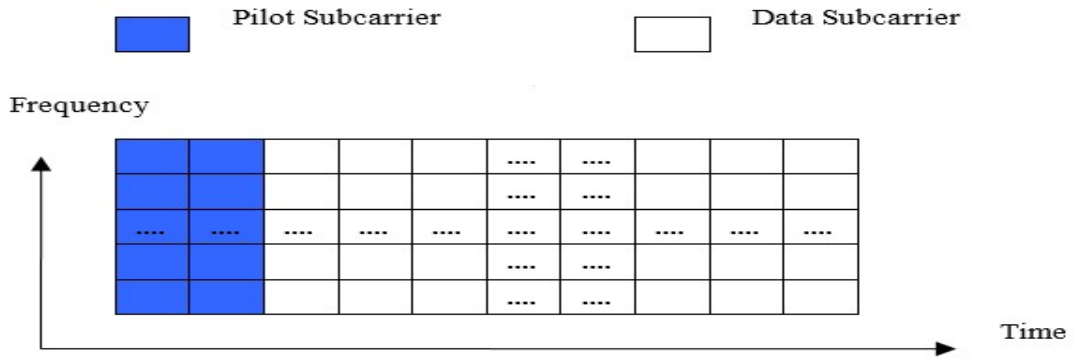


Figure 5-23 OFDM transmission scheme Packet Structure

5.11.1 System Parameters

Bandwidth, bit rate and delay spread are three main conflicting requirementsaffecting the choice of OFDM parameters. Delay spread requirement dictates directly the guard time parameter, depending on the type of coding and modulation. Once the guard time is defined, the symbol duration can be fixed [2][3].

When symbol duration is much larger than guard time, the Signal-to-Noise-Ratio (SNR) loss caused by guard time is reduced.However,when symbol duration is

large, then more subcarriers with smaller subcarrier spacing occurs,which leads to having a larger implementation complexity and more sensitivity to phase noise and frequency offset. In practice, a guard time causes a loss in SNR of only 1-dB if its maximum value is 1/5 of the symbol duration [2][3].

So far, the symbol duration and guard time are fixed.A modulation type, coding rate and symbol rate all determine the bit rate per subcarrier. Division of bit rate requirement by the bit rate per subcarrier determines the number of subcarriers parameter [2][3].

To evaluate the performance of FDPP and HDPP based channel estimation methods for both DWT and FFT based OFDM, common parameters have been chosen as follows:

Generally, a guard time with a tolerable delay spread requirement equal to 200 ns, a safe value of guard time is 800 ns. 800 ns is enough to cover delayed wave in outdoor microcellular communication systems on 5 GHz. To avoid ICI beside avoiding ISI; the guard type will be cyclic extension. For symbol duration, to make the guard time loss smaller than 1 dB, the OFDM symbol duration is determined to be 6 times that of guard time, which is equal to 4.8μs. [2][3]. However, for the DWT-based OFDM, there is no requirement to use a guard interval with a cyclic prefix, and hence the OFDM symbol duration is equal to 4μs

and thus the spectral efficiency is improved [4][72]. For both schemes, the subcarrier spacing is [1/ 4.0 μs] = 250 KHz.

With regards to the number of subcarrier channels, the OFDM signal with 64 subcarrier channels is generated either by a 64-point iFFT circuit for OFDM-FFT or by 6 decomposition stages iDWT for OFDM-DWT. Where 48 subcarrier channels are used for data, and the remaining 16 subcarrier channels having zero values to provide necessary oversampling to avoid aliasing [3][76].

As per the modulation scheme in each subcarrier channel, our aim is to realize a transmission rate in excess of 20 Mbps. moreover,FEC is required to avoid the influence of fading. QPSK cannot use FEC with transmission rate of 20 Mbps. The modulation schemes that can provide a transmission rate over 20 Mbps within a limited bandwidth in the 5 GHz frequency band are 8PSK-OFDM and 16-QAM-OFDM. 8PSK-OFDM performance has been already investigated by many researchers, however, only few researchers have studied the performance of 16-QAM-OFDM. Hence, this research work simulates only 16-QAM-OFDM [2][83].

As per FEC scheme, we use convolutional encoding at the transmitter. In this case, it is recommended to use a soft-decision Viterbi decoding at the receiver to get better performance. The code rate (R) = ½ and the constrain length (K) = 7. In the FEC scheme, K-1 redundant tail bits had to be added to the input data of convolutional encoder at transmitter in order to have a smooth termination in soft-

decision Viterbi decoding at the receiver. In order to calculate the LLR, we utilize the method based on the approximated LLR that was described earlier in this chapter [2][3].

Block interleaving has been used, where it is applied for each sub-packet of OFDM packet. As per the radio channel model, a multipath fading model is considered. With regards to Doppler frequency f_D, a range between 300 to 600 Hz has been adopted. low values of $f_D$such as 50 Hz (3 m/s @ 5GHz) are not considered in this research [2].

Regarding bit rate, a 16-QAM modulation has been used in our OFDM transmission scheme, thus, the bit rate per subcarrier is 4 bit/subcarrier. However, due to the fact that a convolutional encoding with code rate = ½ has been used in the FEC block, the bit rate per subcarrier has been reduced by 2 and become 2 bit/subcarrier. The bit rate per OFDM symbol equals the bit rate per subcarrier multiplied by the number of subcarriers in each OFDM symbol.In our setting, the OFDM symbol has 64 subcarriers, of which only 48 of them is data subcarriers, so the bit rate per OFDM symbol equals to 2×48 = 96 bit/OFDM symbol. In our transmission scheme, symbol duration is equal to 4 μs in the OFDM-DWT case and 4.8 μs in the OFDM-FFT case. Consequently, an OFDM-DWT system can achieve bit rate of24 Mbit/second(96/4.0 μs), whereas OFDM-FFT system can only achieve bit rate of20 M bit/second (96/4.8 μs) =. In our model, so in our system,64 samples in both discrete transform blocks (iFFT/FFT

and iDWT/DWT)have been chosen, hence the sample rate is set to16 MHz (64/4 μs). Finally, the bandwidth which is the product of the number of subcarriers and the subcarrier spacing, thus equals 16 MHz (64subcarrier x 250 KHz subcarrier spacing)[3][4].

CE symbols are in the preamble of the OFDM packet. The number of CE symbols N_{CE} is a trade-off between a short training and better channel estimation performance. Averaging two CE symbols gives a 3-dBlower noise level in the channel estimation performance and estimates the frequency offset [2], hence, in our OFDM transmission simulator , N_{CE} has been chosen to be 2 OFDM symbols.The packet structure of the simulated OFDM transmission scheme is shown in Figure **5-23**.

5.11.2 Comparison between FDPP-DA-based channel estimation and HDPP-DA-based channel estimation for DFT-based OFDM and Wavelet-based OFDM systems

The system modelling and simulation have been carried out by means ofMATLAB (R2012b) computer simulation software to evaluate the system performance quality in terms of BER for both the wavelet-based and DFT-based OFDM transmission schemes under AWGN and multipath Rayleigh fading conditions using FDPP-DA and HDPP-DA channel estimation techniques, where haar, bd2, coif1, sym2, and bior1.3 wavelet are included in the investigative

evaluation. The packet structure is shown in Figure **5-23**. The simulation parameters, listed in Table **5-5**, were set in accordance to those reported in[1][72][79]. Sub-packet size is fixed to 7 OFDM data symbols, and the threshold is fixed to 0.4. Data fieldisset to 100 OFDM symbols and a maximum Doppler frequency $f_D$of 300 Hz.

Table 5-5 Simulation parametersof OFDM FFT/DWT system employing FDPP and HDPP

Modulation	16-QAM
Number of subcarrier s	48 data , 16 null carriers
FEC scheme	Convolutional encoding / soft decision Viterbi decoding , ($R = 1/2, K = 7$)
Discrete Transform	FFT where N_f =64 DWT where decomposition stages L= $\log_2(64)$= 6
Guard interval duration	1/5 OFDM symbol = 800 ns , cyclic extension in case OFDM-FFT Nil in case OFDM-DWT
Number of CE symbols in the preamble N_{CE}	2
Number of OFDM symbols in one packet *N*	100
OFDM symbol duration	4.8 μsec in case OFDM-FFT 4.0 μsec in case OFDM-DWT
Data bit rate	20 M bit/sec in case OFDM-FFT 24 M bit/sec in case OFDM-DWT
Interleaving type	Block interleaving

Interleaving size	Within one OFDM sub-packet [192 * sub packet size] bits
Propagation model	AWGN , multipath Rayleigh fading
Max Doppler frequency f_D	300 Hz
Bit energy / noise density E_b/N_o	8,10,12,……,24 and 26 dB
Channel estimation	HDPP with threshold = 0.4 FDPP with sub-packet = 7 OFDM data symbols

BER is shown in Figure **5-24** and Figure **5-25** as a function of $\mathrm{E_b/N_0}$ with AWGN and multipath Rayleigh fading using HDPP-based and FDPP-based channel estimation method respectively for both OFDM systems DFT-based and DWT-based. In general, BER of wavelet-based OFDM system offers better performance than DFT-based OFDM system except the case in which bior1.3 wavelet mother is used. It is apparent that the best overall performance among the wavelet mothers is proven to bethe db2 wavelet mother for both channel estimation methods. For example, in the HDPP-based method case, it can be seen from Figure **5-24** that for a BER of 10^{-3}, the db2 wavelet-based needs about 2 dB less $\mathrm{E_b/N_0}$ compared with that of the DFT based system, whereas it is only about 1dB less in the case of coif1 and sym2. While in the case of FDPP-based method, a 3dB performance gain is attained when using db2, and 2 dB when using coif1 and sym2 over FFT for the same BER criterion of $10^{-3.}$

On the other hand, FDPP-based channel estimation case out performdthe HDPP-based channel estimation by 10dB under the same BER criterion of 10^{-3} andwhen both are employed in the OFDM-DFT system. However, virtually, under the same basis, 11 dB was attained in the OFDM-DWT(db2) case.

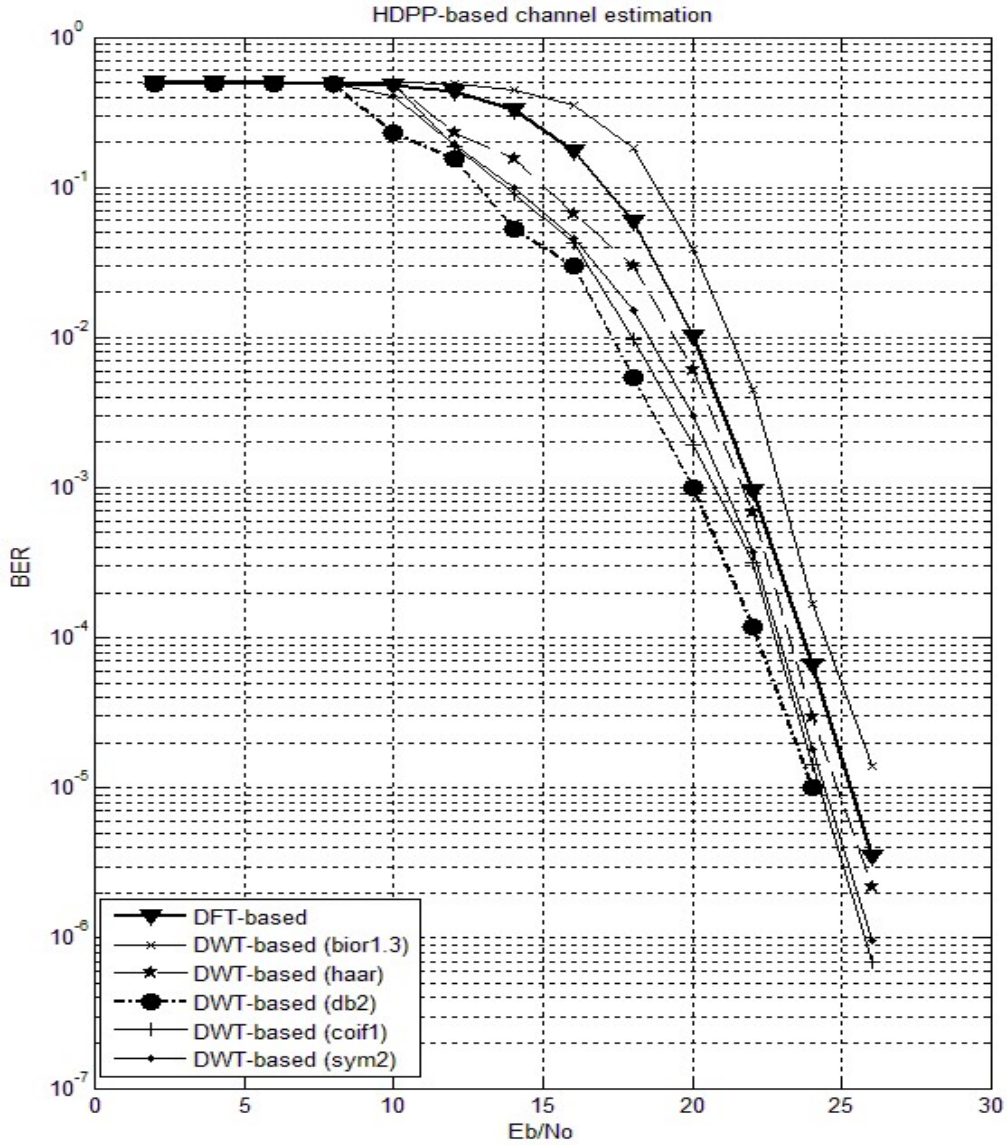


Figure 5-24 BER performance of OFDM system with the HDPP-based channel estimation with threshold = 0.4

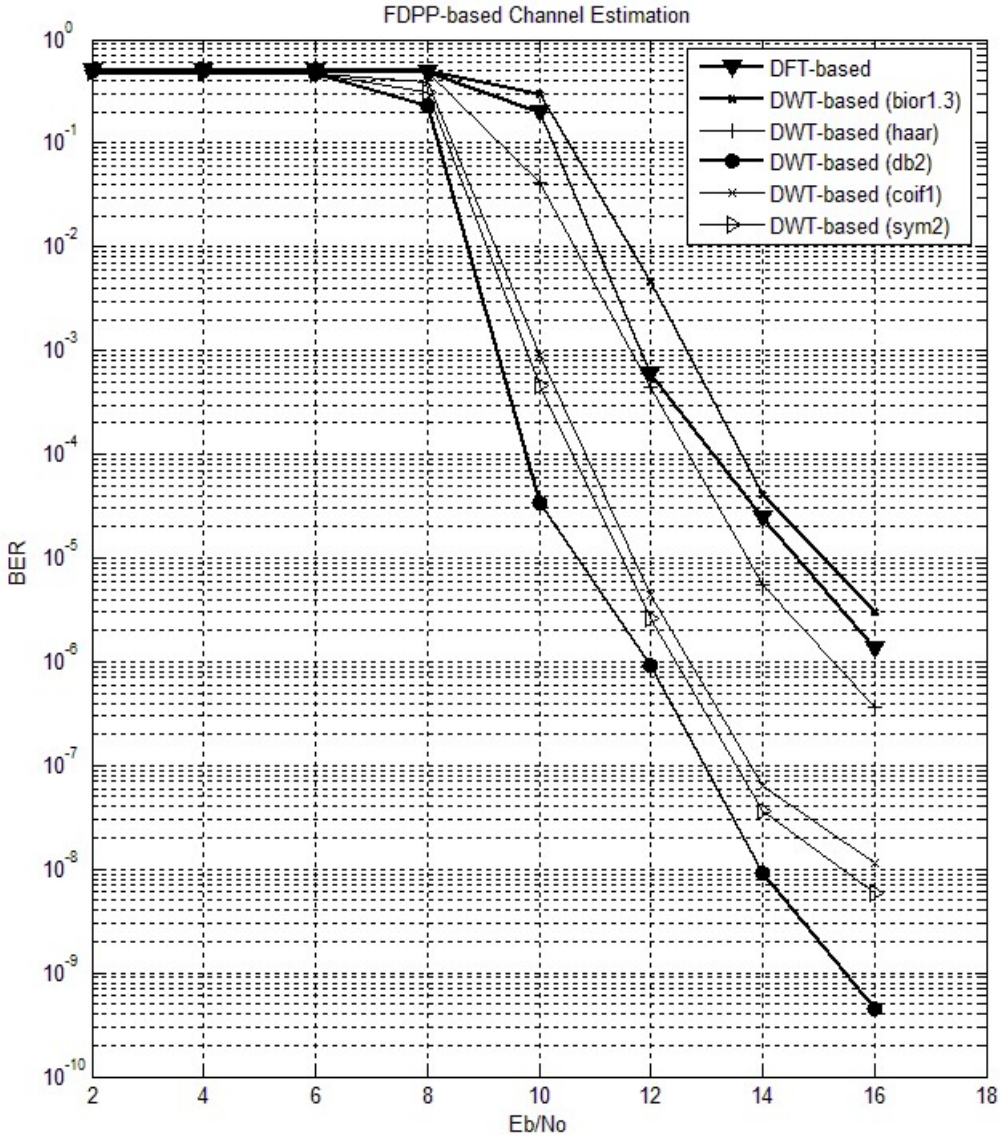


Figure 5-25 BER performance of OFDM system with the FDPP-based channel estimation with sub-packet = 7

5.11.3 Investigation of the effect of the Doppler Frequency on the FDPP-based channel estimation

The OFDM transmission scheme that uses FDPP-based channel estimation was evaluated with the packet structure shown in Figure **5-23** of 20Mbit/s in OFDM-FFT and 24Mbit/s in OFDM-DWT in terms of average BER by means of MATLAB software with simulation parameter valuesas listed in Table **5-6** have been chosen in accordance to those reported in Refs [1][72][79].

Table 5-6 Simulation parametersof OFDM system employing FDPP under Doppler frequency effect

Modulation	16-QAM
Number of subcarrier	48 data , 16 null carriers
FEC scheme	Convolutional encoding / soft decision Viterbi decoding , ($R = 1/2, K = 7$)
Discrete Transform	FFT where N_f =64 DWT where decomposition stages L= $\log_2(64)$= 6
Guard interval duration	1/5 OFDM symbol = 800 ns , cyclic extension in case OFDM-FFT Nil in case OFDM-DWT
Number of CE symbols in the preamble N_{CE}	2
Number of OFDM symbols in one packet *N*	100
OFDM symbol duration	4.8 μ Sec in case OFDM-FFT 4.0 μ Sec in case OFDM-DWT
Data bit rate	20 M bit/sec in case OFDM-FFT 24 M bit/sec in case OFDM-DWT
Interleaving type	Block interleaving
Interleaving size	Within one OFDM sub-packet [192 * sub

	packet size] bit
Propagation model	AWGN , multipath Rayleigh fading
Max Doppler frequency f_D	300, 400, 500, and 600 Hz
Bit energy / noise density E_b/N_o	14 dB
Channel estimation	FDPP with sub-packet = 3,7, and 34 OFDM data symbols

Figure **5-26** shows the average BER performance of the OFDM transmission scheme employing FDPP-based channel estimation versus max Doppler frequency f_D (300 to 600 Hz) for different values of sub-packet sizes (3, 7, and 34), having E_b/N_o fixed at 14 dB. The two approaches of OFDM (FFT-based and DWT-based) are evaluated in this experiment, where db2 mother wavelet selection was based on the best obtained performance from the previous experiment. Data field is 100 OFDM symbols.

It can be seen from Figure **5-26** that when the sub-packet size is its largest value (34 OFDM symbols) the best attained BER performance is when the f_D is as low 300 Hz, whereas the worst obtained performance is when the sub-packet size is as 3 OFDM symbols.This is because when calculating the average channel estimation for each sub-carrier channel in each sub-packet, as the number of channel estimates increases, the noised effect is reduced. Furthermore, the quality of channel coding scheme is directly proportional to the number of coded

symbols, namely, as the number of coded symbols increases, the channel codding quality increases.

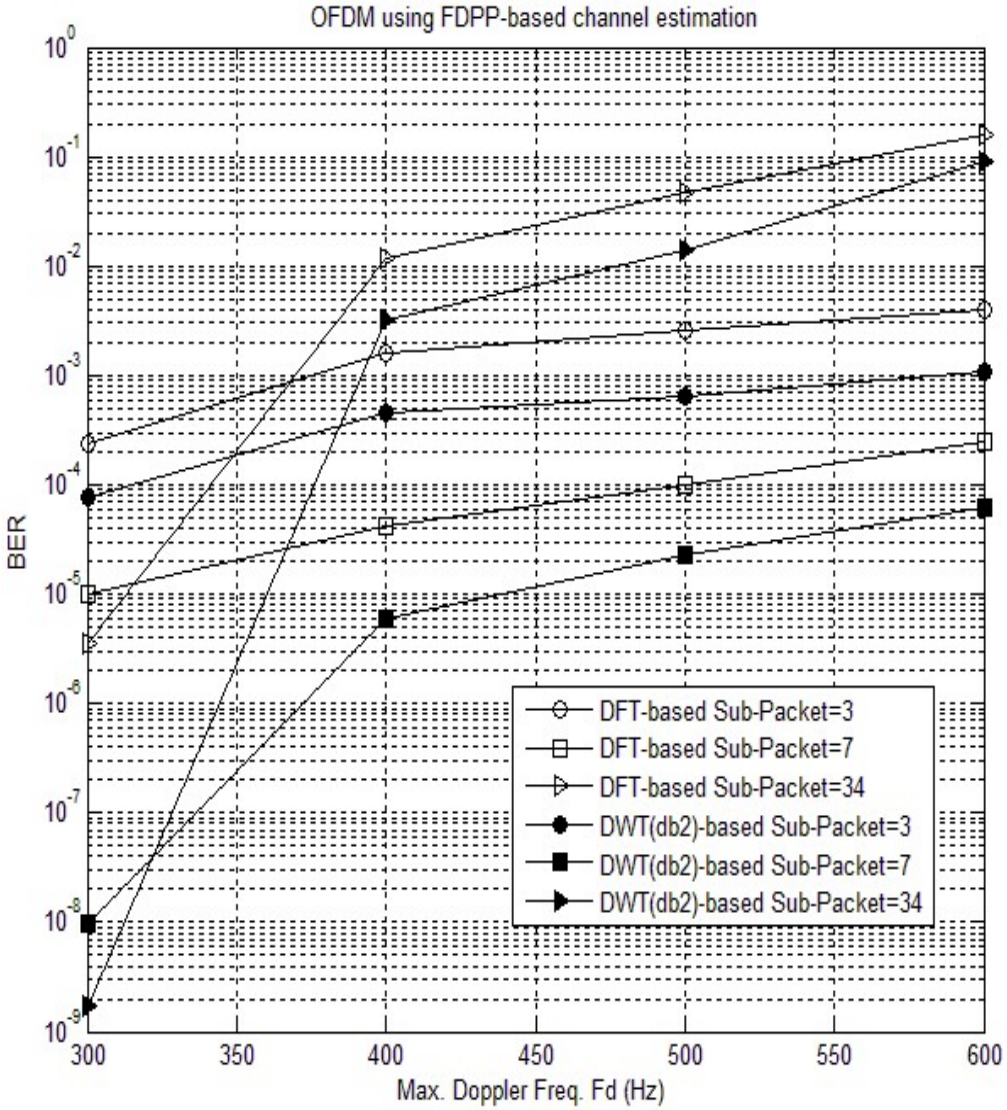


Figure 5-26 BER Performance of OFDM system employing FDPP-based channel estimation for different sub-packet sizes

Since the channel frequency response at the head and tail of any single sub-packet are different, and this disparity get increased with its length particularly at

higher Maximum Doppler frequency. This results in decreasing the time-varying fading tracking. As such, the BER performance when the sub-packet length is 34OFDM symbols is worse than when it is 3and 7.

Accordingly optimum sub packet size should be in the middle [2] as shown in the Figure **5-27**. From Figure **5-26**, 7 OFDM-symbol is found to be the optimum value of sub packet size. This result is true for both OFDM-FFT and OFDM-DWT. However, the BER performance for OFDM-DWT is better than OFDM-FFT when both schemes employ FDPP-based channel estimation over the all rage of f_D and for all values of sub packet as shown in Figure **5-26**.

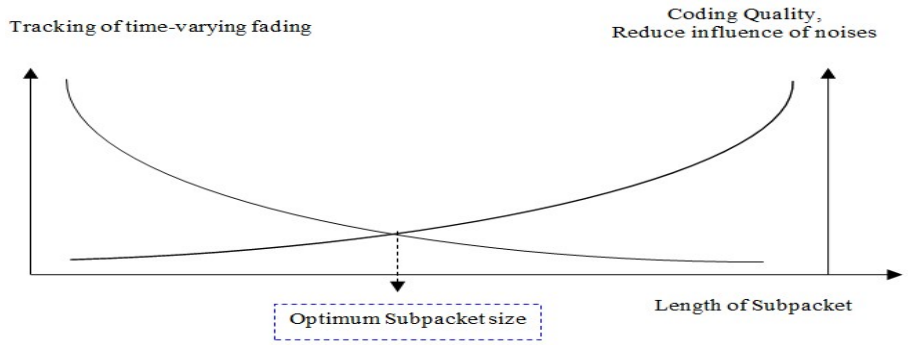


Figure 5-27 Optimum sub-packet size

5.11.4 ECG-PMS system Evaluation

The ECG-PMS system with OFDM transmission using FDPP-based channel estimation was evaluated under AWGN and multipath Rayleigh fading conditions with packet structure shown in Figure **5-23** in terms of the quality of

reconstructed ECG signal by simulation, with simulation parameters listed in Table **5-7**. Each OFDM packet contains 10 ECG symbols, and each ECG symbol contains 59 samples, and the ECG samples were normalized to 16 digits for the precision sake.

A normal heart rate i**s** about 70 beats per minute (BPM), so the sample and hold block stores 70 ECG symbols per minute. In this case 7 OFDM packets are needed to transmit 70 ECG symbols. Accordingly, the active time for our proposed system is 2.8 msec(7×100×4.0 μsec). The frequency of transmission is once per minute and thus the sleep time is (1 minute – 2.8 msec).

Table 5-7 Simulation parametersof ECG-PMS system

Modulation	16-QAM
Number of subcarriers	48 data , 16 null carriers
FEC scheme	Convolutional encoding/soft decision Viterbi decoding, ($R = 1/2, K = 7$)
Discrete Transform	DWT (db2) where decomposition stages L = $\log_2(64)$= 6
Guard interval duration	Nil in case OFDM-FFT
Number of CE symbols in the preamble N_{CE}	2
Number of OFDM symbols in one packet *N*	100
OFDM symbol duration	4.0 μ Sec in case OFDM-DWT
Data bit rate	24 M bit/sec in case OFDM-DWT

Interleaving type	Block interleaving
Interleaving size	Within one OFDM sub-packet [192 * sub packet size] bit
Propagation model	AWGN , multipath Rayleigh fading
Max Doppler frequency f_D	300 Hz
Bit energy / noise density E_b/N_o	15 dB
Channel estimation	FDPP with sub-packet = 7OFDM data symbols

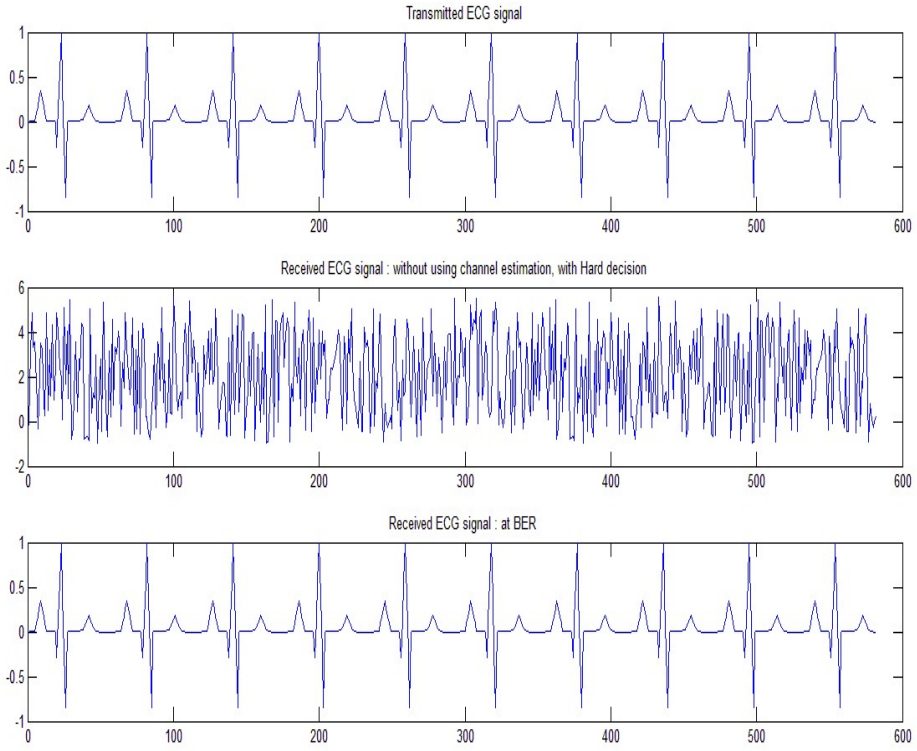


Figure 5-28 Transmitted and received ECG signals

The results obtained are exhibited in Figure **5-28** which show that the received signal is impacted by the channel. However, by using the OFDM-DWT with FDPP-based estimation, the transmitted ECG signal can be reliably recovered

even in the presence of considerable interference and noise. Hence, the substantial BER performance improvement results in considerable better quality of the recovered ECG signal.

5.12 Conclusion

The proposed DWT-based OFDM with FDPP-DA channel estimation for ECG-PMS system has been extensively studied using MATLAB software. The results reveal that the OFDM-DWT system has better performance over OFDM-FFT for both channel estimation techniques; FDPP-DA and HDPP-DA. Among all the wavelets families studied, it has been found that db2 wavelet family has by far the highest performance quality in terms of BER on average. Accordingly the aim of maintaining high quality ECG signal at the hospital site coupled with a low transmitted signal power at patient site in the ECG_PMS system has been realised.

Chapter 6

DWT and FIR filters Architectures for OFDM system on FPGA

6.1 Introduction

The simulation results in chapter five show that the OFDM-DWT systemhas better performance over OFDM-FFTfor FDPP channel estimation technique. However, for a real time system, more attention must be exerted on the hardware implementation aspect. In chapter four, the basic design of the OFDM-FFT was validated and implemented onto FPGAsuccessfully. Accordingly, this chapter proposes an iDWT/DWTbased OFDM system implemented on FPGA using Altera Cyclone IV E chip. High-level design software tools and Altera Quartus II v13.0 are used in the implementation and Verilog HDL was used for realization. The iDWT block was selected as the suitable for OFDM transmitter at patient site in ECG Patient Monitoring System (ECG-PMS). The system uses serial Distributed Arithmetic DA-based architecture to provide low power and efficient hardware utilization. Additionally, the DWT block selected to be suitable for OFDM receiver at hospital site used parallel DA-based architecture to provide high-speed and high-throughput. Both selected architectures are based on DA-based architecture for FIR filter. Compensation study was performed for the

proposed system among the wavelet families to determine the amount of resources consumed, the propagation delay, and the data rate.

The rest of this chapter is organised as follows, section 6.2 presents overview and literature review on the implementation of iDWT/DWT on FPGA for OFDM systems. An overview of FIR filter and its implementation on FPGA is shown in Section 6.3 and Section 6.4 respectively. In Section 6.5, the utilized Distributed Arithmetic algorithm for FIR filter implementation is evaluated with to respect to folding factor and filter order and allocated resources on the selected chip. Section 6.6 presents the proposed iDWT/DWT system implementation. In Section 6.7, the proposed system is investigated among different wavelet families. Finally, a brief conclusion was presented on Section 6.8.

6.2 Literature Review

6.2.1 FT and WT

The traditional Fourier Transform (FT) has the capability to provide only frequency information, hence only gives frequency representation of the signal. Furthermore, FT works only for stationary signals, where the signals are non-stationary in many real world applications. Wavelet Transform (WT) has been proposed as a technique to overcome this drawback [85]. WT has the capability of providing both frequency and time information simultaneously. Consequently,

WT gives a frequency-time representation of the signals. Moreover, WT has been used for processing both non- stationary and stationary signals [85]. Currently, more attention has been paid on the application of DWs in real-time signal processing as a result of these advantages [85]. In the 1980s, some researchers introduced a variety of wavelet functions $\phi(x)$, for example Grossmann and Morlet[86].

6.2.2 DWT

The momentous change in 1989 following the discovery of a Discrete Wavelet Transform (DWT) brought about a new complete representation called wavelet representation that was defined by Mallat in [86]. In the new representation, a significant improvement of the theory for the computation of DWT was achieved using multi-resolution signal decomposition theory.

The improvements on the efficiency of DWT computation over the last 20 years makes it a widely used in different areas of science and engineering. Researchers has demonstrated that DWT is particularly suited to some applications, such as digital signal processing and communications (one-dimensional DWT(1-D DWT))[11][87] image analysis (two-dimensional DWT(2-D DWT)),[12][88] video applications (three-dimensional DWT(3-D DWT))[13][89] and light field compression (four-dimensional DWT(4-D DWT))[14][90].

6.2.3 Link DWT to Filter

An iteration of filters with scaling can be used to realize a wavelet, where the filtering operations determine the resolution of the signal and subsampling (up-sampling and down-sampling) operations determine the scale [91].

6.2.4 Digital Filter

Digital filters are the basic and one of the most essential components of Digital Signal Processing (DSP) and telecommunication systems. Digital filters are classified into two main categories: Infinite Impulse Response (IIR) filter and Finite Impulse Response (FIR) filter. FIR filter can guarantee a linear phase response and system stability, which explains the wide use of FIR filters in a diverse area of DSP [92].

6.2.5 Computation of DWT

The computation of the DWT involves multiple levels of decompositions, which makes the computation of huge volumes of data during the processing such different levels computationally intensive [93]. Therefore, design, validation, and implementation of an efficient Very-Large-Scale-Integration (VLSI) architecture of

the DWT computation for real time applications is considered one of the main challenges faced by engineer designers.

6.2.6 DWT and FIR filter and FPGA

Traditionally, to implement such algorithms like filters and transforms for low rate applications, a programmable DSP chips are used. While on the other hand ASIC chips are used for high rate applications. The true realization of FIR and DWT in digital fabrication can be simplified by implementation of Field-programmable gate array (FPGA). FPGAs are widely used in a variety of applications, mainly in DSP and communications. FPGA can provide a great flexibility and reliability during the design and after shipping of consumer products. The flexibility allows easy maintenance because of its re-configurable logic elements. Also, current FPGAs have a parallel structure that can achieve high speed processing, consequently allows for high speed FIR and DWT realization[92][94].

6.2.7 DWT and FIR performances

The main determinant of the performance (including hardware issue) of whole DSP systems is the filters and transforms. Therefore, experts recommends that more attention is paid on optimization of these units [92].

6.2.8 Hardware real time

And due to the demand for real-time applications and portable devices, the hardware design and implementation has to be strongly realized with specific criteria such as real-time performance and heavy internal memory requirements, certain processing speed, area-efficient, high throughput, and low power consumption[85][95].

6.2.9 FIR on FPGA: Convolution and DA

The FPGA-based FIR filters can be achieved by two main approaches, namely: convolution approach, and Distributed Arithmetic (DA) approach. Implementations based on DA perform better than implementations that are based on convolution approach [96], as explained in details in next sections.

6.2.10 DWT on FPGA

The FPGA-based DWT can be achieved by two main computing approaches: including filter bank approach, which is called the convolution approach and lifting approach. However, convolution–based architectures for DWT require more arithmetic operations compared to the lifting-based architectures because they have shorter critical paths. Furthermore, keeping proper precision in

convolution-based architectures for DWT requires shorter multipliers and adders because of shorter intermediate variable widths compared to lifting-based [94].

There has been a significant amount of research and attention on determining the efficient hardware architectures and implementations of the DWT for different dimensions in VLSI. However, few of such research addressed the issue of an efficient FPGA implementation of DWT/iDWT architecture for signal processing and communications applications, especially OFDM systems has seen little research on this aspect [65][67][97][98].

6.2.11 Proposed work: DWT in designed health system

The aim of this chapter is to implement the proposed iDWT/DWT system by implementation of a DA-based architecture for fast computation of the 1-D DWT/iDWT with high-speed, low-power, area-efficient, efficient-hardware-utilizing, low-complexity, and high-throughput signal, which is utilized in OFDM models of ECG Patient Monitoring System (ECG-PMS) for wireless telemedicine application. This aspect is discussed in detail in the previous chapter.

In the portable wireless body transmitter unit at the patient site, a fully serial DA-based scheme for iDWT is implemented to support a higher hardware utilization and lower power consumption. While, a fully parallel DA-based scheme for DWT

is implemented at the base unit at the hospital/experts' site to support a higher throughput signal by speeding up the clock rate.

Following the completion of the proposed system, the behavioural level of the HDL models of the proposed system is developed and implemented to confirm its correctness in simulation. Then, extend over the simulation to synthesis and implementation for FPGA to confirm its validation.

6.3 FIR Digital Filter

6.3.1 Filter

A filter is a basic process that particularly changes the signal's amplitude-frequency, phase-frequency, and/or shape wave to extract the desired signal. A filter fulfil three main objects: remove/reduce un-wanted feature/components (such as random noise) to increase the quality of a signal, extract desired data/information from signals (like the information lying within a specific range of frequency), and separate two or more combined signals, like the approach used to increase the efficiency of common communication channel. Figure **6-1** shows a block diagram for basic filter [99].

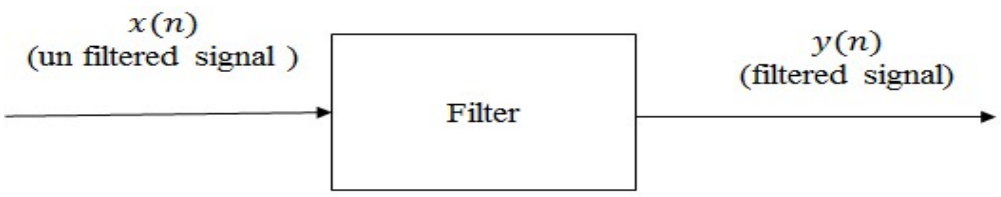


Figure 6-1 Block diagram of a basic filter

6.3.2 Analog Filter and Digital Filter

A Signal can be divided into two main types: Analog signal and Digital signal. An analog signal is a function of a continuous variable time (t) and it may have unlimited/infinite number of values, but a digital signal is a function of discrete variable time (n). Based on this, mainly two types of filters: Analog Filters, and Digital Filters. An analog filter is a mathematical algorithm that operates on an analog input signal x (t) to produce an analog output signal y (t) to gain over filter object, in contrast a digital filter has a digital signal at both its input x (n) and its output y (n). The advantages of Digital filters over Analog filters in [100] leads to the essential roles digital filter plays in Digital Signal Processing System (DSPS), thus a digital filter is recommended over analog filter for many applications such as data transmission, biomedical signal processing [99].

6.3.3 IIR and FIR Filters

Traditionally, common digital filters are classified into two main categories according to their impulse response: Infinite Impulse Response (IIR) filter and Finite Impulse Response (FIR) filter. Figure **6-2** represents both FIR and IIR digital filters by their impulse response sequence, $h(k)$ for $k = 0,1,2,$ ...[99].

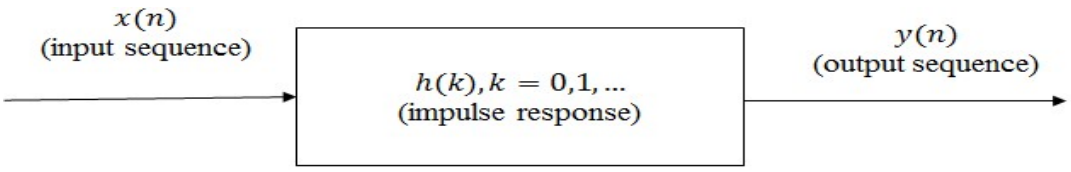


Figure 6-2 A conceptual representation of a digital filter

Output signal are produced by a convolution sum of input signal and impulse response as given in (6.1) and (6.2) for FIR and IIR filter respectively [99].

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \tag{6.1}$$

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \tag{6.2}$$

It is obvious from (6.1)that the impulse response is of finite duration for FIR filter because $h(k)$ has only N values. Also FIR filter operates only on current and past values of input, thus it is called non-recursive filter. However, it is evident from (6.2) that the impulse response is of infinite duration for IIR filter since $h(k)$ has infinite values. Also IIR filter operates on current and past values of input and on past values of output, thus it is called recursive filter [99][100].

An alternative representation for the two filter types is the transfer function for them. (6.3) and (6.4) are the transfer functions for FIR and IIR digital filter respectively, where $H(z)$ is the transfer function of the filter. For FIR filter,$h(k)$

(for $k = 0,1,2,\ldots,N-1$) are the impulse response coefficients of the filter and N is the number of filter coefficients, that is the filter length [99].

$$H(z) = \sum_{k=0}^{N-1} h(k) z^{-k} \tag{6.3}$$

$$H(z) = \sum_{k=0}^{N} b_k z^{-k} (1 + \sum_{k=1}^{M} a_k z^{-k}) \tag{6.4}$$

6.3.4 Choose between FIR and IIR Filters

A comparison between FIR and IIR filters by [100] provide a broad guideline on when to use IIR or FIR filters, by matching the desired application requirements and the relative advantages of the two filter types. Time domain equation on (6.1) shows that FIR filter is a non-recursive filter, and thus implementing FIR filter on this form guarantee that the filter are always stable. However the stability of IIR filter cannot be guaranteed always. FIR filters can provide an exact linear phase response, thus FIR filter are used for many applications that need this important demand such as data transmission and biomedical signal processing [99].

The effects of finite word-length suffer less non-recursive FIR filter than IIR filter, and FIR filter is very simple to implement [99]. These two advantages leads to high number of FPGA chip with the architecture that is suited to FIR filtering.

6.4 FIR Implementation

6.4.1 FIR filter Structure

In order to realize a digital FIR filter, the given differential equation ((6.1)) is presented in the form of block diagram. There are different types of structures used to realize the FIR filter: Direct Form Structure, Cascade Form Structure, Transpose Form Structure, Lattice Structure, and Parallel Filter Structure. Many factors and trade-offs affects the choice between these structures, where the most common structure used to realize no-recursive filter like FIR filters is direct form structure. This structure is very easy to program, it needs a minimum number of components, and requires un-complicated memory access for data [99][101]. In addition, this structure can be implemented efficiently by most FPGA chips because these chips have tools tailored to transversal FIR filtering.

FIR Filter Difference Equation ((6.1)) has been presented by a Direct-Form Structure of FIR filter in Figure **6-3**. Where $x[n]$ and $y[n]$ represents the input and output time series respectively, N is the number of coefficients $h(k)$ of the filter, Z^{-1} is a delay unit to produce the prior input samples, (×) is a multiplier unit, and (+) is an adder unit. $(N-1)$ is called filter order which is also commonly referred to as filter tap [92].

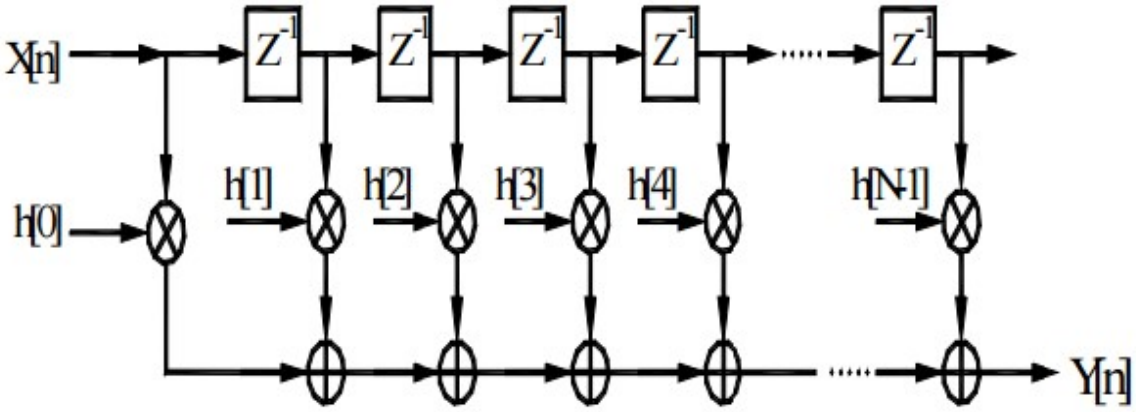


Figure 6-3 Direct FIR filter structure

6.4.2 Convolutional approach

It is obvious that the output $y[n]$ at time n_1 is the summation of all the delayed input samples multiplied by the convenient coefficients, in another words, it is the convolution of the latest N input samples with the filter coefficients. Thus an (N-1) order filter has N coefficients and needs N Multipliers and (N-1) adder. This traditional implementation of FIR is known as convolution approach[92][102].

In convolution approach, each filter tap consists of delay elements, an adder, and a multiplier. This can be implemented using a Multiply-Accumulator (MAC) unit. So N MAC unites needed per input sample to compute the result sample that requires N MAC cycles before the next input sample can be processed. As the filter order increases, the filter throughput is proportionally decreased and this is one of the major drawbacks of the convolution implementation[96][102]. By gathering an N MAC unites in a single MAC engine; a parallel convolution

implantation can be implemented by using N MACs engine. Thus the performance can be speed up by N times [92].

However, multiplication is a repeated addition that makes it the strongest operation. A general purpose multiplier requires large portion of FPGA chip area, especially for high order filter with high power consumption. Thus a convolution approach is expensive to implement in FPGA hardware due to logic complexity, power consumption and area usage[92][102][103].

Most of the running generation of FPGAs handle multiplication operations by having embedded multipliers. However, for high speed requirements the accuracy of the computations is limited because these multipliers size are limited by only 18 bits. Also the number of these multipliers is typically limited [92].

For Linear Time Invariant (LTI) FIR digital filter, the coefficients do not vary with time. Thus all multiplications are with constants that can be implemented by Multiple Constant Multiplication (MCM) unit. So the full flexibility for general purpose multiplier is not required, that leads to the use of algorithms for constant multiplication to reduce the area usage and power consumption[92][104].

6.4.3 Distributed Arithmetic (DA)

Distributed Arithmetic (DA) algorithm is the most suitable alternative for convolution approach in the case of constant coefficients. DA algorithm is suitable forportable applications, because in DA algorithm the costly MACs units are replaced with Look-Up-Tables (LUTs) and shifts that reduces the power consumption [96][102].

The mathematics behind DA has been clearly explained in[92][96][105]. Based on that, DA implementation of MCM can be achieved on FPGA through shift registers, LUT, and scaling accumulator. All possible partial products over the constant coefficients space are pre-computed off-line and stored in the LUT. Therefore, the size of LUT is equal to $2^{(Ne\ -1)}$ (Ne is the number of coefficients). Figure **6-4** and Figure **6-5** show a DA implementation along with a conventional implementation of 4 elements MCM, and Table **6-1** shows coefficient values of DA's LUT [96].

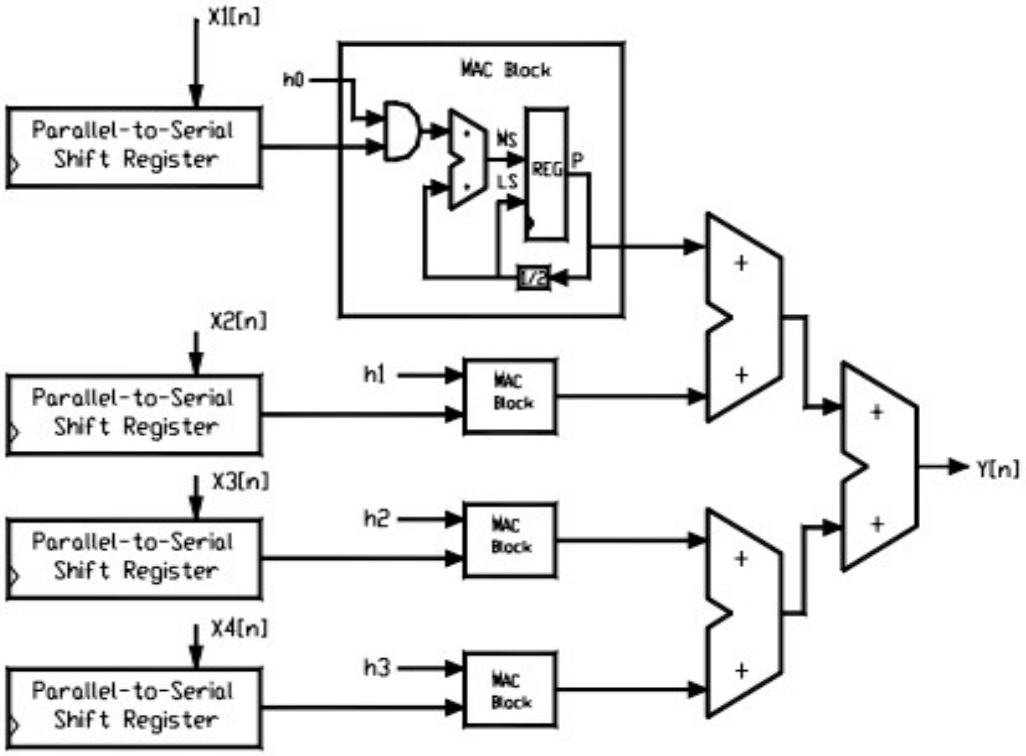


Figure 6-4 Conventional implementation

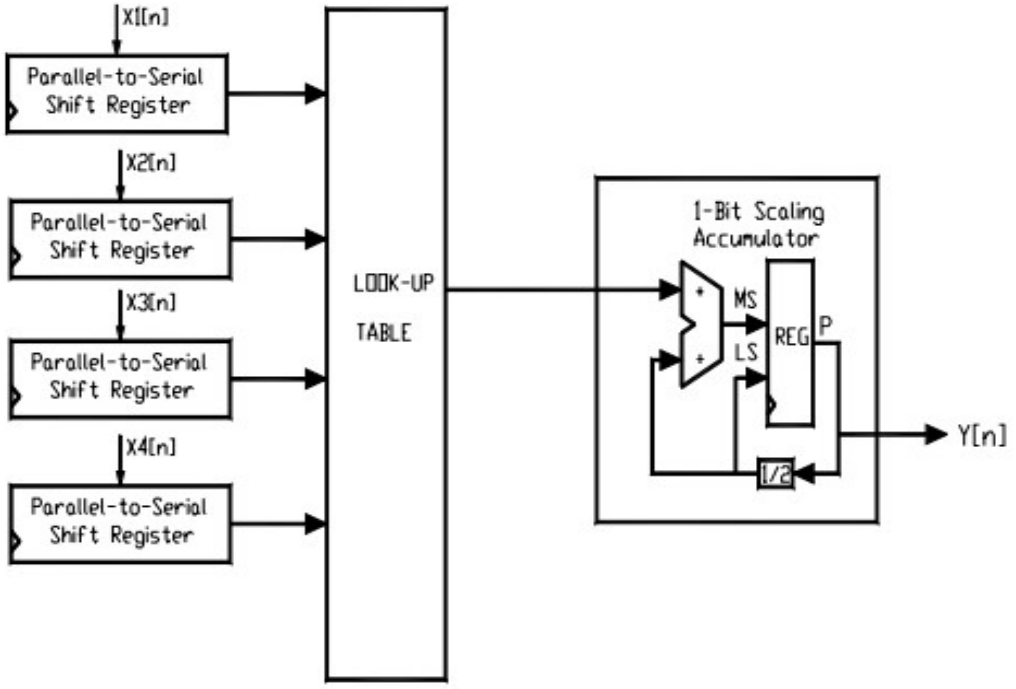


Figure 6-5Distributed Arithmetic implementation

Table 6-1 Coefficient values of DA's LUT.

Address	Data
0000	0
0001	h_0
0010	h_0+h_1
…….	….
1111	h0 + h1 + h2 + h3

By cascading the shift registers, the DA implementation of FIR filter can be achieved as shown in Figure **6-6** . In this architecture the number of coefficients is equal to 4. However, as the number of FIR filter coefficients increases, the LUT size exponentially increases. Thus the LUT access time increases, which consequently decreases the speed of the whole system [96][105].

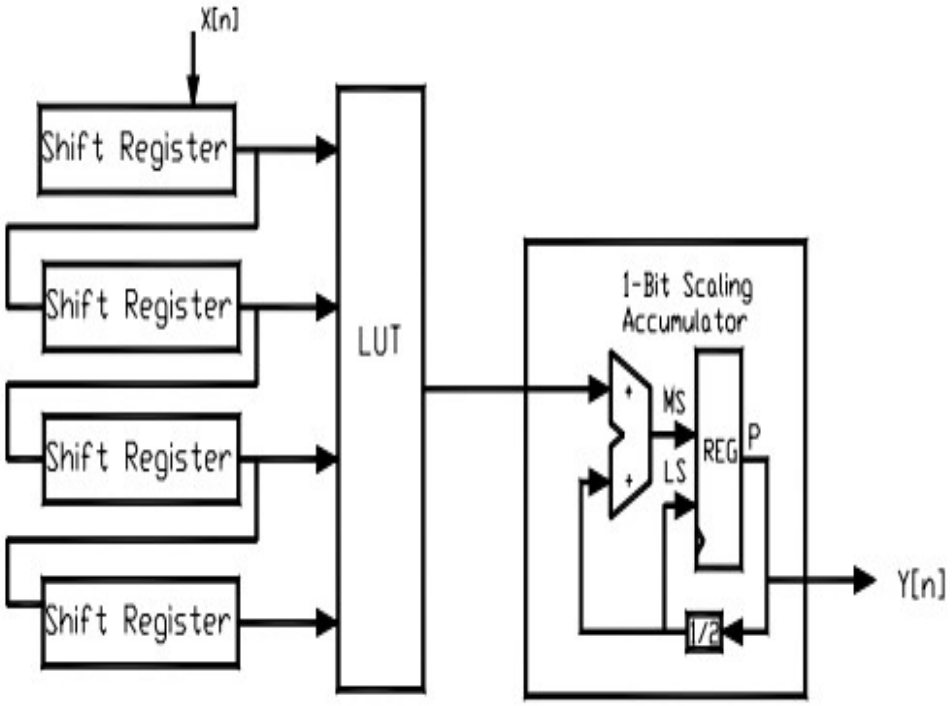


Figure 6-6Serial DA FIR Filter

To overcome this performance limitation, a partitioned LUT technique was used. In Figure **6-7**, the m-bit LUT used in Figure **6-6** have been divided into two m/2-bit LUTs. Then the output of the two LUTs are added before feeding into the scaling accumulator [96][106].

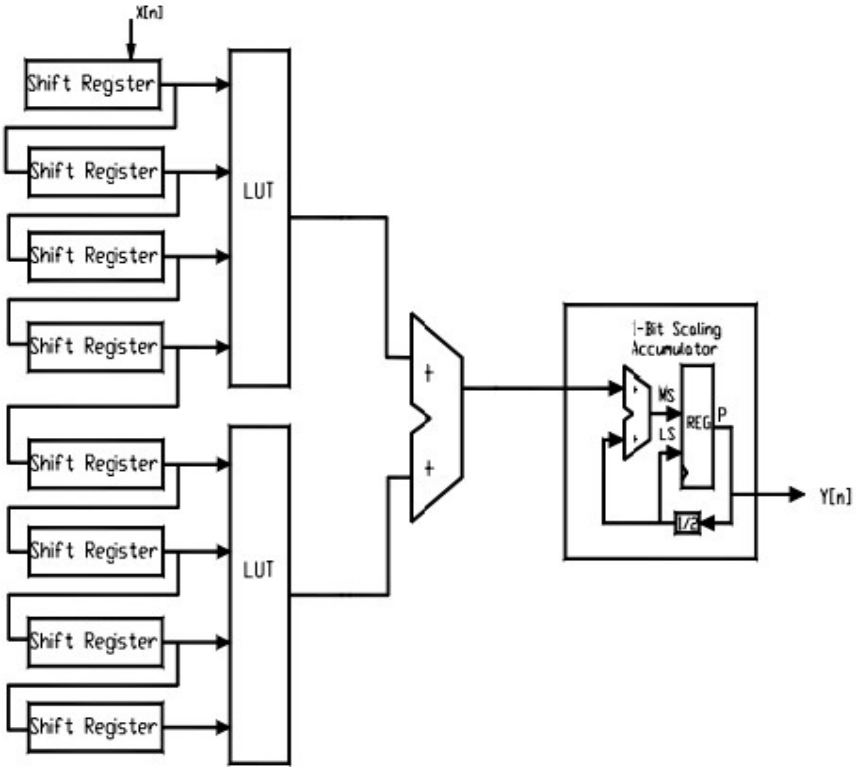


Figure 6-7 DA (with partitioned LUT technique) implementation of the FIR filter

The DA implementation for FIR filter shown in Figure **6-6** is called Serial DA (SDA) architecture because it is serial in nature. If w is the number of bits to represent a precision data input sample, then w clock cycles are required for FIR filter to produce the relevant output. In addition, until the complete current input

sample (all bits) have been processed, the processing of the next input sample cannot commence. Parallel DA (PDA) architecture is proposed to overcome this performance limitation problem[92][96].

6.5 Simulation, Experiments and Results for FIR implementation

The goal of these experiments described in this section was to compare the number of resources consumed by DA method for FIR filter with different filter order. Besides the filter order, an investigation the effects of folding factor on the resources were carried out.

6.5.1 System parameters

The FDATool in MATLAB was used to calculate the filter coefficients. For our experiments, we considered FIR filter of various sizes (4, 8, 16, 32, 64, 128, 512 and 1024 tap filters). We targeted the ALTERA Cyclone IV FPGA chip on DE2-115 board for our experiments. The constant coefficients were normalized to 16 digit of precision and the input samples were assumed to be 16 bits wide. We used the *ALTERA Quartus II Software (V.13)* Environment to perform synthesis and implementation of the designs. The different DA-based architectures for FIR filter was coded in Verilog HDL.

6.5.2 Experiments 01

Table **6-2** shows the resources utilized for the various numbers of bits operated in parallel for the 8-tap FIR filter implemented using the DA algorithm. From the result, we can observe that by increasing the throughput 8 times, the allocated LUT size also increased by 8 times. However the other resources are increased only by about 4 times.

Table 6-2 Filter Synthesis using partly PDA for 8-tap FIR filter

No. of bits operated in parallel	1	2	4	8
LUT total size (bits)	544	1,108	2,216	4,432
Logical Elements (LEs) (used, utilization)	237 (<1%)	321 (<1%)	485 (<1%)	897 (<1%)
Combinational Functions (used, utilization)	178 (<1%)	252 (<1%)	419 (<1%)	770 (<1%)
Registers (used, utilization)	166 (<1%)	200 (<1%)	310 (<1%)	666 (<1%)

6.5.3 Experiments 02

Table **6-3** shows the resources utilized for the various numbers of bits operated in parallel for the 256-tap FIR filter implemented using the DA algorithm. From the result, we can observe that by increasing the throughput 8 times, the allocated LUT size also increased by 8 times. Where, the other resources also increased by more than 8 times (LEs, Combinational Functions, and Registers by 8.6, 7.5, and 10.7 times respectively). Actually, as for parallel implementation, duplicate circuits will be used and hence more resources were needed.

Table 6-3 Filter Synthesis using partly PDA for 256-tap FIR filter

No. of bits operated in parallel	1	2	4	8
LUT total size (bits)	10,626	21,252	42504	85,008
Logical Elements (LEs) (used, utilization)	1,635 (1%)	3,093 (3%)	5,990 (5%)	14,101 (12%)
Combinational Functions (used, utilization)	1,538 (1%)	2,971 (3%)	5,857 (5%)	11,650 (10%)
Registers (used, utilization)	973 (<1%)	1,779 (2%)	3,433 (3%)	10,434 (9%)

6.5.4 Experiments 03

Table **6-4** shows the resources utilized for the various filters tap using fully SDA algorithm. From the result, it can be observed that, in the case of fully SDA, increasing of filter order has a slightly negatively effect on the resource utilization. FIR filter order was increased from 4 tap to 1025 tap, while the LEs increased by 27 times, which is acceptable because it represents on 3% increase on the total LEs in the target FPGA chip.

Table 6-4 Filter Synthesis using fully SDA for variety-tap FIR filter

Filter Order	LUT total size (bits)	Logical Elements (LEs) (used, utilization)	Combinational Functions (used, utilization)	Registers (used, utilization)
4	298	195 (<1%)	138 (<1%)	145 (<1%)
8	554	237 (<1%)	178 (<1%)	166 (<1%)
16	1,002	295 (<1%)	237 (<1%)	202 (<1%)
32	1,830	416 (<1%)	352 (<1%)	267 (<1%)
64	3,208	606 (<1%)	543 (<1%)	364 (<1%)
128	5,840	981 (<1%)	899 (<1%)	595 (<1%)
256	10,626	1,635 (1%)	1,538 (1%)	973 (<1%)
512	19,262	2,903 (3%)	2,816 (2%)	1,688 (1%)
513	20,096	2,944 (3%)	2,854 (2%)	1,742 (2%)
1024	34,440	4,955 (4%)	4,839 (4%)	2,893 (3%)

1025	36,076	5,108 (4%)	5,038 (4%)	2,974 (3%)

6.5.5 Conclusions on these Experiments

DA architecture is a Serial DA by default, which means that it operates only one bit of data input sample on each clock cycle. Thus, SDA architecture can potentially limit the throughput. The throughput was improved by using PDA architecture, which means that it operates on multiple bits in parallel on each clock cycle. Moving from Fully SDA to fully PDA architecture can significantly affect the resource utilization on FPGA, particularly in high order FIR filter. However, its effect is slightlyin the case of low order FIR filter. This is can be observed from the first two experiments.

6.6 Proposed iDWT/DWT system implementation using DA based FIR filter

An efficient way of computing DWT was developed by Mallat[86] in 1988 , where Mallat algorithm (which is also called a Mallat-tree decomposition) was achieved by passing a discrete time-domain signal through a low-pass and high-pass filters as shown in Figure **6-8** (a). Connecting the continuous time multi-resolution to discrete-time filters is considered one of the DWT significance. A sequence $X[n]$ donates the input signal, H_0 donates the high pass filter that produce a detail information $Y_0[n]$, and G_0 donates the low pass filter that produce the approximation information $Y_1[n]$. Both the detail and approximation components

are then down-sampling by removing every second sample to shortening them[91][107].

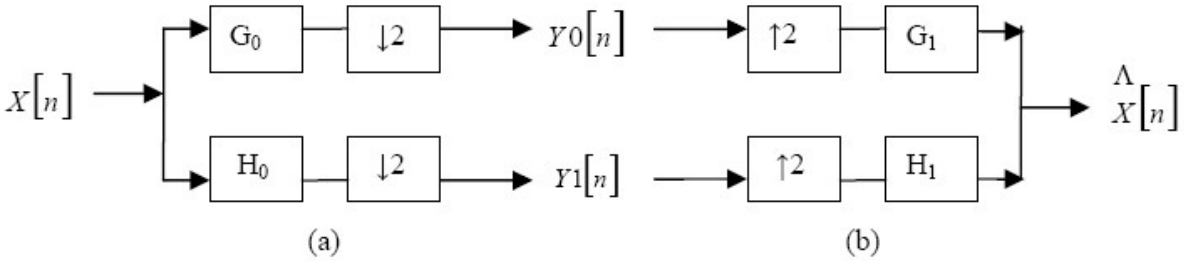


Figure 6-8 Direct form structure of (a) DWT- decomposition/analysis process (b) iDWT-reconstruction/synthesis process

We have seen how the DWT can be analysed or the signal decomposed. Now we will show how the iDWT can be synthesised, or reconstruct the $Y_0[n]$ and $Y_1[n]$ components to assembled it back into the original signal. The approximation component $Y_1[n]$ is up-sampled then passed through the high pass filter H_1, while the detail component $Y_0[n]$ is up-sampled then passed through the low pass filter G_1. After that, they were combined as shown in Figure **6-8** (b). Up-sampling process was achieved by inserting zeros between samples to lengthen the signal components. Choosing the analysis and synthesis filters are the crucial issue for achieving a perfect reconstruction, thus these filters have to satisfy certain conditions[91][107]. In our work, a MATLAB is used to calculate the filters coefficients for the selected wavelet families.

There are different types of structures used to realize the filter bank DWT: Direct Form Structure, Polyphase Structure, Lattice Structure, and Lifting Structure.

Many factors and trade-offs affects the choice between these structures, where the most common structure used to realize filter Bank DWT is direct form structure. This structure is very easy to program, has shorter critical paths, and it needs shorter multipliers and adders. Furthermore, the direct form structure for DWT can be efficiently implemented by most FPGA chips, this is because the direct form structure for DWT consists basically of FIR filters and the FIR filters can also be implemented efficiently by most FPGA chips [99][100][101]. A direct form structure of DWT and iDWT are shown in Figure **6-8** (a) and (b) respectively.

We used a direct form structure for filter bank DWT called a traditional conventional-based DWT architecture. This architecture has more delays and hardware utilization. To overcome this problem, a DA-based DWT architecture is implemented by employing/applying the DA-based FIR implementation on a filter bank–based DWT architecture, hence we achieved a Multiplier-less hardware implementation approach.

The portable device at the patient site requires a lower power consumption to have a longer operation time. Accordingly, the proposed iDWT/DWT system adopts fully SDA-based scheme for iDWT to provide a low-power, area-efficient, and efficient-hardware-utilizing criteria. On the other hand, at the hospital site the model requires a higher speed to allow the medical experts to detect the

emergency case early. Accordingly, the proposed iDWT/DWT system adopts fully PDA-based scheme for DWT to provide high-speed, and high-throughput.

6.7 Simulation, Experiments, and Results for proposed iDWT and DWT architectures

The goal of our experiments in this section was to compare the number of resources consumed by iDWT/DWT system that uses a fully SDA method for iDWT and fully PDA method for DWT using different wavelet mothers. Besides the resources, we also compared the data rate and measured the propagation delay.

6.7.1 Pre Experiments

Firstly, MATLAB was used to calculate the iDWT/DWT coefficients for different wavelet families, where Table **6-5** shows wavelet coefficients for haar wavelet, which we used later in this section.

Secondly, a basic iDWT/DWT using direct form structure was implemented in MALTAB to calculate the BER caused by sub-sampling blocks. Results in Table **6-6** show that the error caused by sub-sampling can be ignored.

Table 6-5Haar Wavelet coefficients

G_0(Lo_D)	H_0(Hi_D)	G_1 (Lo_R)	H_1(Hi_R)
0.7071067811865	-0.7071067811865	0.7071067811865	0.7071067811865
0.7071067811865	0.7071067811865	0.7071067811865	-0.7071067811865

Table 6-6 BER for iDWT-DWT model using direct form structure

Wavelet families	BER
haar	-6.261338147750939e-12
db2	+5.958796545478258e-12
sym2	-4.954735864258757e-12
coif1	-7.085647895214778e-12
bior 1.3	+6.365472536589325e-12

Thirdly, the architectures for fully serial DA-based scheme for iDWT and fully parallel DA-based scheme for DWT was coded in Verilog HDL using the platform of Quartus-II of version 13.0 to confirm their correctness in simulation. The constant coefficients were normalized to 16 digit of precision and the input samples were assumed to be 4 bits wide. Figure **6-9** and Figure **6-10** shows a RTL Schematic Report for fully SDA-based for iDWT and fully PDA-based for DWT respectively. Figure **6-11** and Figure **6-12** shows the simulation output for iDWT and DWT. In both architectures there is a latency at the beginning because of using pipeline registers, however, it is 11 clock cycles in SDA-based and only 4 in PDA-based. Area Utilization report is shown in Figure **6-13** and Figure **6-14**. Max frequencies are shown in the Performance report in Figure **6-15** and Figure **6-16**.

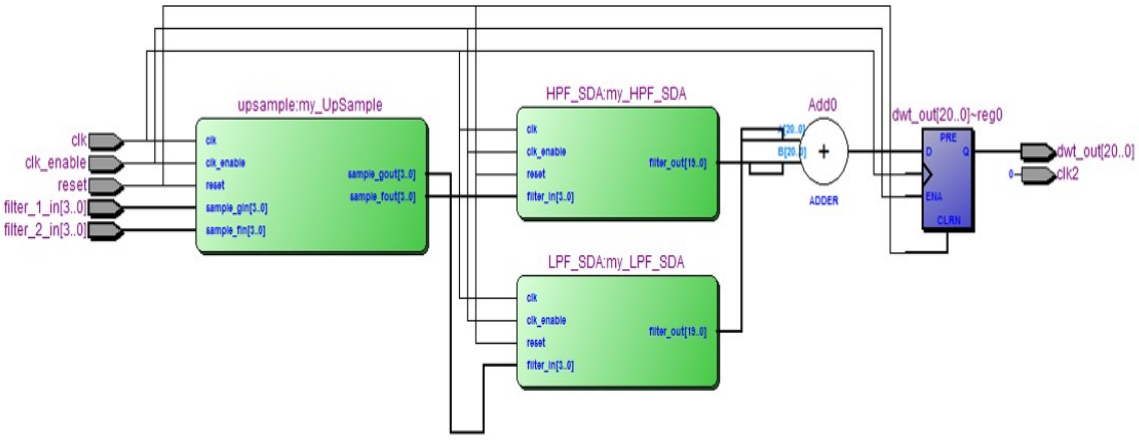


Figure 6-9 RTL Schematic Report for fully SDA-based for iDWT

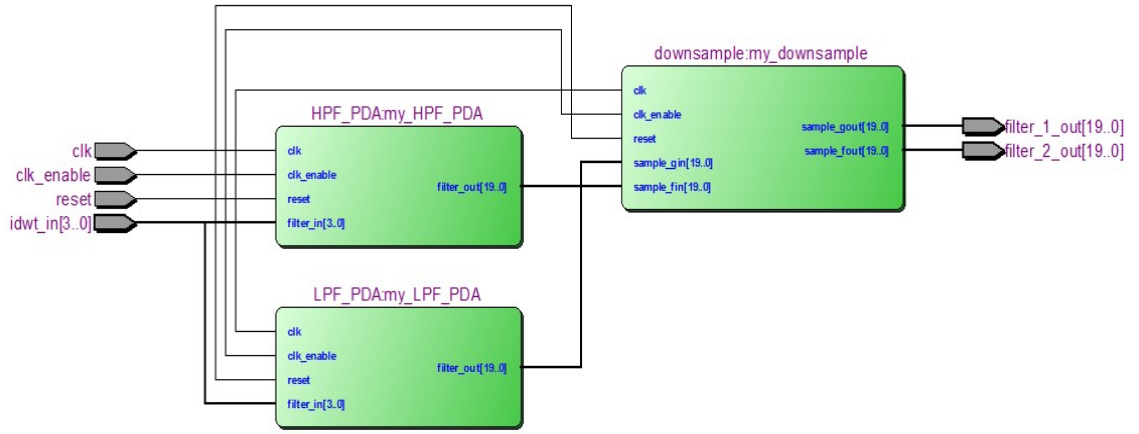


Figure 6-10 RTL Schematic Report for fully PDA-based for DWT

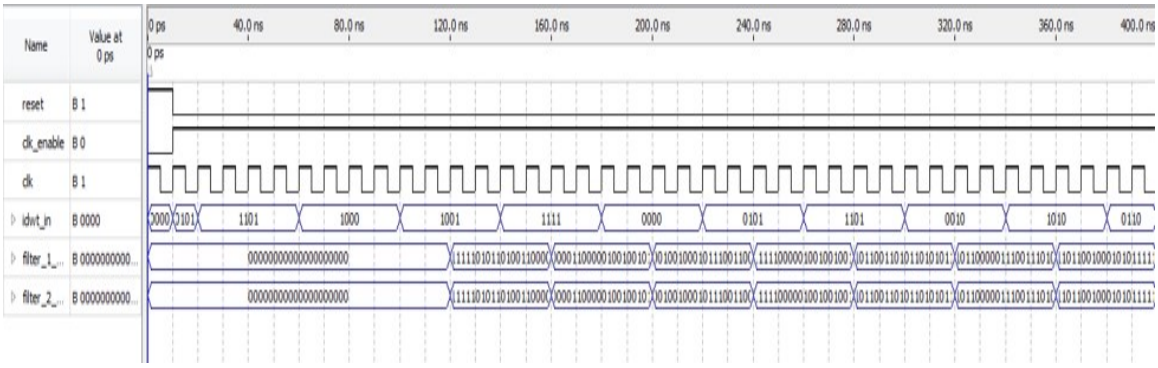


Figure 6-11 Simulation output for fully SDA-based for iDWT

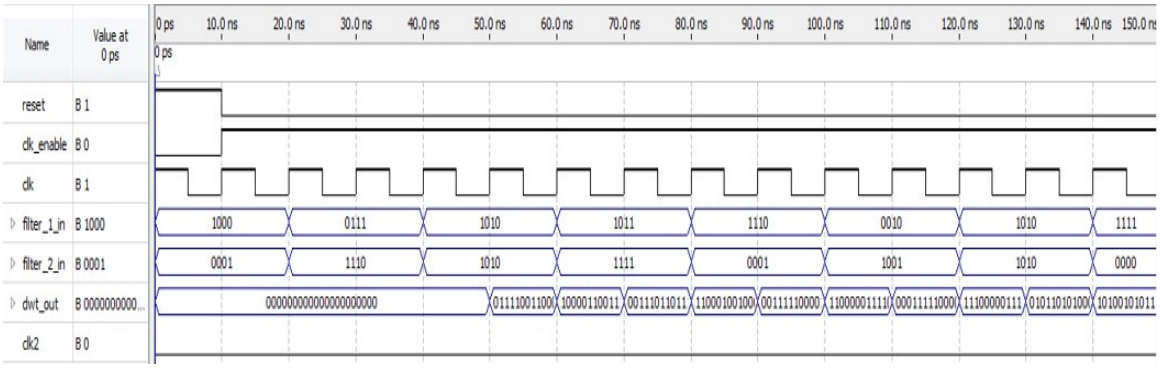


Figure 6-12 Simulation output for fully PDA-based for DWT

Fitter Summary	
Fitter Status	Successful - Mon Jul 20 05:15:57 2015
Quartus II 64-Bit Version	13.0.0 Build 156 04/24/2013 SJ Full Version
Revision Name	iDWT_SDA
Top-level Entity Name	iDWT_SDA
Family	Cyclone IV E
Device	EP4CE115F29C7
Timing Models	Final
Total logic elements	125 / 114,480 (< 1 %)
Total combinational functions	43 / 114,480 (< 1 %)
Dedicated logic registers	112 / 114,480 (< 1 %)
Total registers	112
Total pins	47 / 529 (9 %)
Total virtual pins	0
Total memory bits	0 / 3,981,312 (0 %)
Embedded Multiplier 9-bit elements	0 / 532 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 6-13 Area Utilization Fitter Summary report for fully SDA-based for iDWT

Fitter Summary	
Fitter Status	Successful - Mon Jul 20 05:58:41 2015
Quartus II 64-Bit Version	13.0.0 Build 156 04/24/2013 SJ Full Version
Revision Name	DWT_PDA
Top-level Entity Name	DWT_PDA
Family	Cyclone IV E
Device	EP4CE115F29C7
Timing Models	Final
Total logic elements	165 / 114,480 (< 1 %)
Total combinational functions	155 / 114,480 (< 1 %)
Dedicated logic registers	151 / 114,480 (< 1 %)
Total registers	151
Total pins	33 / 529 (6 %)
Total virtual pins	0
Total memory bits	0 / 3,981,312 (0 %)
Embedded Multiplier 9-bit elements	0 / 532 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 6-14 Area Utilization Fitter Summary report for fully PDA-based for DWT

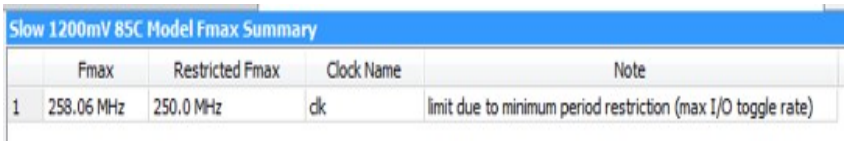


Figure 6-15Fmax. Summary Report for fully SDA-based for iDWT

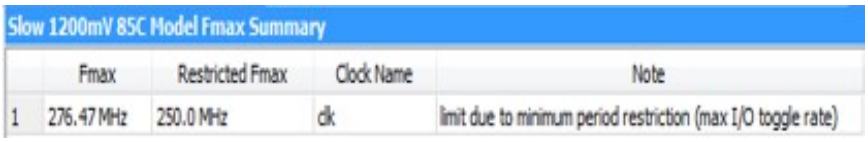


Figure 6-16Fmax. Summary Report for fully PDA-based for DWT

6.7.2 Experiments

Fourthly, investigation of the effects of wavelet families on DA based architecture for iDWT/DWT proposed system. Table **6-7** shows the resources utilized, data rate, and propagation delay for the various wavelet families for the iDWT was implemented using the fully SDA method, while Table **6-8** for the DWT was implemented using the fully PDA method.

Table 6-7 Compassion of Resources Usage, Data Rate and Propagation delay for fully PDA-based for DWT

Parameters	Haar	db2	sym2	coif1	bior1.3
Total logic elements used (Used, Available, Utilization)	171 / 114,480 (< 1 %)	277 / 114,480 (< 1 %)	304 / 114,480 (< 1 %)	532 / 114,480 (< 1 %)	447 / 114,480 (< 1 %)
Total combinational functions used	155 / 114,480 (< 1 %)	260 / 114,480 (< 1 %)	296 / 114,480 (< 1 %)	516 / 114,480 (< 1 %)	441 / 114,480 (< 1 %)
Total registers used	151 /	174 /	176 /	190 /	185 /

	114,480 (< 1 %)	114,480 (< 1 %)	114,480 (< 1 %)	114,480 (< 1 %)	114,480 (< 1 %)
Fmax: Input Bit Rate (M bit/sec) Out. Symbol Rate (M symbol/sec)	250MHz 250*4 250	199MHz 199*4 199	219MHz 219*4 219	163MHz 163*4 163	187MHz 187*4 187
Propagation delay (Minimum period)	8 ns	17 ns	17 ns	17 ns	17 ns

Table 6-8 Compassion of Resources Usage and Data Rate for fully SDA-based for iDWT

Parameters	Haar	db2	sym2	coif1	bior1.3
Total logic elements used	196 / 114,480 (< 1 %)	105 / 114,480 (< 1 %)	151 / 114,480 (< 1 %)	257 / 114,480 (< 1 %)	251 / 114,480 (< 1 %)
Total combinational functions used	74 / 114,480 (< 1 %)	58 / 114,480 (< 1 %)	69 / 114,480 (< 1 %)	129 / 114,480 (< 1 %)	119 / 114,480 (< 1 %)
Total registers used	173 / 114,480 (< 1 %)	79 / 114,480 (< 1 %)	120 / 114,480 (< 1 %)	189 / 114,480 (< 1 %)	189 / 114,480 (< 1 %)
Fmax: Input Bit Rate (M bit/sec) Out. Symbol Rate (M symbol/sec)	239 MHz 239 239/4	250 MHz 250 250/4	250 MHz 250 250/4	187 MHz 187 187/4	208 MHz 208 208/4
Propagation delay (Minimum period)	7 ns	7 ns	7 ns	7 ns	7 ns

With respect to fully PDA-based architecture for DWT, from the result in Table **6-7** , we can observe a reduction of about 60% in the number of logical elements and combinational functions when using haar wavelet instead of coif1 or bior1.3, and about 40% reduction compared to using db2 or sym2. Moreover, we observed about 50% reduction in the minimum period when using haar. For the data rate, the worst rate was observed for coif1 wavelet family, and the highest was observed for haar.

With respect to fully SDA-based architecture for iDWT, from the result in Table **6-8** , we can observe a reduction of about 20% in the number of logical elements and combinational functions when using haar wavelet instead of coif1 or bior1.3, however, an increase of about 60% was observed when compared to db2 or sym2. The minimum period is same for all wavelet families. For the data rate, the worst rate was observed for coif1 wavelet family, and the highest was observed for bior1.3.

This is because the wavelets families are different in terms of number of coefficients for their filters. For example, haar wavelet has 2 coefficients for each filter, db2 has 2 and coif1 have 6. Moreover, section 7.5 shows that as the number of filter order increases the negative effect on consumed resources increases.. In the case of wavelet families, we observed same number of coefficients, contrary to the case of coif1 and bior1.3 where both has 6 coefficients for each filter, some filters for bior1.3 has zeros coefficients, and that makes a slight reduction in the resources utilization over coif1.

With respect to Sym2 wavelet, from the results in Table **6-7** and Table **6-8**, we can observe a significant reduction in the resources consumed and minimum period in case of using fully SDA-based. From another point of view, the symbol rate is N times more in case of fully PDA-based over the fully SDA-based, where N is the number of bit in each input sample.

Finally, realizing it successfully with the FPGA device Cyclone IV E family, EP4CE115F29C7 Device form Altera Crop validated the proposed iDWT/DWT system.

6.8 Conclusion

iDWT/DWT system using DA-based architectures are proposed in this chapter to be used for OFDM-DWT model used in the ECG-PMS, where a fully serial DA-Based are used for iDWT to support low-power and efficient-hardware-utilizing, and a fully Parallel DA-Based are used for DWT to support high-speed and high-throughput. The proposed system has been realized in Verilog HDL and implemented using Altera EP4CE115F29C7 Cyclone IV E FPGA device.

The proposed system was investigated among different wavelet families. However, our results could not be compared with the other researches results because of using different technology in terms of hardware and software. In addition, their proposed systems for different applications lead to different main requirements.

Chapter 7

Conclusions and Recommendations for Future Work

7.1 Summary

For the provision of high data rates along with robustness against the adverse effects of multipath fading and inter-symbol interference, current and projected radio communications systems recourse to the adoption of OFDM scheme. There available many OFDM standards such as 802.11 and 802.16[9][66][68]. The spectral efficiency and BER performance of an OFDM system can be improved if DWT transforms were used instead of FFT[4].

Over the years, FPGA has been used in a wide range of communication applications market venues having different requirements in terms of power efficiency, performance, and cost. On top of the attractive merit of reconfiguration, FPGA also possesses the combined features found in ASIC and processor, and hence FPGA has been recommended in the implementation of OFDM systems on FPGA chips[9].

The challenges facing design engineers during the design, validation, and implementation of an OFDM transceiver on an FPGA chip are how to improve speed and area on the chip simultaneously by optimizing a number of resources such as LUTs and memory allocated on the target FPGA chip to achieve an effective high performance design cost that is highly sought in wireless communication application[9][66][68].

7.2 Summary of the thesis conclusions

The conclusions of this work can be summarised by each Chapter as follows:

Chapter 2: many issues regarding the digital hardware has been explored in this chapter. Firstly, a PLD evolution is viewed, then an introducing of FPGA architectures is described, after that a comparison between FPGA and other hardware is done, next the CAD flow is explained, and finally an available commercial FPGA is overview. The chapter outcome with a strong recommendation of using FPGA for high speed digital applications.

Chapter 3: an OFDM using FFT algorithm has been presented in this chapter. Firstly, the advantages of multicarrier multi-carrier modulation over the single carrier are shown. Then, the components of OFDM are explored and their functions. After that, explained how OFDM works against multi path fading by

inserting a guard interval with cyclic prefix. The chapter outcome with a strong recommendation of using OFDM for future wireless communication systems.

Chapter 4: an OFDM system using FFT algorithm has been implemented on an Altera EP2C35F672C6 Cyclone II FPGA chip. Altera Quartus II design tool and Math-Work Simulink software are used to design each component of OFDM system, and some Mega Core functions provided by Altera are used. The designs are validated through the configuration on the target FPGA chip, where the allocated resources are investigated and addressed. The results shows that most of the allocated resources are covered by iFFT/FFT and soft decision Viterbi decoder modules, while other modules cover less than 1% of the available resources in the target FPGA chip.

Chapter 5: This chapter proposes an OFDM system to be used in the ECG Patient Monitoring System (ECG-PMS) for wireless telemedicine applications. The system needs a level of transmitted power at the patient site, and achieves a high BER performance at the hospital base station. This achieved by implementing a wavelet-based OFDM system that uses channel estimation based on the improved Data-Aided technique called Forward Error Correction coding Decoded Pseudo Pilot (FDPP). The performance of wavelet-based OFDM systems and Discrete Fourier Transform (DFT)-based OFDM system using FDPP-based as well as Hard Decision Pseudo Pilot channel estimation techniques are compared. The system model was studied using MATLAB

software in which the average BER was addressed for randomized data, and the difference between the transmitted and received signals provide a measure of the ECG signal quality. A better BER performance achieved by using OFDM-DWT system than OFDM-FFT system, where the best performance achieved when using db2 wavelet. Thus our system requirements achieved successfully.

Chapter 6: an implementation of iDWT/DWT system used in OFDM-DWT system is proposed in this chapter to meet the ECG-PMS system requirements, where the DA architectures are used. The iDWT/DWT system in designed in Verilog HDL language using ALTERA Quartus II Software (V.13), validated and configurator on Altera EP4CE115F29C7 Cyclone IV E FPGA chip. The allocated resources, propagation delay, and data rate are addressed for the proposed system for different wavelet families. The comparison result shows that db2 wavelet is the best choose for iDWT/DWT system, even that haar wavelet has better result at DWT block.

7.3 Suggestions for Future Work

7.3.1 Design on an FPGA-Based MIMO Transceiver

To boost the wireless communication systems capacity, a current and next wireless communication schemes tend to use Multiple Input Multiple Output (MIMO) systems

[108]. There are two general types of MIMO: Multi-antenna type and Multi-user MU-MIMO type [109].

In Multi-antenna type where the multiple antenna can be used at the transmitter a receiver sides, the spatially separated antennas in a rich multipath scattering environment obtain a spatial diversity which is exploited by MIMO system to transmit multiple data streams in same frequency band. Thus a capacity gain or diversity gain in obtained depends on how MIMO system is implemented [108][110].

Using Spatial multiplexing techniques in MIMO antenna system to increase channel capacity at higher Signal to Noise Ratio (SNR) makes the receiver more complex [109]. Thus, to handle efficiently the effects of multi-path channel the MIMO is incorporated with OFDM or OFDM Access (OFDMA). As an example, MIMO and OFDMA are combined in IEEE 802.16e standard [109].

In the area of MIMO algorithms and protocols, there is a lot of published paper and reported works based on theoretical / simulation to provide a superior BER for a given SNR [8]. For such algorithms, few reported works such as [8][108][111][112] paid attention on the actual hardware practical implementation in real time system [8]. The objective of the first stage of our future work is to implemented MIMO-OFDM system including channel estimation circuit and pay an extra attention on hardware complexity by measuring the difficulty of fitting the

design on the low end FPGA chip or allocated resources on high end FPGA chip where there are very few reported works in this are such as done by [8][112].

[8][108][111]used FPGAs in their work because FPGA are a suitable prototyping platform for multiple antenna systems because of its attitudes and features such as embedded multiplier, high densities, high level of parallelism [113].The following are the steps to achieve this future work:

A) Solid review on the signal processing of MIMO system [110].
B) Implementation and configuration of MIMO system[8][108][111][112]..
C) Tests several scenarios of MIMO systems and state recommendation of the collected measured data.
D) A written report that discusses the outcomes and finding will be presented.

7.3.2 FPGA application for Location Based Services (LBS)

In recent years, the advances in wireless communication technology have led to the rapid development of sensors that networked through wireless links to obtain wireless sensors networks (WSNs) and then adding the mobility to implement mobile wireless sensors networks (MWSNs) to provide applications of remote monitoring and tracking of mobile object [114][115][116][117].

The need for localization is considered the most significant challenges for MWSNs used in indoor/outdoor applications which are created for civilian, health, military, environmental and commercial purposes[115][116].

There are many measurement techniques in MWSN; indoor/outdoor; localization developed in the past such as Global Positioning System (GPS), Angle-of-arrival (AOA), Time-different-of-Arrival (TDOA), Received signal strength (RSS), and Fingerprinting measurements and others as surveyed in[114][116][118]. Upon the requirements for each particular application, a localization algorithm is implemented using one of the measurement localization techniques, and sometimes using more than one which is called hybrid algorithm[114][115][116][118].

In the area of indoor/outdoor localization algorithms, there are a lot of published paper and reported works based on theoretical / simulation to improve one or more of quality criteria for localization method such as localization accuracy, availability and consistency, cost and complexity, power consumption, latency [118]. And very few reported works paid attention on the actual hardware practical implementation in real time system for those algorithms, where the main challenge faces the engineer designer is to execute these algorithms on embedded systems[119].

There are many applications on tracking of mobile objects like Patient Location Tracking system (PLTS) which is applied at outdoor and Child Tracking system(CTS) which is applied at indoor[115][119]. The objective of the second stage of our future work is to upgrade MIMO-OFDM system into MIMO-OFDM system including indoor localization circuit used RF signal and RSS method and applied for PLTS application[115] The system will be implemented on FPGA chip which is recommended in [119]. The objective of this work is design, validate, and implement this system onto the FPGA chip to provide a high performance cost effective solution. And pay more attention on hardware complexity by measuring the difficulty of fitting the design on the low end FPGA chip or allocated resources on high end FPGA chip where there are very few reported works on this area. The following are the steps to achieve this future work:

A) Review on location based services that can be both indoor and outdoor environment[82][114][115][116][117][118].

B) Possible prototyping of indoor CTS onto FPGA chips [119].

C) Testing and analysing the outcome of the collected data and make recommendations.

D) A written report that discusses the outcomes and finding will be presented.

Chapter 8

References

8.1 Bibliography

[1] H. S. O. Migdadi, A. U. Priantoro, and M. F. Mohammad, "Channel Estimation for OFDM-based Wireless Broadband in Highly Mobile Environment," in *Proceedings International Conference on Electrical Engineering and Informatics*, Institut Teknologi Bandung, Indonesia, June 2009, pp. 996-999.

[2] H. S. O. Migdadi, "Channel estimation techniques in mobile OFDM for beyond 3rd generation (B3G) communication systems," Master of Science in Computer and Information Engineering, Kulliyyah of Engineering, International Islamic University Malaysia, Malaysia, 2007.

[3] R. V. Nee and R. Prasad, *OFDM for Wireless Multimedia Communications* Artech House Boston London, 2000.

[4] A. Ghaith, R. Hatoum, H. Mrad, and A. Alaeddine, "Performance analysis of the Wavelet-OFDM new scheme in AWGN channel," in *Communications and Information Technology (ICCIT), 2013 Third International Conference on*, 2013, pp. 225-229.

[5] M. K. Gupta and S. Tiwari, "Performance evaluation of conventional and wavelet based OFDM system," *AEU-International Journal of Electronics and Communications,* vol. 67, pp. 348-354, 2013.

[6] S. Mohandass and G. Umamaheswari, "Biomedical Signal Transmission using OFDM-based Cognitive Radio for Wireless Healthcare Applications," *SmartCR,* vol. 4, pp. 147-159, 2014.

[7] T.-C. Shih, T.-W. Chen, W.-H. Sung, P.-Y. Tsai, and C.-Y. Lee, "An energy-efficient OFDM-based baseband transceiver design for ubiquitous healthcare monitoring applications," in *SOC Conference (SOCC), 2011 IEEE International*, 2011, pp. 371-375.

[8] H. S. Kim, W. Zhu, J. Bhatia, K. Mohammed, A. Shah, and B. Daneshrad, "AN EFFICIENT FPGA BASED MIMO-MMSE DETECTOR " presented at the 15th European Signal Processing Conference (EUSIPCO 2007), Poznan, Poland, Sep 2007.

[9] L. Gopal, D. W. S. Tze, and N. Z. Ishak, "Design of an FPGA-based OFDM transceiver for DVB-T standard," in *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, 2011, pp. 193-197.

[10] I. Kuon and J. Rose, "Area and Delay Trade-offs in the Circuit and Architecture Design of FPGAs," presented at the ACM Symposium on FPGAs, February 2008.

[11] I. Kuon and J. Rose, "Automated Transistor Sizing for FPGA Architecture Exploration," presented at the DAC '08, ACM/IEEE Design Automation Conference, June 2008.

[12] I. Kuon, R. Tessier, and J. Rose, "FPGA Architecture: Survey and Challenges," *Foundations and Trends in Electronic Design Automation,* vol. 2, pp. pp. 135-253. , 2008

[13] S. Brown and J. Rose, "FPGA and CPLD architectures: a tutorial," *Design & Test of Computers, IEEE,* vol. 13, pp. 42-57, 1996.

[14] (Accessed: Nov 2011). *PROM-Programmable Read Only Memory* [Online]. Available: http://www.fpgacentral.com/pld-types/prom-programmable-read-only-memory
[15] (Accessed: Nov 2011). *PLA - Programmable logic arrays* [Online]. Available: http://www.fpgacentral.com/pld-types/pla-programmable-logic-arrays
[16] R. H. Katz. (Accessed: Nov 2011). *Programmable Logic* [Online]. Available: http://poppy.snu.ac.kr/~kchoi/class/lc_intro/programmable_logic.pdf
[17] (Accessed: Nov 2011). *PAL - Programming Array Logic* [Online]. Available: http://www.fpgacentral.com/pld-types/pal-programming-array-logic
[18] (Accessed: Nov 2011). *Types of Programmable Logic Simple Programmable Devices SPLDs* [Online]. Available: http://labspace.open.ac.uk/mod/resource/view.php?id=360467
[19] (Accessed: Nov 2011). *History of the programmable logic* [Online]. Available: http://www.fpgacentral.com/docs/fpga-tutorial/history-programmable-logic
[20] (Accessed: Nov 2011). *FPGA Architecture* [Online]. Available: http://www.fpgacentral.com/docs/fpga-tutorial/fpga-architecture
[21] (Accessed: Dec 2011). *Field-programmable gate array* [Online]. Available: http://en.wikipedia.org/wiki/Field-programmable_gate_array
[22] (Accessed: Dec 2011). *FPGA logic blocks* [Online]. Available: http://www.fpgacentral.com/docs/fpga-tutorial/fpga-logic-blocks
[23] (Accessed: Dec 2011). *Lookup table* [Online]. Available: http://en.wikipedia.org/wiki/Lookup_table
[24] (Accessed: Dec 2011). *FPGA routing matrix* [Online]. Available: http://www.fpgacentral.com/docs/fpga-tutorial/fpga-routing-matrix
[25] (Accessed: Dec 2011). *Other FPGA resources* [Online]. Available: http://www.fpgacentral.com/docs/fpga-tutorial/other-fpga-resources
[26] (Accessed: Dec 2011). *FPGA configuration memory* [Online]. Available: http://www.fpgacentral.com/docs/fpga-tutorial/fpga-configuration-memory
[27] (Accessed: Dec 2011). *Flash memory* [Online]. Available: http://en.wikipedia.org/wiki/Flash_memory#Floating-gate_transistor
[28] (Accessed: Dec 2011). *CMOS* [Online]. Available: http://en.wikipedia.org/wiki/CMOS
[29] AlteraCorporation. (Feb 2007, Accessed:Dec 2011). Cyclone II Device Handbook, Chapter 2. Cyclone II Architecture [Online]. *1*. Available: http://www.altera.com/literature/hb/cyc2/cyc2_cii5v1.pdf
[30] (Accessed: Dec 2011). *Integrated circuit* [Online]. Available: http://en.wikipedia.org/wiki/Integrated_circuit
[31] (Accessed: Dec 2011). *Application-specific integrated circuit* [Online]. Available: http://en.wikipedia.org/wiki/Application-specific_integrated_circuit
[32] I. Kuon and J. Rose, "Measuring the Gap Between FPGAs and ASICs," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on,* vol. 26, pp. 203-215, 2007.
[33] H. Ouarda, "FPGA and Field Programmable Devices architectures : A tutorial," *INTERNATIONAL JOURNAL of CIRCUITS, SYSTEMS and SIGNAL PROCESSING,* vol. 5, pp. 529-536, 2011.
[34] AlteraCorporation. (May 2011, Accessed: Oct 2011). Quartus II Introduction Using VHDL Designs [Online]. Available: ftp://ftp.altera.com/up/pub/Altera_Material/11.0/Tutorials/VHDL/Quartus_II_Introduction.pdf

[35] (Accessed: Oct 2011). *Types of Programmable Logic Hardware Design and Development* [Online]. Available: http://labspace.open.ac.uk/mod/resource/view.php?id=360469

[36] (Accessed: Dec 2011). *FPGA synthesis and implementation (Xilinx design flow)* [Online]. Available: http://www.fpgacentral.com/docs/fpga-tutorial/fpga-synthesis-and-implementation-xilinx

[37] Xilinx®Inc. (1995–2009, XST Synthesis Overview. [Online]. *(Accessed: Dec 2011)*. Available: http://www.xilinx.com/support/documentation/sw_manuals/xilinx11/ise_c_using_xst_for_synthesis.htm

[38] AlteraCorporation. (Nov 2011, Accessed:Dec 2011). Quartus II Handbook Version 11.1, Chapter 1: Simulating Altera Designs. [Online]. *3*. Available: http://www.altera.com/literature/hb/qts/qts_qii53025.pdf

[39] P. Hornyack and M. Martin. (Jan 2006, Accessed:Dec 2011). *Xilinx ModelSim Simulation Tutorial* [Online]. Available: http://www.cis.upenn.edu/~milom/cse372-Spring06/simulation/

[40] Xilinx®Inc. (1995–2010, Accessed: Dec 2011). Implementation Overview for FPGAs. [Online]. Available: http://www.xilinx.com/support/documentation/sw_manuals/xilinx12_3/ise_c_implement_fpga_design.htm

[41] Xilinx®Inc. (1995–2009, Accessed: Dec 2011). Timing Simulation. [Online]. Available: http://www.xilinx.com/support/documentation/sw_manuals/xilinx11/ise_c_simulation_timing.htm

[42] Jeff. (July 2011, Accessed: Jan 2012). *List and comparison of FPGA companies* [Online]. Available: http://www.fpgadeveloper.com/2011/07/list-and-comparison-of-fpga-companies.html

[43] O. Šcekic. (2005, Accessed: Jan 2012). *FPGA Comparative Analysis* [Online]. Available: http://home.etf.rs/~vm/os/vlsi/razno/Altera%20vs%20Xilinx%20%28Scekic%29%20.pdf

[44] AlteraCorporation. (1995-2012, Accessed: Jan 2012). Devices. [Online]. Available: http://www.altera.com/devices/dvcs-index.html

[45] Xilinx®Inc. (March 2012, Accessed: Jan 2012). 7 Series FPGAs Overview: Advance Product Specification. [Online]. *v1.10*. Available: http://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf

[46] AlteraCorporation. (1995-2012, Accessed: Jan 2012). About Stratix Series High-End FPGAs. [Online]. Available: http://www.altera.com/devices/fpga/stratix-fpgas/about/stx-about.html

[47] AlteraCorporation. (1995-2012, Accessed: Jan 2012). About Arria FPGA Series. [Online]. Available: http://www.altera.com/devices/fpga/arria-fpgas/about/arr-about.html

[48] AlteraCorporation. (1995-2012, Accessed: Jan 2012). About Altera's Cyclone FPGA Series. [Online]. Available: http://www.altera.com/devices/fpga/cyclone-about/cyc-about.html

[49] AlteraCorporation. (Jan 2011, Accessed: Jan 2012). Getting Started with Altera. [Online]. Available: http://www.altera.com/support/read-me-first.pdf

[50] AlteraCorporation. (1995-2012, Accessed: Jan 2012). All Development Kits. [Online]. Available: http://www.altera.com/products/devkits/kit-dev_platforms.jsp

[51] AlteraCorporation. (1995-2012, Accessed: Jan 2012). End Markets. [Online]. Available: http://www.altera.com/end-markets/end-index.html

[52] AlteraCorporation. (1995-2012, Accessed: Jan 2012). Altera Development and Education Boards. [Online]. Available: http://www.altera.com/education/univ/materials/boards/unv-dev-edu-boards.html
[53] AlteraCorporation. (1995-2012, Accessed: Jan 2012). DE2 Development and Education Board. [Online]. Available: http://www.altera.com/education/univ/materials/boards/de2/unv-de2-board.html
[54] AlteraCorporation. (1995-2012, Accessed: Jan 2012). DE3 Development and Education Board. [Online]. Available: http://www.altera.com/education/univ/materials/boards/de3/unv-de3-board.html
[55] AlteraCorporation. (1995-2012, Accessed: Jan 2012). DE4 Development and Education Board. [Online]. Available: http://www.altera.com/education/univ/materials/boards/de4/unv-de4-board.html
[56] H. HARADA, T. YAMAMURA, and M. FUJISE, "A New Estimation Method of Propagation Characteristics Using Pilot-Data-Inserted OFDM Signals for High-Mobility OFDM Transmission Scheme," *IEICE TRANSACTIONS on Communications,* vol. E85-B, pp. 882-894, May 2002.
[57] L. Litwin and M. Pugel. (Jan 2011, Accessed: July 2011). The principles of OFDM. [Online]. Available: http://images.rfdesign.com/files/4/0101Puegel30.pdf
[58] R. R. Mosier and R. G. Clabaugh, "Kineplex, a bandwidth-efficient binary transmission system,," *Trans. AIEE,* pp. 723–728, Jan. 1958
[59] "Orthogonal Frequency Division Multiplexing," U.S. Patent No. 3, 488,4555, filed November 14, 1966, issued Jan. 6, 1970.
[60] P. Lau. (Sep 2010, Accessed: March 2011). State Tables and State Diagrams. [Online]. Available: http://www.ee.usyd.edu.au/tutorials/digital_tutorial/part3/t-diag.htm
[61] kitz. (2003-2012, Accessed: March 2011). ADSL Data - Error Correction. [Online]. Available: http://www.kitz.co.uk/adsl/error_correction.htm
[62] S. Sharma and S. Kumar, "BER Performance Evaluation of FFT-OFDM and DWT-OFDM," *International Journal of Network and Mobile Technologies,* vol. 2, pp. 110-116, 2011.
[63] S. J. Vaughan-Nichols. (Nov 2002, Accessed: March 2012). OFDM's New Uses. [Online]. Available: http://www.wi-fiplanet.com/tutorials/article.php/1502361/OFDMs-New-Uses.htm
[64] I. Poole. (2015, Accessed: Sep 2015). 5G Modulation & Waveforms. [Online]. Available: http://www.radio-electronics.com/info/cellulartelecomms/5g-mobile-cellular/modulation-waveforms.php
[65] M. Veena and M. S. Swamy, "Low Power Pipelined Dwt-Idwt Architecture for Ofdm System on Fpga," *Procedia Engineering,* vol. 30, pp. 466-474, 2012.
[66] S. Kaur and R. Mehra, "FPGA Implementation of OFDM Transceiver using FFT Algorithm," *International Journal of Engineering Science and Technology (IJEST),* vol. 4, pp. pp. 1532-11537, April 2012
[67] D. Varugheese and N. Muniraj, "FPGA Implementation of New Adaptive DWT-IDWT Lifting Technique for OFDM," *European Journal of Scientific Research,* vol. 75, pp. 92-103, 2012.
[68] J. Garcia and R. Cumplido, "On the design of an FPGA-based OFDM modulator for IEEE 802.11a," in *Electrical and Electronics Engineering, 2005 2nd International Conference on*, 2005, pp. 114-117.
[69] C. Langton. (July 1999, Accessed:July 2011). Signal Processing and Simulation Newsletter. [Online]. Available: http://www.complextoreal.com/convo.htm

[70] AlteraCorporation. (Nov 2011, Accessed: Jan 2012). FFT MegaCore Function [Online]. *11.0*. Available: http://www.altera.com/literature/ug/ug_viterbi-compiler.pdf
[71] AlteraCorporation. (May 2011, Accessed: Jan 2012). Viterbi Compiler [Online]. Available: http://www.altera.com/literature/ug/ug_fft.pdf
[72] A. Kiani and S. Mousavi, "Performance Assessment of DFT-OFDM and DWT-OFDM Systems in the Presence of the SSPA and Fading Channel."
[73] H. Zhang, D. Yuan, M. Jiang, and D. Wu, "Research of DFT-OFDM and DWT-OFDM on different transmission scenarios," *IEEE ICITA,* pp. 8-11, 2004.
[74] H. J. Taha and M. Salleh, "Performance analysis of QAM-modulation parameters on wavelet packet transform (WPT) and FFT-OFDM system," in *Communications (MICC), 2009 IEEE 9th Malaysia International Conference on*, 2009, pp. 1-5.
[75] M. Oltean, "Wavelet OFDM performance in flat fading channels," *Scientific Bulletin of University Politehnica Timisoara, ETC Series,* vol. 52, pp. 167-172, 2007.
[76] M. Oltean and M. Nafornita, "Wavelet OFDM performance in frequency selective fading channels," in *IEEE 8th Int. Conf. on Communications (COMM)*, 2010, pp. 343-346.
[77] H.-G. Jeon, H.-K. Song, and E. Serpedin, "Walsh coded training signal aided time domain channel estimation for MIMO-OFDM systems," *Communications, IEEE Transactions on,* vol. 56, pp. 1430-1433, 2008.
[78] D. Imamura, K. I. H. Sudo, and G. Ohta, "A study of adaptive channel estimation for MMAC/OFDM systems," presented at the Proc. IEICE Gen. Conf., March 2000.
[79] R. Funada, H. Harada, Y. Kamio, and S. Shinoda, "A channel estimation method for a highly mobile OFDM wireless access system," *IEICE transactions on communications,* vol. 88, pp. 282-291, 2005.
[80] J. Song, W. Ding, F. Yang, H. Yang, J. Wang, X. Wang*, et al.*, "Indoor hospital communication systems: An integrated solution based on power line and visible light communication," in *Faible Tension Faible Consommation (FTFC), 2014 IEEE*, 2014, pp. 1-6.
[81] K. Thenmozhi and V. Prithiviraj, "Suitability of Coded Orthogonal Frequency Division Multiplexing (COFDM) for multimedia data transmission in wireless telemedicine applications," in *Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on*, 2007, pp. 288-292.
[82] S. H. Chew, P. A. Chong, E. Gunawan, K. W. Goh, Y. Kim, and C. B. Soh, "A Hybrid Mobile-based Patient Location Tracking System for Personal Healthcare Applications," in *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*, 2006, pp. 5188-5191.
[83] H. Harada, "A feasibility study of 16QAM-OFDM transmission scheme under the standard model for the multimedia mobile access communication systems," *Proc. PIMRC'99, Sept.,* vol. 2, pp. 905-910, 1999.
[84] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Communications, 1995. ICC'95 Seattle,'Gateway to Globalization', 1995 IEEE International Conference on*, 1995, pp. 1009-1013.
[85] D. Shah and C. Vithlani, "Efficient implementations of discrete wavelet transforms using FPGAs," *International Journal of Advances in Engineering & Technology,* vol. 1, 2011.
[86] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 11, pp. 674-693, 1989.

[87] M. Martone, "Multiresolution sequence detection in rapidly fading channels based on focused wavelet decompositions," *Communications, IEEE Transactions on,* vol. 49, pp. 1388-1401, 2001.
[88] S.-C. B. Lo, H. Li, and M. T. Freedman, "Optimization of wavelet decomposition for image compression and feature preservation," *Medical Imaging, IEEE Transactions on,* vol. 22, pp. 1141-1151, 2003.
[89] G. Minami, Z. Xiong, A. Wang, and S. Mehrotra, "3-D wavelet coding of video with arbitrary regions of support," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 11, pp. 1063-1068, 2001.
[90] M. A. Magnor, A. Endmann, and B. Girod, "Progressive Compression and Rendering of Light Fields," in *VMV*, 2000, pp. 199-204.
[91] D. Sripath, "Efficient Implementations of Discrete Wavelet Transforms Using FPGAs," 2003.
[92] S. Mirzaei, A. Hosangadi, and R. Kastner, "FPGA implementation of high speed FIR filters using add and shift method," in *Computer Design, 2006. ICCD 2006. International Conference on*, 2007, pp. 308-313.
[93] C. Zhang, C. Wang, and M. O. Ahmad, "A pipeline VLSI architecture for fast computation of the 2-D discrete wavelet transform," *Circuits and Systems I: Regular Papers, IEEE Transactions on,* vol. 59, pp. 1775-1785, 2012.
[94] M. Maamoun, R. Bradai, A. Meraghni, and R. Beguenane, "Low cost VLSI discrete wavelet transform and FIR filters architectures for very high-speed signal and image processing," in *Cybernetic Intelligent Systems (CIS), 2010 IEEE 9th International Conference on*, 2010, pp. 1-6.
[95] C. Zhang, C. Wang, and M. O. Ahmad, "A pipeline VLSI architecture for high-speed computation of the 1-D discrete wavelet transform," *Circuits and Systems I: Regular Papers, IEEE Transactions on,* vol. 57, pp. 2729-2740, 2010.
[96] A. M. Al-Haj, "Fast discrete wavelet transformation using FPGAs and distributed arithmetic," *International Journal of Applied Science and Engineering,* vol. 1, pp. 160-171, 2003.
[97] D. K. Varugheese and N. Muniraj, "Modified Lifting Based DWT/IDWT Architecture for OFDM on Virtex-5 FPGA," *Global Journal of Researches In Engineering,* vol. 12, 2012.
[98] N. S. Bhat, "FPGA based DWT-IDWT implementation of OFDM on UWB Systems."
[99] E. C. Ifeachor and B. W. Jervis, *Digital signal processing: a practical approach*: Pearson Education, 2002.
[100] H. Kumar, "FPGA Implementation of Digital FIR Filter," THAPAR UNIVERSITY, 2012.
[101] A. Sharma, "VLSI Implementation of Pipelined FIR filter," THAPAR UNIVERSITY, 2013.
[102] M. Yazhini and R. Ramesh, "Fir filter implementation using modified distributed arithmetic architecture," *Indian Journal of Science and Technology,* vol. 6, pp. 4485-4491, 2013.
[103] H. Yoo and D. V. Anderson, "Hardware-efficient distributed arithmetic architecture for high-order digital filters," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, 2005, pp. v/125-v/128 Vol. 5.
[104] S. W. Smith, "The scientist and engineer's guide to digital signal processing," 1997.
[105] S. S. Velukar and M. Parlewar, "FPGA Implementation of Fir Filter using Distributed Arithmetic Architecture for DWT," *International Journal of Computer Applications,* vol. 92, 2014.

[106] P. R. KUMAR and M. MEDURI, "THE IMPLEMENTATION OF HIGH ORDER MATCHED FIR FILTER WITH DISTRIBUTED ARITHEMETIC," *International Journal of VLSI and Embedded Systems-IJVES,* vol. 04, 2013.

[107] M. Misiti, Y. Misiti, G. Oppenheim, and J.-M. Poggi, "Matlab Wavelet Toolbox User\'s Guide. Version 3," 2004.

[108] A. Burg, E. Beck, M. Rupp, D. Perels, N. Felber, and W. Fichtner, "FPGA implementation of a MIMO receiver front-end for the UMTS downlink," in *Broadband Communications, 2002. Access, Transmission, Networking. 2002 International Zurich Seminar on*, 2002, pp. 8-1-8-6.

[109] SydaapTechnologiesPvt.Ltd. Accessed: March 2012). Multiple Input Multiple Output (MIMO) [Online]. Available: http://sydaap.com/docs/MIMO.pdf

[110] G. L. Stuber, J. R. Barry, S. W. McLaughlin, L. Ye, M. A. Ingram, and T. G. Pratt, "Broadband MIMO-OFDM wireless communications," *Proceedings of the IEEE,* vol. 92, pp. 271-294, 2004.

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

[112] S. Reddy.P and R. Reddy.G, "Design and FPGA Implementation of Channel Estimation Method and Modulation Technique for MIMO System " *European Journal of Scientific Research* vol. 25, pp. pp. 257-265, 2009.

[113] L. G. Barbero and J. S. Thompson. Accessed: March 2012). FPGAs Enable Next-Generation Multiple-Antenna Wireless Communications. [Online]. Available: http://www.see.ed.ac.uk/~jst/sphere/alpha_data_wp.pdf

[114] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *Computer Networks,* vol. 51, pp. 2529-2553, 2007.

[115] A. Rozyyev, H. Hasbullah, and F. Subhan, "A Survey of Child Tracking Techniques in Wireless Sensor Network " *International Journal of Research and Reviews in Computer Science (IJRRCS),* vol. 1, pp. pp. 59-66, Dec 2010.

[116] I. Amundson and X. D. Koutsoukos, "A survey on localization for mobile wireless sensor networks," in *MELT'09 Proceedings of the 2nd international conference on Mobile entity localization and tracking in GPS-less environments . Springer-Verlag Berlin, Heidelberg ©2009* 2009 pp. pp. 235-254

[117] M. Naderan, M. Dehghan, and H. Pedram, "Mobile object tracking techniques in wireless sensor networks," in *Ultra Modern Telecommunications & Workshops, 2009. ICUMT '09. International Conference on*, 2009, pp. 1-8.

[118] L. Hui, H. Darabi, P. Banerjee, and L. Jing, "Survey of Wireless Indoor Positioning Techniques and Systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on,* vol. 37, pp. 1067-1080, 2007.

[119] M. C. Sacchetin, J. J. Lopes, D. F. Wolf, J. L. Silva, and E. Marques, "Analysis and Implementation of Localization and Mapping Algorithms for Mobile Robots Based on Reconfigurable Computing " *International Journal of Latin American Applied Research,* vol. 37, pp. pp. 31-34, 2007.

Author's Publication Record

- **International Conferences**

- **H.S.O. Migdadi,** R.A. Abd-Alhameed, E.A.A. Qaralleh, H.A. Obeidat, J.M. Noras, M.J. Ngala,"*FIR Implementation on FPGA: Investigate the FIR order on SDA and PDA algorithms"* Sixth International Conference on Internet Technologies & Applications(ITA15), Wrexham, North Wales,UK. 8-11 September 2015. *(in Press)*

- **H.S.O. Migdadi,** R.A. Abd-Alhameed, N. N. Khatib, H.A. Obeidat, J.M. Noras, "*Channel Estimation for OFDM FFT/DWT in Multi-Carrier Modulation used in Wireless Telemedicine*",15th IEEE International Conference on Computer and Information Technology to be held in Liverpool, 26-28 Oct 2015. *(in Press)*

- **Hassan Migdadi**, AkhmadUnggulPriantoro, MasrulFaizal Mohammad, "*Channel Estimation for OFDM-based Wireless Broadband in Highly Mobile Environment*" International Conference on Electrical and Informatics, Bandung, Indonesia, 17-19 June 2007.

- Y.A.S. Dama, R. A. Abd-Alhameed, S.M.R. Jones, T. Sadeghpour, **H.S.O. Migdadi,** and P.S. Excell*"A New Approach to Quasi-Orthogonal Space-Time Block Coding Applied to Quadruple MIMO Transmit Antennas"*. Proceedings of the Fourth International Conference on Internet Technologies and Applications (ITA 11), Wrexham, North Wales, UK. 6-9 September 2011.

- A. Adeel, R.A. Abd-Alhameed, O.O. Anoh, J.M. Noras, **H.S.O. Migdadi**, and P.S. Excell*"Performance Comparison of OFDM and Wavelet Based OFDM for Multi-Path Fading Channels*", Proceedings of the Fourth International Conference on Internet Technologies and Applications (ITA 11), Wrexham, North Wales,UK. 6-9 September 2011.

- R. Asif, R. A. A. Abd-Alhameed, O. O. Anoh,YAS Dama, **H.S. Migdadi**, J. N. Noras, A. S. Hussaini, and J. Rodringuez, "Performance Comparison between DWT-OFDM and FFT-OFDM Using Time Domain Zero Forcing Equalization," *International Conference on Telecommunications and Multimedia*, TEMU, Aug. 2012.

- S.W.A. Shah, **H.S. Okleh**, A. S. Hussaini1, R.A. Abd-Alhameed, J. M. Noras, and J. Rodriguez, "*FPGA Implementation of Modern Wireless Communication System"* 2012 IEEE 17th International Workshop on Computer Aided Modeling

and Design of Communication Links and Networks (CAMAD), 17-19 September 2012 Barcelona Spain.

- R Asif, N.T. Ali, **H.S.O. Migdadi**, R.A. Abd-Alhameed, A.S. Hussaini, Tahereh S. Ghazaany, S. Naveed, JM Noras, PS Excell, J. Rodriguez. *"Wavelet Based MIMO-Multicarrier System Using Forward Error Correction and Beam Forming"* Proceedings of the Fifth International Conference on Internet Technologies and Applications (ITA 13), Wrexham, North Wales,UK. 10-13 September 2013.

- K. O. O. Anoh, N.T. Ali, **H.S.O. Migdadi**, R.A. Abd-Alhameed, T. Ghazzany, S. M. R. Jones, JM Noras, PS Excell "*Performance Comparison of MIMO-DWT and MIMO-FrFT Multicarrier Systems"* Proceedings of the Fifth International Conference on Internet Technologies and Applications (ITA 13), Wrexham, North Wales,UK. 10-13 September 2013.

- R.A. Abd-Alhameed, K.O.O. Anoh, I. Ahmad, **H.S.O. Migdadi**, R. Asif, N.A. Jan, T. Ghazaany, S. Zhu, F. Elmegri, M. Bin-Melha, A. Atojoko, J.M. Noras, S.M.R. Jones, C.H. See, H. Alhassan and M.B. Child *"A Review of Location Based Services: Current Developments, Trends and Issues"* Proceedings of the Fifth International Conference on Internet Technologies and Applications (ITA 13), Wrexham, North Wales, UK. 10-13 September 2013.

- Y.A.S.Dama, R.A.Abd-Alhameed, SMR Jones, **H.S.O. Migdadi**, Ogbonnaya.O.Anoh, *"Performance Evaluation of Adaptive Modulation over WiMAX*", Under Processing.

- RaedAbd-Alhameed, Y.A.S Dama,**Hassan Migdadi**, E Elkazmi, WafaShuaieb, WalaaHammoudeh, Ahmed Masri, EshtiwiAbdulmula,*"A new approach for implementing QO-STBC over OFDM*",WISATS 2015 - 7th EAI International Conference on Wireless and Satellite Systems, Bradford, UK, 6-7 July 2015.

- Arfan Ghani, Chan H. See,**HassanMigdadi**,Raed A. A.Abd-Alhameed, James M Noras, *"Reconfigurable Neurons – Making the Most of Configurable Logic Blocks (CLBs)"*Sixth International Conference on Internet Technologies & Applications(ITA15), Wrexham, North Wales,UK. 8-11 September 2015. *(in Press)*