



4-2007

WiMAX Software Defined Radio

Assad Saleem
Western Michigan University

Follow this and additional works at: https://scholarworks.wmich.edu/masters_theses



Part of the Electrical and Computer Engineering Commons

Recommended Citation

Saleem, Assad, "WiMAX Software Defined Radio" (2007). *Masters Theses*. 1408.
https://scholarworks.wmich.edu/masters_theses/1408

This Masters Thesis-Open Access is brought to you for free and open access by the Graduate College at ScholarWorks at WMU. It has been accepted for inclusion in Masters Theses by an authorized administrator of ScholarWorks at WMU. For more information, please contact wmu-scholarworks@wmich.edu.



WiMAX SOFTWARE DEFINED RADIO

by

Assad Saleem

A Thesis
Submitted to the
Faculty of The Graduate College
in partial fulfillment of the
requirements for the
Degree of Master of Science in Engineering (Electrical)
Department of Electrical and Computer Engineering

Western Michigan University
Kalamazoo, MI
April 2007

WiMAX SOFTWARE DEFINED RADIO

Assad Saleem, M.S.E.

Western Michigan University, 2007

As processors and configurable devices are getting faster and smaller, they are giving rise to better solutions for various communications systems and standards. In this paper, the implementation of a software defined radio for WiMAX is presented. This paper highlights the system model for simulation to achieve better understanding of the system, which was developed in MATLAB. Also, included in the paper is the discussion of the physical implementation of this communication system. The system is build around National Instruments' PCI 5640 hardware which features a Xilinx Virtex-II FPGA. Naturally, LabVIEW was the tool used to implement the hardware/software co-design.

ACKNOWLEDGMENTS

All praises be to God who has blessed me with all what I have. I am fortunate to have parents who took extreme care of me in every way possible. Only because of this I was able to attain what I have today and conduct such research. I have special thanks to Dr. Bazuin whose devoted support during all phases of the work, and invaluable information made it possible to accomplish this research work successfully. Also, many thanks to Dr. Dong who provided the tools required for the implementation of this work.

Assad Saleem

Copyright by
Assad Saleem
2007

TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	ii
LIST OF TABLES.....	v
LIST OF FIGURES.....	vi
CHAPTER	
1 INTRODUCTION	1
1.1 Need, Requirement and Significance.....	1
1.1.1 OFDM.....	2
1.1.2 WiMax, IEEE- 802.16 Standard	2
1.1.3 Software Radios and Rapid Prototyping.....	3
1.1.4 NI Tools for MIMO Research and Development	3
1.2 Research Overview	3
2 OVERVIEW	5
2.1 Software Defined Radio.....	5
2.2 OFDM.....	7
2.3 802.16 and WiMAX.....	9
2.4 Tools for Rapid Prototyping	10
2.4.1 NI PCI 5640R	11
2.4.2 LabVIEW Software and Host.....	12
3 DETAILED DESCRIPTION.....	13
3.1 OFDM Symbol Structure.....	13
3.1.1 OFDM Theoretical Development	15
3.2 WIMAX.....	21
4 SIMULATION.....	23
4.1 OFDM Signaling.....	23
4.1.1 OFDM Time-domain Signals	24
4.1.2 OFDM Received Signals	25
4.1.3 OFDM Constellation Plots.....	27
4.2 OFDM with Channel Impairments	29
4.2.1 OFDM Constellation Plots.....	36
5 IMPLEMENTATION.....	38
5.1 Development using LabVIEW and NI Hardware.....	38
5.2 Graphical User Interface of the WiMAX SDR.....	38
5.3 Block Diagram of the Graphical User Interface of the WiMAX SDR.....	39
5.4 Block Diagram of the OFDM System	41
5.5 OFDM Modulator Block Diagram.....	42

Table of Contents -- Continued

CHAPTER		
5.6	Pilot Inserter Block Diagram	43
5.7	Block Diagram of the Transceiver	43
5.8	OFDM Demodulator Block Diagram	45
6	SYSTEM RESULTS	46
6.1	OFDM Signal.....	47
7	SUMMARY AND CONCLUSIONS	49
7.1	Results of this Work.....	49
7.2	Future Work	49
7.3	Conclusion	50
APPENDIX.....		51
MATLAB Files.....		51
BIBLIOGRAPHY.....		64

LIST OF TABLES

1: Primitive parameter definitions [12].....	21
2: Derived parameter definitions [12].....	22
3: Parameters of transmitted signal [12].....	22
4: Filling OFDM bins.....	24
5: Channel noise.....	30
6: Channel delay	31
7: Phase correction.....	34
8: Parameters of OFDM signal	47

LIST OF FIGURES

1: Emergence of various wireless and communications standards [16]	1
2: Signal processing intensity versus flexibility [18].....	5
3: Evolving FPGA generations [18]	6
4: Evolving DSP generations [19]	7
5: Spectrum of a FDM (frequency division multiplexing) communication system [1]	8
6: Spectrum of an OFDM (orthogonal frequency division multiplexing) communication system [1].....	8
7: Basic structure of an OFDM system [4]	9
8: WiMAX transceiver [1].....	10
9: PCI 5640R card [10].....	11
10: PCI-5640R block diagram [10].....	11
11: Computer system used to implement the research project	12
12: OFDM spectrum [16].....	13
13: OFDM signal generation [1].....	14
14: OFDM symbol structure and sub-channelization [4]	14
15: OFDM symbol time structure showing insertion of Cyclic Prefix [4]	15
16: OFDM signal generation [1].....	16
17: OFDM signal demodulation [1].....	17
18: Adjacent Symbol Interference (ASI) symbol searing due to channel [1].....	18
19: Insertion of Guard Interval between adjacent symbols to suppress ASI [1]	18
20: Cyclic Prefix inserted in the Guard Interval to suppress Adjacent Channel Interference (ACI) [1].....	19
21: Cyclic Prefix insertion [16].....	19

List of Figures -- Continued

22: Spectrum of (a) an individual carrier and (b) an OFDM signal [16].....	20
23: Basic OFDM transmitter [16].....	21
24: OFDM spectrum prior to transformation and transmission.....	24
25: Time domain representation of OFDM signal at the transmitter.....	25
26: Time domain representation of OFDM signal at the receiver	26
27: OFDM spectrum after reception	26
28: Constellation plot prior to transmission.....	27
29: Constellation plot showing only the pilot tones prior to transmission	28
30: Constellation plot after the reception, with no channel distortion	28
31: Constellation plot showing only the pilot tones after the reception, with no channel distortion.....	29
32: Constellation plot at receiver with channel noise and delay.....	31
33: Constellation plot just for pilot tones at receiver with channel noise and delay ..	32
34: OFDM spectrum at receiver with channel noise and delay	32
35: OFDM spectrum difference between the transmitted and received signal	33
36: OFDM spectrum after reception with noise	35
37: Error calculated by taking the difference between the transmitted and received signal spectrum	35
38: Constellation plots after the reception, with channel distortion	36
39: Constellation plot showing only the pilot tones after the reception, with channel distortion.....	37
40: Graphical User Interface (GUI) of the system.....	39
41: Block diagram of GUI	40

List of Figures -- Continued

42: Block diagram of the OFDM system.....	41
43: OFDM modulator	42
44: OFDM pilot inserter.....	43
45: Block diagram of the transceiver	44
46: OFDM demodulator.....	45
47: System setup for testing.....	46
48: Channel and connecting transmitter and receiver, also oscilloscope probe on the channel	46
49: Two OFDM signal bursts	47
50: Four OFDM signal bursts	48
51: OFDM frame structure for WiMAX [4].....	49

1 INTRODUCTION

The advancements in integrated circuits technology, in the past few decades, have revolutionized the communications industry. RF and mixed signal ICs have made it possible to realize the continuous signal waveforms in discrete time domain. Furthermore, as the ICs are getting denser, highly efficient at minimal power utilization; it is becoming increasingly possible to implement the communication system in software. This paper presents the design of such a system from concept to physical implementation.

Competition among American, European, and Asian counter parts further fueled the race of advanced communications all over the world. Due to this, there exist many communication standards. One of the most recent techniques involves orthogonal frequency division multiplexed (OFDM) signaling and systems.

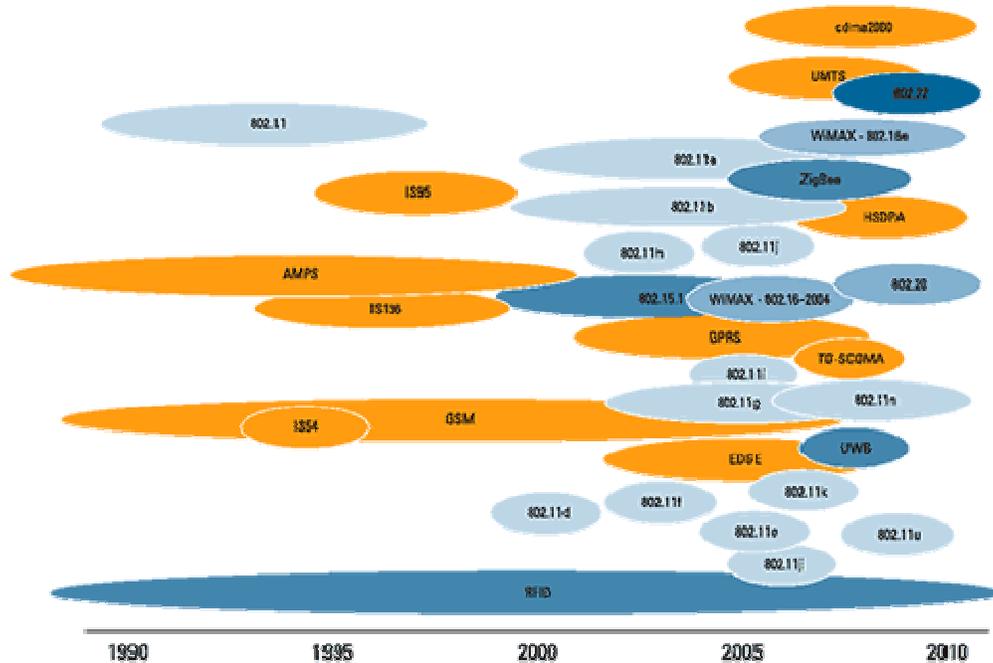


Figure 1: Emergence of various wireless and communications standards [16]

For development, laboratory and future systems, a reconfigurable communication system approach comes in very handy in this situation, as the same hardware can be reconfigured and/or reprogrammed for different communication scheme. This thesis has developed a reconfigurable laboratory system for OFDM signal transmission and reception, building toward the WiMax, IEEE 802.16 system simulation and studies.

1.1 Need, Requirement and Significance

This work has focused on two aspects of advanced communications, the emerging WiMax standard and OFDM communications and the implementation of

software radio components to support rapid prototyping and laboratory research in MIMO communications.

1.1.1 OFDM

Modern communication is becoming more and more wireless. Wireless signal transmission is far from ideal conditions. OFDM's inherent immunity to channel imperfections, due to many sub-channels, makes it an ideal choice for wireless transmission. The fading effects in each sub-channel are easily removed by assuming it to be a constant across all sub-channels. Also, what makes it more desirable is the fact that this system can be implemented using existing hardware.

Orthogonal Frequency Division Multiplexing (OFDM) is a multi-carrier transmission scheme and its concept dates back to 1960s. But it was quite recent since it was adapted as a useful technique for communication systems. There are three main reasons as to why new emerging systems like WiMAX and DSL use this concept. Firstly, it utilizes the available bandwidth very efficiently and secondly, it is lesser vulnerable to channel distortions compared to other high bandwidth schemes. And lastly, it can be realized using fast, low cost, and easily available DSP hardware.

Because of these advantages it is being widely used in the communication industry. Such as, global standard for asymmetric digital subscriber line (ADSL), European standard for digital audio broadcasting (DAB), Wireless local area networks (LANs), IEEE 802.11, and IEEE 802.16. WiMAX is also based on the Orthogonal Frequency Division Multiplexing (OFDM), which is the reason for its better multi-path performance in non-line-of-sight situations.

1.1.2 WiMax, IEEE- 802.16 Standard

IEEE 802.16 Air interface standard is the base for WiMAX technology. It is being widely adopted for fixed broadband wireless metropolitan area networks (MAN). There are over 150 WiMAX labs across the globe. Low data latency and efficient data multiplexing are essential for high data throughput. Such are the required features of broadband data services like VoIP and streaming video with high quality of service (QoS). This will result into transparency of quality of service between broadband wired service and Mobile WiMAX, which is necessary and desired for the success of Mobile internet application for Mobile WiMAX. Examples of benefiting broad band wired services are Cable, and DSL.

Other prominent features of WiMAX are its capabilities for high data throughput, low cost deployment, scalable architecture, friendly IPR structure, open standard approach, and healthy ecosystem. Mobile WiMAX allows flexible network architecture and a common wide area broad band access technology, which will result into the convergence of fixed and mobile broad band networks.

1.1.3 Software Radios and Rapid Prototyping

Although the third generation of wireless networks and services is just getting to the market, engineers are already working on the next generation, known as the fourth generation of communication systems. Over the course of time many different communication standards have emerged because of the competition among communication industries in various parts of the world. It is speculated that the future of communication technology will offer a seamless integration of voice, data, and video accessible across wide range of platforms. Designing communication systems for this generation is more challenging because of the fact that there no single standard to follow.

Fortunately, the integrated circuits technology have matured enough that signal processing operations can be performed using reprogrammable and reconfigurable hardware. It is very useful in this situation because the same hardware can be reprogrammed for an entirely different system. This means that manufacturers will enjoy less time to market, cost reduction, and will be able to rapidly prototype new standards. This will result in better services for consumers at comparatively cheaper prices.

Therefore, reprogrammable FPGA or software radio solutions can be very beneficial to the manufactures as well as end customers. This is particularly true in for research and development platforms and the university laboratory environment.

1.1.4 NI Tools for MIMO Research and Development

National Instruments provide hardware and software tools for engineers. One such tool is “Communications Systems Design Pioneer Program.” This hardware contains all the necessary and required components to build a communication system though this requires a host computer and appropriate software on the host. This software is also provided by the National Instruments which is used to design and implement the communication systems. Dr. Dong has acquired the above mentioned hardware and software to conduct his research in OFDM and MIMO communication systems.

The result of this research has developed a system which provides a ground to further investigate the communication systems. It can be used to research and study OFDM communication systems, software defined radios, real-time systems, IF and RF signals for research. The communication system developed in this research provides the tools to testing needs and requirements for Dr Dong’s MIMO research.

1.2 Research Overview

This paper presents the mathematical modeling and simulation, and firmware/hardware implementation of a Multiple Input Multiple Output (MIMO) Orthogonal Frequency Division Multiplexing (OFDM) communication system.

Section 2 provides an overview of software radios, OFDM techniques and WiMax standard.

The project started with a research and understanding of OFDM communication systems, especially the mathematics on which it is based upon. The mathematical model was used to develop the simulation algorithm in MATLAB. The detailed mathematics and signal formatting required for WiMax is described in section 3. Section 4 presents the MATLAB simulation algorithm and results. In the beginning the basic OFDM system was implemented and gradually more and more pieces were added to it to help analyze the close to the real world results. This software helped a lot to understand the OFDM system and its response to various system parameters. Knowledge acquired through these stages of development was used to build a physical prototype.

Though there were new challenges faced like partitioning of the algorithm into hardware and software pieces, understanding the hardware architecture of the platform, a new programming language style, software development environment and various ready to use modules. This was overcome by attending seminars and workshops, contacting the tool manufacturer, studying various manuals, and application notes and examples.

Finally, a successful system was built and tested. And it was demonstrated to the Faculty advisors.

2 OVERVIEW

2.1 Software Defined Radio

A software defined radio uses general purpose computer and/or reconfigurable hardware to perform signal processing operations. Their ability to change radio protocol in real-time just by changing the software have found their way into cell phone market and into the military. At present software defined radio technology provides an excellent option for rapid prototyping a multitude of communication standards. However, in the long run it is expected to become mainstream technology in the communication industry.

The hardware for a software defined radio comprise of a RF front end, ADC (analog to digital converter), DAC (digital to analog converter), DUC(digital up-converter), DDC(digital down-converter), processor and/or reconfigurable hardware. The RF front end converts the RF signal to (and from) IF signal, which is then converted to (and from) digital form using ADC (DAC). The digital data is then processed using some processor and/or reconfigurable hardware. The concept of software defined radio technology became reality only because of highly efficient hardware, which includes: DACs, ADCs, Digital Signal Processors (DSPs) or general purpose CPUs, reconfigurable hardware known as FPGAs. Therefore, we can achieve software reprogrammable flexibility for complex communication systems.

The following figure gives a glimpse of different signal processing operation intensity (complexity and required operation rate) as compared to the hardware complexity, allowing the appropriate hardware elements to be selected.

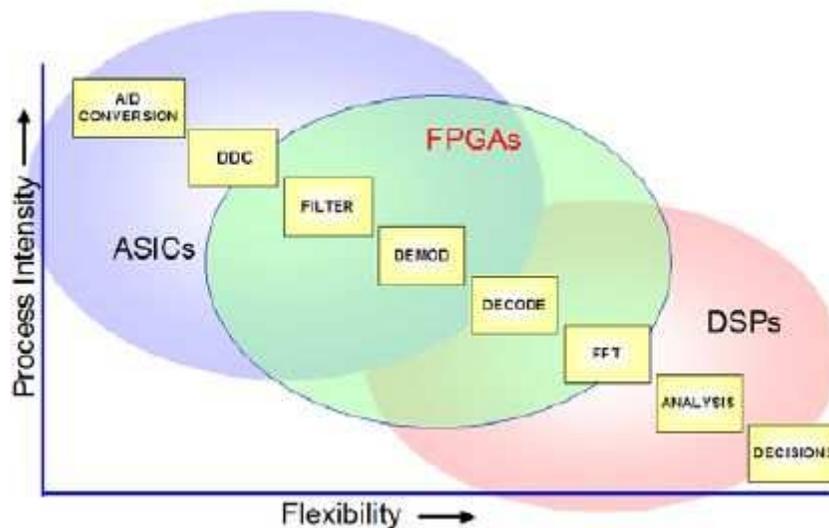


Figure 2: Signal processing intensity versus flexibility [18]

Many software radios use a combination of the hardware elements shown. For example, high speed A/D converters are followed by digital downconverters (DDC), a field programmable gate array (FPGA), and either a digital signal processor (DSP) or personal computer.

The next figure shows the evolution of a highly sophisticated FPGA component family from Xilinx Inc. It is obvious to note that a significant amount of reprogrammable hardware can fit into these chips. Beside this, other significant features are on-chip RAM and multi-core CPUs. Features like these have made it possible to transform an analog communication system into digital one. As these devices get more advanced, it will be possible to build a software only radio.

	Virtex E	Virtex II	Virtex II-Pro	Virtex 4 LX	Virtex 4 SX	Virtex 4 FX
	XCV 600E	XC2V 3000	XC2V P100	XC4V LX100	XC4V SX55	XC4V FX100
Logic Cells	15,552	32,256	99,216	110,592	55,296	94,896
Max Block RAM	295k	1,728k	7,992k	4,320k	5,760k	6,768k
Max I/O User Pins	512	720	1164	960	640	768
18 x 18 Multipliers	-	96	444	96	512	160
Clock Mgmt Blks	-	12	12	12	8	12
405 PowerPCs	-	-	2	-	-	2
Rocket I/O Serial	-	-	20	-	-	20
Gbit ENET Ports	-	-	-	-	-	4

Figure 3: Evolving FPGA generations [18]

Similarly, Digital Signal Processors (DSPs) are also gaining advance features. It is obvious from the following figure that DSPs are becoming high performance and low power. They can support many communication schemes including broadband wireless WiMAX. Therefore, DSPs are essential part of embedded communications and handheld communication devices like cellular telephones. Following chart shows the family of DSP processors for Telecom from Texas Instrument.

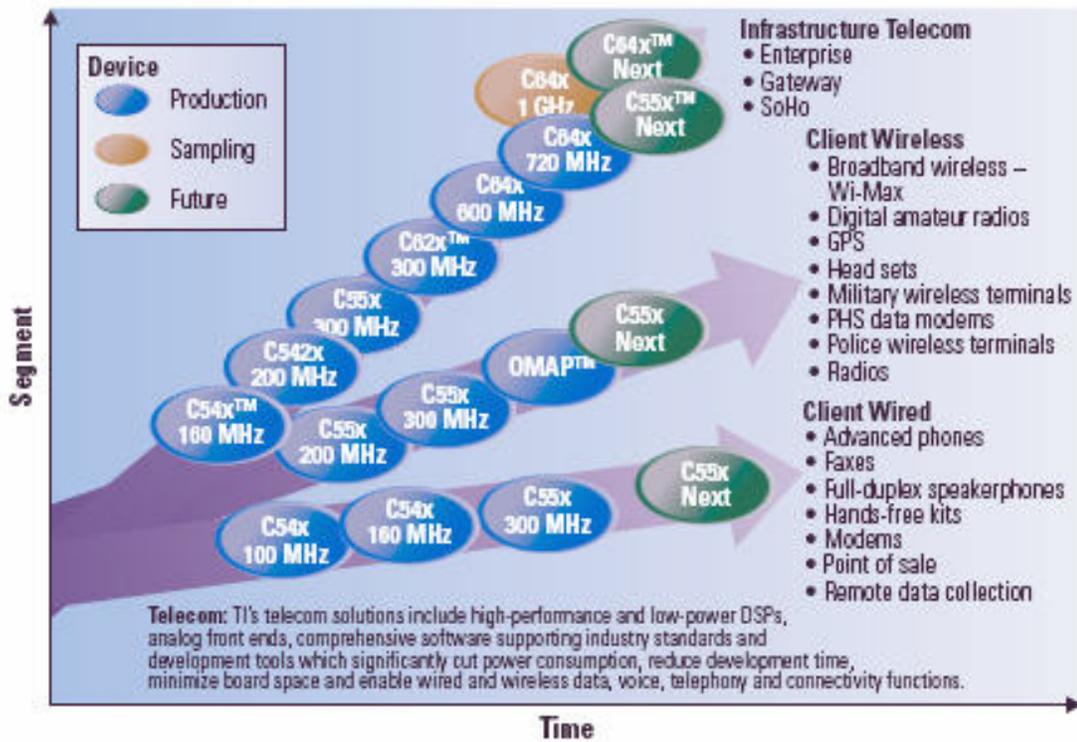


Figure 4: Evolving DSP generations [19]

2.2 OFDM

Desirable feature of an advanced communication system are providing high bandwidth, supporting multi-path and resilience to channel impairments. OFDM signaling technique has proved to very successful to meet these demands because of its characteristics to utilize the available bandwidth highly efficiently and lesser prone to channel distortions. It can also support the multi-path communication for wired and wireless signaling. Following discussion describes the OFDM communication starting with simpler Frequency division multiplexing (FDM) system.

Frequency division multiplexing (FDM) of multiple signals is widely used in communication system. In a FDM communication system more than one signal are transmitted over a single channel. This is achieved by dividing the available bandwidth into many sub-channels. Examples of such system are wireless systems, cable.

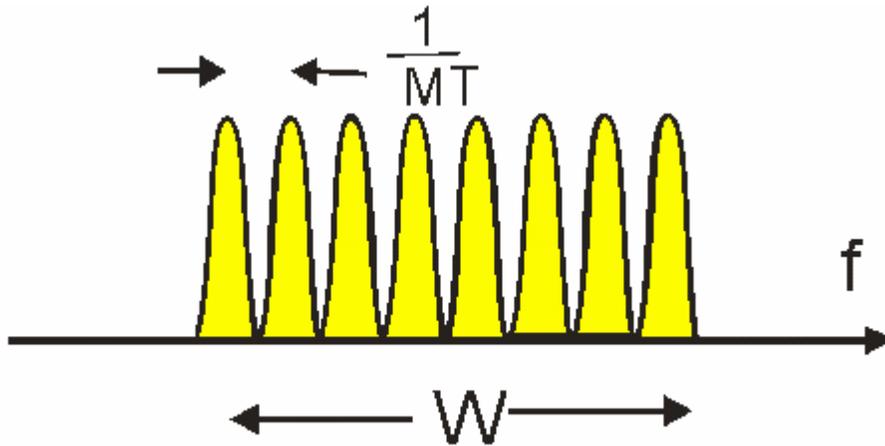


Figure 5: Spectrum of a FDM (frequency division multiplexing) communication system [1]

One such system is the advanced mobile phone systems (AMPS) initially deployed as the cellular telephone standard in the U.S. Multiple users communicate simultaneously on assigned frequency channels with base stations that use FDM receiver and transmitters to support the communications. AMPS used 30 kHz channels distributed within two 25 MHz frequency bands.

In case of an Orthogonal FDM (OFDM) system the available bandwidth is also divided into several sub-channels but the spacing between any two adjacent sub-carriers provides orthogonality. Orthogonality provides a reduced channel bandwidth with minimal inter-symbol interference between channels. This increases the number of sub-carriers that can be transmitted over a single channel or in other words higher spectral efficiency. Other advantages are its capability to be less prone to multi-path channel distortions, and RF interference resilience.

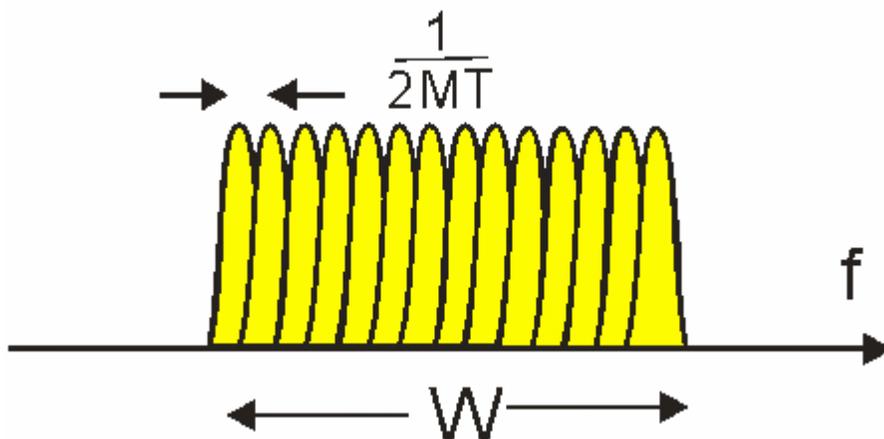


Figure 6: Spectrum of an OFDM (orthogonal frequency division multiplexing) communication system [1]

OFDM waveforms are readily constructed using forward and inverse discrete Fourier transforms (DFT). In a similar approach to telephony FDM transmultiplexers [3], digital data symbols are placed into DFT spectral bins. An inverse DFT is performed on the spectral bins to create a time waveform. The time waveform has a prefix added to the time waveform that is equal to a defined time period at the end of the waveform, a cyclic prefix. The received time waveform is then received and processed using a forward DFT to extract the original bins. Due to inherent time delays, the bins are phase shifted. Reference pilot tones, included in the original spectral bins, are used to compute and remove the phase shift.

The widely used IEEE-802.11a (WiFi) standard for wireless Ethernet employs OFDM. It uses 64 sub-channel or orthogonal carriers, cyclic prefix which less than or equal to $\frac{1}{4}$ of the symbol. It can support four different modulation schemes namely BPSK, QPSK, 16-QAM, and 64-QAM. All the bins are not used for data. As there some are used as guard bin in order to eliminate Inter Symbol Interference (ISI). And pilot tones are inserted to estimate phase delays.

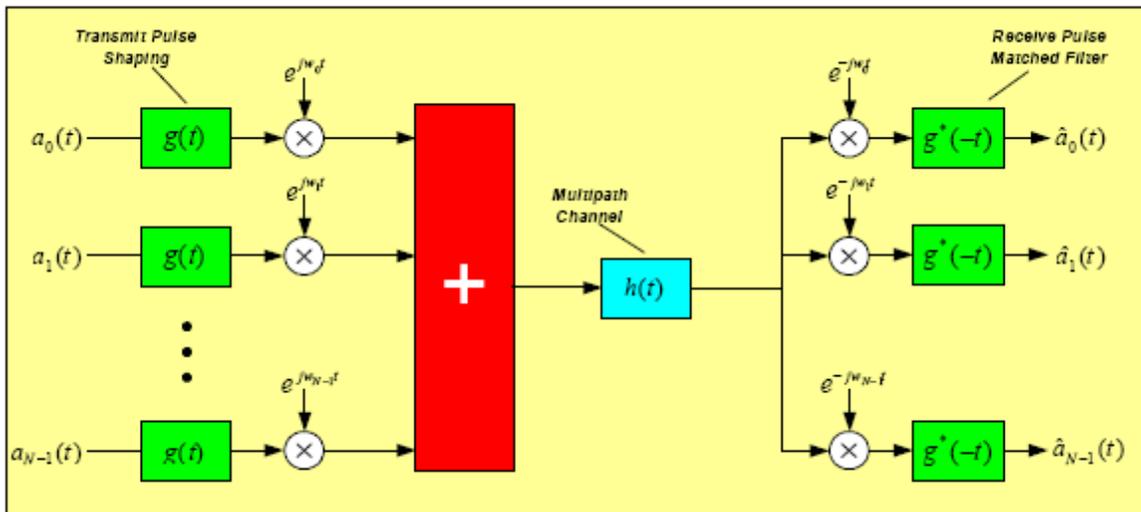


Figure 7: Basic structure of an OFDM system [4]

2.3 802.16 and WiMAX

WiMAX is the acronym for “Worldwide for Microwave Interoperability Access,” which is also referred as IEEE 802.16. Because it is the standard set by IEEE for Broadband Wireless Metropolitan Area Network Access. Its purpose is to provide an alternative solution to the existing DSL, cable, and T1/E1 technology for the last mile access. Internet hot-spots will also benefit from it. It is capable of providing broadband access over larger distances. IEEE approved its first version in 2001 which was later published in year 2002. This was only suitable for fixed line-of-sight connectivity and providing access within the range of 5 kilometers. Later version improved these drawback by introducing lower frequencies 2 GHz to 11GHz (previously 10GHz to 66GHz), and extending the accessibility to 50Km at a rate of 75 Mbits/s. The other improvement was its capability to connect non-line-of-sight points. It is less expensive than the DSL technology and also provides more

bandwidth than DSL does. For that reason it will be very suitable for multimedia and faster internet accessibility.

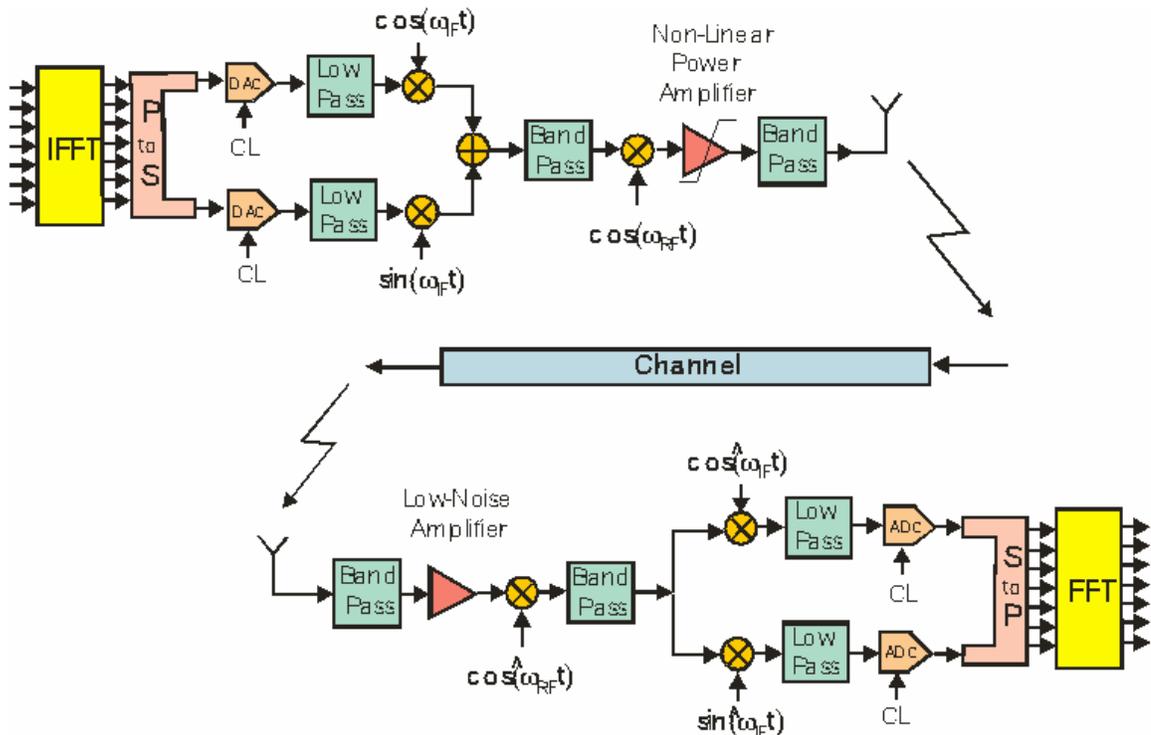


Figure 8: WiMAX transceiver [1]

As shown in the previous figure, the WiMAX technology is dependent upon OFDM methodology.

2.4 Tools for Rapid Prototyping

National Instruments provides a wide range of software and hardware tools to support both system simulation and rapid hardware prototyping. For communications, the LabVIEW software environment can be combined with real-time programmable hardware based on the NI Communications System Design Pioneer Program.

“The National Instruments Communications System Design Pioneer Program is designed for communications researchers and designers who need a platform for researching communications system design, software-defined radio, and emerging wireless technologies and techniques.” [10] Dr. Liang Dong has acquired the “Communication Systems Design Pioneer Program” along with necessary software from National Instruments for research at WMU.

2.4.1 NI PCI 5640R

The communication system is build around National Instruments' PCI 5640R card, shown in the following figure. The NI PCI-5640R card features an IF transceiver based on Xilinx FPGA.



Figure 9: PCI 5640R card [10]

The NI PCI-5640R contains two IF input channels consisting of high rate ADCs and digital downconverter (DDCs), two IF output channels consisting of high rate DACs and digital upconverters (DUCs), a Xilinx XXXXX FPGA, and a PCI bus interface to a host (PC). A block diagram of the card is shown in the following figure. The interface also has four DMA channel to support the communication between host and the card.

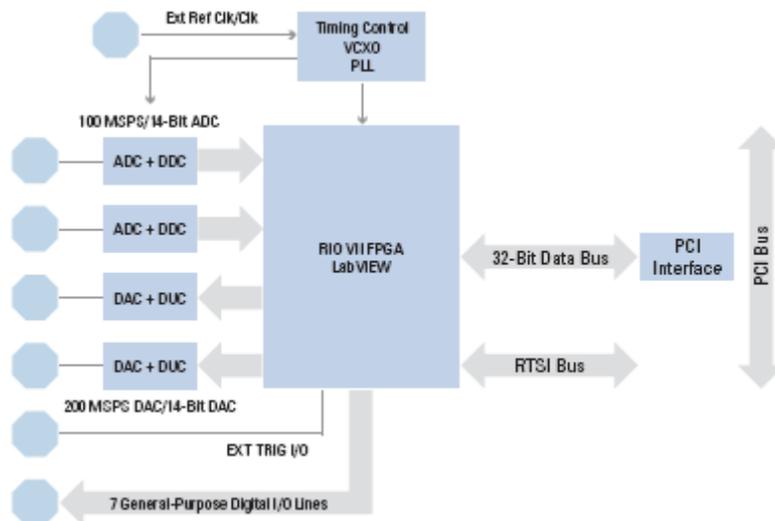


Figure 10: PCI-5640R block diagram [10]

For operation, the NI 5640R requires a computer. It is attached on the PCI bus of a computer in order to program its functionality and make it work. National Instruments also provides the basic software platform and libraries.

2.4.2 LabVIEW Software and Host

LabVIEW Full Development System and LabVIEW FPGA Module are essential components to develop a communication system. Other available modules can speed up the development process, namely, LabVIEW Modulation Toolkit, LabVIEW Spectral Measurements Toolkit, LabVIEW Digital Filter Design Toolkit, and the LabVIEW Signal Processing Toolkit.

The computer used for the project, from DELL, was specifically acquired by Dr. Liang Dong to support wireless communications research and the hosting of available hardware and software modules. Following figure shows the hardware and software (operating system) details.



Figure 11: Computer system used to implement the research project

3 DETAILED DESCRIPTION

This section provides a preview of technologies used to develop the system. This includes concepts about Software Defined Radio (SDR), and general description of WiMAX system and standards.

3.1 OFDM Symbol Structure

OFDM divides the spectrum of a given channel into several sub-channels, typically in the range of 64 to 1024. The resulting bandwidth of each sub-channel is correspondingly reduced. These sub-channels are equally spaced and the spacing between two adjacent sub-channels is orthogonal to provide maximum use of the bandwidth.

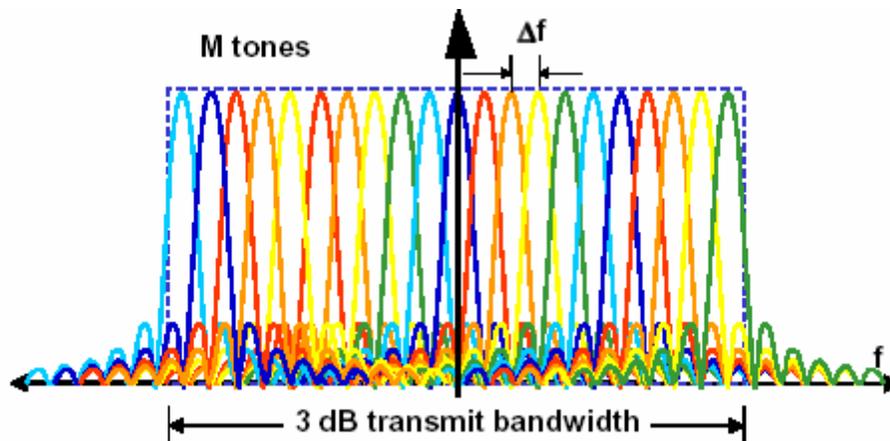


Figure 12: OFDM spectrum [16]

Modulation (and demodulation) of so many sub-channels can be achieved by simply a Discrete Fourier Transform (DFT) operation, which can easily be implemented using efficient Fast Fourier Transform (FFT) algorithms.

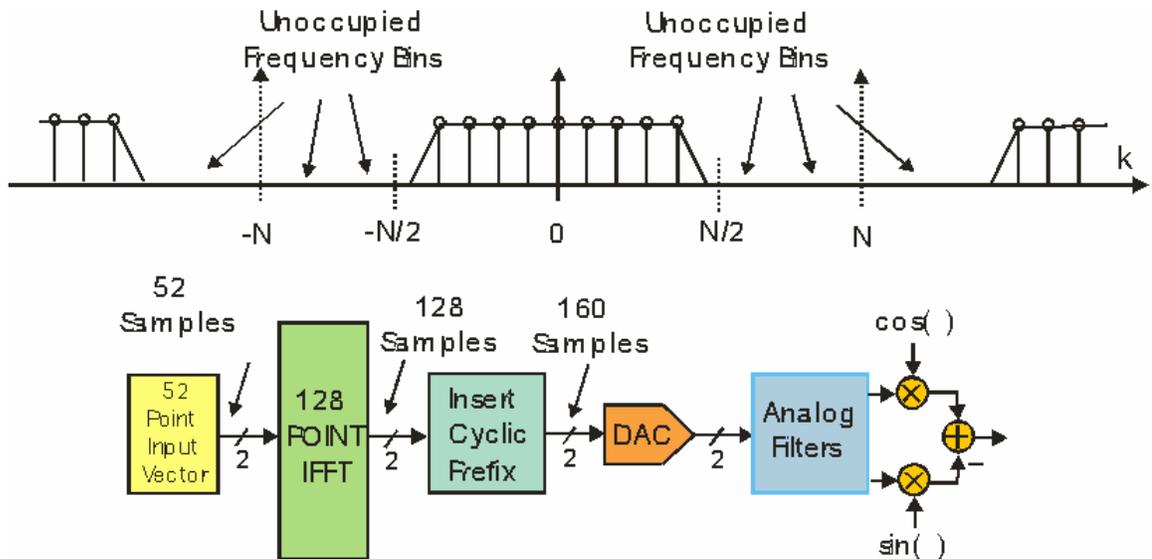


Figure 13: OFDM signal generation [1]

Each sub-channel is assigned a specific role. Phase and amplitude of the carrier depends upon the modulation scheme used. Once the spectrum is constructed, it is converted to a time sequence using an Inverse Discrete Fourier Transform (IDFT) operation. As we know, an FFT converts a time sequence to its frequency domain whereas an IFFT converts the spectrum of a signal into its time domain sequence. The use of Inverse Fast Fourier Transform (IFFT) algorithms makes the process very efficient. Further, it also ensures that the sub-channels are orthogonally apart. The data points in frequency domain, which are to be converted to time sequence using IFFT, are known as bins.

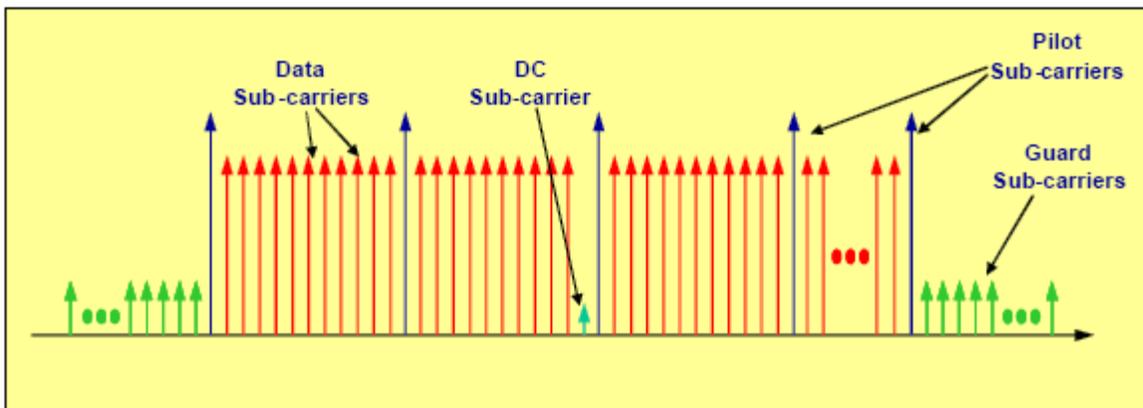


Figure 14: OFDM symbol structure and sub-channelization [4]

To reduce inter-symbol and inter-carrier interference a guard interval is inserted in each symbol which makes the period of transmitted symbol longer than the active symbol. To ensure this, the period of guard symbol is kept longer than the delay of any echoes present in the signal. This reduces the data capacity of the signal but not so much as the OFDM provides so many sub-carriers. Cyclic prefix insertion is another name for this scheme.

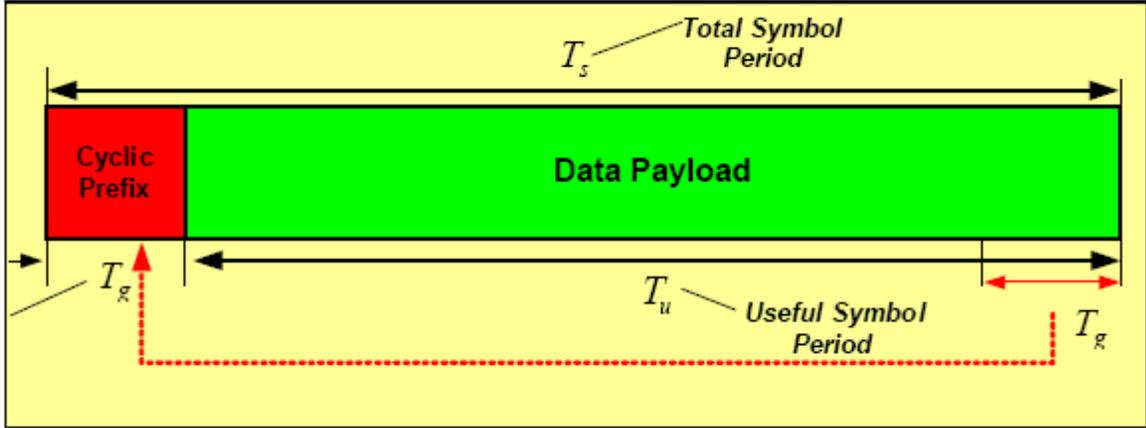


Figure 15: OFDM symbol time structure showing insertion of Cyclic Prefix [4]

Following section describes an OFDM system in more details including the math behind it.

3.1.1 OFDM Theoretical Development

3.1.1.1 Orthogonal Carriers [16]

Orthogonality of any two adjacent sub-carriers is the main feature of any OFDM system. Mathematically, two functions $f(x)$ and $g(x)$ are orthogonal over a the period $[a, b]$ if and only if they satisfy the following equation

$$\int_a^b f(x) \cdot g(x)^* \cdot dx = 0 \quad (1)$$

If $f(x)$ and $g(x)$ represent signals then the left hand side of above equation is the common energy of their spectra.

Reminder, OFDM uses sinusoidal carriers. Let's assume two sinusoidal functions $\exp(j \cdot 2\pi \cdot n \cdot f_0 \cdot t)$ and $\exp(j \cdot 2\pi \cdot m \cdot f_0 \cdot t)$ then,

$$\frac{1}{T} \cdot \int_T e^{j2\pi \cdot n \cdot f_0 \cdot t} \cdot e^{-j2\pi \cdot m \cdot f_0 \cdot t} \cdot dt = \begin{cases} 0, & m \neq n \\ 1, & m = n \end{cases} \quad (2)$$

where, $T = 1/f_0$.

Orthogonality of all the harmonics of a sinusoid of frequency f is shown in the above equation. Orthogonal carriers for OFDM signal are generated using this property.

3.1.1.2 OFDM: Signal Generation [16]

The input bit stream is mapped into complex symbols (IQ symbols) based on an m -ary constellation, which is then used to modulated to generate OFDM signal. Sequence of complex symbols is the reduced form of the bit stream.

Mapping the input bit stream to generate N symbols, and modulating N orthogonal sinusoidal carriers we get the following OFDM symbol

$$s(t) = \sum_{k=0}^{N-1} m[k] \cdot e^{j \frac{2\pi}{T} k t} \quad (3)$$

where, T is the active symbol period, N is the number of carriers, and m[k] is the kth symbol in the message symbol sequence for k in [0, N-1].

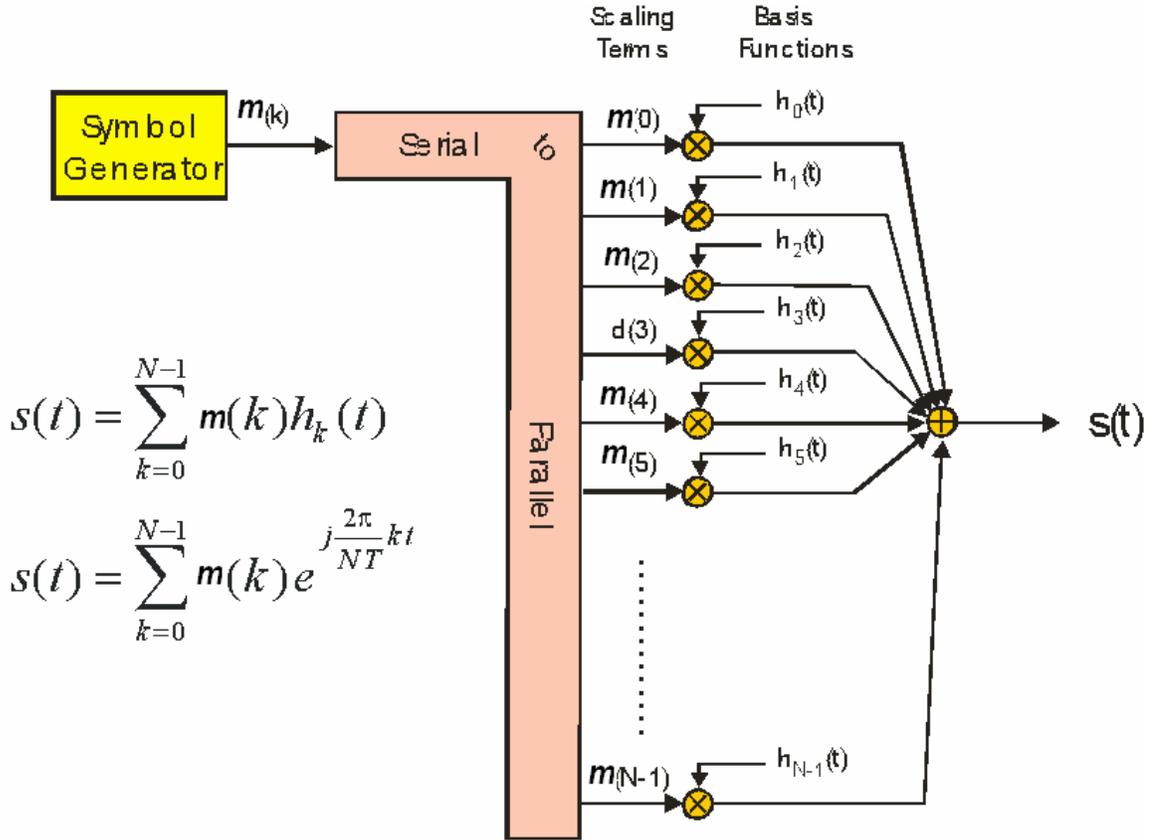


Figure 16: OFDM signal generation [1]

3.1.1.3 OFDM: Signal Demodulation [16]

Since all the carriers are orthogonal to each other and noting the above representation of an OFDM symbol, a symbol which was used to modulate a particular carrier say the ith harmonic of the fundamental frequency, can be recovered or demodulated just by integrating that carrier frequency.

$$\frac{1}{T} \cdot \int_T \left[\sum_{k=0}^{N-1} m(k) \cdot e^{j \frac{2\pi}{T} k t} \right] \cdot e^{-j \frac{2\pi}{T} i t} \cdot dt = m[i] \quad (4)$$

Although the spectra of carriers overlap but still the modulated symbols can be extracted from the carriers, as is shown by above mathematical equation.

OFDM symbol $s(t)$ can be denoted in discrete time as

$$s[n] = \sum_{k=0}^{N-1} m[k] \cdot e^{j\frac{2\pi}{N}k \cdot n} \quad (5)$$

where, continuous time t is replaced by discrete time n , and continuous time active symbol period T is replaced by N . One can recognize the Inverse Discrete Fourier Transform (IDFT) in the above expression. Hence, an OFDM symbol can be generated from a sequence of IQ symbols by taking their Inverse Discrete Fourier Transform (IDFT).

Also, we get the following expression if we replace integral by summation, “ t ” by “ n ,” and “ T ” by “ N ” in expression 4.

$$\frac{1}{N} \cdot \sum_{k=0}^{N-1} s[n] \cdot e^{-j\frac{2\pi}{N}i \cdot n} = \hat{m}[i] \quad (6)$$

where, $\hat{m}[i]$ is the estimate of the symbol modulating the carrier whose frequency is the product of the fundamental frequency and “ i .” As it is obvious that the above expression is the Discrete Fourier Transform (DFT), therefore OFDM signals can be demodulated by a DFT.

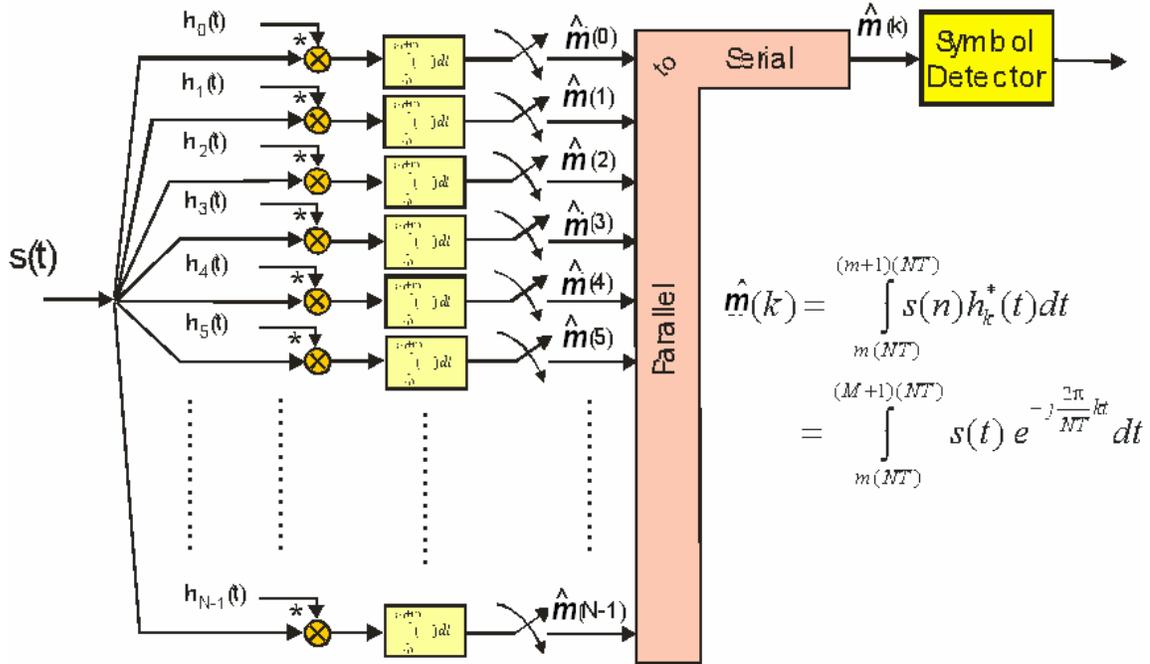


Figure 17: OFDM signal demodulation [1]

3.1.1.4 OFDM Guard Interval and Cyclic Prefix

When transmitting simple OFDM symbols, the duration of carriers is only within the OFDM symbol duration T . This period “ T ” is also known as correlation period.

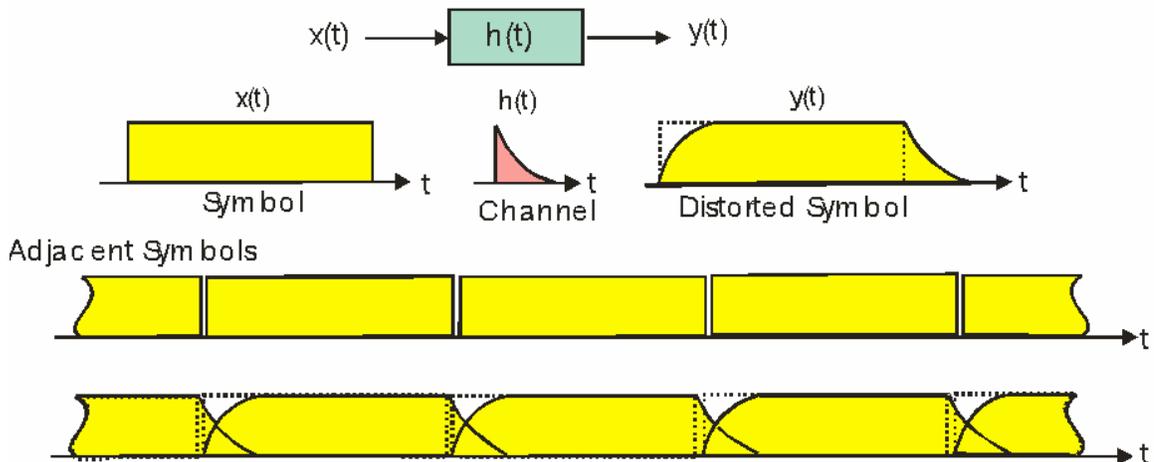


Figure 18: Adjacent Symbol Interference (ASI) symbol searing due to channel [1]

In channels which have frequency selective delay, some carriers can take longer duration to receive at the other end than others. This phenomenon can result into zero amplitude carriers for some part of the correlation interval.

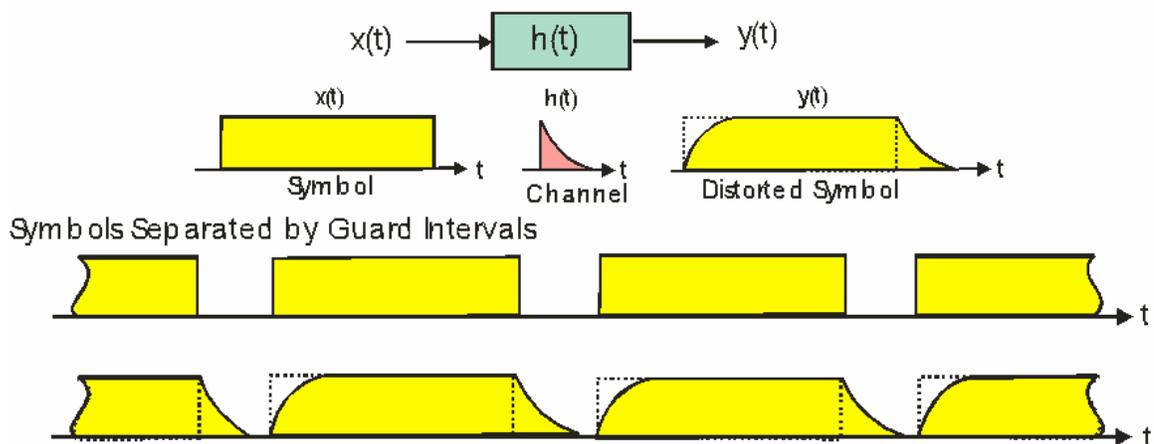


Figure 19: Insertion of Guard Interval between adjacent symbols to suppress ASI [1]

Every carrier should have an integer number of cycles in the correlation (integration) period in order to maintain orthogonality between any two received carriers. All the carriers are extended in time to minimize the effect of delay on the orthogonality of the carriers. This extension in time is known as “Guard interval insertion” or “cyclic prefix.” The period of this extension depends on the channel’s root mean square delay.

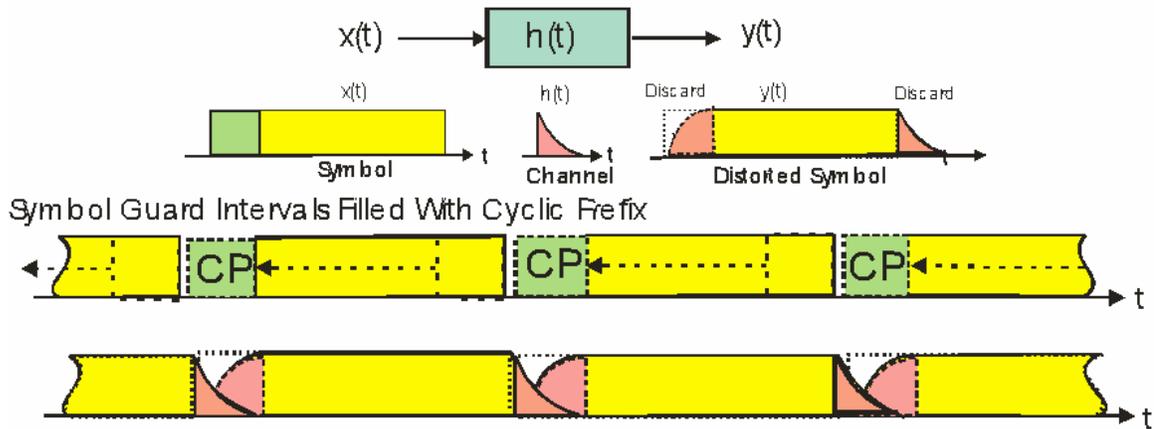


Figure 20: Cyclic Prefix inserted in the Guard Interval to suppress Adjacent Channel Interference (ACI) [1]

Also, cyclic prefix helps reduce the effect of channel transfer function effects from a linear convolution to a cyclic convolution. The channel effect, which is circular convolution, can be reduced by dividing the demodulator (DFT) by an estimate of channel transfer function. Since the circular convolution of two functions is the same as the product of their DFTs.

Cyclic prefix of two orthogonal OFDM carriers is shown in the figure.

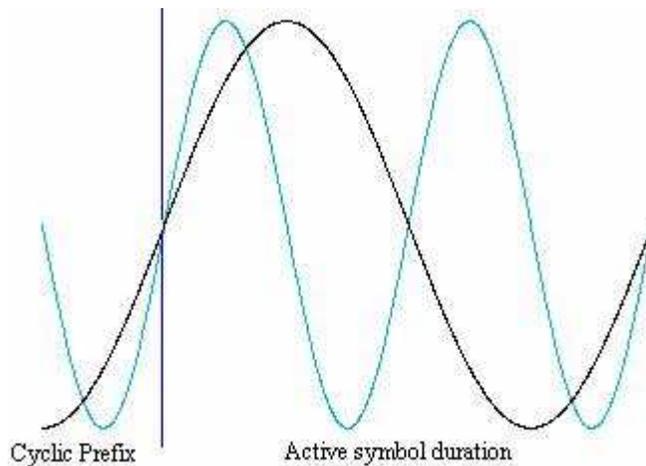


Figure 21: Cyclic Prefix insertion [16]

3.1.1.5 OFDM Spectrum [16]

Let's take a simple example of mapping a bit onto a carrier. In this example the presence or absence of the carrier in the interval T represents the bit. OFDM symbol in the time domain can be obtained by the product of (1) the sum of N orthogonal sinusoids, and (2) the rectangular pulse of interval T .

$$s(t) = \left[\sum_{k=0}^{N-1} b_k \cdot c_k(t) \right] \cdot \text{rect}\left(\frac{t}{T}\right) \quad (7)$$

where, c_k is the k^{th} carrier, $c_k(t) = e^{j2\pi f_k t}$, $\text{rect}\left(\frac{t}{T}\right)$ is the rectangular pulse over the interval $[-T/2, T/2]$, and b_k is the bit stream.

Spectrum of a rectangular (square) pulse is a sinc function, and that of a carrier is a set of impulses. Therefore, the spectrum of an OFDM symbol is the convolution of these two.

$$S(f) = \left[\sum_{k=0}^{N-1} b_k \cdot \delta(f - f_k) \right] * T \cdot \text{sinc}(f \cdot T) \quad (8)$$

In the above equation, there is a weighted sinc function and a shifted sum. Definition of a sinc function is as follows.

$$\text{sinc}(f \cdot T) = \frac{\sin(\pi \cdot f \cdot T)}{\pi \cdot f \cdot T} \quad (9)$$

The figure on the left hand side is the spectrum of one carrier in the OFDM signal whereas the figure on right hand side is the spectrum of the OFDM signal.

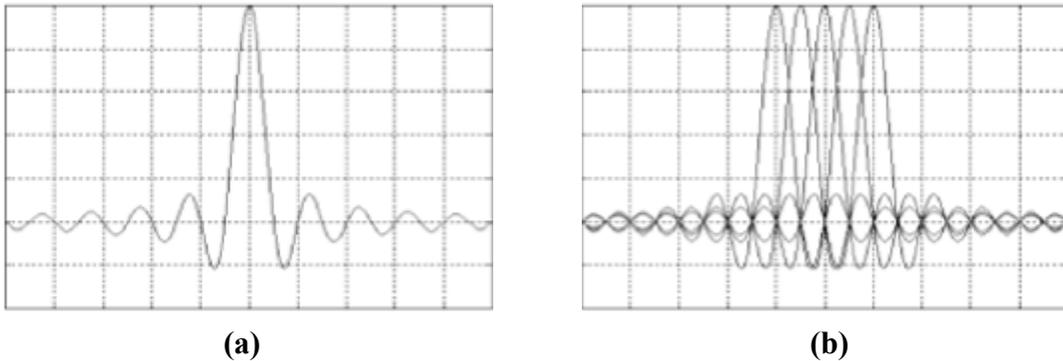


Figure 22: Spectrum of (a) an individual carrier and (b) an OFDM signal [16]

3.1.1.6 OFDM transmitter

If we put together all the pieces explained so far in a particular order, we can get an OFDM transmitter. The figure below shows one such transmitter.

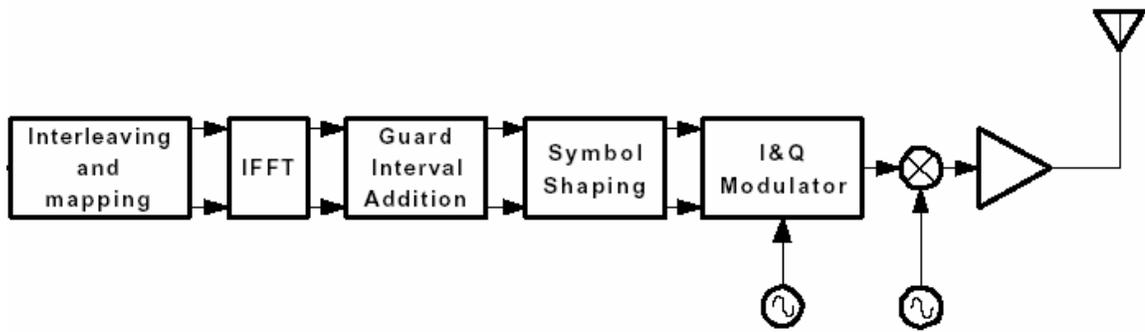


Figure 23: Basic OFDM transmitter [16]

The purpose of the “Symbol Shaping” block in the above given diagram is to remove high frequencies and to interpolate the basic OFDM signal. The output of this operation is a complex baseband signal. This complex signal is then fed into an “I & Q Modulator” which up-converts it into Intermediate Frequency (IF). The IF signal is then up-converted by the “RF Up Converter” to be transmitted.

3.2 WIMAX

The Institute for Electrical and Electronics Engineers (IEEE) has set a standard for different OFDM parameters, which is 802.16 IEEE standard for Local and Metropolitan Area Networks commonly known as 802.16 or WiMAX. This section describes the system parameters used to build the system.

Following table shows the primitive parameters which will then be used to define derived parameters.

Table 1: Primitive parameter definitions [12]

BW	nominal channel Band Width
N_{used}	number of used sub-carriers
n	Sampling Factor
G	ratio of Cyclic Prefix to useful time

The next table gives the definition of derived parameters by using the primitive parameters, which are defined in the previous table.

Table 2: Derived parameter definitions [12]

N_{fft}	Smallest power of two greater than N_{used}
Sampling Frequency (F_s)	$\text{floor}(n * \text{BW}/8000) * 8000$
Sub-carrier spacing (ΔF)	F_s / N_{fft}
useful symbol time (T_b)	$1/\Delta F$
Cyclic Prefix time (T_g)	$G * T_b$
OFDM Symbol time (T_s)	$T_b + T_g$
Sampling time	T_b / N_{fft}

In the previous two tables we have definitions of primitive and derived parameters. To clarify it even further, the next table pertains the specific values for the previously defined parameters.

Table 3: Parameters of transmitted signal [12]

N_{fft}	256
N_{used}	200
n	if(mod(BW,1.75)= =0) n=8/7; else if(mod(BW,1.5)= =0) n=86/75; else if(mod(BW,1.25)= =0) n=144/125; else if(mod(BW,2.75)= =0) n=316/275; else if(mod(BW,2.0)= =0) n=57/50; else n=8/7;
G	1/4, 1/8, 1/16, 1/32
Number of lower frequency sub-carriers	28
Number of higher frequency sub-carriers	27
Frequency Offset Indices of guard sub-carriers	-128, -127,..., -101 +101, +102,..., +127
Frequency Offset Indices of pilot carriers	-88,-63,-38,-13,13,38,63,88 (Note that pilot sub-carriers are allocated only if two or more sub-channels are allocated)

4 SIMULATION

MIMO-OFDM communication system was modeled and simulated using MATLAB software. After developing initial model for the OFDM communication system, best effort was made to follow IEEE specification for 802.16.

Simulation Goals:

- Better understanding of OFDM and the OFDM time symbols
- Develop IFFT to FFT relationship of spectral bin symbols
- Time sample time delays in the received path requiring both a cyclic prefix and pilot tones
- Noise and the effects of timing mismatch
- Arbitrary delay timing in the received path and pilot tone computations
- A general tool for OFDM education and future research

The simulation software was designed in way which makes it easier to modify key system parameters. For example, number of sub-channels, pilot tone assignment to sub-channels, pilot tones, etc. This feature proved very helpful during the model building of the system. The system can be easily simulated for any suitable number of sub-channels. Similarly, being able to play with pilot tones, helped to improve the system model. The system was iteratively simulated and improved for Effects of

4.1 OFDM Signaling

The following table shows the method followed to generate data to fill OFDM bins. These OFDM bins were modulated to generate the OFDM symbol or time sequence to be transmitted over channel.

Table 4: Filling OFDM bins

```
fd1=zeros(Bins_fft,1);
dx1ref=floor(sqrt(Mqam)*rand(Data_setsize,1))-(sqrt(Mqam)/2-0.5);
dylref=floor(sqrt(Mqam)*rand(Data_setsize,1))-(sqrt(Mqam)/2-0.5);

% Normalize by the symbols by the constellation average power
dx1=dx1ref/AvgMagSum;
dyl=dylref/AvgMagSum;

%
% Form the Pre-OFDM array
% inject data
% inject pilot tones
% (Note: data should be corrected for the pilot tones)
fd1((1+Doffset):(Data_setsize+Doffset))=dx1+j*dyl;
fd1(p_ind+Doffset+1)=p_vect;

% fill the conjugate positions prior to ifft
fd1(Bins_fft:-1:(Bins_fft/2+2))=conj(fd1(2:Bins_fft/2));
```

Following is the spectrum of the OFDM symbol that before transmission. The pilot tones can be easily recognized in this diagram.

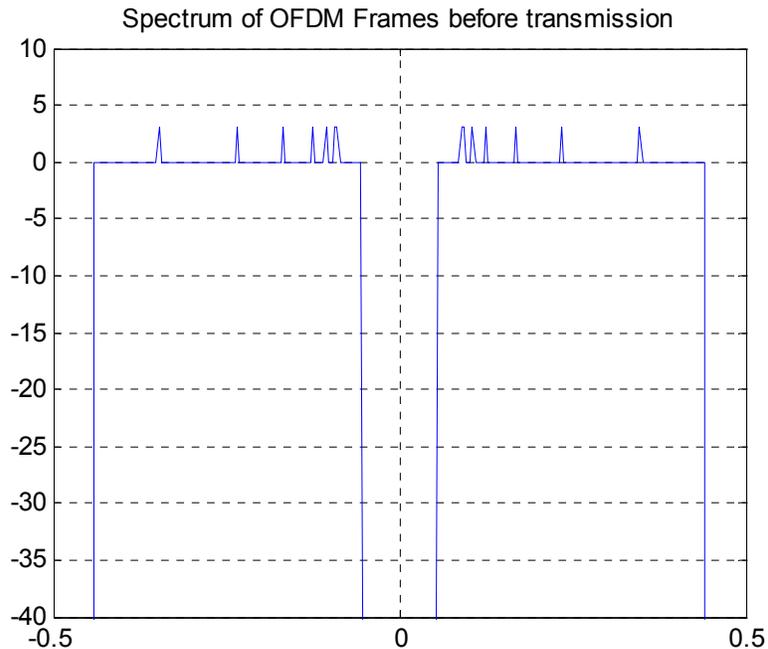


Figure 24: OFDM spectrum prior to transformation and transmission

4.1.1 OFDM Time-domain Signals

The following figure shows the time sequence of the OFDM signal or time domain OFDM symbol. The time domain OFDM signal sequence before

transmission is the result of OFDM modulation or simply IDFT. This figure also shows the effect of other required preprocessing which is injection of pilot tones, and addition of cyclic prefix.

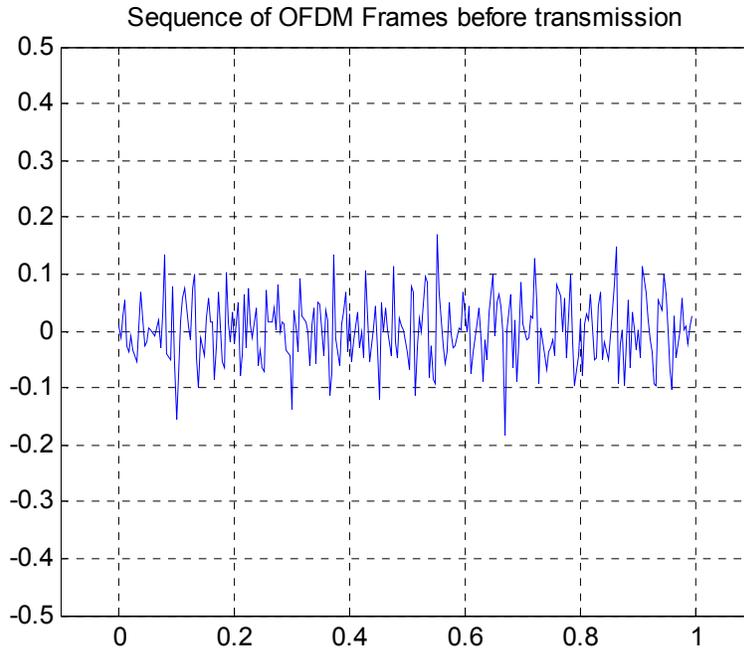


Figure 25: Time domain representation of OFDM signal at the transmitter

4.1.2 OFDM Received Signals

The following figure shows the received OFDM signal ready for processing. A quick visual comparison of two figures showing the OFDM time domain signal might lead one into believing that they don't match. But after a careful analysis one could tell that the received signal is equivalent to the transmitted signal if a part of the received on the left hand side is detached and attached on the right hand side. It is because of the fact that the transmitted signal has the additional part of the cyclic prefix. The receiver locks onto the transmitted signal and keeps receiving until the desired length. Additional redundant part of the cyclic prefix provides enough time for the receiver to start receiving. So the part which was at the end on the transmitted side may end up on the beginning of the receiver side. Still, the data is preserved and demodulation (DFT) generates the same spectrum as on the transmitter end. The additional part gets removed on the receiver side but parts of the time sequence are rearranged.

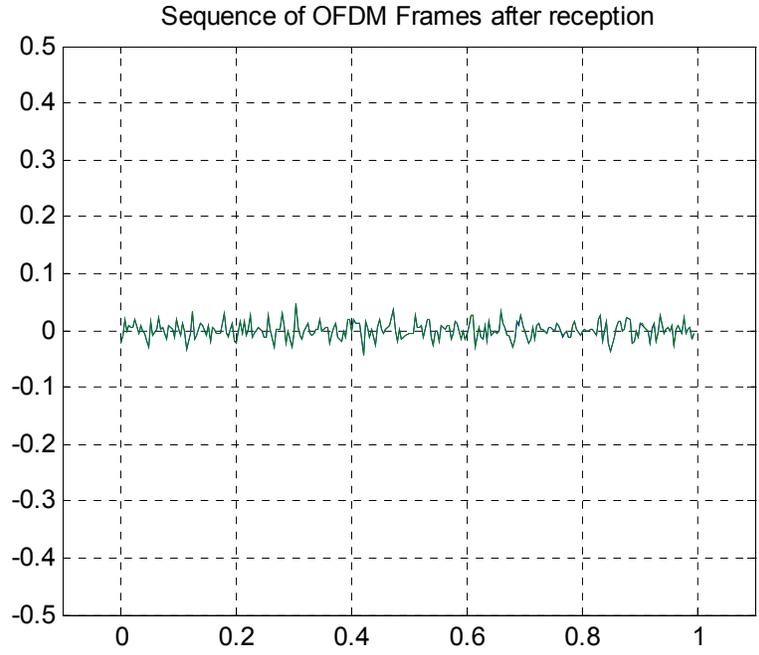


Figure 26: Time domain representation of OFDM signal at the receiver

The following figure shows the exact difference between the transmitted OFDM signal and received OFDM signal. Again, keep in mind that this is figure shows the result of a simulation. In other words, the original OFDM signal was convoluted with the channel transfer function to approximate the channel simulation.

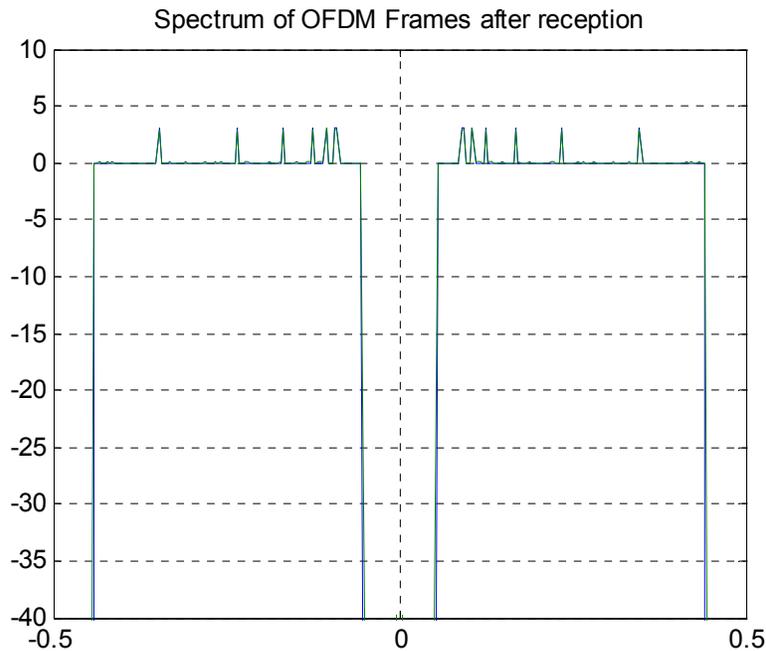


Figure 27: OFDM spectrum after reception

The OFDM signal in the previous diagram is the recovered signal. Since the transmitted signal contains the redundant part of cyclic prefix. Also, because of the channel distortion some data gets changed from its original value. Therefore, the signal is chopped back to its original length. Then pilot tones are used to restore the phase distortion due to channel time delays.

4.1.3 OFDM Constellation Plots

Constellation plots are another way to visualize the communication signals. Following is the constellation plot of transmitted OFDM signal. The inner four points represent the data whereas the outer four points are pilot tones.

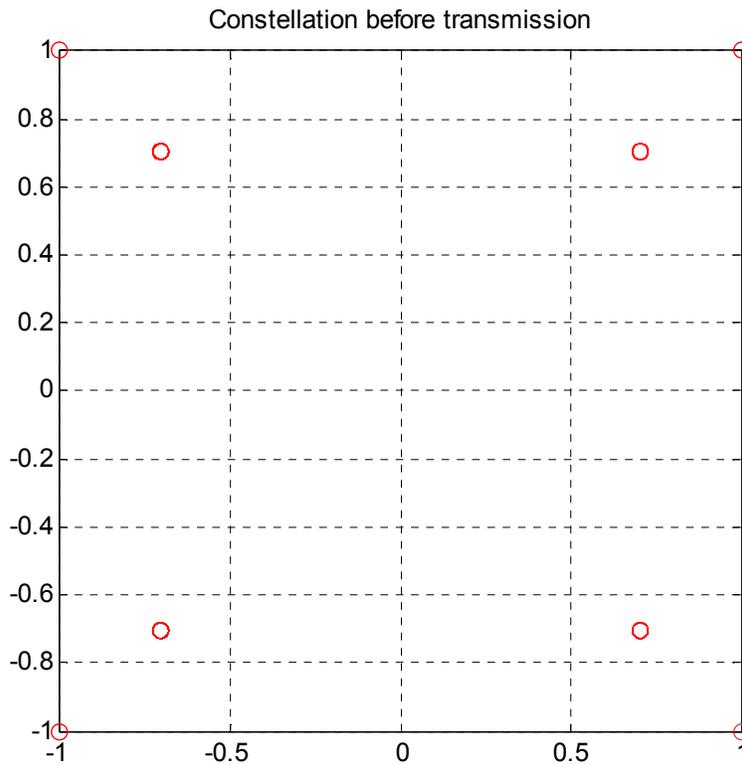


Figure 28: Constellation plot prior to transmission

The following plot exclusively shows the pilot tone constellation of transmitted signal.

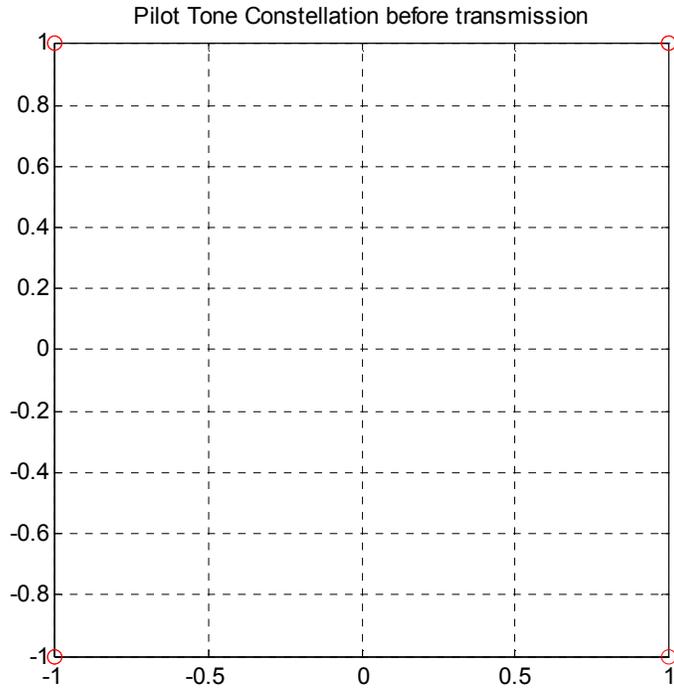


Figure 29: Constellation plot showing only the pilot tones prior to transmission

The following two plots are for the received OFDM signal but without any channel impairments. In other an OFDM signal received over an ideal channel.

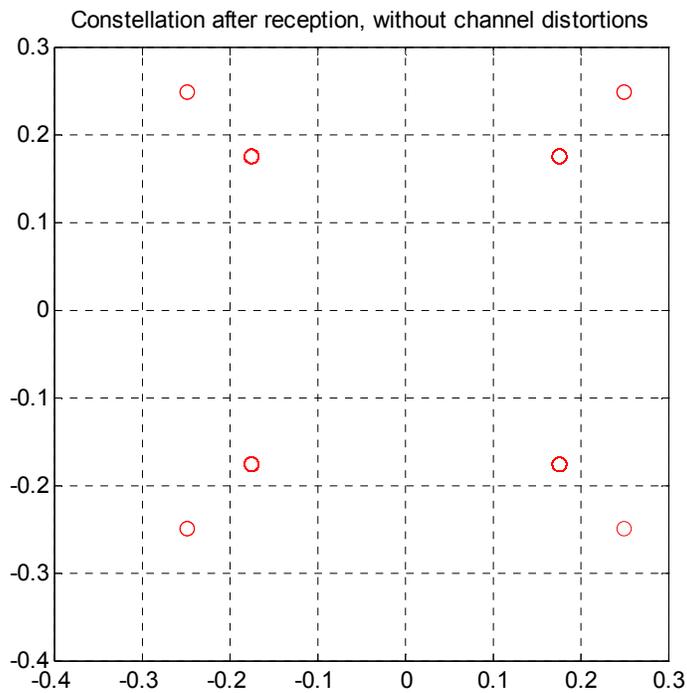


Figure 30: Constellation plot after the reception, with no channel distortion

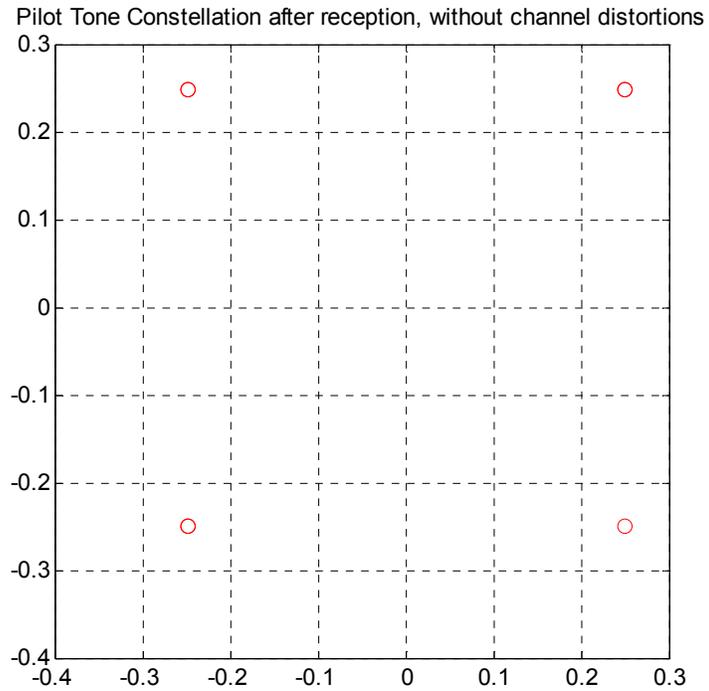


Figure 31: Constellation plot showing only the pilot tones after the reception, with no channel distortion

4.2 OFDM with Channel Impairments

Simulation software is designed in a way to be able to quickly experiment with different levels of channel noise and analyze the results. The following table depicts the additions of channel noise to the received signal to achieve more realistic results.

Table 5: Channel noise

```
function [dlchan_n] =Channel(dl_twice,i);

% Channel Distortion
%
% Routine call:      dlchan_n =Channel(dl_twice)
%
% Input parameter:  dl_twice (no channel)
% Output parameter: dlchan_n (data distorted by channel)

global Bins_fft;global Doffset;global Data_setsize;global chan;
global SNRdB;global Mpsk;global Mqam;global p_num;global p_vect;
k=i;

normnoise=randn(2*k*Bins_fft,1);
normnoise=sqrt(2*(Data_setsize/2+2)/(Bins_fft/4))*normnoise/sqrt(sum(normnoise.^2));
normnoise=normnoise/(10^(SNRdB/20));

chan=[1 .2 0 0.02 0 j*0.1];
chan=chan/sum(chan);
dlchan=conv([zeros(2*Bins_fft,1); dl_twice]',chan);
%dlchan_n=dlchan+0.0435*(randn(1,4*Bins_fft+5)+j*randn(1,4*Bins_fft+5));

dlchan_n=dl_twice+normnoise;

return
```

The next table shows how the channel delays were simulated for the received channel.

Table 6: Channel delay

```
function [d3,timeoff] =DelayInterp(d1,timeoff);

% Received data time delay (offset)
%
% Routine call:      d2 =Delay(d1)
%
% Input parameter:  d1
% Output parameter: d3

global interpfactor;
global Bins_fft;

if nargin<2

timeoff=interpfactor*32+floor(interpfactor*Bins_fft/2*rand(1));
end

d2 = d1(timeoff:timeoff+interpfactor*Bins_fft-1);
d3 = d2(1:interpfactor:end);

return
```

The next figure shows the effects of channel delays and channel noise on the received OFDM signal. The following constellation plot shows that channel caused heavy phase distortions and there is lot of noise present in the signal. Due to these impairments the communication system requires to counter or restore these effects which will be later discussed in this chapter.

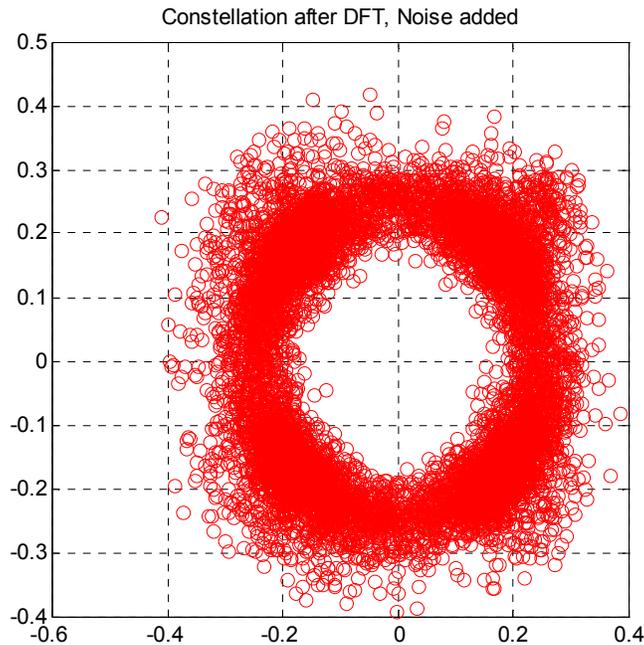


Figure 32: Constellation plot at receiver with channel noise and delay

The next figure shows only the pilot tones extracted from the signal.

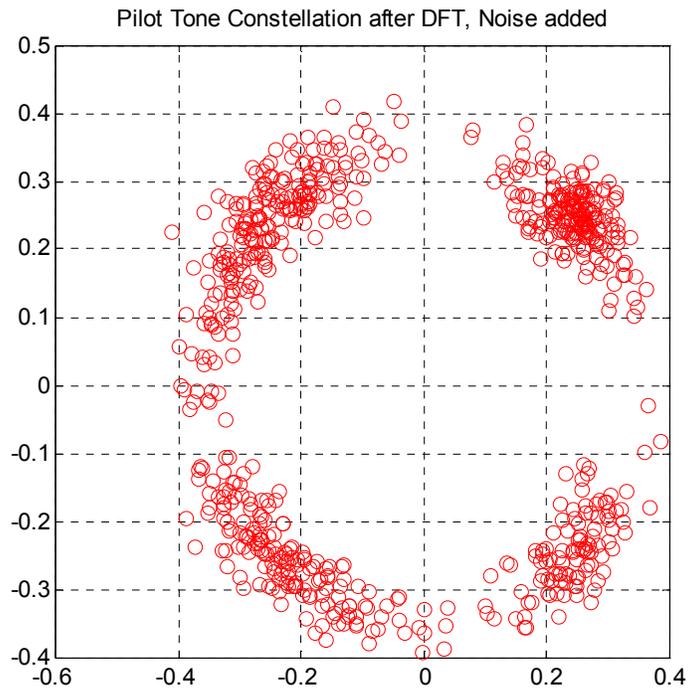


Figure 33: Constellation plot just for pilot tones at receiver with channel noise and delay

The next figure shows the spectrum of this heavily distorted OFDM signal.



Figure 34: OFDM spectrum at receiver with channel noise and delay

Next figure tells the difference in the spectrum of the original (transmitted) and distorted (received) signal.

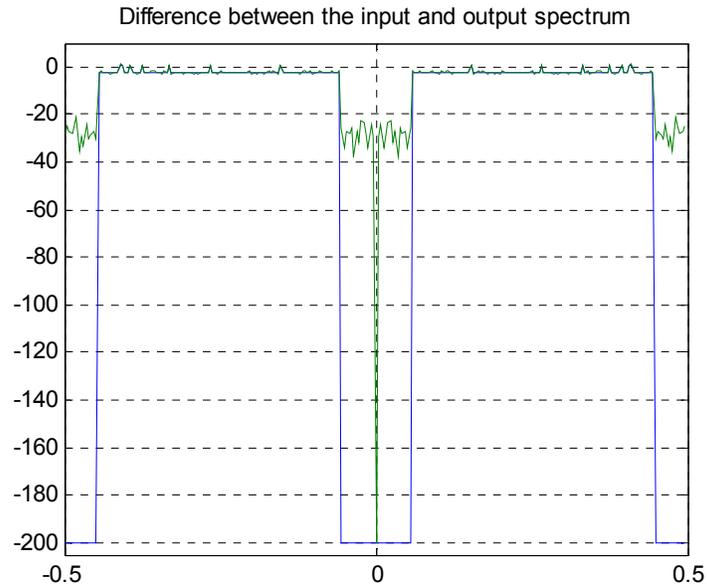


Figure 35: OFDM spectrum difference between the transmitted and received signal

Pilot tones were used to restore the phase distortion caused by the channel. Since the correct phase of pilot tones is known so this is used to estimate the distortion due to channel. Then using this estimate a correction vector is computed which is applied to the OFDM signal to minimize the channel effects. The following table shows the phase correction algorithm.

Table 7: Phase correction

```

function [fd_correct] =CorrectPhase(fd1, fd2);

%   Compute Phase Correction Vector
%
%   Routine call:      fd_correct=CorrectPhase(fd_shifted)
%
%   Input parameter:  fd1
%   Input parameter:  fd2 (with phase shifted)
%   Output parameter: fd2 (phase corrected using pilot tones)

global Bins_fft;global Doffset;global Data_setsize;global
SNRdB;global Mpsk;
global Mqam;global p_num;global p_ind;global p_vect;global
timeoff;
global pbinsteps;
global refphased phasesteps phase2
global interpfactor

% pilot tones
ptone_fd1=fd1(p_ind+Doffset+1);
ptone_fd2=fd2(p_ind+Doffset+1);

phaseptone=ptone_fd2./ptone_fd1;
refphased=360*(timeoff-1)/(Bins_fft);
ptonepdiff=angle(phaseptone)*180/pi;
pdptonepdiff=diff(ptonepdiff);
pdptonepdiff2=pdptonepdiff + 360*(pdptonepdiff<0);

phaseshift=(pdptonepdiff2./pbinsteps);
for mmm=2:(length(p_ind)-1)
    padd=round((phaseshift(mmm)-1)-
phaseshift(mmm))*pbinsteps(mmm)/360);
    pdptonepdiff2(mmm)=pdptonepdiff2(mmm)+360*padd;
    phaseshift(mmm)=pdptonepdiff2(mmm)/pbinsteps(mmm);
end

phasesteps=mean(phaseshift);
CorrectPVect=exp(-sqrt(-1)*(p_ind+Doffset)*(phasesteps*pi/180));
phase2=mean(angle((ptone_fd2.*CorrectPVect)./ptone_fd1));
CorrectPVect=CorrectPVect*exp(-sqrt(-1)*phase2);
angle((ptone_fd2.*CorrectPVect)./ptone_fd1);

CorrectVect(1:Bins_fft/2,1)=exp(-sqrt(-1)*(0:Bins_fft/2-
1)*phasesteps*pi/180));
CorrectVect=CorrectVect*exp(-sqrt(-1)*phase2);
CorrectVect(Bins_fft:-
1:(Bins_fft/2+2),1)=conj(CorrectVect(2:Bins_fft/2));
% Apply the phase correction vector
fd_correct =fd2 .* CorrectVect;

return

```

The following figure shows the spectrum of corrected OFDM signal.

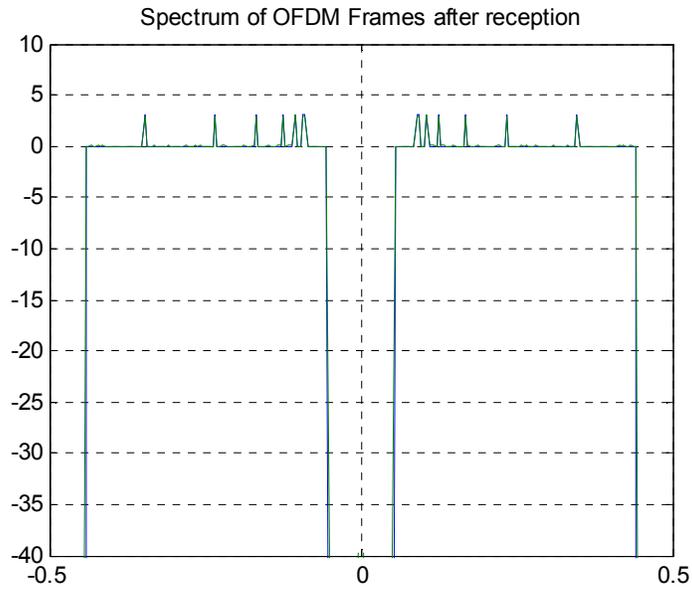


Figure 36: OFDM spectrum after reception with noise

The next figure tells the difference between the spectrum of the original (transmitted) and corrected OFDM signal on the receiver side. The difference level for both magnitude and phase response is minimized to the desired values.

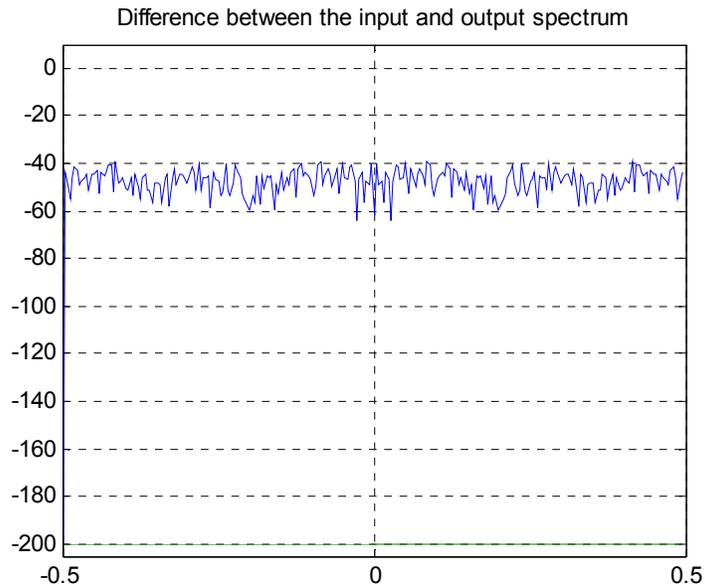


Figure 37: Error calculated by taking the difference between the transmitted and received signal spectrum

4.2.1 OFDM Constellation Plots

The next two plots show the constellation plot of received OFDM signal with channel distortions. The received constellation points are not as crisp as the transmitted constellation points. This is due to channel distortions. More noise in channel means larger a fuzzier constellation and vice versa. In this case, the error is in the acceptable range meaning that the system can clearly determine the constellation point quadrants.

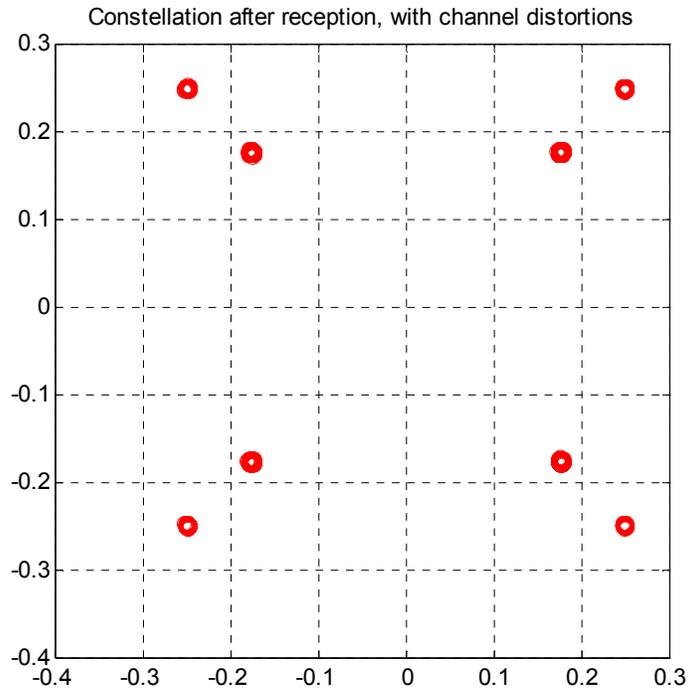


Figure 38: Constellation plots after the reception, with channel distortion

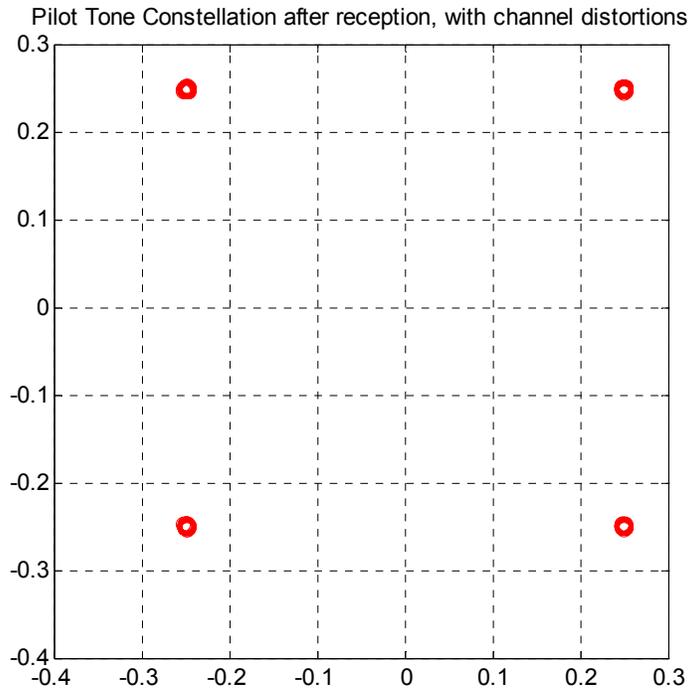


Figure 39: Constellation plot showing only the pilot tones after the reception, with channel distortion

5 IMPLEMENTATION

The OFDM communication system simulation which was designed using MATLAB was implemented using National Instruments' hardware and software. The tools to implement the system were provided by Dr. Dong.

NI Implementation Goals:

- Better understanding of NI hardware and software
- Testing requirements for MIMO research
- OFDM tools available
- Real-time, burst outputs
- Real-time OFDM symbol reception
- Real-time transmission and reception
- The OFDM lab capability
- A general tool for OFDM education and future research

5.1 Development using LabVIEW and NI Hardware

Implementation of a communication system required the understanding of NI hardware and software. This knowledge was acquired through various sources such as; tutorials, manual, workshops, example materials etc. Some of the most important pieces for the implementation are discussed next in the section

5.2 Graphical User Interface of the WiMAX SDR

The following figure shows the Graphical User Interface for the implemented systems. This front panel shows the spectrum of the signal being transmitted and received. On the bottom right, it tells the constellation points of transmitted and received signal. Transmitted and received point can be distinguished by their colors, red for transmitted and which for the received signal.

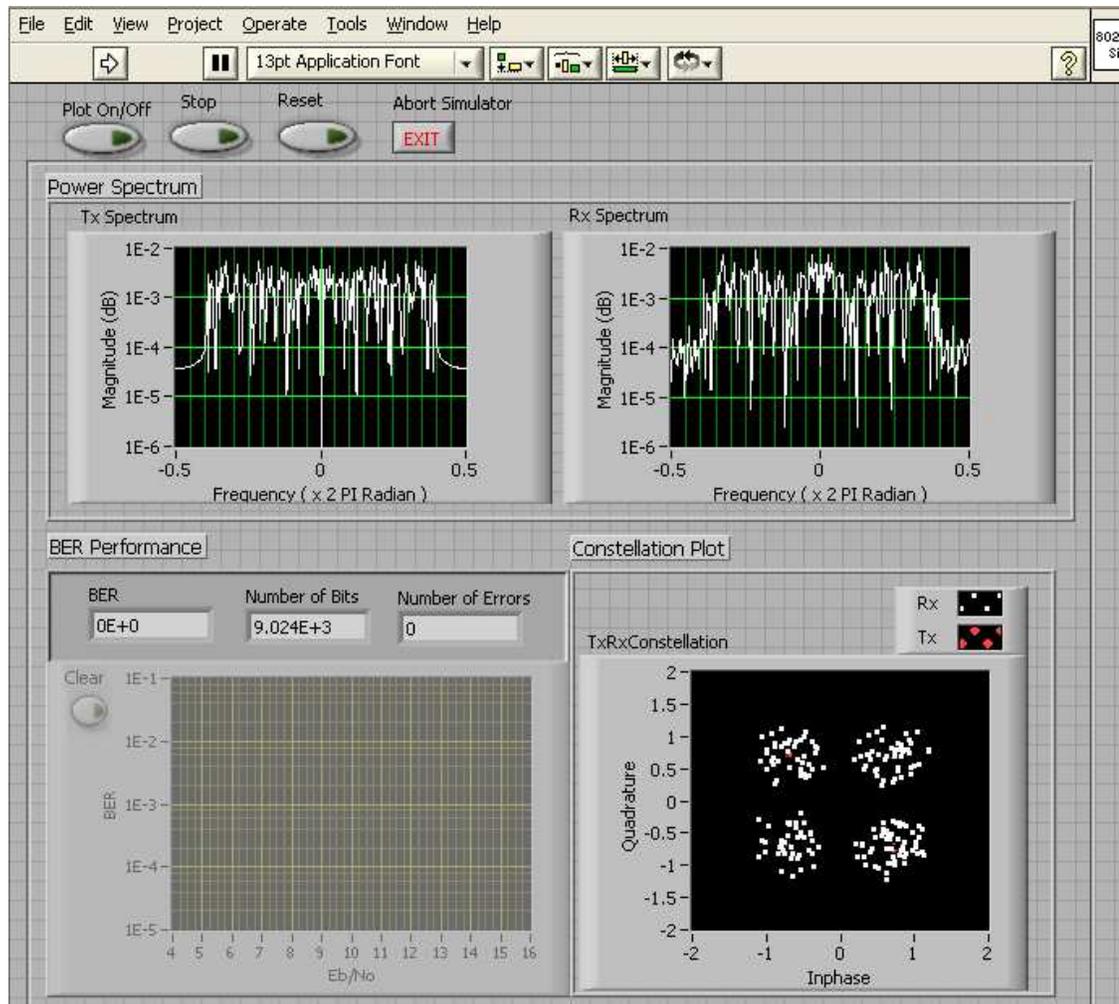


Figure 40: Graphical User Interface (GUI) of the system

Beside, run, stop, abort the GUI also gives an option to turn off the plots. Since we know that to compute and plot takes away lot of processing time therefore this control option is included in order fully utilize the system resource for the actual transmission and reception tasks. In this case the signal can be monitored through an oscilloscope.

5.3 Block Diagram of the Graphical User Interface of the WiMAX SDR

This next figure shows the programming (block diagram) of the GUI previously shown. This figure and most of the other block diagrams, shown later, are rotated in order to reveal all the details.

This is block diagram at the highest level which is the user interface to the OFDM communication system. It shows the two case structures inside an infinite loop. One case structure translates user inputs while the other structure selects the appropriate OFDM system.

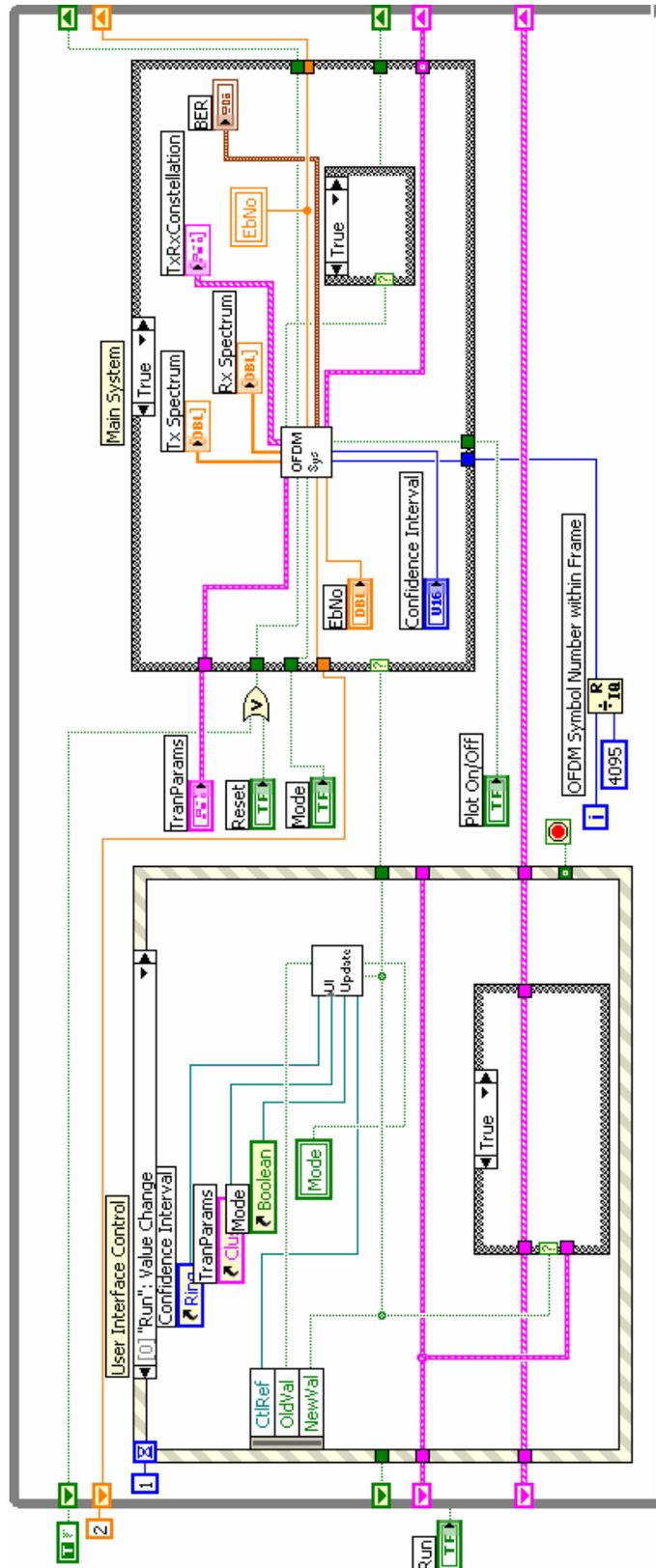


Figure 41: Block diagram of GUI

5.4 Block Diagram of the OFDM System

This figure shows the block diagram of the “OFDM Sys” in the previous figure. This is the main OFDM system where the whole sequence of communication system is defined. The little square blocks or modules are known as Virtual Instruments (VI) in LabVIEW language. These can be effectively used to reduce the complexity of the system.

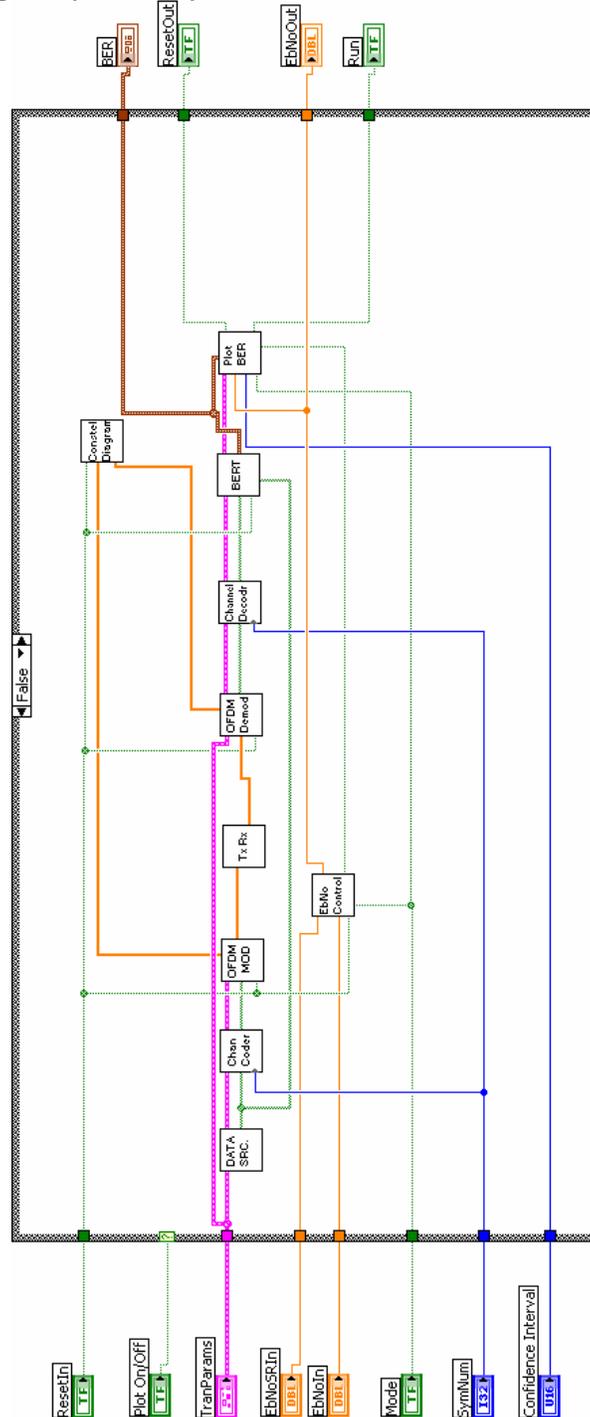


Figure 42: Block diagram of the OFDM system

5.5 OFDM Modulator Block Diagram

The “OFDM MOD” VI is used to generate the OFDM signal. QAM mapping, pilot insertion, modulation, and cyclic prefix insertion is handled by this VI.

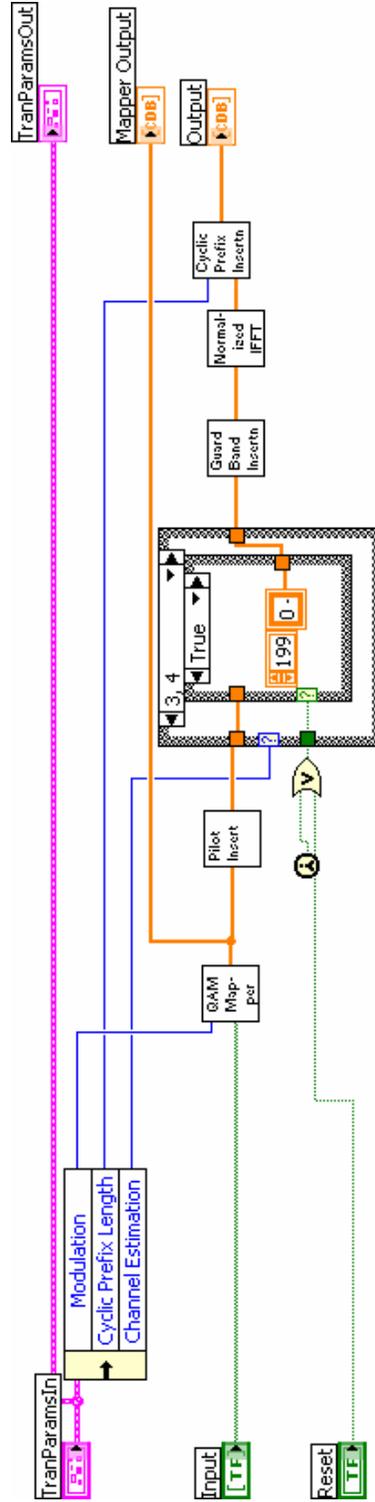


Figure 43: OFDM modulator

5.6 Pilot Inserter Block Diagram

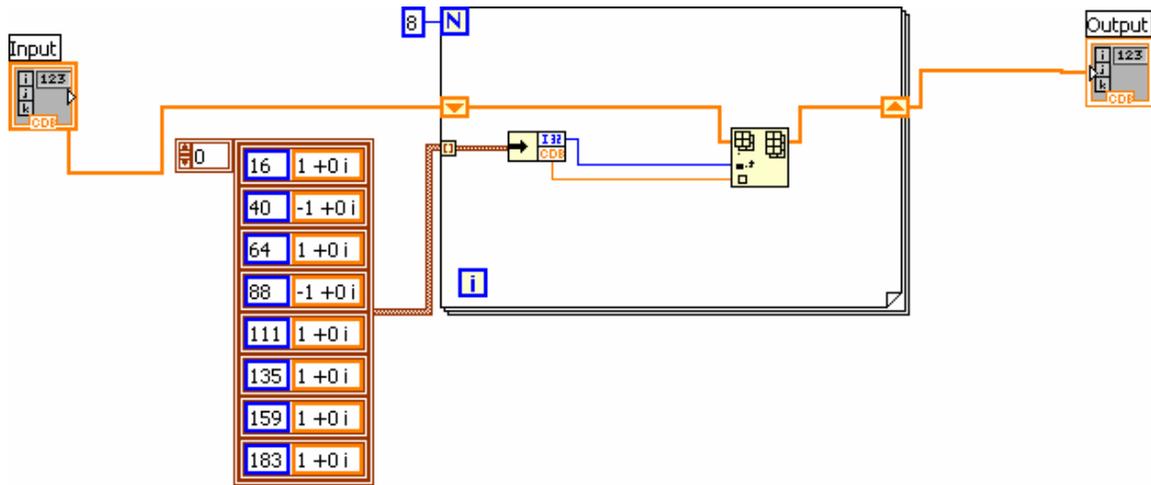


Figure 44: OFDM pilot inserter

The figure above shows the insertion of pilot tone indices and the values inserted in these locations.

5.7 Block Diagram of the Transceiver

The following figure shows the “TxRx” VI. This figure shows the transfer of data between host computer and the FPGA on the NI card. It also controls the data transfer on the FPGA. Notice the reference to FPGA is opened and closed only once. This figure also shows the computation of data for plotting both on the transmitter and the receiver end. That’s why one can identify two similar pieces in the figure. For the faster processing of data another VI was designed without data computation for the plotting.

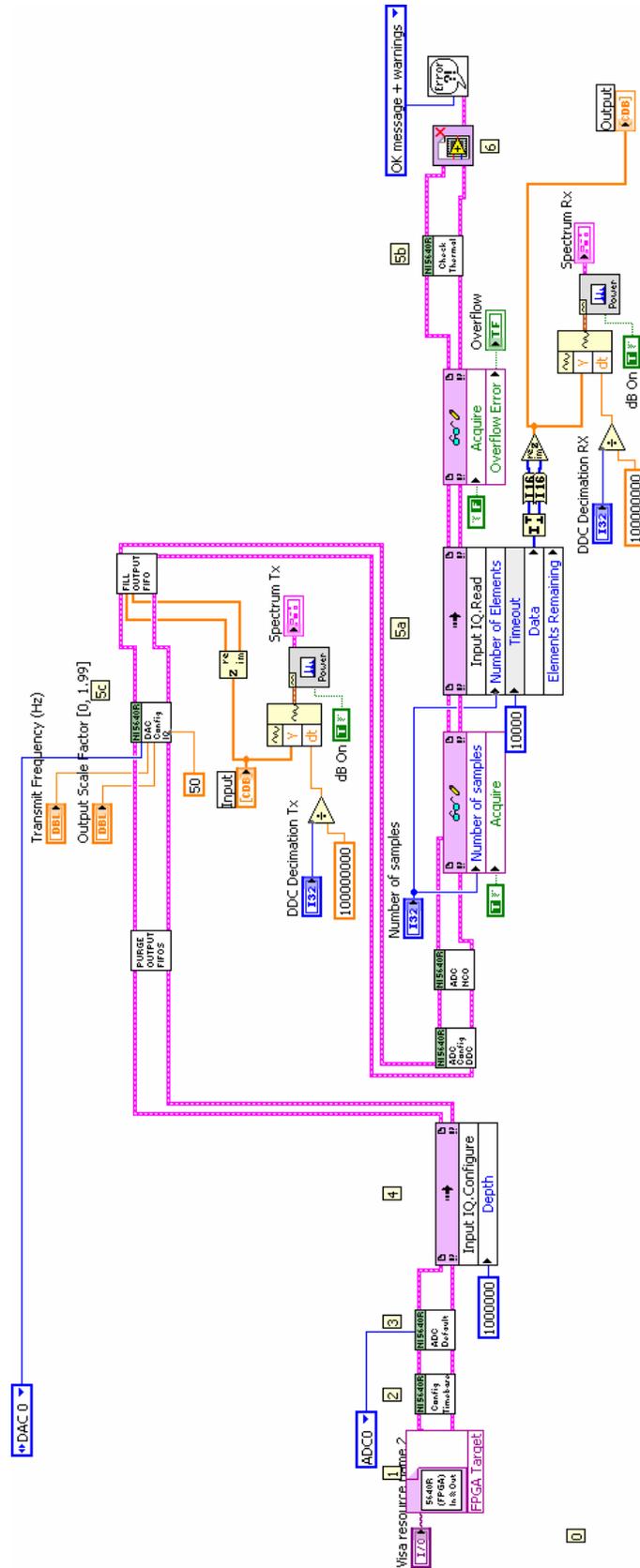


Figure 45: Block diagram of the transceiver

5.8 OFDM Demodulator Block Diagram

“OFDM Demod” VI is to extract the data from OFDM signal. It removes channel impairment using “Channel Estimate,” “Channel Equalize,” and “Channel Decode” VIs. Other notable VIs are “Cyclic Prefix Remove,” “Normalized FFT,” “Guard band Removal,” and “Pilot Extract.” Their functionality is obvious from their names.

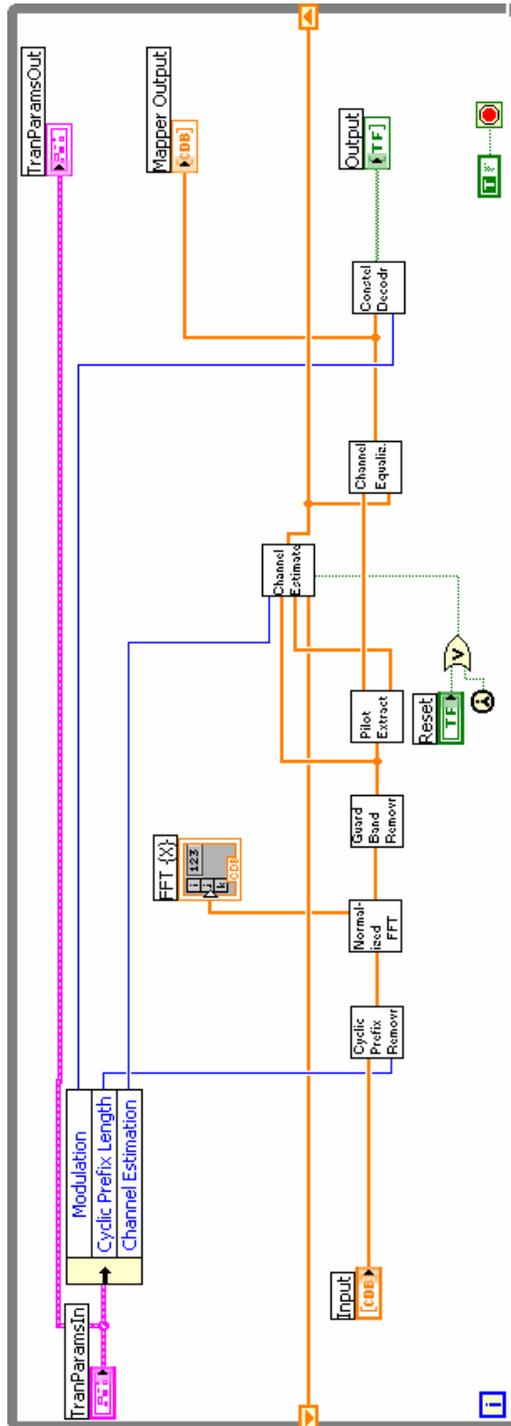


Figure 46: OFDM demodulator

6 SYSTEM RESULTS

In order to verify the communication system, the output channel was connected to the input. The channel was also probed using an oscilloscope. Following figure tells the system configuration.

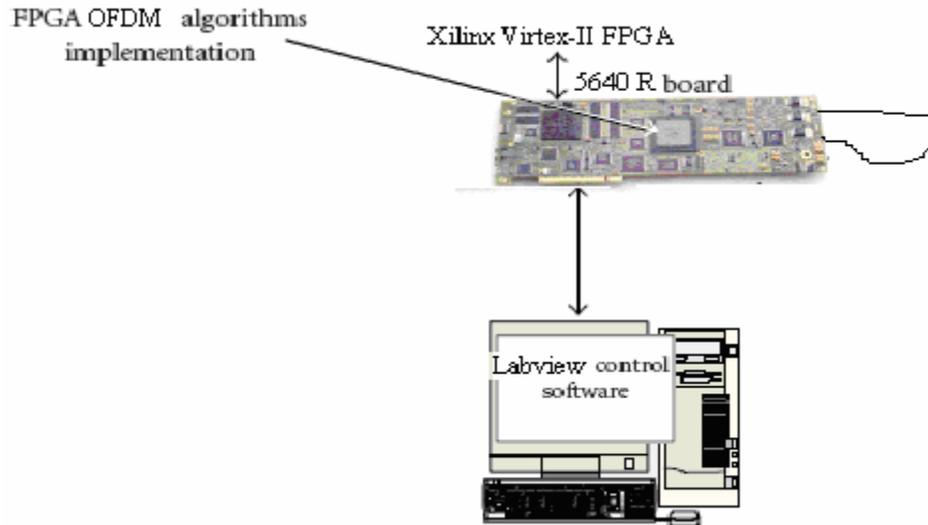


Figure 47: System setup for testing

The following picture shows the output of the card going back into the input channel, and the oscilloscope connection.

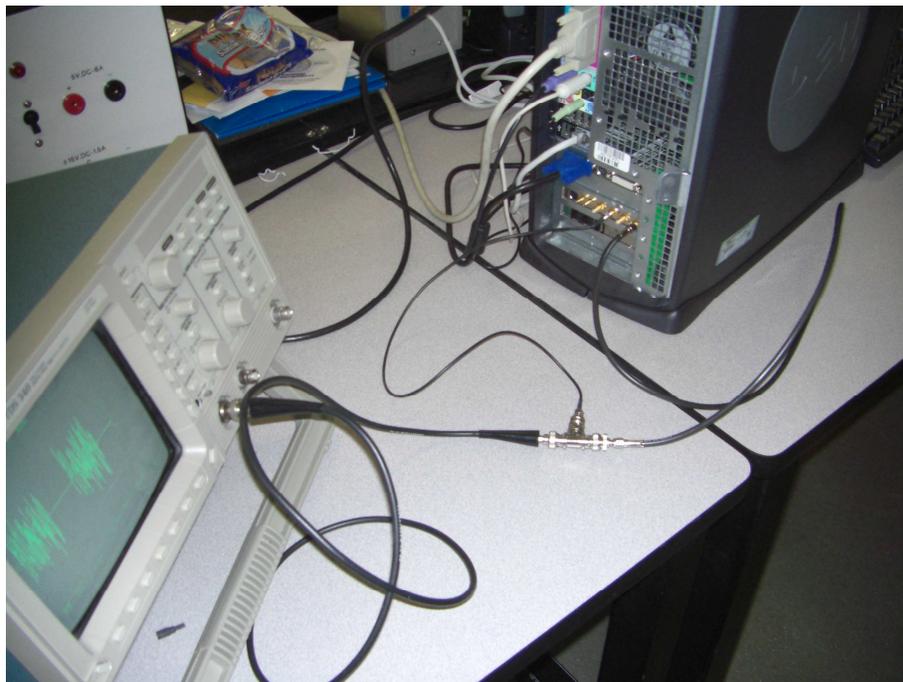


Figure 48: Channel and connecting transmitter and receiver, also oscilloscope probe on the channel

6.1 OFDM Signal

The signal characteristics in terms of the WiMAX specifications are following:

Table 8: Parameters of OFDM signal

N_{fft}	256
N_{used}	200
G	1/4
Number of lower frequency sub-carriers	28
Number of higher frequency sub-carriers	27
Frequency Offset Indices of guard sub-carriers	-128, -127, ..., -101 +101, +102, ..., +127
Frequency Offset Indices of pilot carriers	-88, -63, -38, -13, 13, 38, 63, 88

The following two pictures show the OFDM signal being transmitted over the wire line channel.

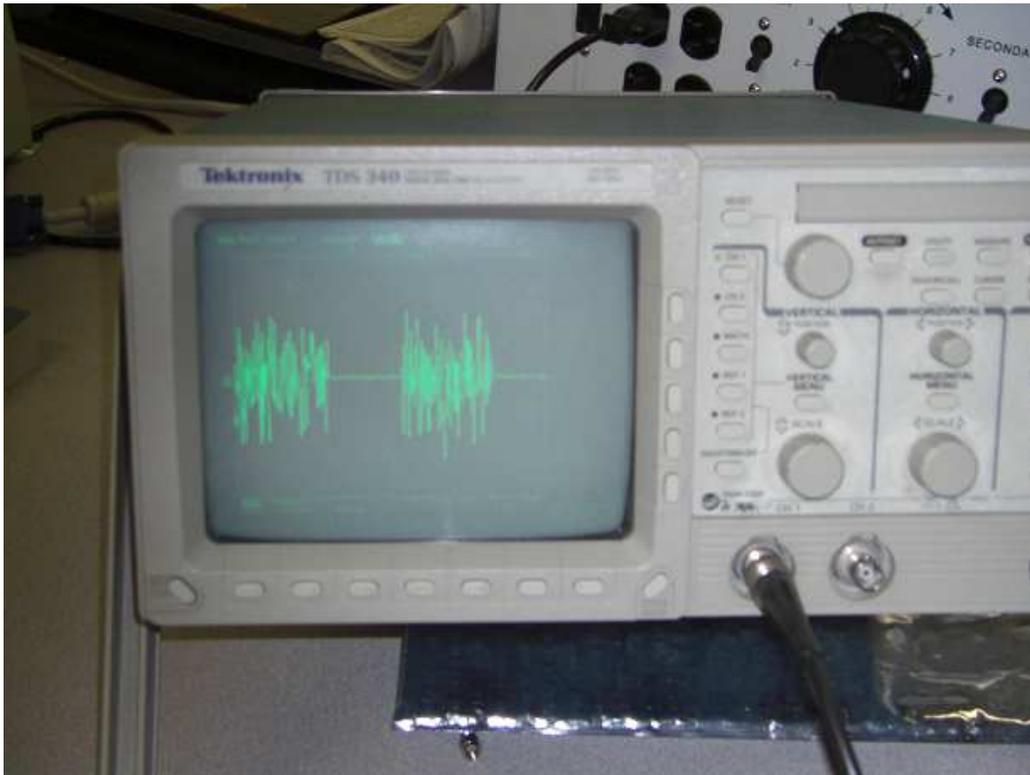


Figure 49: Two OFDM signal bursts



Figure 50: Four OFDM signal bursts

7 SUMMARY AND CONCLUSIONS

7.1 Results of this Work

The work undertaken for this research is a significant contribution in the field of modern communication systems. The system is implemented using software mainly with some reconfigurable hardware which is a newer technique for the implementation. Also, OFDM technique and WiMAX standard has been very recently introduced by the IEEE.

The project can be used as an example and a hands-on learning tool in many courses in the department of Electrical and Computer at Western Michigan University; like ECE 6640 Digital Communications, ECE 4600/5950 Communication Systems, ECE 5550 Digital Signal Processing, ECE 6950 Mobile Communications, ECE 6950 Multi-rate Signal Processing, and ECE 5150 Real-time Computing.

The work presented here will also add benefit to the department graduate research and development efforts. Current and future students can use this system as their foundation to build more complex and sophisticated systems. It can be used to experiment and learn about various techniques used to implement the project.

7.2 Future Work

The system build provides an excellent tool for Dr. Dong's Laboratory to further investigate the use of OFDM communication, and to experiment with MIMO communication systems. The existing system can also be used to build OFDM Frames structures. One such OFDM frame structure consisting of many OFDM symbols is show in the following figure.

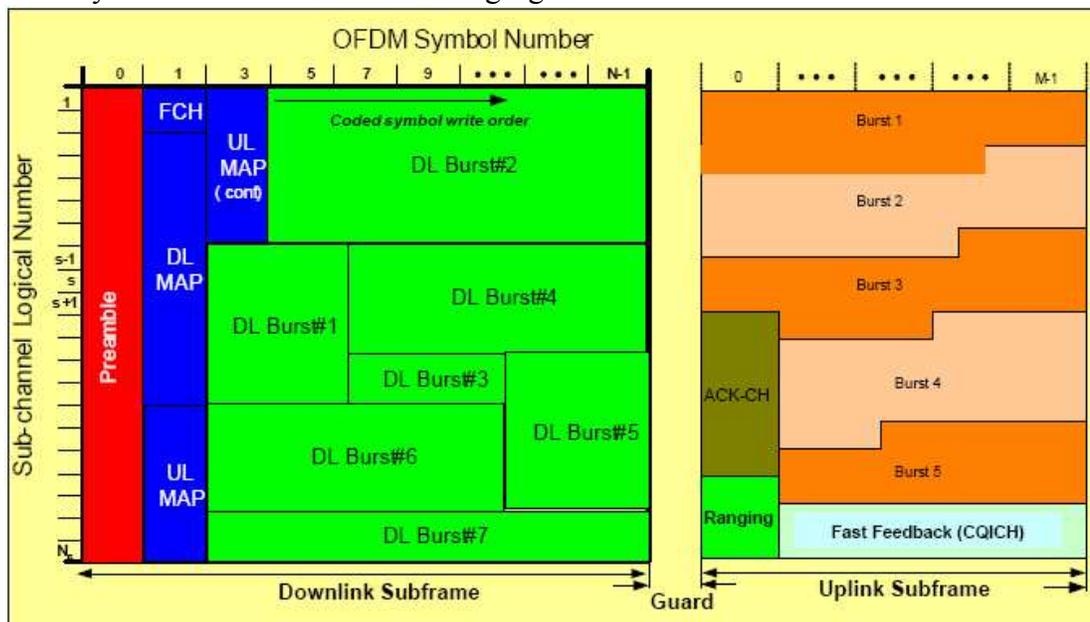


Figure 51: OFDM frame structure for WiMAX [4]

7.3 Conclusion

This work provided me the opportunity to learn and have hand-on experience on the state of the art technologies. The key concepts learned are; OFDM based communication, Software defined radios, and hardware/software co-design. OFDM technique has been playing a vital role in digital communication system and it will be even more significant in the future. The future generation communication standards like WiMAX, DSL are based in it. Similarly, software defined radios will be the essential part of designing any communication system. It is the most feasible choice for rapidly prototyping any new communication standard. What makes it even more significant is the fact that this technique can be used in the end user product also.

APPENDIX

MATLAB Files

Main.m

```
%function 802.16_2004 WirelessMAN -OFDM PHY
% (reference "802.16_2004.pdf" Table 213, Page No.s 428,429, 430)
% 8.3.2 OFDM Symbol Parameters and transmitted signal
%
% 8.3.2.1 Primitive Parameter definitions
% BW: nominal channel BandWidth
% Nused: number of used subcarriers
% n: Sampling Factor
% G: ratio of CP to "useful time"
%
% 8.3.2.2 Derived Parameter definitions
% Nfft: Smallest power of two greater than Nused
% Sampling Frequency: Fs = floor(n.BW/8000)*8000
% subcarrier spacing: deltaF = Fs/Nfft
% useful symbol time: Tb = 1/deltaF
% CP time: Tg = G*Tb
% OFDM Symbol time: Ts = Tb + Tg
% Sampling time: Tb/Nfft
%
% 8.3.2.4 Parameters of Transmitted Signal
% Nfft = 256
% Nused = 200
% n:
%           if(mod(BW,1.75)==0) n=8/7;
%           else if(mod(BW,1.5)==0) n=86/75;
%           else if(mod(BW,1.25)==0) n=144/125;
%           else if(mod(BW,2.75)==0) n=316/275;
%           else if(mod(BW,2.0)==0) n=57/50;
%           else n=8/7;
% G: 1/4, 1/8, 1/16, 1/32
% Number of lower frequency subcarriers: 28
% Number of higher frequency subcarriers: 27
% Frequency Offset Indices of guard subcarriers: -128, -127,...,-101
%                                               +101,+102,...,+127
% Frequency Offset Indices of pilot carriers -88,-63,-38,-
% 13,13,38,63,88
% Note that pilot subcarriers are allocated only if two or more
% subchannels
% are allocated
%
% 8.3.3.6 Preamble structure and modulation
% 1st preamble in the downlink PHY PDU, as well as the initial
% ranging
% preamble, consists of following two consecutive OFDM symbols:
% The 1st OFDM symbol uses only the indices of which are a multiple
% of 4
% The 2nd OFDM symbol utilizes only even subcarriers
% Resulting in the following time domain (Downlink and network
% entry)
% preamble structure:
% |CP |64|64|64|64|CP | 128 | 128|
```

```

% <Tg><----Tb----><Tg><----Tb---->

clc
clear all
close all

global Bins_fft;global Doffset;global Data_setsize;global
SNRdB;global Mpsk;
global Mqam;global p_num;global p_ind;global p_vect;global k;global
timeoff;
global pbinsteps;global chan;global interpfactor
global refphased phasesteps phase2

Bins_fft=256;
Doffset=14;
Data_setsize = Bins_fft/2 - 2*Doffset;
SNRdB=50;
Mpsk=8;
Mqam=4;

debugplot=0;

% Frequency Offset Indices of guard subcarriers
%guard_temp1=[-128;-127;-126;-125;-124;-123;-122;-121;-120;-119;-
118;-117;-116;-115;-114;-113;-112;-111;-110;-109;-108;-107;-106;-
105;-104;-103;-102;-101];
%guard_temp2=[-127;-126;-125;-124;-123;-122;-121;-120;-119;-118;-
117;-116;-115;-114;-113;-112;-111;-110;-109;-108;-107;-106;-105;-
104;-103;-102;-101];
%guard_ind=[guard_temp1;-flipud(guard_temp2)];

% Pilot Tone Generation
p_ind=[9;10;13;18;29;46;75];%p_ind=[13;38;63;88];
p_num=length(p_ind);
pbinsteps=diff(p_ind);
px_vect=sign(cos(pi*(0:p_num-1)'/2+pi/4));
py_vect=sign(sin(pi*(0:p_num-1)'/2+pi/4));
p_vect=[px_vect+j*py_vect];

% Interpolation Factor
interpfactor=4;

figure(1)
figure(2)
figure(3)
figure(4)
figure(5)
set(1,'position',[465 533 450 350])
set(2,'position',[ 10 533 450 350])
set(3,'position',[ 10 100 450 350])
set(4,'position',[465 100 450 350])
set(5,'position',[920 216 450 350])
%pause

```

```

%%
mm=0;
for mm=1:100

    % Form the Pre-OFDM array, with data, and pilot tones
    fd1=FillBins();

    % Interpolate
    fdli=Interpolate(fd1);

    dl=real(iffd(fd1)); %just for plotting
    dli=real(iffd(fdli));

    % Complete Cycle Cyclic Prefix
    dli_twice=[dli;dli];

    % Raw, and Channel Distorted Signals
    dlichan_r=dli_twice;
    dlichan_n=Channel(dli_twice,interpfactor);

    % Received data time delay (offset)
    [d2,timeoff] =DelayInterp(dlichan_r);
    [d2_n,timeoff] =DelayInterp(dlichan_n,timeoff);

    % OFDM Tone Computation
    fd2 =fft(d2);
    fd2_n =fft(d2_n);

    % Compute & Apply Phase Correction Vector using pilot tones
    [fd2correct phasesteps_r phase2_r]=CorrectPhase(fd1, fd2);
    [fd2correct_n phasesteps_n phase2_n]=CorrectPhase(fd1, fd2_n);

    [refphased phasesteps_r phasesteps_n phase2_r phase2_n]
    [phasesteps_r-phasesteps_n phase2_r-phase2_n]

    % For Debugging
    ptest(mm,:)= [refphased phasesteps_r phasesteps_n phase2_r
    phase2_n];

    % Pilot Tones
    ptone_fd1=fd1(p_ind+Doffset+1);
    ptone_fd2correct=fd2correct(p_ind+Doffset+1);
    ptone_fd2correct_n=fd2correct_n(p_ind+Doffset+1);

    ff=(-0.5:1/Bins_fft:.5-1/Bins_fft);
    p_ff=0.5*(0:1/p_num:1-1/p_num);
    %i_ff=(-0.5:1/(interpfactor*Bins_fft):.5-
    1/(interpfactor*Bins_fft));

    % Constellation Data
    cnstl_fd1(mm,:)=fd1;
    cnstl_fd2correct(mm,:)=fd2correct;

```

```

cnstl_fd2correct_n(mm,:)=fd2correct_n;

% Pilot-Tone-Contellation Data
cnstl_ptone_fd1(mm,:)=ptone_fd1;
cnstl_ptone_fd2correct(mm,:)=ptone_fd2correct;
cnstl_ptone_fd2correct_n(mm,:)=ptone_fd2correct_n;

% OFDM Bins Error Computation
error_bins=fd1-4*fd2correct;
error_bins_n=fd1-4*fd2correct_n;

figure(1)
plot(ff,Spec(fd1));
title('Spectrum of OFDM Frames before transmission')
axis([-0.5 0.5 -40 10]);grid;

figure(2)
plot((0:1/(Bins_fft):1-1/(Bins_fft)),real([d1]));
title('Sequence of OFDM Frames before transmission')
axis([-0.1 1.1 -0.5 0.5]);grid;

figure(3)
plot((0:1/(Bins_fft):1-1/(Bins_fft)),real([d2 d2_n]));
title('Sequence of OFDM Frames after transmission')
axis([-0.1 1.1 -0.5 0.5]);grid;

figure(4)
plot(ff,[Spec(4*fd2correct) Spec(4*fd2correct_n)]);
title('Spectrum of OFDM Frames after reception')
axis([-0.5 0.5 -40 10]);grid;

figure(5)
%plot(ff.',dBv(abs([error_bins error_bins_n])))
plot(ff.',Spec([error_bins error_bins_n]))
title('Difference between the input and output spectrum')
axis([-0.5 0.5 -205 10]);grid;

pause(.01)
end
fprintf('End of the loop.\n')
%% Constellation Plots
figure(6)
plot(cnstl_fd1(:,(Doffset+1):(Doffset+Data_setsize)),'or')
grid
axis('square')
title('Constellation before IDFT')

figure(7)
plot(cnstl_fd2correct(:,(Doffset+1):(Doffset+Data_setsize)),'or')
grid
axis('square')
title('Constellation after DFT, No Noise')

```

```

figure(8)
plot(cnst1_fd2correct_n(:, (Doffset+1):(Doffset+Data_setsize)), 'or')
grid
axis('square')
title('Constellation after DFT, Noise added')

%% Pilot Tones Constellation Plots
figure(9)
plot(cnst1_ptone_fd1, 'or')
grid
axis('square')
title('Pilot Tone Constellation before IDFT')

figure(10)
plot(cnst1_ptone_fd2correct, 'or')
grid
axis('square')
title('Pilot Tone Constellation after DFT, No Noise')

figure(11)
plot(cnst1_ptone_fd2correct_n, 'or')
grid
axis('square')
title('Pilot Tone Constellation after DFT, Noise added')
%%
ptesterror=ptest(:,2)-ptest(:,1)-ptest(:,3);
figure(20)
plot(ptesterror)

```

Spec.m

```
function [specfd] =Spec(fd);

% Decimation
%
% Routine call:      specfd =Spec(fd)
%
% Input parameter:  fd
% Output parameter: specfd

global Bins_fft;global Doffset;global Data_setsize;global
SNRdB;global Mpsk;global Mqam;global p_num;global p_vect;global k;

specfd = fftshift(dBv(abs(fd)));

return
```

Interpolate.m

```
function [fdli] =Interpolate(fdl);

% Form the Pre-OFDM array
% Inject Data
% Inject Pilot Tones
% (Note: data should be corrected for the pilot tones)
%
% Routine call:      fdli =Interpolate(fdl)
%
% Input parameter:   fdl
% Output parameter:  fdli

global Bins_fft;global Doffset;global Data_setsize;
global SNRdB;global Mpsk;global Mqam;global p_num;
global p_vect;global interpfactor;

% Interpolation performed in the frequency domain
% The original spectral bins are in the region 1:N/2 and
% (K-1)*Interp+(N:-1:N/2+2)
%
fdli=zeros(interpfactor*Bins_fft,1);
fdli(1:Bins_fft/2)=fdl(1:Bins_fft/2);
fdli(((interpfactor-1)*Bins_fft)+(Bins_fft:-
1:(Bins_fft/2+2)))=conj(fdli(2:Bins_fft/2));

return
```

FillBins.m

```
function [fd1] =FillBins();

% Form the Pre-OFDM array
% Inject Data
% Inject Pilot Tones
% (Note: data should be corrected for the pilot tones)
%
% Routine call:      fd1 =FillBins()
%
% Input parameter:  none
% Output parameter: fd1

global Bins_fft;global Doffset;global Data_setsize;global
SNRdB;global Mpsk;global Mqam;global p_num;global p_ind;global
p_vect;global k;

fd1=zeros(Bins_fft,1);
dx1ref=floor(sqrt(Mqam)*rand(Data_setsize,1))-(sqrt(Mqam)/2-0.5);
dylref=floor(sqrt(Mqam)*rand(Data_setsize,1))-(sqrt(Mqam)/2-0.5);

% Normalize by the symbols by the constellation average power
dx1=dx1ref/AvgMagSum;
dyl=dylref/AvgMagSum;

%dex=exp(i2pi*(floor(Mpsk*rand(Data_setsize,1))/Mpsk));
%dex=exp(i2pi*rand(1,Data_setsize));
%
% Form the Pre-OFDM array
% inject data
% inject pilot tones
% (Note: data should be corrected for the pilot tones)
fd1((1+Doffset):(Data_setsize+Doffset))=dx1+j*dyl;
fd1(p_ind+Doffset+1)=p_vect;

% fill the conjugate positions priot to ifft
fd1(Bins_fft:-1:(Bins_fft/2+2))=conj(fd1(2:Bins_fft/2));

return
```

DelayInterp.m

```
function [d3,timeoff] =DelayInterp(d1,timeoff);

% Received data time delay (offset)
%
% Routine call:      d2 =Delay(d1)
%
% Input parameter:  d1
% Output parameter: d3

global interpfactor;
global Bins_fft;

if nargin<2
    timeoff=interpfactor*32+floor(interpfactor*Bins_fft/2*rand(1));
end

d2 = d1(timeoff:timeoff+interpfactor*Bins_fft-1);
d3 = d2(1:interpfactor:end);

return
```

dBv.m

```
function [snr_dB] =dBv(vratio);

% Routine to convert a matrix of scalar voltage ratios to
decibels.
% The output power ratings are limited to ±200 dB, and are
computed as
% 20*log10 of the absolute value of the input voltage ratios.
%
% Routine call:      snr_dB=dBv(vratio)
%
% Input parameter:   vratio   Matrix of power ratios
% Output parameter:  snr_dB   Decibel representation of `snr`

% Original dB routine written 10/30/91, by Brian Agee

snr_dB=20*log10(min(10^(10),max(10^(-10),abs(vratio))));

return
```

CorrectPhase.m

```
function [fd_correct phasesteps phase2] =CorrectPhase(fd1, fd2);

% Compute Phase Correction Vector
%
% Routine call:      fd_correct=CorrectPhase(fd_shifted)
%
% Input parameter:  fd1
% Input parameter:  fd2 (with phase shifted)
% Output parameter: fd2 (phase corrected using pilot tones)

global Bins_fft;global Doffset;global Data_setsize;global
SNRdB;global Mpsk;
global Mqam;global p_num;global p_ind;global p_vect;global timeoff;
global pbinsteps;
global refphased phasesteps phase2
global interpfactor

% pilot tones
ptone_fd1=fd1(p_ind+Doffset+1);
ptone_fd2=fd2(p_ind+Doffset+1);

phaseptone=ptone_fd2./ptone_fd1;
refphased=360*(timeoff-1)/(interpfactor*Bins_fft);
ptonepdiff=angle(phaseptone)*180/pi;
pdptonepdiff=diff(ptonepdiff);
pdptonepdiff2=pdptonepdiff + 360*(pdptonepdiff<0);

phaseshift=(pdptonepdiff2./pbinsteps);
for mmm=2:(length(p_ind)-1)
    padd=round(((phaseshift(mmm)-
phaseshift(mmm))*pbinsteps(mmm))/360);
    pdptonepdiff2(mmm)=pdptonepdiff2(mmm)+360*padd;
    phaseshift(mmm)=pdptonepdiff2(mmm)/pbinsteps(mmm);
end

phasesteps=mean(phaseshift);
CorrectPVect=exp(-sqrt(-1)*(p_ind+Doffset)*(phasesteps*pi/180));
phase2=mean(angle((ptone_fd2.*CorrectPVect)./ptone_fd1))
%CorrectPVect=CorrectPVect*exp(-sqrt(-1)*phase2);
angle((ptone_fd2.*CorrectPVect)./ptone_fd1);

CorrectVect(1:Bins_fft/2,1)=exp(-sqrt(-1)*(0:Bins_fft/2-
1)*(phasesteps*pi/180));
%CorrectVect(1:Bins_fft/2,1)=exp(-sqrt(-1)*(0:Bins_fft/2-
1)*(refphased*pi/180));
%CorrectVect=CorrectVect*exp(-sqrt(-1)*phase2);
CorrectVect(Bins_fft:-
1:(Bins_fft/2+2),1)=conj(CorrectVect(2:Bins_fft/2));
% Apply the phase correction vector
fd_correct =fd2 .* CorrectVect;

return
```

Channel.m

```
function [dlchan_n] =Channel(dl_twice,i);

% Channel Distortion
%
% Routine call:      dlchan_n =Channel(dl_twice)
%
% Input parameter:  dl_twice (no channel)
% Output parameter: dlchan_n (data distorted by channel)

global Bins_fft;global Doffset;global Data_setsize;global chan;
global SNRdB;global Mpsk;global Mqam;global p_num;global p_vect;
k=i;

normnoise=randn(2*k*Bins_fft,1);
normnoise=sqrt(2*(Data_setsize/2+2)/(Bins_fft/4))*normnoise/sqrt(sum(normnoise.^2));
normnoise=normnoise/(10^(SNRdB/20));

chan=[1 .2 0 0.02 0 j*0.1];
chan=chan/sum(chan);
dlchan=conv([zeros(2*Bins_fft,1); dl_twice]',chan);
%dlchan_n=dlchan+0.0435*(randn(1,4*Bins_fft+5)+j*randn(1,4*Bins_fft+5));

dlchan_n=dl_twice+normnoise;

return
```

AvgMagSum.m

```
function [avgmagsumnum] =AvgMagSum();

%
%
% Routine call:      avgmagsum =AvgMagSum()
%
% Input parameter:  none
% Output parameter: avgmagsum

global Bins_fft;global Doffset;global Data_setsize;global
SNRdB;global Mpsk;global Mqam;global p_num;global p_vect;global k;

dxv=flipud((1:sqrt(Mqam))'-(sqrt(Mqam)/2+0.5));
dyv=flipud((1:sqrt(Mqam))'-(sqrt(Mqam)/2+0.5));
magsum=0;
for ii=1:sqrt(Mqam)
    for iii=1:sqrt(Mqam)
        magsum=magsum+(dxv(ii).^2+dyv(iii).^2);
    end
end
avgmagsumnum=sqrt(magsum/Mqam);

return
```

BIBLIOGRAPHY

- [1] Harris, F., Orthogonal Frequency Division Multiplexing OFDM, Vehicular Technology Conference, 2004.
- [2] Vaidyanathan, P.P., Filter Banks in Digital Communications, IEEE, 2001.
- [3] Vaidyanathan, P.P., and Vrcelj, B., Transmultiplexers as Precoders in modern digital communication: a tutorial review, 1999.
- [4] WiMAX Forum, Mobile WiMAX – Part 1: A Technical Overview and Performance Evaluation, 2006.
- [5] WiMAX Forum, Mobile WiMAX – Part 1: A Comparative Analysis, 2006.
- [6] WAVE Report, OFDM Tutorial available at www.wave-report.com/tutorials/OFDM.htm, 2001.
- [7] National Instruments, Getting Started with LabVIEW available at www.ni.com, 2005.
- [8] National Instruments, LabVIEW Fundamentals available at www.ni.com, 2005.
- [9] National Instruments, LabVIEW: FPGA Module User Manual available at www.ni.com, 2003.
- [10] National Instruments, NI Communications Systems Design Pioneer Program available at www.ni.com, 2006.
- [11] National Instruments, NI PCI-5640R Specifications available at www.ni.com, 2006.
- [12] IEEE Computer Society, and IEEE Microwave Theory and Techniques Society, 802.16 IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems, 2004.
- [13] IEEE Computer Society, and IEEE Microwave Theory and Techniques Society, 802.16.2 IEEE Recommended Practice for Local and Metropolitan Area Networks Coexistence of Fixed Broadband Wireless Access Systems, 2004.
- [14] IEEE Computer Society, and IEEE Microwave Theory and Techniques Society, 802.16 IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems Amendment 1 : Management Information Base , 2005.

- [15] IEEE Computer Society, and IEEE Microwave Theory and Techniques Society, 802.16 IEEE Standard for Local and Metropolitan Area Networks Part 16 : Air Interface for Fixed Broadband Wireless Access Systems Amendment 2 : Physical and Medium Access Control Layers for Combined Fixed and Mobile operation in Licensed Bands, Corrigendum 1 , 2005.
- [16] National Instruments Developers Zone, “Gaining Spectral Efficiency with OFDM”, <http://zone.ni.com/devzone/cda/tut/p/id/3370> .
- [17] Wong, I., Han, K., and Doyle, A., IEEE 802.16a Simulator available at <http://users.ece.utexas.edu/~iwong/IEEE80216aSim.htm>, 2004.
- [18] Hosking, R. H., Putting FPGAs to Work Software Radio Systems, Part 1 available at www.rfdesignline.com, 2007.
- [19] Texas Instruments, Telecom Overview, available at www.ti.com, 2007.