

**A Low-complexity Linear and Iterative Receiver
Architecture for Multi-antenna Communication Systems**

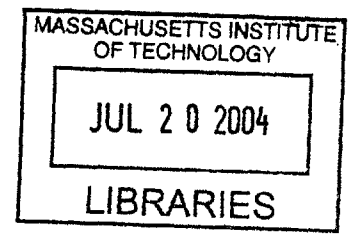
by

David Louis Milliner

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 20, 2004
[June 2004]

Copyright 2004 David L. Milliner. All rights reserved.



The author hereby grants to M.I.T. permission to reproduce and
distribute publicly paper and electronic copies of this thesis
and to grant others the right to do so.

Author David L. Milliner
Department of Electrical Engineering and Computer Science
May 20, 2004

Certified by _____
Dr. Manish Goel, Ph.D.
VI-A Company Thesis Supervisor

Certified by _____
Professor Moe Win
Charles Stark Draper Assistant Professor
M.I.T. Thesis Supervisor

Accepted by _____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

**A Low-complexity Linear and Iterative Receiver
Architecture for Multi-antenna Communication Systems**

by
David Louis Milliner

Submitted to the
Department of Electrical Engineering and Computer Science

May 20, 2004

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

Multi-antenna systems have been shown to significantly improve channel capacity in wireless environments. The focus of this thesis is on the design of low-complexity multi-antenna receiver architectures for communication networks and their demonstration in a real-time wireless environment. Our practical realization of an orthogonal frequency-division multi-antenna receiver is capable of several forms of linear and iterative detection. Our implementation is based on a division-free reformulation of standard minimum mean-squared-error detection algorithms and uses complex dot-products as the basic building blocks of a folded-pipelined architecture. This folded-pipelined architecture provides significant area savings over non-folded approaches. The demonstration of our receiver architecture is carried out on a rapid-prototyping FPGA communication system. This prototype is used to validate our design and complement theoretical and simulated results with real-time laboratory measurements in a typical office environment.

Thesis Supervisor: Moe Win

Title: Charles Stark Draper Assistant Professor

Acknowledgments

There are many individuals at Texas Instruments and MIT who have contributed to my work over the past year. Below are small but representative samples of the people who have helped make my thesis possible.

First of all, I am grateful to Manish Goel for his discerning insight and industrious guidance. Manish has been an exceptional mentor, manager, and friend from the day I started out on this endeavor. His daily advice and unwavering support of my work will never be forgotten. Additionally, Manish's initial development in areas pertinent to this research served as a stimulus and source of reference for much of the work presented in this thesis. I would like to thank Michael Polley for giving me the opportunity to join the Communications Systems Laboratory and the Wireless Broadband Architectures team. Mike's contributions as a technical advisor and as a sounding board for my ideas have been indispensable. I am indebted to the members of my lab group Muhammad Ikram, Mike DiRenzo, David Magee, Srinath Hosur, and Hank Eilts. I am grateful to Mike and Dave for their guidance using the MIMO Testbed which facilitated and advanced the work presented in this thesis. I am also grateful for their constant optimism and sincere interest in my academic and professional affairs. I would like to thank Muhammad and Sri for their initial development of the simulation platform used in this work and their assistance in answering my questions as I extended this platform to meet the needs of my research. Additionally, I would like to thank Don Shaver and Bob Hewes for their full support of my project and the wireless research occurring at Texas Instruments. Finally, I would like to thank Randy Cole for serving as my first manager at TI and integrating me

into the DSP R&D Center. Randy provided me with my earliest opportunity to succeed at Texas Instruments and for this I will always be grateful.

I wish to thank my thesis advisor, Prof. Moe Win, for his constant support and honest advice along the way. I am most thankful for the high standards he set for me and his charge to always strive for improvement. Moe's suggestions and technical contributions have been invaluable to the final production of this work and to the overall success of my thesis. Additionally, I would like to thank Watcharapan Suwansantisuk for his comments. I am also grateful to my academic advisors Prof. Jesus del Alamo and Soosan Behesti for their guidance over the past five years. I would like to thank Lydia Weremensi, Kathleen Sullivan, and Prof. Markus Zahn, for their administration of the VI-A program. I would also like to thank Anne Hunter and Vera Sayzew at MIT and Lucrecia Ladwig at Texas Instruments for their patience and rapid solutions to the numerous administrative issues I have encountered in the completion of my thesis.

I would like to thank my parents for their love and support. I am quite confident this thesis would not have been possible without them. Their constant nurturing support has been the largest contributing factor to the successful completion of my undergraduate and graduate studies to date. Finally, I wish to thank my grandparents. Their support of my academic endeavors serves a perpetual source of energy over my efforts.

Table of Contents

ABSTRACT	2
ACKNOWLEDGMENTS	3
TABLE OF CONTENTS	5
TABLES AND FIGURES	7
1 INTRODUCTION	8
2 BACKGROUND	12
2.1 Overview of Core Technologies	12
2.1.1 Orthogonal Frequency Division Multiplexing	12
2.1.2 Multiple Input Multiple Output Systems	14
2.1.3 Wireless Local Area Networks	15
2.1.4 Fading and Channel Models	16
2.2 Multi-Antenna System Model	16
2.3 Design Methodology	19
2.3.1 Design Tools	19
2.3.1.1 MATLAB (The MathWorks)	19
2.3.1.2 ModelSim (Mentor Graphics)	20
2.3.1.3 PrecisionC (Mentor Graphics)	20
2.3.1.4 Synplify Pro (Synopsis)	21
2.3.1.5 Virtex-II FPGA (Xilinx)	21
2.3.2 Methodology Overview	22
2.3.3 Floating-Point Design	23
2.3.4 Fixed-Point Design	23
2.3.5 RTL Design	24
2.3.6. Real-Time Validation	25
3 LOW COMPLEXITY DETECTION ALGORITHMS	26
3.1 Detection Algorithms Overview	26
3.1.1 Zero-Forcing Linear Detection	26
3.1.2 Linear Minimum Mean-Squared Error	27
3.1.3 Iterative Detectors	28
3.1.5 Maximum Likelihood	28
3.2 Linear MMSE Detection Equations	29
3.3 Computational Complexity	30
3.4 Reformulation of MIMO Detection Equations	32

	6
3.5 Iterative MMSE Algorithm Support	33
4 FINITE PRECISION IMPLEMENTATION	34
4.1 Simulation Setup	35
4.2 Finite Precision Results	37
4.3 ASIC Gate Counts	41
5 MULTI-ANTENNA TESTBED AND REAL-TIME RESULTS	46
5.1 Multi-antenna Prototype	46
5.2 Time Domain Processing/FFT Module	48
5.3 Channel Estimation	49
5.4 MIMO FPGA/Core	49
5.5 FEC Module	51
5.6 Real-time Wireless Measurements	51
6 CONCLUSIONS AND FUTURE WORK	56
6.1 Summary	56
6.2 Future Work	58
6.3 Concluding Remarks	59

List of Tables and Figures

FIG. 2-1. MIMO SYSTEM MODEL.....	17
FIG. 2-2. DESIGN METHODOLOGY FLOW CHART.....	22
TABLE 1. POSSIBLE SYSTEM DATA RATES.....	36
FIG. 4-1. FIXED-POINT BIT-WIDTH STUDY FOR BER FOR $N_T = 2$ AND $N_R = 3$	38
FIG. 4-2. BER FOR LMMSE DETECTOR (MULTIPLE CONFIGURATIONS).....	40
FIG. 4-3. FIXED-POINT IMPLEMENTATION LOSS OF LMMSE AND IMMSE WHERE $N_T = 2$ AND $N_R = 3$	41
FIG. 4-4. ESTIMATED GATE COUNTS FOR 2X3 20MHZ ARCHITECTURE.....	43
FIG. 4-5. ASIC GATE COUNTS PRODUCED FROM SYNTHESIS OF RTL IMPLEMENTATIONS.....	44
FIG. 5-1. 3X3 MIMO TRANSCEIVER SYSTEM	47
FIG. 5-2. MIMO TRANSCEIVER PROTOTYPE.....	48
FIG. 5-3. INPUT/OUTPUT CONNECTIONS OF MIMO FPGA	50
FIG. 5-4. FORWARD ERROR CORRECTION FPGA BLOCK DIAGRAM	51
FIG. 5-6. 2X2 VS. 2X3 COMPARISON FOR TRIALS 1 AND 2 AT 72MB/S.	55

CHAPTER 1

INTRODUCTION

MULTIPLE antenna wireless communication systems have been studied for well over a decade because of their ability to improve spectral efficiency, increase data rates, and strengthen the robustness of communication systems in fading environments, particularly those characterized by richly scattered multi-path. In recent years the theoretical understanding and standardization of multi-antenna techniques has received considerable attention [1], [2]. As the industry demand for multiple input multiple output (MIMO) wireless networks continues to grow, their design and implementation become increasingly important. While multi-antenna systems have been an active research area for a number of years now, there has been less emphasis on practical realization of these systems. In particular, the design and demonstration of Systems-on-a-Chip (SoC) multi-antenna wireless networks is only starting to receive attention. Moreover, the computation performed by these systems has remained almost entirely off-line. This thesis explores the practical realization of linear and iterative receiver architectures for multi-antenna communication systems which are designed for real-time operation over a wireless channel.

One potential application of MIMO technology is in next generation wireless local area networks (WLAN). WLAN is an excellent application for multi-antenna

systems because typically WLAN deployments are found in indoor environments which are characterized by richly scattered multi-path. Current WLAN standards such as IEEE 802.11a [3] and IEEE 802.11g [4] are based on orthogonal frequency division multiplexing (OFDM). It is likely that future 802.11 standards such as 802.11n will incorporate a combination of MIMO and OFDM technologies.

While the majority of the research thrust in this thesis is on WLAN systems, the techniques and architectures developed in this work are applicable more generally. Specifically, while MIMO communication systems perform better in rich scattering environments typically found in indoor wireless networks other possible future applications of MIMO-OFDM might include emerging Metropolitan Area Network (MAN) standards such as IEEE 802.20 and cellular technologies such as the upcoming fourth generation (4G) mobile wireless systems.

Detection in multi-antenna networks is an area that has been heavily studied for many years now. The primary multi-antenna detectors we will examine in this thesis have cubic complexity based on the number of antennas in the system. Specifically, we will focus on multi-antenna detection using the linear and iterative detectors which satisfy the zero forcing and minimum mean-squared-error criteria. As a consequence of the cubic nature of these algorithms it is essential that there exist an efficient architectural implementation for MIMO detection. This statement will become increasingly relevant as multi-antenna systems move toward higher dimension implementations. For this reason we develop a division-free modular approach to standard linear detection for multi-antenna systems which allows for a low-complexity architectural implementation.

A division-free implementation is important because the division operation is widely accepted as an expensive processor function in terms of area and latency.

This division-free reformulation of standard multi-antenna detection lends itself to a modular fixed-point structure that can be generalized to MIMO systems with an arbitrary number of transmitters and receivers. We take this generalized fixed-point structure and develop a hardware implementation which is then validated on a multi-antenna OFDM rapid-prototyping communication system operating in real-time over a wireless channel.

The remainder of this thesis is organized as follows. In Chapter 2, we review some of the key concepts from multi-antenna communication theory and survey prior work and implementations pertaining to MIMO-OFDM systems. We then present the system descriptions and assumptions used throughout this thesis. We conclude the chapter with a description of the design methodology used in developing the research found in this thesis. We begin Chapter 3 by reviewing standard detection algorithms and then develop low-complexity detection algorithms based on a division-free modular reformulation of these standard detectors. In Chapter 4 we present a finite-precision implementation for the algorithms developed in Chapter 3 and floating-point and finite-precision simulation results to validate our implementation. Additionally, we present a novel “folded-pipelined” architecture implemented in hardware which is based on our finite-precision implementation. Chapter 5 serves as validation of our design by describing our laboratory prototype and presenting results from actual real-time wireless measurements performed in a typical office environment and their comparison to

simulations. We use the results obtained from our laboratory prototype as validation of our low-complexity multi-antenna receiver design.

CHAPTER 2

BACKGROUND

In this chapter we will review some of the key concepts from multi-antenna communication theory and survey prior work and implementations pertaining to MIMO-OFDM systems. We then present the system descriptions and assumptions we will use throughout this thesis and conclude the chapter with a description of the design methodology used.

2.1 Overview of Core Technologies

We now present an overview of the core technologies utilized in this thesis. Specifically we summarize orthogonal frequency division multiplexing, multiple-input multiple-output systems, wireless local area networks, and channel models.

2.1.1 Orthogonal Frequency Division Multiplexing

Orthogonal frequency division multiplexing (OFDM) has been shown to be an effective technique in combating multi-path fading environments encountered by wireless communication networks and improving data rates in these environments¹. OFDM is over forty years old and has been successfully implemented in many high profile applications such as asymmetric digital subscriber lines (ADSL) services, high frequency (HF) radios, digital audio broadcasting (DAB), digital terrestrial TV broadcasting (DVB in Europe

¹ A description of path loss is provided in Section 2.1.4.

and ISDB in Japan), IEEE 802.16a [5] and of course IEEE WLAN. OFDM is currently a very popular choice for future wireless applications such as IEEE 802.11n, Ultra Wideband (UWB) standardization efforts, cellular and personal communication system (PCS) data, IEEE 802.20 and 4G networks [6, 7].

The main strengths of OFDM are its spectral efficiency, robustness against narrowband interference, and robustness in multi-path environments. The most popular form of OFDM is multicarrier OFDM which works by dividing a transmission bandwidth into narrow sub-carriers which are transmitted in parallel. High spectral efficiency can be maintained by performing inverse fast Fourier transform (IFFT) and fast Fourier transform (FFT) operations to ensure that sub-carriers do not interfere with one another. Additionally, robustness against narrowband interference is maintained by the orthogonal nature of transmission whereby if a narrowband interferer is present in the narrowband allocation of a sub-carrier or small number of sub-carriers then that sub-carrier can be ignored and compensated for through the use of forward error correction² (FEC). Finally, the use of a cyclic prefix to preserve the orthogonal relationship between sub-carriers provides receivers with the ability to capture multi-path energy more efficiently [7].

We choose to use OFDM in our system implementation because it combines the desirable networking properties of maintaining orthogonal transmission within a wireless sectorization while obtaining universal frequency reuse across this sectorization through interference averaging [8].

² We mention FEC techniques employed by our design in Chapter 4.

2.1.2 Multiple Input Multiple Output Systems

MIMO systems have emerged in recent years as one of the most exciting technologies for academics and industry professionals who are engaged in communication systems and wireless networks. In particular, multi-antenna systems have received a large amount of attention as a consequence of their ability to improve system performance across a wide array of overlapping metrics including system throughput, reliability of transmission, degrees of freedom, and reach. While there are generally tradeoffs in MIMO networks between these system improvements the general performance of a MIMO system in comparison to that of a single input single output (SISO) transmission scheme is superior. As the cost of semiconductor production continues to decrease and the desire of users for improved networking capabilities continues to increase the prevalence of wireless networks with MIMO capabilities will see an explosion in the next decade and beyond.

MIMO systems differ from SISO systems in that channel inputs and outputs for SISO systems are scalar valued whereas they are vector valued in the MIMO case. This adds to system complexity as not only must MIMO systems deal with traditional wireless hurdles such as noise and inter-symbol interference (ISI), but there is now the additional burden of handling interference between inputs. Additional difficulties introduced in MIMO communication systems are high algorithmic complexity, cost of multiple radio chains, antenna diversity, system robustness, and time to market. Indeed, there are many issues to be addressed in MIMO systems that are not present in SISO configurations. Primarily these issues are resolved using smart detection and transmission schemes. In particular some of the more appealing aspects of MIMO systems allowing them to handle

difficulties involved with multi-antenna communication are diversity gain, spatial multiplexing, and interference cancellation.

In this thesis we are interested primarily in the design of receivers in multi-antenna networks. As such we will focus our attention on the presentation of multi-antenna detection strategies and their formulation. Chapter 3 presents a survey of the most popular and relevant multi-antenna detection strategies and a derivation for reducing their implementation complexity. Additionally, we present our multi-antenna system model in Section 2.2 of this chapter. For a more thorough explanation of MIMO-OFDM systems see [9]

2.1.3 Wireless Local Area Networks

While there are many potential applications of MIMO-OFDM technologies, the original target application of the work presented in this thesis is wireless local area networks. IEEE 802.11n is promising to standardize MIMO-OFDM into WLAN protocol, a decision which could result in the first mass market consumer applications incorporating a combination of these two key wireless technologies. For this reason we briefly discuss this application.

WLAN deployments provide high-speed wireless connections to the internet whether they are found in an enterprise or home environment. WLAN technology allows users to roam throughout their home or office facilities while maintaining networking capabilities without a physical tethering to their network. Additionally, WLAN deployments can be found in locations where people require networking capabilities such as hotels, airports, coffee shops and restaurants. The cost of deploying wireless networks is significantly less than wireline deployments. This is another reason for the popularity

of WLAN. As the base number of WLAN users continues to grow so to will the need for improved network throughput and reliability. As a first mass market consumer application of MIMO-OFDM our prototype presented in Chapter 5 is targeted for IEEE 802.11n.

2.1.4 Fading and Channel Models

A plethora of channel models exist in the literature on wireless communications. The most popular of these models is the Rayleigh fading channel model. This is primarily a function of the widespread acceptance of this model to describe fading effects for a broad array of wireless channels as well as the fact that it is quite tractable mathematically. Ricean channel models were also investigated in this work although simulation results performed using this model were not significantly different from the Rayleigh fading channel model results. For a thorough description of these channel models the author recommends [10, 11]. Additionally, one modeling approach gaining in popularity is cluster modeling [12].

2.2 Multi-Antenna System Model

The multi-antenna system investigated in this paper consists of N_T transmitting and N_R receiving antennas. We assume a single tap impulse response as a sufficient channel representation. This is a typical model for an orthogonal frequency division modulation system with no inter-carrier interference. We therefore assume proper timing and carrier recovery at the receiver. The derivations presented in the paper are valid for each carrier in an OFDM system.

Throughout this paper, vectors and matrices are indicated in bold. The superscripts T and H denote the transpose and conjugate transpose respectively of a matrix or a vector. The determinant of a matrix A is denoted by $\det(A)$. The adjoint of a matrix A is denoted by $\text{adj}(A)$. The trace of a matrix A is denoted by $\text{tr}[A]$.

Fig. 2-1. depicts our multi-antenna system consisting of N_T transmitting and N_R receiving antennas. The configuration of such a system is denoted by $N_T \times N_R$. For example a MIMO system consisting of two transmit antennas and three receiver antennas has a 2x3 configuration.

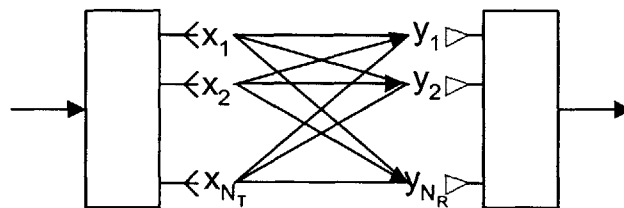


Fig. 2-1. MIMO System Model

The N_R -dimensional signal \mathbf{y} at the output of the receiving antennas in flat fading can be written as [2,13-16]

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w}. \quad (2-1)$$

Here \mathbf{x} is the N_T -dimensional vector with complex components that represent the transmitting signals, and \mathbf{w} is an N_R -dimensional vector with zero-mean Gaussian entries having equal variance that represent additive white noise. The discrete function $\mathbf{y}[n]$ is the Fast Fourier Transform of the vector \mathbf{y} seen by the receivers. The channel matrix \mathbf{H} , defined by

$$\mathbf{H} \equiv \begin{bmatrix} | & | & & | \\ \mathbf{h}_1 & \mathbf{h}_2 & \cdots & \mathbf{h}_{N_T} \\ | & | & & | \end{bmatrix}, \quad (2-2)$$

is an $N_R \times N_T$ random matrix with complex elements. The element $\{h_{ij}\}$ in the channel matrix denotes the gain of the radio channel between the j th transmitting antenna and the i th receiving antenna. We denote the j th column of \mathbf{H} by \mathbf{h}_j , which is the N_R -dimensional propagation vector corresponding to the j th transmitted signal [17].

In the case of a MIMO OFDM system we let n be the index of the OFDM symbol count, and k be the sub-carrier number within that OFDM symbol. The symbol estimate for the i th bit-stream is denoted $\hat{s}_i[n, k]$ and its corresponding reliability metric is denoted $\text{scale}_i[n, k]$. Because processing each sub-carrier is independent, we will suppress the index k in subsequent equations.

We define the covariance of the noise vector \mathbf{w} to be

$$\mathbf{R}_{\mathbf{w}\mathbf{w}} = E\{\mathbf{w}\mathbf{w}^H\} = \begin{bmatrix} \sigma_{11}^2 & 0 & 0 & 0 \\ 0 & \sigma_{22}^2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \sigma_{N_R N_R}^2 \end{bmatrix}. \quad (2-3)$$

The total power P_T of the transmitted vector \mathbf{x} is held constant regardless of the number of transmitting antennas. The total transmitted power is simply the trace of the covariance matrix $\mathbf{R}_{\mathbf{x}\mathbf{x}}$:

$$P_T = \text{tr}[\mathbf{R}_{\mathbf{x}\mathbf{x}}] = \sum_{m=1}^{N_T} \sigma_{x_m}^2. \quad (2-4)$$

In this thesis, we consider the scenario in which the transmitter has no channel state information (CSI), and thus all antennas transmit with the same power [17]

$$\sigma_{x_1}^2 = \sigma_{x_2}^2 = \dots = \sigma_{x_{N_T}}^2 = \sigma_x^2. \quad (2-5)$$

2.3 Design Methodology

A solid design flow is essential when undertaking the implementation of any digital design no matter how trivial. Much research has been focused solely on design flow for communication systems [18, 19]. In this section we present software and hardware tools used to design, simulate, and validate our hardware implementation as well as the design methodology used to develop and verify all of the work presented in this thesis. We begin by presenting an overview of the design tools followed by an overview of the overall design methodology and then discuss each of the major steps in this methodology.

2.3.1 Design Tools

This subsection provides a description of the tools used in the design methodology outlined above. In particular we discuss MATLAB, ModelSim, PrecisionC, Synplify Pro, and the Virtex-II FPGA.

2.3.1.1 MATLAB (The MathWorks)

MATLAB is a programming language for technical computing. MATLAB enables data exploration, algorithmic creating and visualization and graphing capabilities. For the purposes of this thesis MATLAB was used to develop initial floating-point

algorithmic formulations, structured floating-point development, fixed-point design, and as a reference for hardware language implementations.

2.3.1.2 ModelSim (Mentor Graphics)

ModelSim is the industry standard simulation program for digital design. Developed by Mentor Graphics, ModelSim allows for hardware description languages (HDLs) such as VHDL and Verilog HDL to be compiled and simulated through the use of a testbench. A testbench is simply another HDL file which sets and changes inputs appropriately so that the system responses can be verified using the design simulator. This simulation can then be viewed in a waveform window to verify that the code for a particular digital design performs as expected.

Hardware description languages are a convenient device independent representation of digital logic. In this research we used Verilog as our HDL of choice because it is more of a behavioral language than VHDL and is less strongly typed making it easier to use. Verilog designs consist of interconnected modules. We describe the modules used in our system and their validation in Chapter 4. For a detailed description of programming in the Verilog language please see the following references [20-21].

2.3.1.3 PrecisionC (Mentor Graphics)

The PrecisionC software tool from Mentor Graphics is an interesting design tool which was used in a number of different capacities for this research. One primary use of this tool is the ability to take fixed-point C code and generate functional register transfer

language (RTL)³ code without the need for a “hand-coded” HDL implementation. Additionally, the tool can be used for design exploration. Through the use of a graphical user interface (GUI) architectural design decisions, such as whether or not to unroll high-level coding loops or pipeline computations, can be made. It is through this design exploration that we first validated our approach for a folded-pipelined architecture.

Another capability of the PrecisionC tool which we exploited in our research is the ability to estimate gate counts and latency of digital designs. We use these results to compare our hand-coded implementation and verify the efficiency and compactness of our design. The results of this comparison and a finite-precision study conducted using this tool are presented in Chapter 4.

2.3.1.4 Synplify Pro (Synopsys)

The Synplify Pro tool from Synopsys was the program used to synthesize logic for the field programmable gate array (FPGA) using our real-time system prototype. The FPGA we used was a Xilinx Virtex-II chip which is described subsequently in the next sub-section. Once a design is synthesized it is necessary to map this logic onto the FPGA. This process involves placing logic and routing wires among logic units on the FPGA. Additionally, timing analysis is helpful in predicting the maximum clocking frequency possible for a given design. This information is exploited in Chapter 4.

2.3.1.5 Virtex-II FPGA (Xilinx)

A FPGA is like an electronic breadboard that is wired together by an automated synthesis tool [21]. The FPGAs used in this work are Xilinx Virtex-II FPGA devices

³ Throughout this thesis we will use RTL and HDL interchangeably.

with part number XC2V6000FF1152-5. This particular FPGA consists of six million transistors allowing for significantly complex digital designs.

2.3.2 Methodology Overview

Communication system algorithms as we will see in Chapter 3 are typically specified as floating-point (infinite precision) operations. The HDL implementations of these algorithms however rely on fixed-point (finite precision) approximations of the infinite precision representation to reduce the hardware cost. Fig. 2-2 depicts the high level flow of our top-down design methodology. In general we start with a conception of what algorithm we would like to implement. We take this algorithm and create a floating-point representation which we then implement. Based on this implementation we move to a fixed-point design also known as a finite-precision implementation. We can use the structure of this fixed-point implementation to derive and implement a hardware implementation. We then test and validate this implementation on a FPGA. A detailed description of each of these steps is provided in the following subsections.

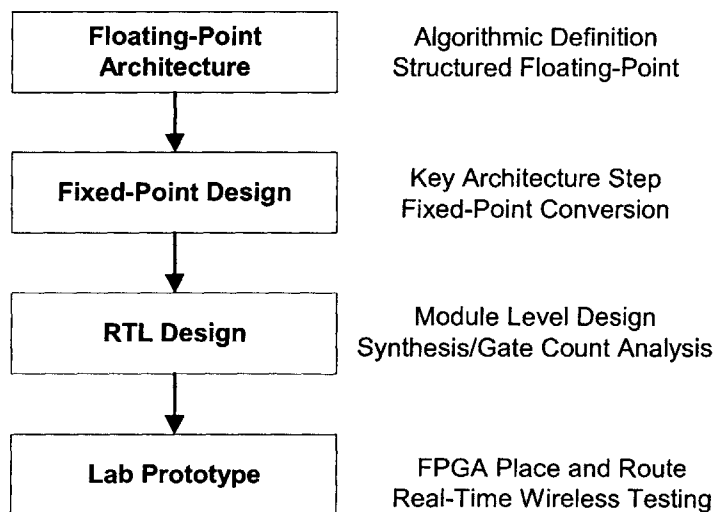


Fig. 2-2. Design Methodology Flow Chart

We should note that it is absolutely vital to test and validate a design at every step during the process outlined in Fig. 2-2. Indeed, a design that is not tested is no design at all. Moreover, we can amplify this sentiment by stating that if a design is not verified after each iteration then it will almost certainly not work when moving from one iteration to the next. While there are no arrows going in the upwards direction in Fig. 2-2 we note that at each step in the design process and after each iteration we validated results against prior steps in the design flow. A detailed description of these results is provided in Chapter 4.

2.3.3 Floating-Point Design

Given that communication system algorithms are typically specified using infinite precision operations it is necessary to begin at the top level floating-point design and work down towards a structure that can be implemented using finite precisions. This strategy typically involves structuring the floating-point algorithm in such a way that the algorithm only utilizes hardware functions that are available to the final integrated circuit. These operations include like additions, multiplications, and shifts. Once this structure is established the process of finite precision conversion can begin. This design step is described next.

2.3.4 Fixed-Point Design

Perhaps the most important and generally the most time consuming step in a top-down digital design flow is fixed-point conversion. This conversion requires the determination of fixed-point data types at each signal node. Specifically, decisions must

be made relating to the wordlength, truncation mode and overflow mode [22, 23] used for every calculation.

Two different methods of fixed-point design were used in this thesis. One approach is to convert structured floating-point MATLAB code to fixed-point MATLAB code and the other involves converting floating-point C code to a fixed-point C implementation which was written in the fixed-point C language “PrecisionC”. The two methods can be made to be functionally equivalent, a step which was done for this work. In both fixed-point conversion methods a pseudo-floating point representation for the fixed-point design was used. This representation maintains bit precisions for both the mantissa and the exponent of what is essentially a fixed-point number represented in standard floating-point representation. A more detailed description of the fixed-point conversion process and performance results of our finite-precision design is presented in Chapter 4.

2.3.5 RTL Design

Once the author was comfortable with the fixed-point design and the performance results associated with this design an RTL implementation was developed. Before beginning the coding process for this step in the design flow, a hierarchical module level structure was established which formed the basis of our hardware design. Once this module level design was established each block in the hierarchy was coded in Verilog. After coding in Verilog each step in the computational data path was verified to meet the design specification of being functionally equivalent to the fixed-point design in both MATLAB and fixed-point C. Once this specification was satisfied the hardware design

could be synthesized for initial validation on an FPGA or production on an application specific integrated circuit (ASIC).

In our development we synthesized for both FPGA and ASIC implementations. The synthesis process provided us with a number of useful benefits. The first benefit is the obvious fact that we can use our synthesized design to port onto a processor. Second, synthesis provides us with a gate count prediction on the target semiconductor process. Finally, synthesis results provide us with timing analysis whereby we can establish the operational frequency of our implementation.

Additionally, we would like to note in this section that the PrecisionC tool was used to develop RTL code as well. In our design flow we validated the RTL produced from our fixed-point C implementation to be functionally equivalent to the RTL generated as an output of our hand-coded HDL implementation.

2.3.6. Real-Time Validation

The most exciting design step in our flow was the ability to validate our design using a real-time wireless system prototype. Once the hardware design was synthesized we were able to integrate our implementation into the wireless prototype, described in Chapter 5, to validate the system functionality of our design and determine through quantitative and qualitative means the performance of our detection algorithm implementation.

CHAPTER 3

Low Complexity Detection Algorithms

In this chapter we present an overview of standard multi-antenna detection algorithms and develop a low-complexity reformulation of these standard algorithms based on a division-free modular reformulation.

3.1 Detection Algorithms Overview

There are numerous detection algorithms for estimating transmitted symbols in a MIMO communication scheme. The simplest detector for MIMO channels is the linear detector. In this thesis we are interested primarily in the algorithmic structure of the zero-forcing (ZF) and minimum mean-squared-error (MMSE) linear detectors. In particular, we are interested in structuring these algorithms in such a way as to reduce their computational complexity in hardware. Along with the LMMSE and ZF algorithms some attention will be given to the iterative zero forcing (IZF), and iterative minimum mean-squared-error (IMMSE) algorithms. The MMSE receiver was introduced in [24]. An overview of maximum-likelihood (ML) and decision feedback detection algorithms can be found in [25].

3.1.1 Zero-Forcing Linear Detection

The zero-forcing linear detector often referred to as the linear decorrelator or decorrelating detector, works by removing inter-stream interference by projecting the

received signal \mathbf{y} onto a subspace orthogonal to the one containing all other data streams. This is the subspace orthogonal to the one spanned by the vectors $\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_{N_T}$ [8]. The linear decorrelator chooses a matrix \mathbf{C} which completely removes interference without regard to the effects of noise enhancement. Noise enhancement occurs as a result of the ZF detector compensating for losses in signal energy as a result of the detector removing the fraction of the received signal lying in the interference subspace. The matrix \mathbf{C} is chosen such that $\mathbf{C}\mathbf{H} = \mathbf{I}$ and will exist given $N_R \geq N_T$ and the columns of \mathbf{H} are linearly independent. The zero-forcing (ZF) linear detector is given by the $N_T \times N_R$ Moore-Penrose pseudoinverse matrix:

$$\mathbf{C} = \mathbf{H}^\dagger = (\mathbf{H}^* \mathbf{H})^{-1} \mathbf{H}^* . \quad (3-1)$$

We note that when \mathbf{H} is invertible \mathbf{H}^\dagger reduces to \mathbf{H}^{-1} . For a more detailed description of the ZF linear detector see [8, 26].

3.1.2 Linear Minimum Mean-Squared Error

An important drawback of the ZF linear detector is that it forces interference to zero, regardless of the interference strength. The minimum mean-squared-error linear detector (LMMSE) works by striking a balance between signal energy in the interference subspace and increased interference. In fact, the MMSE detector achieves an optimal balance of noise enhancement and interference suppression [26]. The linear MMSE detector is given by the matrix:

$$\mathbf{C} = (\mathbf{H}^* \mathbf{H} + N_0 \mathbf{I})^{-1} \mathbf{H}^* \quad (3-2)$$

where N_0 is the noise matrix and \mathbf{I} is the $N_R \times N_R$ identity matrix. A more detailed description of the LMMSE and ZF detection equations can be found in subsection 3.2.

3.1.3 Iterative Detectors

Both the MMSE and ZF detection criterion can be applied iteratively to determine the transmitted symbols from the value received by the vector of receive antennas. Rather than applying a bank of decorrelators or LMMSE receivers where each element in the bank estimates a parallel data stream as is done in the linear detector, the iterative detector uses a similar bank of decorrelators or LMMSE receivers which utilize successive cancellation of streams. This process works by iteratively applying the decorrelators or LMMSE receivers to an ordered set of received symbols. The ordering of the symbols is determined by an ordering algorithm which computes the symbol with the most energy after each iteration of the detection and places this symbol as the next to be computed. After each iteration of the detection each estimated symbol is subtracted off from the received signal and the process is iterated until all symbols have been estimated. For more on iterative detection see [8].

3.1.5 Maximum Likelihood

The most accurate detection algorithm is the Maximum Likelihood (ML) algorithm which is described in many digital communication textbooks [25]. Implementing the ML algorithm for constellations of size larger than that of QPSK turns out to be very expensive in terms of algorithmic complexity which maps to undesirable effects such as increased area and latency. Reducing the complexity of ML estimation remains largely an open research topic although there is significant research underway in

a category of algorithms that are referred to as Near-ML or Decision Feedback algorithms [27]. Near-ML algorithms approximate Maximum Likelihood performance with significant savings in terms of computational complexity. We do not concern ourselves with these algorithms in this work.

3.2 Linear MMSE Detection Equations

An unbiased minimum mean-squared-error (MMSE) estimate for a transmitted quadrature amplitude modulated (QAM) symbol in a multiple stream system with an $N_T \times N_R$ configuration is computed using

$$\hat{s}_i[n] = \frac{\mathbf{h}_i^H \mathbf{R}_{N_T}^{-1} \mathbf{y}[n]}{\mathbf{h}_i^H \mathbf{R}_{N_T}^{-1} \mathbf{h}_i} \quad (3-3)$$

where \hat{s}_i is the estimated QAM symbol for the transmitted symbol s_i , \mathbf{h}_i is the i th column of the channel matrix \mathbf{H} , and

$$\mathbf{R}_{N_T} = \sum_{i=1}^{N_T} \mathbf{h}_i \mathbf{h}_i^H + \mathbf{R}_{ww} . \quad (3-4)$$

$\mathbf{y}[n]$ is the received signal. In the case of an OFDM system, $\mathbf{y}[n]$ represents one of the points in the Fast Fourier Transform (FFT) computation at the receiver. As the limit \mathbf{R}_{ww} approaches zero, we arrive at the detector with zero forcing criterion. The equations above can be derived by solving $\min_{W_i^H} E[|s_i - \hat{s}_i|^2]$ where $\hat{s}_i = W_i^H \mathbf{y}$ is the linear estimator.

3.3 Computational Complexity

In order to calculate the computational complexity of (3-3) we first examine the common term

$$\mathbf{h}_i \mathbf{h}_i^H = \begin{bmatrix} h_{i1} h_{i1}^* & h_{i1} h_{i2}^* & \cdots & h_{i1} h_{iN_R}^* \\ h_{i2} h_{i1}^* & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ h_{iN_R} h_{i1}^* & \cdots & \cdots & h_{iN_R} h_{iN_R}^* \end{bmatrix}. \quad (3-5)$$

This term can be shown to have $N_R \left(\frac{N_R + 1}{2} \right)$ complex multiplications. Because there are N_T of the terms $\bar{h}_i \bar{h}_i^*$ in the computation of \mathbf{R}_{N_T} , computing \mathbf{R}_{N_T} requires $N_T \left(\frac{N_R^2 + N_R}{2} \right)$ complex multiplications. Computing the inverse of (3-3) requires αN_R^3 complex multiplication for some $\alpha > 0$. The complexity of the multiplication between the $1 \times N_R$ vector \mathbf{h}_i^H and the already computed $N_R \times N_R$ matrix $\mathbf{R}_{N_T}^{-1}$ requires N_R^2 additional complex multiplications. Multiplication with the computed $1 \times N_R$ vector $\mathbf{R}_{N_T}^{-1}$ and either the $1 \times N_R$ vectors $\mathbf{y}[n]$ in the numerator or \mathbf{h}_i in the denominator requires another N_R complex multiplications. The number of complex multiplications required for each symbol estimation is therefore $N_T \left(\frac{N_R^2 + N_R}{2} \right) + \alpha N_R^3 + N_R^2 + 2N_R$. Each complex multiplier requires four real multipliers. Therefore computing $\hat{s}_i[n]$ in (3-3) requires $4 \left(N_T \left(\frac{N_R^2 + N_R}{2} \right) + \alpha N_R^3 + N_R^2 + 2N_R \right)$ real number multiplications. We note that the complexity of matrix inversion contributes only once to the overall complexity of our

algorithm because our design is structured to reuse this term. Because there are N_T symbols to estimate we have $4 \left(N_T \left(\frac{N_R^2 + N_R}{2} \right) + \alpha N_R^3 + N_T (N_R^2 + 2N_R) \right)$ real multiplications. If we allow $N_T = N_R = N$ in an NxN multi-antenna configuration, then the overall complexity in terms of real multiplications for QAM symbol estimation using standard MMSE detection simplifies to

$$(6 + 4\alpha)N^3 + 10N^2. \quad (3-6)$$

For a 3x3 multi-antenna configuration with 48 data tones, 20 MHz bandwidth and OFDM symbol duration of 4 μ s, the processing requirement is 3024+1296 α million multiplies/sec. This computational requirement is higher than that achievable by standard general purpose processors due to the cubic computational complexity of the multi-antenna linear detection algorithm which is dependent on the number of receiver antennas in the wireless network. Computing the estimates $\hat{s}_i[n]$ for a MIMO system with three or more receiving antennas requires a processing power that exceeds that of the current highest performance general purpose processors. We therefore chose to structure the formulation of our algorithm for an application specific processor.

A primary reason for designing application specific processors is that many wireless applications are intended to run on low-power or power sensitive devices. By designing a custom processor we are able to reduce power consumption by producing a small and low-complexity design. In the next section we present an algorithmic reformulation of (3-3) which we will later map into a fixed-point design and a Verilog implementation.

3.4 Reformulation of MIMO Detection Equations

In this subsection we propose a modular division-free dot-product reformulation of the LMMSE and ZF equations for MIMO detection. Using this reformulation we can achieve a low-complexity architecture which will be described in Section IV. Our reformulation is based on (3-3). We arrange this equation into a modular division-free structure based on complex dot-product multiplications. In order to obtain a division-free multi-antenna detection (DIFMAD) formulation we use the matrix inversion lemma to write $\mathbf{R}_{N_T}^{-1}$ recursively in the following form:

$$\mathbf{R}_k^{-1} = \mathbf{R}_{k-1}^{-1} - \frac{\mathbf{R}_{k-1}^{-1} \mathbf{h}_k \mathbf{h}_k^H \mathbf{R}_{k-1}^{-1}}{1 + \mathbf{h}_k^H \mathbf{R}_{k-1}^{-1} \mathbf{h}_k}. \quad (3-7)$$

Applying recursion to $\mathbf{R}_{N_T}^{-1}$ we decrement k through $N_T, N_T - 1, N_T - 2, \dots, 1$. The recursion terminates at $\mathbf{R}_1^{-1} = \mathbf{R}_w^{-1}$. To compute the inverse of the noise estimation matrix \mathbf{R}_{ww} we note that \mathbf{R}_{ww} is diagonal and composed of entries $\sigma_{11}, \sigma_{22}, \dots, \sigma_{N_R N_R}$. The inverse can therefore be computed using

$$\mathbf{R}_{ww}^{-1} = \frac{\text{adj}(\mathbf{R}_{ww})}{\det(\mathbf{R}_{ww})} = \frac{\text{adj}(\mathbf{R}_{ww})}{\sigma_{11}^2 \sigma_{22}^2 \cdots \sigma_{N_R N_R}^2} = \frac{\text{adj}(\mathbf{R}_{ww})}{\text{tr}[\mathbf{R}_{ww}]}. \quad (3-8)$$

We also wish to avoid division in our hardware implementation as the division operation is a costly processor function. Moreover, a division-free approach provides us with other desirable system qualities. In particular, a division-free solution maintains a higher degree of precision, uses less chip area, and computes the output faster than a division based design does. To implement a division-free system, we begin by removing

the division operation from denominators of terms with the same form as the denominator in (3-7) which are represented as

$$1 + \mathbf{h}_{N_T}^H \mathbf{R}_{N_T-1}^{-1} \mathbf{h}_{N_T} . \quad (3-9)$$

We move all terms of this form into the numerator and maintain their multiplication by subsequent terms from their own denominator in a reliability metric. This reliability metric, which we refer to as the scale value, is stored until the end of the detection algorithm computation. The approach outlined in equations (3.7) and (3.8) forms the basis for the implementation presented in Chapter 4.

Additionally, for more information on the matrix inversion lemma the author suggests [28]. Furthermore we note that similar methods to the ones presented in this chapter were simultaneously and independently developed at Bell Laboratories in a paper on fast V-BLAST [29].

3.5 Iterative MMSE Algorithm Support

The formulation presented in the previous subsection can easily be extended to iterative MMSE algorithms, since in each step we are looking at solving equations of the form of (3-3) with a varying number of summation terms in the computation of the $\mathbf{R}_{N_T}^{-1}$ matrix. The number of steps required to compute the inverse varies linearly with the number of steps required to execute the iterative MMSE algorithm.

CHAPTER 4

Finite Precision Implementation

This Chapter presents a low-complexity fixed-point implementation for our multi-antenna receiver and uses simulation results to verify this implementation. Our implementation is based on the algorithms developed in the previous chapter. Additionally, we present a novel “folded-pipelined” architecture implemented in hardware which is based on our finite-precision implementation.

We begin the section by presenting our simulation setup. We then present a bit-width study on internal precisions necessary for complex dot-product multipliers in our fixed-point structure. This study is followed by the presentation of a generalized fixed-point structure based on the DIFMAD algorithm which we derived in the previous section. This fixed-point structure can be referred to as a folded-pipelined architecture. Additionally, we include a discussion of our Verilog hardware description language (Verilog HDL) implementation and application specific integrated circuit (ASIC) gate counts for various hardware implementations.

Throughout the analysis in this section we assume a pseudo floating point representation with parameters (B_m, B_e) where B_m is the number of bits required to represent the mantissa and B_e is the number of bits required to represent the exponent. Additionally, we present finite-precision simulation results for the 2x3 MIMO configuration. Our design approach can be used for any multi-antenna configuration.

4.1 Simulation Setup

We simulate the Rayleigh flat-fading channel, which has three resolvable paths, transmitting over a frequency range of 2-2.4GHz using 20MHz sub-carriers. BER performance is calculated by averaging over 30,000 packets of size 200 bytes each. The number of carriers used is 64, with 48 data carriers and 4 pilot carriers. The OFDM symbol duration is 4 μ s. An FFT length of 64 and a cyclic prefix length of 16 are used. Simulations were carried out under BPSK, QPSK, 16-QAM, and 64-QAM signaling schemes at various data rates. The data rate is computed by

$$\text{data rate} = \# \text{ of bits per symbol} * \text{code rate} * \frac{\# \text{ of data tones per burst}}{\# \text{ of pilot tones per burst}} * N_T. \quad (4-1)$$

Table 1 depicts the attainable data rates computed under the simulation and prototyping environments used for the various MIMO configurations. Symbol energies are adjusted appropriately for each modulation scheme. Additionally, the IEEE 802.11a interleaver is used [4].

Table 1. Possible System Data rates.

DATA RATES			
Throughput (Mb/s)	System Configuration	Modulation Technique	Coding Rate
6	SISO	BPSK	1/2
9	SISO	BPSK	3/4
12	SISO	QPSK	1/2
18	SISO	QPSK	3/4
24	SISO	16-QAM	1/2
36	SISO	16-QAM	3/4
48	SISO	64-QAM	2/3
54	SISO	64-QAM	3/4
12	2x2 or 2x3	BPSK	1/2
18	2x2 or 2x3	BPSK	3/4
24	2x2 or 2x3	QPSK	1/2
36	2x2 or 2x3	QPSK	3/4
48	2x2 or 2x3	16-QAM	1/2
72	2x2 or 2x3	16-QAM	3/4
96	2x2 or 2x3	64-QAM	2/3
108	2x2 or 2x3	64-QAM	3/4
18	3x3	BPSK	1/2
27	3x3	BPSK	3/4
36	3x3	QPSK	1/2
54	3x3	QPSK	3/4
72	3x3	16-QAM	1/2
108	3x3	16-QAM	3/4
144	3x3	64-QAM	2/3
162	3x3	64-QAM	3/4

Floating point simulations are obtained from solving for the QAM symbol estimation. We solve for the floating-point QAM symbol estimation by employing standard floating point arithmetic operations such as matrix inversion/pseudoinversion and multiplication in Matlab. The results from our floating point simulations are then used as a performance reference for our fixed-point design, as well as to outline some performance trends we expect to see in our real-time implementation. Fixed-point simulations are used to calculate our finite-precision implementation loss and to validate our DIFMAD algorithmic implementation in comparison to the floating-point solution of (3-3).

4.2 Finite Precision Results

In this subsection we present a bit-width study to determine internal precisions necessary for complex dot-product multipliers in our fixed-point structure. We use the pseudo floating point representation described above where we represent mantissa and exponent precisions separately. Fig. 4-1 presents results of our bit-width study for LMMSE detection at the maximal 2x3 throughput for our system (108Mb/s). These results indicate that an internal precision at the inputs to our complex dot-product modules of only fifteen bits is sufficient to maintain a negligible fixed-point implementation loss. Specifically, we maintain an implementation loss of less than 0.2dB at a Frame Error Rate (FER) operating point of 10^{-2} . Another study was conducted for IMMSE detection with similar results. Additionally, from Fig. 4-1 we see that obtaining performance within 0.2dB of our floating-point precision implementation requires an exponent precision of six bits. A similar study was conducted using an exponent precision of five bits. Because the performance of this study was severely

degraded beyond an acceptable implementation loss at an FER of 10^{-2} we chose not to present these results. Furthermore, the area of our design is a function of mantissa precision far more than it is of exponent precision. Additionally, the maximum area of complex dot-product modules required in our design is independent of the number of transmit antennas. Therefore, internal precisions for complex dot-product inputs determined for the 2x3 LMMSE precision study in Fig. 4-1 are the same as those necessary for the 3x3 LMMSE detection.

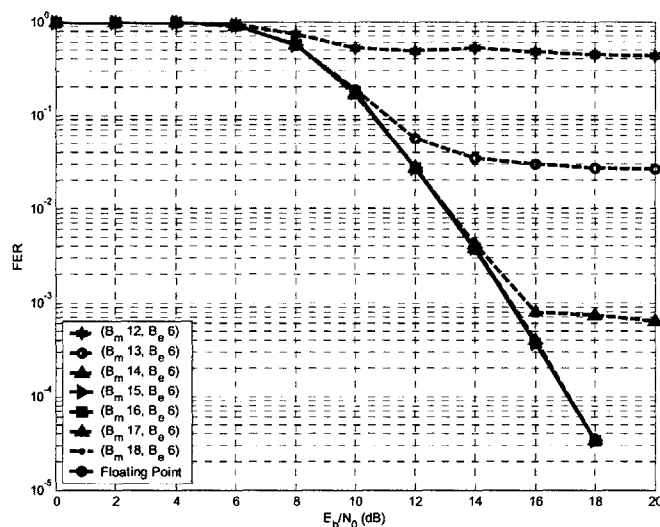


Fig. 4-1. Fixed-Point Bit-width Study for BER for $N_T = 2$ and $N_R = 3$

From this point onward we chose to present results for fixed-point simulations using a mantissa precision of eighteen bits and an exponent precision of six bits. We choose to maintain eighteen bits for the mantissa precision as opposed to the possible fifteen bits primarily because the Xilinx Virtex-II field programmable gate array (FPGA) used in our real-time prototype consists of eighteen-bit wide multiplier logic units. By maintaining an eighteen-bit mantissa we can easily map eighteen-bit wide Verilog multipliers used in our finite-precision implementation onto the FPGA. Furthermore, we

have shown that our simulated performance using a finite-precision of fifteen-bits at the most demanding throughput our 2x3 system handles (108 Mb/s) performance equally well as an eighteen-bit representation.

We now present the simulation results of our floating-point and fixed-point implementations of various MIMO systems. We use these simulation results to validate our generalized fixed-point design and demonstrate the efficiency of our implementation. Specifically, using the simulation parameters provided earlier in this section and using floating-point and fixed-point simulations we determine the BER at various E_b/N_0 values for each multi-antenna system configuration. Here E_b/N_0 is the ratio of the energy per bit (E_b) to the spectral noise density (N_0) at the receiver. Fig. 4-2 presents floating-point simulation results for the linear detector operating at 108Mb/s under MMSE criterion. In order to maintain constant data rates across various system configurations we need to appropriately vary the modulation scheme and the coding rate. In this particular case 64-QAM constellations with rate $\frac{3}{4}$ were used in the simulations of 2x2 and 2x3 configurations while 16-QAM constellations with rate $\frac{3}{4}$ were used in the simulations of 3x3 configurations.

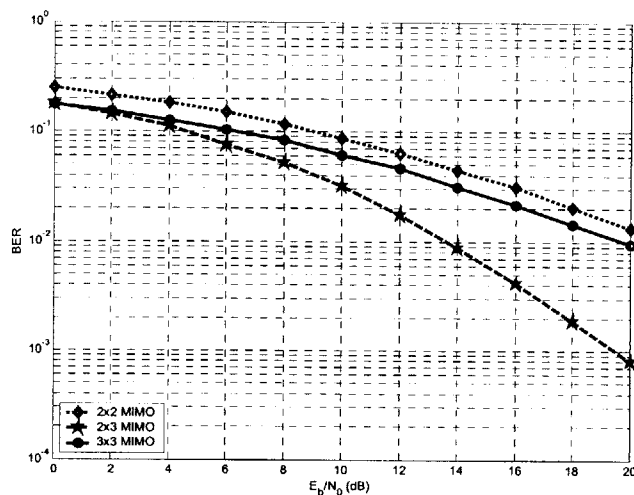


Fig. 4-2. BER for LMMSE detector (multiple configurations).

From the results of Fig. 4-2, at the same data rate, we observe the BER of the 2x3 system to be lower than that of the 3x3 system. We attribute this trend to the rate-diversity trade-off between the 2x3 configuration and the 3x3 system. We can intuitively think of this trade-off as follows. If we are given two MIMO systems with identical transmitting conditions and identical receiving conditions then the system with more receiver antennas will have a lower BER as a consequence of the additional receive diversity. The benefit of this improved diversity comes at the cost of decreasing the maximum obtainable throughput due to a reduction in the number of degrees of freedom where the number of degrees of freedom is defined as $\min(N_T, N_R)$. The result in Fig. 4-2 implies that for a fixed number of receiver antennas it is advantageous to transmit data by using the minimum number of antennas which will allow for the desired data rate. Without channel information and a feedback mechanism; however, a strategy of transmission using the minimum number of transmit antennas will not always result in the optimal performance.

The floating-point and fixed-point BER for LMMSE and IMMSE detection are depicted in Fig. 4-3 for a 2x3 MIMO system operating at 108Mb/s. Both the LMMSE and IMMSE fixed-point implementations use internal complex dot-product precisions of 18 bits. From this plot we observe a fixed-point implementation loss of less than 0.2 dB at a BER of 10^{-3} . This result verifies our DIFMAD algorithmic structure as well as our finite-precision implementation.

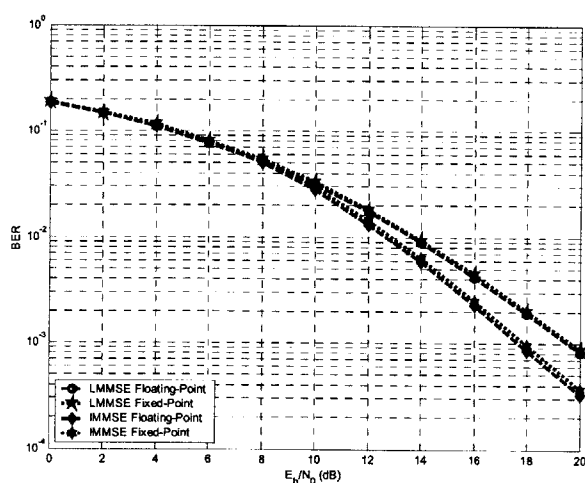


Fig. 4-3. Fixed-point Implementation Loss of LMMSE and IMMSE where $N_T = 2$ and $N_R = 3$.

4.3 ASIC Gate Counts

In this subsection we present a number of results pertaining to gate counts for custom processors using our finite-precision implementation. We begin with a study which determines ASIC gate counts across variable mantissa precisions ranging B_m from 18 bits down to 12 bits. We also compare ASIC gate counts of hand-coded Verilog HDL implementations for various receiver system configurations across a range of clocking frequencies. We then discuss these results.

In order to determine the impact of internal precisions on area we conducted a study using the PrecisionC register transfer language (RTL) synthesis tool from Mentor Graphics. This tool is capable of taking fixed-point C code and generating VHDL or Verilog HDL code. We went through this process and validated HDL code generated by the tool as functionally equivalent to our hand-coded Verilog implementation. The PrecisionC tool is capable of using the generated HDL to estimate gate counts for a given design. Fig. 4-4a presents the estimated gate counts for a 2x3 system that operates at 40MHz and uses variable mantissa bit-widths. While these results are only estimates of actual gate counts we found the percentage difference between gate count results obtained for each mantissa bit-width implementation to be relatively accurate. Fig. 4-4b supports this assertion by depicting actual ASIC gate counts which were produced from a synthesis of the same HDL used to generate the estimates presented in Fig. 4-4a. We notice one deviation from this trend between the gate count for an internal precision of 15-bits and 14-bits where the gate count goes up when we expect it to go down. This can be attributed to the tool not always finding the optimal design and inserting an additional pipelining stage thus increasing latency and, as a consequence, gate count. The synthesis tool used in our research is the Synplify Pro synthesis tool from Synopsis.

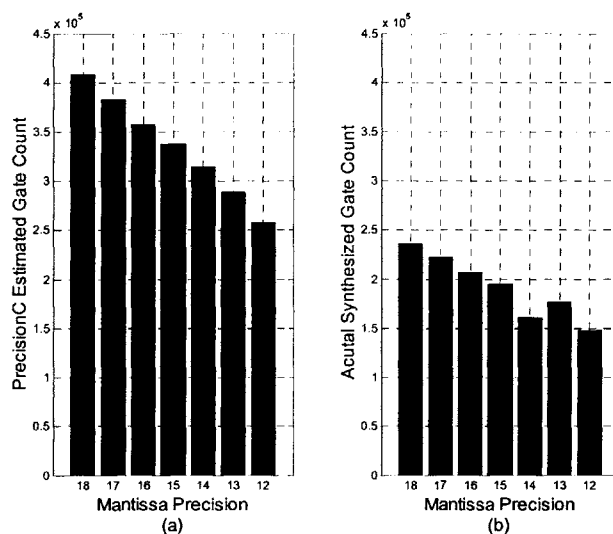


Fig. 4-4. Estimated Gate Counts for 2x3 20MHz Architecture from PrecisionC tool for varying mantissa precisions.
 (a) depicts estimated gate counts generated from the PrecisionC Tool
 (b) depicts actual synthesized gate counts

We now present gate counts generated from the synthesis of our hand-coded Verilog implementation for various MIMO receiver configurations operating at specific clocking frequencies. We then compare these results against both estimated gate counts computed by the PrecisionC tool and actual gate counts computed by synthesizing HDL generated by this tool. Fig. 4-5 depicts gate counts results for three different system configurations operating at various clocking frequencies. The leftmost bar in the graph represents the gate count for a 3x3 system configuration designed for 20MHz operation requiring one clock cycle for each complex dot-product computation. The middle bar in the graph is representative of the gate count for a 3x3 system configuration designed for 60MHz operation requiring three clock cycles for each complex dot-product computation. The rightmost bar in the graph represents the gate count for a 2x3 system configuration operating at 20MHz requiring one clock cycle for each complex dot-product computation. By computing dot products in parallel as we do in the 3x3

configuration operating at 60MHz we are able to reduce the gate count by over a factor of two as compared to the 20MHz design. We achieve an overall gate count of less than 200,000 gates for a 3x3 LMMSE receiver for our hand-coded Verilog implementation.

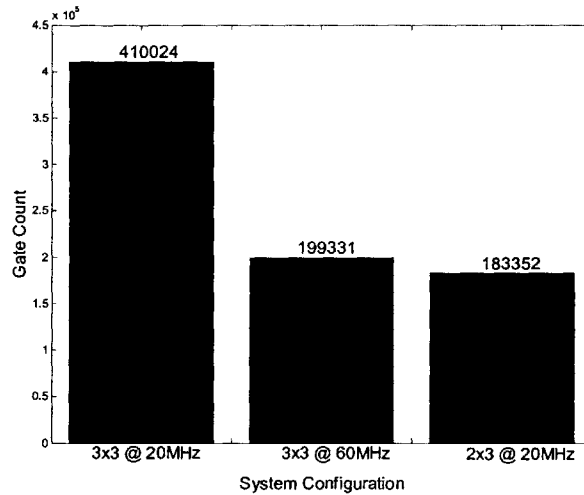


Fig. 4-5. ASIC Gate Counts produced from synthesis of RTL implementations

Comparing this actual gate count, which is a result of the synthesis of our hand-coded design, to the estimated gate count and to the actual synthesized gate count from the PrecisionC tool we can make a number of observations. First, comparing the synthesized gate count for our hand-coded HDL design at 20MHz and the actual synthesized gate count from HDL code generated using Verilog where $B_m = 18$ in both designs we can see that the hand-coded design performs better than the tool generated design for the specific case of the 2x3 receiver operating at 20MHz. Our hand-coded design has a gate count of 183352 as compared to 236043 for the tool generated HDL which is a difference of over 50,000 gates. We are confident that with sufficient optimization of our fixed-point C code which we passed through the PrecisionC tool to

generate functionally equivalent HDL code we could have matched or beaten the size of our hand-coded design in terms of gate count.

We note at this time why we see a significantly smaller gate count for the 60MHz 3x3 configuration as compared to the 20MHz configuration. We are performing operations in parallel for our 3x3 configuration and therefore can reduce the number of complex dot-product modules required in our design. Interestingly, we see that the area of the 60MHz 3x3 system configuration nearly matches that of the 2x3 configuration operating at 20MHz. The difference between the two configurations is on the order of only 10,000 gates. The implication of this result is that by reducing the number of complex dot-product modules in our architecture and performing these operations serially with a faster clocking frequency, a practice known as “folding”, we have been able to achieve significantly higher data rates for essentially the same cost in area. In the case of 2x3 and 3x3 designs we obtain maximal data rates 50% higher (162Mb/s as compared to 108Mb/s) because we are using three transmitters as compared to two. In other words, we have gained one degree of freedom for the same cost in area by folding our design. Generally, we can always obtain lower gate counts with a folded design as compared to an unfolded design. Due to the cubic nature of MMSE detectors and the trend in industry toward incorporating MIMO technologies, we observe that even as the cost of transistors in digital circuits continues to drop rapidly as MIMO systems move to higher dimensions it is apparent that folded architectures will become increasingly relevant in multi-antenna receivers.

CHAPTER 5

Multi-Antenna Testbed and Real-Time Results

This chapter serves as a validation of our design by describing our laboratory prototype and presenting results from actual real-time wireless measurements performed in a typical office environment and their comparison to simulations. We use the results obtained from our laboratory prototype as a validation of our low-complexity multi-antenna receiver design. We begin the chapter with an overview of the multi-antenna prototype which covers the structure of the prototype and a description of its important components such as the time-domain/FFT module, the channel estimation module, the MIMO block responsible for implementing the DIFMAD algorithm, and the forward error correction module. We then describe the configuration of our measurement setup. We conclude the section with a presentation and discussion of the real-time wireless measurements obtained in a typical office environment.

5.1 Multi-antenna Prototype

While the system aspects of this prototype are not the focus of this thesis, we present a summary of the entire prototype for the purposes of understanding how our multi-antenna receiver architecture interfaces into a wireless network. Our prototype can be employed to implement up to a 4x4 system. Fig. 5-1 presents a 3x3 MIMO transceiver implemented on this prototype. Fig. 5-2 is a picture of one MIMO prototype

with three antennas. The radio blocks consist of radio frequency circuits required for up-conversion and down-conversion of baseband signals, a power amplifier for the transmit path and low-noise amplifiers (LNA) for the receive path. This system contains an analog front end (AFE) for converting digital baseband signals to and from analog baseband signals. The physical layer (PHY) field programmable gate array (FPGA) implements time-domain algorithms for automatic gain control, digital timing recovery, as well as FFTs and IFFTs. The frequency domain data are passed from the PHY FPGA to the PHY digital signal processor (DSP) on the receiver side for channel estimation and tracking. The uncompensated frequency domain data along with channel and noise variance estimates are passed to the MIMO FPGA, which implements channel compensation algorithms. The compensated data from the MIMO FPGA are sliced using a soft-slicer and passed to the Forward Error Correction (FEC) FPGA which implements interleaver, de-interleaver, convolution encoder, Viterbi decoder, scrambler, and descrambler algorithms. The descrambled data are passed to the media-access control (MAC) accelerator FPGA which along with the MAC DSP implements MAC protocol and a connection to the host PC.

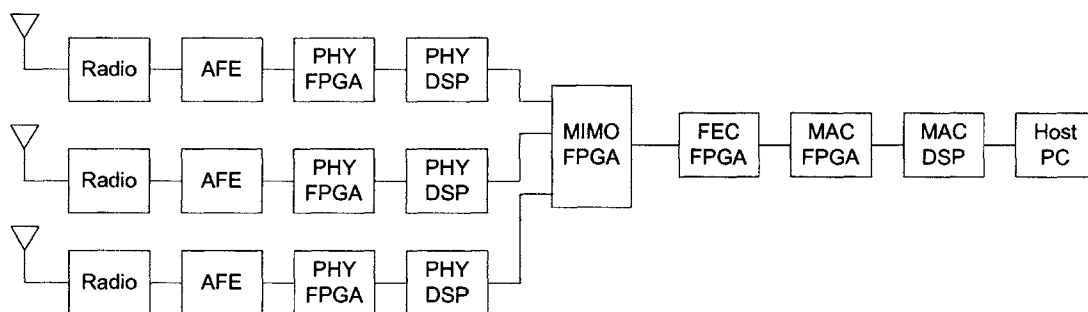


Fig. 5-1. 3x3 MIMO Transceiver System.

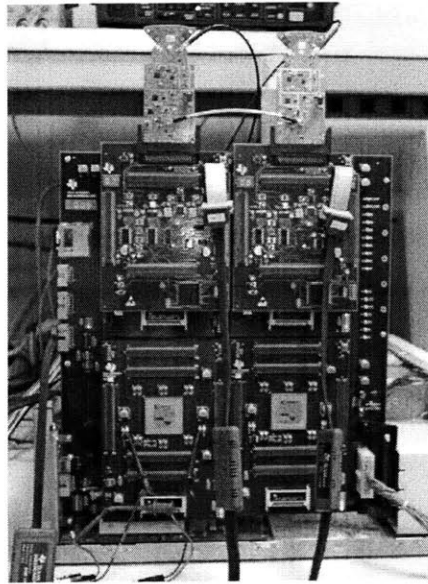


Fig. 5-2. MIMO Transceiver Prototype

5.2 Time Domain Processing/IFFT Module

The time-domain processing module consists of transmit and receive portions. In the transmit portion, frequency domain data are transferred from a Texas Instruments TMS320C6416 DSP, referred to as a PHY DSP, via a 16-bit synchronous external memory interface (EMIF) running at an 80 MHz clock frequency. These data are zero-padded with 192 zeros before feeding into a 256-point IFFT module. A 64 sample cyclic prefix is added at the output of the IFFT module. The output of the cyclic prefix module is then passed through a time-domain window module before sending samples to digital-to-analog converters running at 80 MHz.

The receiver side of the time-domain processing consists of an automatic gain control (AGC) module, timing synchronization and frequency synchronization blocks, an FFT block and an interface to the PHY DSP. The AGC module is responsible for maintaining the optimum dynamic range at the input of the analog-to-digital converters.

Additionally, timing recovery circuits are used to determine the FFT window placement. The frequency recovery circuits are used to estimate and compensate for frequency errors. A 256-point FFT module is employed in our system but only 64 points of this fast transform are passed to the PHY DSP via a 16-bit synchronous EMIF interface operating at 80 MHz.

5.3 Channel Estimation

The C6416 PHY DSP operates at 600 MHz and is responsible for estimating the channel impulse response during transmission of the long sequence. After transmission of a long sequence the PHY DSP keeps track of timing and frequency errors. This estimation block then compensates for these errors before passing the data to the MIMO FPGA, which is the topic of the next subsection.

5.4 MIMO FPGA Core

The MIMO FPGA, also referred to as the MIMO core, implements the DIFMAD algorithm proposed in this paper. A QAM mapper is also implemented on this FPGA for transmission. The MIMO core estimates transmitted QAM symbols and computes a corresponding reliability metric for multiple data streams. This metric is passed to the forward error correction module described in the next subsection. This block takes the following inputs: fast FFT outputs from each receive chain, a channel estimation matrix from each transmit antenna to each receive antenna, and noise variance estimations for each receive antenna. These inputs are used to compute (3-3) where $\mathbf{y}[n]$ is the vector of fast FFT outputs from the receive chains, \mathbf{H} is the channel estimation matrix, and \mathbf{R}_{ww} is the diagonal matrix of the noise variance estimations from each receive antenna. The

MIMO core employs antenna combining and interference estimation/cancellation techniques to process the inputs and generates QAM symbol estimates and reliability metrics (including bit probabilities, if desired), which are passed to a symbol demapper to generate bit metrics for forward error correction (FEC) decoding. The system level block diagram in Fig. 5-3 shows the MIMO core and the neighboring blocks.

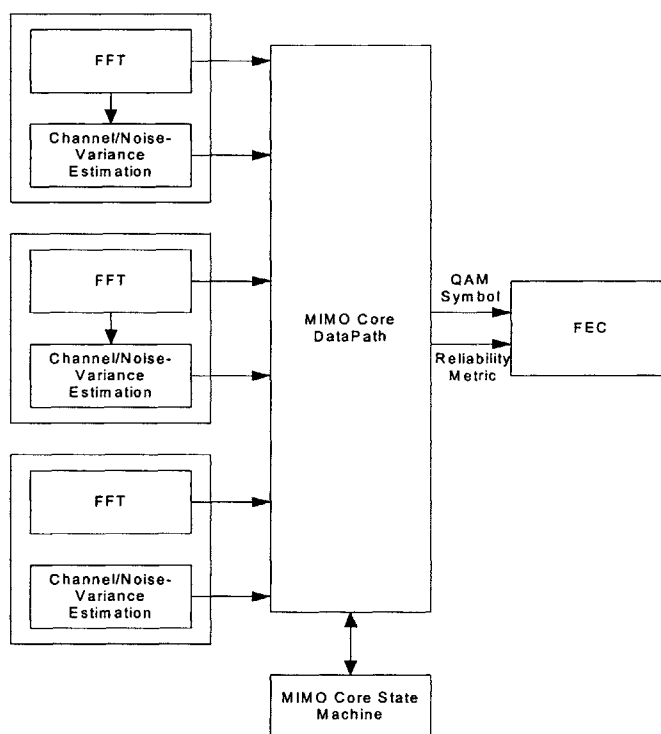


Fig. 5-3. Input/Output Connections of MIMO FPGA

The MIMO core is implemented on a Xilinx Virtex-II FPGA device with part number XC2V6000FF1152-5. All the interfaces to this FPGA are operating at 80MHz clock speed while the MIMO core operates any clock speeds ranging from 20-60 MHz.

5.5 FEC Module

The main blocks in the FEC are depicted in Fig. 5-4. The throughput of the FEC is determined by the Viterbi decoder which is the most complex block in the FEC. To achieve high data rates, several algorithm and architectural optimizations are employed. A word level parallelism is exploited in the Viterbi decoder by implementing a radix-4 algorithm, where two information bits are processed at each clock by applying a look-ahead scheme to the conventional radix-2 trellis [30], [31]. Other more advanced algorithms can be employed as well. For a detailed description of the FEC see [32].

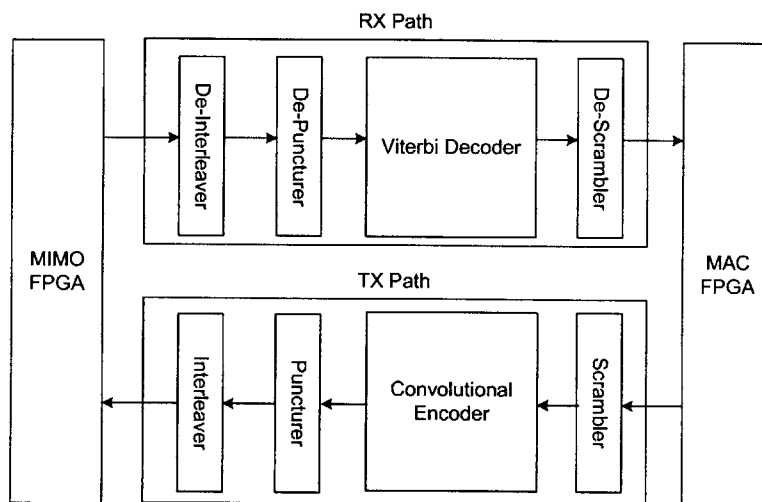


Fig. 5-4. Forward Error Correction FPGA Block Diagram

5.6 Real-time Wireless Measurements

This subsection presents measurements obtained using our system prototype. These measurements are used to validate our low-complexity receiver architecture and to provide insight into practical multi-antenna propagation issues encountered during real-time wireless operation. We collected these measurements on the first floor of the Texas Instruments South Campus building in Dallas, Texas USA. All walls and floors are made

of concrete and there are desks and storage shelves along the walls. For each receiver location, measurements were taken using multiple orientations for the transmitters and receivers. The antennas used for the transmitters and receivers are omnidirectional with a transmitter antenna spacing of roughly two meters and a fixed receiver spacing of approximately one-half wavelength. The transmitting power was varied so that the SNR changes at increments of 3dB from a relative SNR of 0dB to 27dB and BER was measured at the receiver. BER measurements were taken at the baseband and therefore include system effects. QAM symbol data generation for transmission over a wireless channel is performed by a Rhode & Schwarz vector signal generator SMIQ with a frequency range including the band of interest 2-2.4 GHz. We justify classifying our close range links as multipath due to significant interferes and suitable transmitter spacing relative to the propagation distance. Additionally, our results indicate a lower BER for MIMO measurements compared to SISO measurements. This is not the case in line of sight (LOS) propagation.

In order to demonstrate the functionality of our receiver architecture as well as highlight the tradeoff between the data rate and the diversity of multi-antenna systems, we compare the real-time wireless results of the 2x2 system with those of the 2x3 system. These measurements were taken continuously at the output of the MIMO FPGA and averaged over a one minute interval for a 2x2 system and then for a 2x3 system. The system configuration consists of three receiving antennas and two transmit antennas which we call Rx1, Rx2, Rx3, Tx1, and Tx2 respectively. Antennas Rx1 and Rx2 were used for 2x2 reception and all three receiver antennas were used for 2x3 reception. Both Tx1 and Tx2 were used to transmit in both configurations.

In this paper we focus on two specific trials which we will refer to as trial 1 and trial 2. In both trials the receiver prototype was located in the same location which is less than 10 meters from the transmit antennas which were separated by approximately 10 meters. In each trial measurements were taken for 2x2 and 2x3 systems after observing the channel capacity metric. The channel condition metric is a function of the channel matrix condition number with a range from 0 to 1 with 0 representing fully correlated channels and 1 representing orthogonal channels. Measurements were taken at 18Mb/s, 36Mb/s, 72Mb/s and 108 Mb/s across the relative transmitter SNR range of 0dB to 27dB. In trial 1 the system antennas were oriented randomly and in trial 2 the antennas were carefully oriented to obtain a desired effect which we will discuss shortly. In trial 1 the channel capacity metric was observed to be greater than 0.9 from 2 GHz to 2.25 GHz after which the metric rolled off to 0.5 in the higher frequencies of the band from roughly 2.3 to 2.4GHz in the 2x3 configuration. Additionally, the SISO BER performance results from each transmit antenna to each receive antenna was observed to be within an order of magnitude of each other. In trial 2 the antennas at each end of the wireless link were positioned to induce poor reception at receive antenna Rx2 and reliable reception at Rx3 given positioning constraints on Rx3 relative to Rx2. This was accomplished by observing single input single output (SISO) BER measurements from each transmit antenna. The channel capacity metric for trial 2 was essentially flat compared to trial 1 in a 2x3 system configuration. Additionally the integral of the channel capacity metric across all frequencies in trial 2 was higher than that in trial 1. While we use the channel capacity metric in this subsection we note that this metric was only used as a rough, but suitable, indicator of the channel's quality.

Fig. 5-5a and Fig. 5-5b presents results from trial 1. Fig. 5-5c and Fig. 5-5d present results from trial 2. Fig. 5-5a and Fig. 5-5c were obtained using a 2x2 configuration while Fig. 5-5b and Fig. 5-5d were obtained with a 2x3 setup. Fig. 5-6a and Fig. 5-6b depict BER performance results for the same two trials and compare 2x2 to 2x3 performance on the same graphs for 72Mb/s operation.

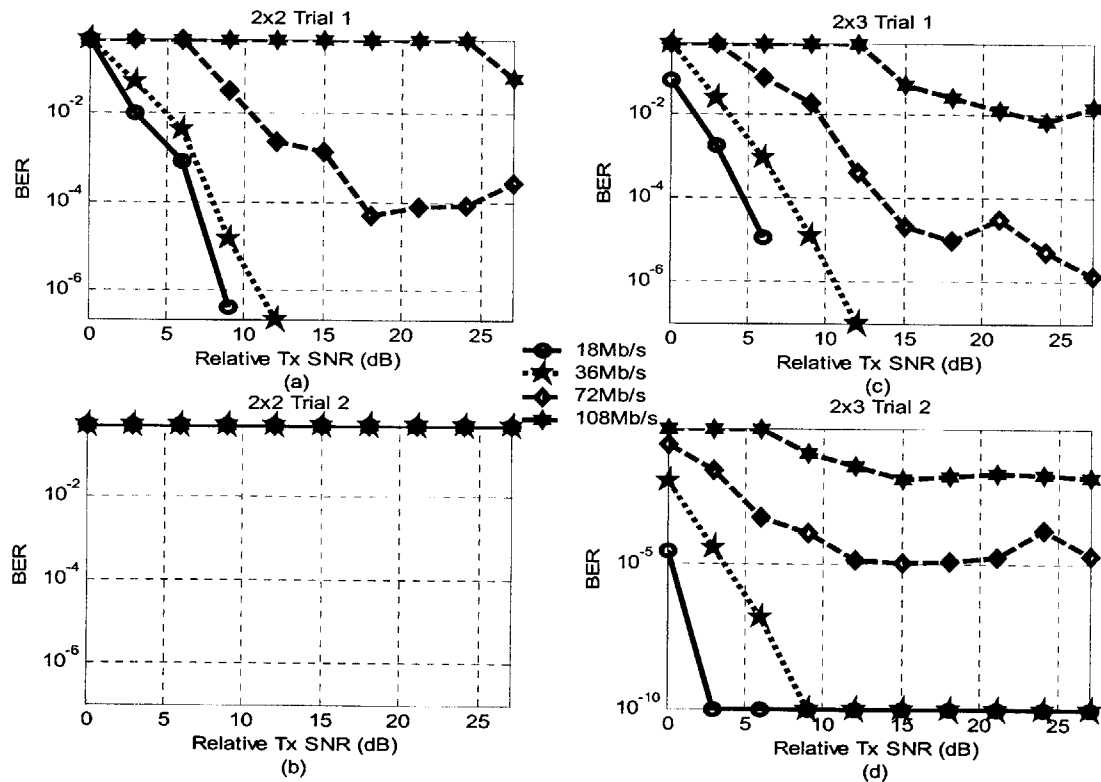


Fig. 5-5. Real-time Wireless Measurements

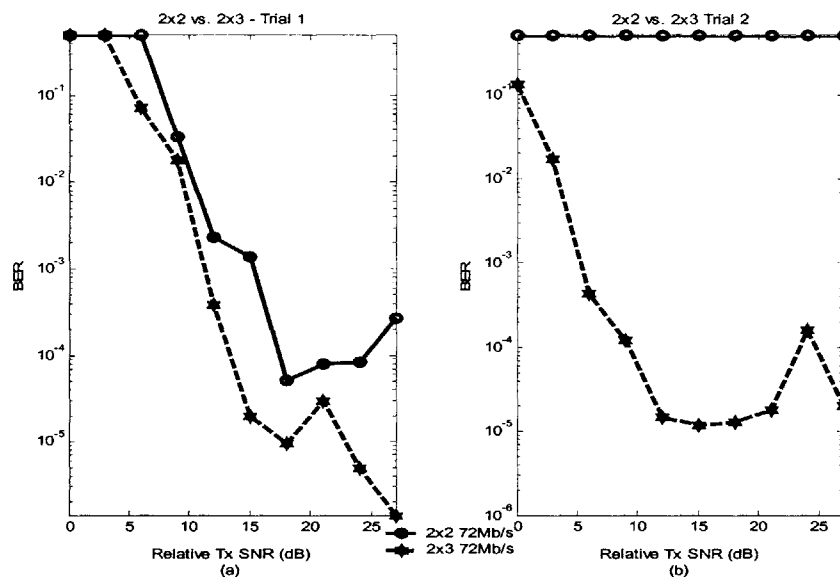


Fig. 5-6. 2x2 vs. 2x3 Comparison for Trials 1 and 2 at 72Mb/s.

While the results in Fig. 5-5 and Fig. 5-6 are obtained for specific cases we make some interesting observations on multi-antenna wireless propagation. First we note that in both trials the 2x3 BER is significantly lower than the 2x2 BER. In the case of Trial 2 the 2x3 system is performing several orders of magnitude better than the 2x2 configuration. From Fig. 5-5c we see that the 2x2 system has a BER of $\frac{1}{2}$ regardless of the transmit SNR. The value of $\frac{1}{2}$ is not an averaged result, but an indication that the BER tester lost synchronization at some point during the one minute measurement interval. This loss of synchronization is a result of extremely poor reception at Rx2 as a result of the careful placing of this antenna to induce such a condition. While trial 2 is more of a canonical example than a typical wireless situation it meets the objectives of validating our receiver architecture and providing an insight into practical multi-antenna propagation issues.

CHAPTER 6

Conclusions and Future Work

In this thesis, we presented the design of a low-complexity multi-antenna OFDM receiver and demonstrated this design on a real-time wireless MIMO network. We have investigated several topics which include algorithmic complexity reduction, fixed-point design, Verilog implementation, and rapid-prototyping of a multi-antenna receiver. We considered a generalized complexity computation for LMMSE and ZF detection and structured these algorithms using the DIFMAD algorithm thereby completely removing divisions from their computation. Doing so, we were able to reduce the hardware complexity for these detection schemes. Furthermore, we presented a novel folded-pipelined architecture which was simulated for floating-point and finite-precision implementations and demonstrated through real-time wireless measurements.

6.1 Summary

In Chapter 2 we reviewed key concepts from multi-antenna communication theory particularly pertaining to MIMO-OFDM systems and WLAN. Additionally, we presented system descriptions and assumptions used throughout this thesis and concluded with a description of the design methodology used for our research. Chapter 3 presented an overview of standard multi-antenna detection algorithms and developed a low-complexity reformulation of these standard algorithms based on a division-free modular

reformulation. This reformulation was developed to reduce the complexity of an ASIC implementation as a result of the computational requirements for detection in multi-antenna systems.

Chapter 4 presented a finite-precision implementation based on the algorithms developed in Chapter 3. It also presented floating-point and finite-precision simulation results validating this implementation. We discussed the rate-diversity tradeoff of MIMO systems in the context of our results and found the implementation loss of our finite-precision design to be well within our target specifications. Further results in this section indicated that an acceptable implementation loss can be maintained by using a complex dot-product bit-width of only 15-bits. Chapter 4 also presented a novel “folded-pipelined” architecture implemented in hardware which was based on our finite-precision implementation. ASIC gate counts were shown which clearly demonstrated the savings of a folded architecture over an unfolded approach.

Chapter 5 then served as validation of our design by describing our laboratory prototype and presenting results from actual real-time wireless measurements performed in a typical office environment and their comparison to simulations. Specifically, our prototype description focused on time-domain processing, FFT processing, channel estimation, the MIMO FPGA, and the FEC block with a major emphasis on the MIMO Core. We used the results obtained from our laboratory prototype as validation of our low-complexity multi-antenna receiver design. Additionally, we studied the rate-diversity tradeoff and presented a configuration where we demonstrated the utility of using the maximum number of receive antennas. We proved the feasibility of implementing MIMO-OFDM techniques into future 802.11 and other wireless

standardization efforts by demonstrating a low-complexity, high bandwidth, and reliable receiver architecture.

6.2 Future Work

We have presented the design, simulation, and demonstration of MIMO-OFDM receiver architectures. While our implementation was only taken to the prototyping stage it is likely that it will eventually be fully integrated into a mass-market ASIC. The challenge of developing an ASIC which incorporates MIMO-OFDM techniques into mainstream technology is an exciting one. The author looks forward to seeing the final product when it is ready for market deployment.

Our implementation assumed no channel state information. The incorporation of open-loop or closed-loop MIMO into our architecture could significantly increase system performance and reduce power consumption. In the end, MIMO systems incorporating loop-back methods will surely make their way into mainstream technology. Additionally, the use of singular value decomposition (SVD) could also significantly improve system performance. Exploration of this technology and its architectural implementation would be a useful area of focus for future research.

We further suggest the exploration of our architecture in higher dimensions. We would like to examine 4x4 systems and higher order systems using wider bandwidth allocation of 40MHz and wider. As the cost of transistors continues to decrease the production of higher order and consequently higher bandwidth multi-antenna networks becomes a cost-effective option. Along these lines we would like to tweak our current design further to reduce gate counts and power consumption. Additionally, we would like to explore the deployment of our prototype in a multi-node configuration studying the

communication across a wider variety of environments and orientations. By performing these additional wireless measurements we could achieve a better understanding of our system.

6.3 Concluding Remarks

Wireless communication is an enabling technology. It allows people to communicate more freely with others in a work or social setting. Each day the world breaks evermore free of wired limitations lured by the attraction of wireless operation. MIMO systems will continue this trend of enabling a more free flow of information by allowing for higher bandwidth, increased reach, and added robustness. As the demand for these appealing wireless properties grows it is essential that low-complexity architectures exist to allow for small form factor and power sensitive devices which are sure to increase in prevalence over the next decade and beyond. In this thesis we have provided a description of what these architectures might look like and demonstrated their functionality. In the coming years it is the hope of the author that these architectures will find their way into consumer products thus extending and improving the enabling technology that is wireless communication.

REFERENCES

-
- [1] J. H. Winters, "On the capacity of radio communications systems with diversity in Rayleigh fading environments" *IEEE J. Select. Areas Commun.*, vol. JSAC-5, pp871-878, June 1987.
 - [2] G. J. Foschini and M. J. Gans, "On limits of wireless communications in fading environments when using multiple antennas," *Wireless Pers. Commun.*, vol. 6, pp. 311-335, 1998.
 - [3] IEEE 802.11a standard, ISO/IEC 8802-11:1999/Amd 1:2000(E).
 - [4] IEEE 802.11g-2003 standard.
 - [5] Local and Metropolitan Area Networks—Part 16, Air Interface for Fixed Broadband wireless Access Systems, IEEE Standard IEEE 802.16a.
 - [6] L. J. Cimini, Jr. and Y. Li, "Orthogonal Frequency Division Multiplexing for Wireless Channels". Online Available http://users.ece.gatech.edu/~liye/ECE6602/ofdm_tutorial.pdf.
 - [7] A. Batra et al., "TI Physical Layer Proposal: Time-Frequency Interleaved OFDM," IEEE P802.15-03/141r3-TG3a, May 2003.
 - [8] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, 2003, unpublished.
 - [9] G. L. Stuber, J. R. Barry, S. W. McLaughlin, Y. (G.) Li, M. A. Ingram, and T. G. Pratt, "Broadband MIMO-OFDM Wireless Communications," Invited Paper, Proceedings of the IEEE, vol. 92, no. 2, pp. 271-294, February 2004.
 - [10] T. S. Rapaport, *Wireless Communications*, Prentice-Hall 2nd Edition 2002. pp 210-212.
 - [11] R. B. Ertel, P. Cardieri, K. W. Sowerby, T. S. Rappaport, J. H. Reed, "Overview of Spatial Channel Models for Antenna Array Communication Systems," *IEEE Pers. Commun.*, Feb. 1998.
 - [12] A. A. M. Saleh and R. A. Valenzuela, "A Statistical Model for Indoor Multipath Propagation", *IEEE Journal Commun.*, vol. SAC-5, No. 2., Feb 1987.

- [13] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. J.*, pp. 41-59, Autumn 1996.
- [14] G. J. Foschini, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Commun.*, Vol. 6. no. 3, pp. 311-335, Mar. 1998.
- [15] E. Telatar, "Capacity of multi-antenna Gaussian channels," *Europ. Trans. Telecomm.*, vol. 10, no. 6, pp. 585-595, Nov.-Dec. 1999.
- [16] A. F. Molisch, M. Steinbauer, M. Toeltsch, E. Bonek, and R. Thoma, "Capacity of MIMO systems based on measured wireless channels", *IEEE JSAC 20*, 561-569 (2002).
- [17] M. Chiani, M. Z. Win, A. Zanella, "On the Capacity of Spatially Correlated MIMO Rayleigh Fading Channels," *Proc. IEEE Trans. Info. Theory*, Vol. 49, Issue 10., pp 2363-2371, October 2003.
- [18] C. Chang, K. Kuusilinna, B. Richards, A. Chen, N. Chan, R. W. Brodersen, "*Rapid Design and Analysis of Communication Systems Using the BEE Hardware Emulation Environment*," *Proc. IEEE Rapid System Prototyping Workshop*, June 2003.
- [19] J. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*. Prentice-Hall, 2003.
- [20] Samir Palnitkar, *Verilog HDL*, Pearson Education (2nd Edition) 2003.
- [21] Donald Thomas, Philip Moorby, *The Verilog hardware Description Language*, 5th Edition, Kluwer Academic Publishers. 2003.
- [22] N. Zhang, B. Haller, and R. W. Brodersen, "Systematic architecture exploration for implementing interference suppression techniques in wireless receivers," *Proc. IEEE Workshop on Signal Processing Systems*, LA. October 2000.
- [23] C. Shi and R. W. Brodersen "*An Automated Floating-point to Fixed-point Conversion Methodology*," ICASSP Presentation 2003.
- [24] U. Madhow, M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA", *IEEE Trans. Commun.*, Vol. 42(12), pp. 3178-3188, December 1994 .
- [25] J. G. Proakis, *Digital Communications*, McGraw-Hill 4th Edition 2001. pp 242-247.
- [26] J. Barry, *MIMO Communications*, Unpublished.2003.

- [27] D. Waters, and J. Barry, "Noise-Predictive Decision-Feedback Detection for Multiple-Input Multiple-Output Channels," submitted to IEEE Transactions on Signal Processing, July 2003.
- [28] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD. Johns Hopkins, p. 51, 1996.
- [29] B. Hassibi, "A Fast Square-root implementation for BLAST," *Conference Record of the Thirty-Fourth Asilomar Conference on Signals, System and Computers*, 2000, pages 1255-59.
- [30] P. J. Black and T. H. Meng, "A 140-Mb/s, 32-State, Radix-4 Viterbi Decoder," *IEEE Journal of Solid-State Circuits*, vol. 27, no. 12, pp. 1877-1885, Dec. 1992.
- [31] S. Lee, M. Goel, and N. Shanbhag, "High-Speed FEC FPGA Design for MIMO-OFDM based WLAN modems".
- [32] G. Fettweis and H. Meyr, "High-Speed Parallel Viterbi Decoding: Algorithm and VLSI-Architecture" *IEEE Comm. Magazine*, pp. 46-55, May 1991.

David L. Milliner was born in New Orleans, Louisiana in 1981. He received the S. B. degree in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology (MIT) in 2003. He is pursuing the M.Eng degree in Electrical Engineering and Computer Science at MIT for which this thesis is a partial requirement. David was a member of the VI-A Program where he has worked jointly with MIT and Texas Instruments from 2001-2004 on digital audio signal processing (Digital Audio Branch) and wireless broadband architectures (Communications Systems Laboratory) projects in the DSPS R&D Center. His general research interests include broadband wireless communication systems, baseband architectures, MIMO-OFDM networks, ASIC design, and systems-on-a-Chip (SoC). David holds a pending patent on simplification of LMMSE detection.

David was awarded the Bell Northern Undergraduate Research Award in 2001 for best Digital Laboratory Project by the Massachusetts Institute of Technology after which he went on to serve as a teaching assistant for this class. David is a Student Member of the IEEE (S'04).