Reduction of Implementation Complexity in MIMO-OFDM Decoding for V-BLAST Architecture

by

Tariq Nanji

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2010

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This dissertation documents alternative designs of the Zero Forcing decoding algorithm with Successive Interference Cancellation (ZF-SIC) for use in Vertical Bell Laboratories Layered Space Time Architecture (V-BLAST) Multiple Input Multiple Output (MIMO) Orthogonal Frequency Division Multiplexing (OFDM) systems, in an effort to reduce the computational complexity of the receiver. The development of a wireless platform utilizing this architecture intended for use in an indoor wireless multipath environment was created to analyze the multipath environment. This implementation is the result of efforts from several individuals within the CST group. My contributions are documented in this dissertation.

In order to obtain channel state information (CSI), a training sequence is sent with each incoming frame. A pseudo-inverse operation is performed on the channel matrix and applied to each OFDM symbol that was received. Performing this operation on each tone and across each OFDM symbol is computationally inefficient in a MIMO configuration. If the number of pseudo-inverses can be reduced while maintaining acceptable levels of bit error, the processing time of each frame can be decreased.

Traditionally, tests of the performance of ZF-SIC have been conducted with simulations modelling a multipath channel. In this thesis, CSI is observed using an open loop platform developed for MIMO-OFDM communications. The rate of change of the channel is observed for different multipath environments. The proposed methods of decoding require modifications to ZF-SIC. The suggested changes are only applicable to a MIMO OFDM based method of data transmission. The most effective method of reducing decoding complexity and maintaining an acceptable number of bit errors was observed to occur in the time domain rather than in the frequency domain. For selecting frames and averaging frames in the time domain it was determined that the optimal number of OFDM symbols per frame is 1932 and 174, respectively.

Acknowledgements

I would like to express my gratitude and appreciation towards Professor Amir Khandani, who has provided his continuous support and insight into my research interests and activities. It has been an honour to work under his tutelage and be apart of a group that has exposure to technologies that are years ahead of industry. His efforts over the last few years have been greatly appreciated and valuable to my growth both personally and as his student.

I would also like to acknowledge, Raphael Hernandez, Ethan Zheng, Mohsen Baratvand, Arash Tabibiazar, Sandra Ramautarsingh, and Hamed Holisaz for sharing their knowledge and assisting with the development of the wireless platform.

I would like to thank my friends: Jaimal Soni, James Ho, Christie Kong, Jason Ho, and Alan Kuurstra for making my graduate experience memorable and being like family; Dan Weisser, Fahreen Velji, and Rajiv Tanna for finding the humour in any situation; Imran Premji, Sameer Jadavji, Mirabelle Nakhla, and Ratika Gandhi for keeping my head on straight and always being there for me.

Finally, and most importantly, I would like to thank my parents, Bashir and Yasmin Nanji for their lifelong support and continuous belief in me. Without their love and support I would not be where I am today.

To my parents, for their love, endless support, and encouragement.

Contents

Appendicies

List of Tables

List of Figures

Chapter 1

Introduction

Increased data rates in wireless communications systems are becoming a requirement facilitated by larger populations of people joining the new revolution of wireless devices. This demand dictates the use of large bandwidths of information transfer. There are many methods which have been used to significantly reduce receiver complexity in broadband wireless systems, the most notable being OFDM. Standards employing an OFDM based physical layer include IEEE 802.11a/g wireless local area network (WLAN) standard, the IEEE 802.16 fixed broadband wireless access (BWA) standard, and the European digital audio and video broadcasting standards DAB and DVB-T, respectively [7].

In an effort to improve spectral efficiency in high data rate systems, MIMO wireless systems have been receiving a great deal of attention. This technology employs multiple antennas at both the transmitting and receiving side of the radio link. Combining both strategies, MIMO-OFDM has been, among others, under consideration for use in the IEEE 802.11n high-throughput working group. This group aims to extend data rates of the IEEE 802.11a/g based WLANs past the observable maximum of 54 Mbits/s [8].

Superior data rates can be achieved using spatial multiplexing in environments that support multipath. In the case of multipath frequency selective fading channels, much like an indoor office environment, the V-BLAST architecture can be combined with OFDM to achieve high data rate transmission [2]. The conventional method of detection in the V-BLAST architecture uses an ordered serial Zero Forcing (ZF) nulling technique with Successive Interference Cancellation (SIC). To improve the detection process an optimal order of symbol decoding is determined which maximizes the minimum post-detection signal-to-noise ratio (SNR) of all data streams [11].

1.1 Addressing Implementation Complexity

In a MIMO spatial multiplexing system, the maximum diversity gain is determined by the number of receiving antennas n_r [21]. This diversity is only available if all channel coefficients are independent. Each data symbol is transmitted over n_r independent fading channels and the larger n_r is, the smaller the likelihood is of all channels fading simultaneously. Coupled with OFDM, the number of paths that are available is increased and as a result the reliability of detection can be improved. A fundamental issue with using an architecture that merges MIMO and OFDM together lies in the receiver complexity, more specifically, in the detection process. There are many types of detection methods for this type of architecture: Maximum Likelihood Detection (MLD), ZF, ZF-SIC, Minimum Mean Square Error (MMSE) Equalization, Fincke-Phost Sphere Decoding, and Sphere Projection are among the most commonly suggested [21]. This thesis focuses on the ZF-SIC detection algorithm due to its known reduced implementation complexity and its suggested use in the V-BLAST architecture [2]. The ZF-SIC algorithm uses ordered detection to improve its ability to decode incoming data correctly. As data is spread across frequency, time, and space in MIMO OFDM systems, the level of decoding complexity greatly increases. Each frequency tone has a channel matrix associated to it resulting in each tone having to negate the channel effects. This increased level of complexity severly slows down the processing time at the receiver in which the most involved operation is calculating the pseudo-inverse. Reducing the number of inverse operations that occur can effectively reduce this time and result in faster processing times at the receiver.

1.2 Thesis Contributions

The proposed alternate framework of decoding tackles the issues outlined in the previous section. A known concern in wireless communications is that simulations have been the means by which wireless systems have been made operable. With MIMO communications, it is uncertain as to the true nature of the channel and educated approximations have been made as to how the multipath environment behaves. The implemented system presented in this thesis allows for real-time measurements to be conducted in a multipath environment. As a result, the true nature of this channel can be seen in the CSI provided by each received frame. Rather than using simulations to obtain performance measures, real data tested in a multipath environment can be used. With obtainable CSI, improvements can be made to MIMO decoding algorithms. A detailed analysis is provided regarding the ZF-SIC decoding algorithm and ways in which knowing how the channel behaves in a multipath environment can assist in reducing the processing time of detection. Analysis was conducted in the time domain and in the frequency domain given that CSI can be obtained over both parameters. Through this analysis it was determined that reducing the number of inversions that occur in the frequency domain is not worth the loss in bit error performance that results in performing this operation. Additionally, in the time domain, it was shown that the CSI can be maintained for a series of ten frames of four OFDM symbols being sent 200 μs apart from one another. If OFDM symbols were to fill the gaps between frames, and assuming phase tracking was accounted for, over 1900 OFDM symbols can be sent that can share the same CSI. The propsed analysis on the ZF-SIC detection algorithm can greatly reduce the complexity of the receiver through reducing the overall number of pseudo inverse operations. Through extensive analysis, results presenting the bit error rates (BER) of different schemes of inversion reduction over varied SNR values are presented as a means of verification.

This thesis is organized as follows. Chapter 2 provides background information on multiple antenna wireless communications. In Chapter 3, the system implementation of the developed framework is presented. The ZF-SIC algorithm, measurement procedures, and potential improvements are given in Chapter 4. Chapter 5 provides the conclusive remarks which summarize the contributions of this research.

Chapter 2

Background

This chapter outlines the proposed framework of MIMO decoding by touching on the technology and infrastructure required to understand the underlying concepts. The developed technology uses these fundamental principles in its implementation and the proposed algorithms can be applied towards reducing the computational complexity of MIMO OFDM detection. There are two domains in which the computational complexity can be reduced: the frequency domain and the time domain. To evaluate the true effectiveness of altering decoding algorithms in either case, a realizable MIMO channel is required.

2.1 Multiple Input Multiple Output

MIMO can be categorized into three different areas: precoding, spatial multiplexing, and diversity coding. Precoding refers to the spatial processing that is performed at the transmitter (also known as beamforming). The goal of spatial processing is to reduce the effect of multipath fading from constructive interference of the signals being transmitted. Multipath is a propagation phenomenon that is characterized by the arrival of multiple versions of the same signal from different locations shifted in time due to having taken different transmission paths of varying lengths [13]. Precoding is used when a receiver consists of more than one antenna and beamforming cannot maximize the signal level across all receiving antennas. In addition, precoding requires that the transmitter has knowledge of the CSI. Spatial multiplexing gain refers to using the degrees of freedom in a communication system by sending independent symbols in parallel over multiple spatial channels [10]. This technique is a powerful one for increasing channel capacity for higher SNRs and can be used in either a closed loop system or an open loop system. Diversity coding requires a single stream of data to be encoded with a space-time code prior to transmission. Space-time codes create orthogonality among the data being sent to the receiver. No CSI is required to be known at the transmitter to take advantage of this mechanism of MIMO. The diversity order of a MIMO transmission system over an independant and identically distributed (i.i.d.) Rayleigh channel with n_t transmit antennas and n_r receive antennas at high SNR is given by [10]:

$$n_r - n_t + 1; \tag{2.1}$$

Thus, the diversity order can be improved by increasing the number of receive antennas and can be degraded by increasing the number of transmit antennas [10]. However, increasing the number of transmit antennas increases spatial multiplexing gain so there is generally a trade off made in the design process.

There are different architectures used for MIMO communications that can be used when trying to take advantage of this technology. Bell Labs Layered Space Time Architecture (BLAST), Per Antenna Rate Control (PARC), and Selective Per Antenna Rate Control (SPARC) are among the most notable [2]. Spatial multiplexing causes the complexity of the receiver to drastically increase and as a result techniques such as OFDM are used to handle the multipath channel. The basic MIMO channel model is shown in Fig. 2.1, below.

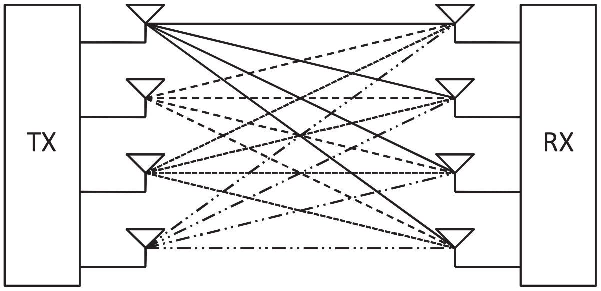


Figure 2.1: MIMO Channel Model

The transmitter sends several streams of data from multiple antennas. The number of possible paths that the channel will allow is $n_t n_r$ where n_t is the number of transmit antennas and n_r is the number of receiver antennas. It is the receivers job to decode the information received and output the original data set. A MIMO system can be modelled as [4]:

$$y = \mathbf{H}x + n \tag{2.2}$$

where y is the received vector, x is the transmitted vector, n is the noise vector, and H is the channel matrix. For the case of a 4 x 4 MIMO system, the general form of H is given by the following:

$$\mathbf{H} = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & h_{1,4} \\ h_{2,1} & h_{2,2} & h_{2,3} & h_{2,4} \\ h_{3,1} & h_{3,2} & h_{3,3} & h_{3,4} \\ h_{4,1} & h_{4,2} & h_{4,3} & h_{4,4} \end{bmatrix} \tag{2.3}$$

where h_{ij} represents the CSI received from transmitter antenna i to receiver antenna j.

2.2 Orthogonal Frequency Division Multiplexing

Frequency Division Multiplexing (FDM) uses multiple subcarriers within the same channel. The total data rate sent across the channel is divided among these subcarriers. The main advantage of FDM over single carrier modulation is that narrowband frequency interference will only affect particular subcarriers rather than the entire carrier. Due to the lowered information rate on each subcarrier, the symbol periods will be larger and this will add further protection against noise and reflections [13]. FDM usually requires guard intervals between subcarriers to combat Inter Carrier Interference (ICI).

Orthogonal Frequency Division Multiplexing (OFDM) requires all subcarrier frequencies to be orthogonal to one another. In other words, from tone to tone there is no cross over, or ICI, and they can be considered uncorrelated from one another. This creates a higher level of spectral efficiency as there is no need for a guard interval. Each of these subcarriers will contain data that is being sent from the transmitter. If the transmitter and receiver are not frequency synchronized, the received tones will no longer be orthogonal to one another and ICI will result. In the case of a multipath environment, this synchronization is even more of a concern as reflections will appear at different frequency offsets. A frequency selective channel is considered to be constant over each OFDM subcarrier if each tone is sufficiently narrow banded [4]. Effectively, OFDM converts a frequency selective channel into a series of flat fading sub-channels [12]. Fig. 2.2, illustrates the separation of one carrier into multiple subcarriers.

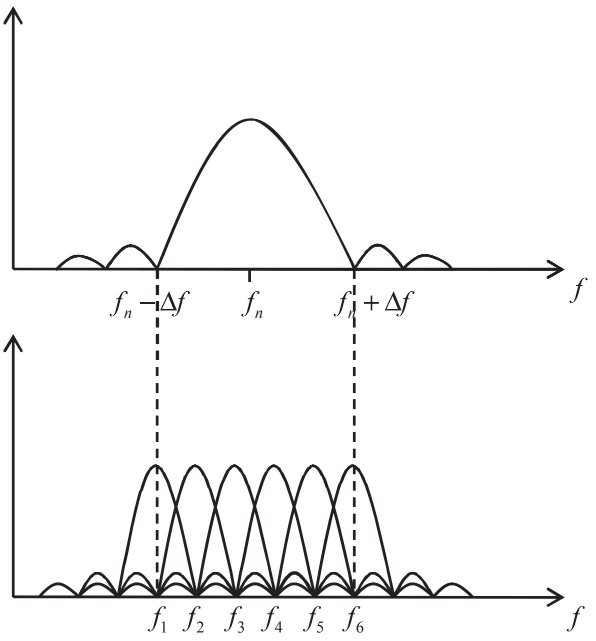


Figure 2.2: OFDM Subcarrier Splice

The interference of subcarriers on one another is zero if the subcarrier frequencies are perfectly orthogonal. As shown in Fig. 2.2, the side lobes of each subcarrier cancel to zero at the peaks of all other subcarriers and do not cause ICI.

Fig. 2.3, illustrates a simple block diagram of an OFDM system. An OFDM symbol is the sum of a number of orthogonal subcarriers. Each sub-carrier contains its own baseband data. In general, the transmitter and receiver architectures of a system utilizing OFDM use an Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT) block, respectively, to convert the frequency tones into a time domain signal to be sent over the air or vice versa. At the transmitter, this transform maps the input signal to a set of orthogonal subcarriers, or basis functions, of the FFT. At the receiver, the signals are then combined to form an estimate of the originally transmitted signal. Since the basis functions of the FFT block are uncorrelated, the correlation calculation performed in the FFT block will only see the energy for that particular subcarrier.

The input to the IFFT block maps each bit of data to a set of orthogonal sinusoidal

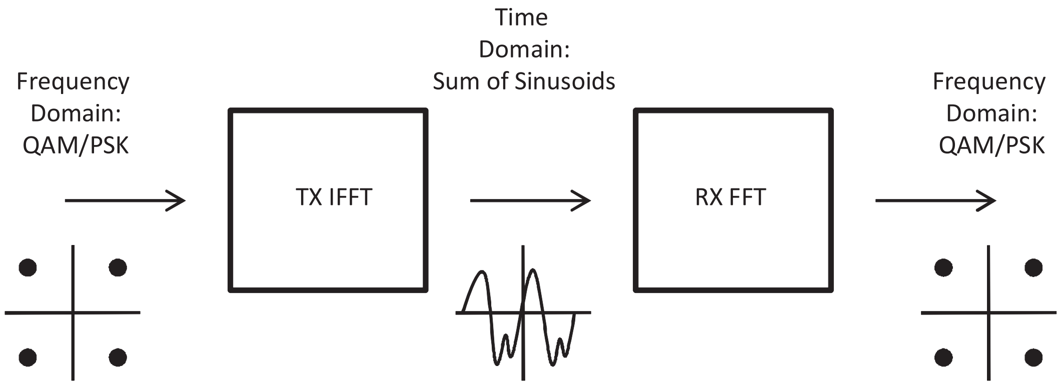

Frequency Domain: QAM/PSK
TX IFFT
Time Domain: Sum of Sinusoids
RX FFT
Frequency Domain: QAM/PSK

Figure 2.3: OFDM System Model

signals and these sinusoidal signals are summed together to be sent over the channel. The FFT block accepts this sum of sinusoids and separates the signal into its orthogonal subset of frequency domain bits. Each of these bits corresponds to a particular modulated signal mapping such as BPSK, QPSK, 16-QAM, or 64-QAM.

2.3 MIMO OFDM

Since it has been observed that wideband wireless channels are frequency selective in nature [12], OFDM can be used in a MIMO system to take advantage of this. MIMO takes advantage of the spatial and temporal dimensions of transmission and adding OFDM into this framework includes the frequency dimension. The general structure for a MIMO OFDM transmitter and receiver is outlined in [12].

As previously explained, a multipath channel is one in which multiple versions of the same transmitted signal arrive at the receiver delayed in time from one another. This type of channel will pose two main problems for a system using OFDM, Inter Symbol Interference (ISI) and ICI [13]. In single carrier systems the ISI is only due to the previous symbol, however, in a multipath environment the ISI can be due to several other symbols. ICI is the interference experienced by an OFDM′s symbols subcarriers. To minimize ISI, a guard interval of sufficient length is generally employed between OFDM symbols, however, a guard interval does not prevent ICI from occurring. To combat ICI, the guard interval is replaced by a cyclic prefix of the last L samples of the OFDM symbol. In effect, the cyclic prefix behaves much like a guard interval in preventing ISI and similarly makes the OFDM symbol appear periodic to the receiver which assists in reducing ICI in the convolution operation of the FFT block in the receiver.

2.4 V-BLAST

It has been shown in [1,2,3,4] that if a wireless channel is rich in multipath scattering it is capable of producing large capacities with a relatively low number of bit errors. Various techniques have been developed to take advantage of this property including BLAST. This architecture takes advantage of Space Division Multiplexing (SDM) or Space Division Multiple Access (SDMA). SDM is inherent to a MIMO system because multiple antennas are being used to transmit data across the wireless channel at the same frequency. To ensure error-free decoding is possible multiple receive antennas are required, again, an inherent property of a MIMO system.

There are two common encoding methods for MIMO spatial multiplexing, horizontal and vertical encoding[5]. In horizontal encoding each data stream is independently encoded and transmitted by different antennas. Vertical encoding uses a single encoder to spread information across all antennas. V-BLAST utilizes the horizontal encoding method. The term "vertical", as mentioned, does not refer to the encoding method used but it refers to the method in which the detection at the receiver is performed. The V-BLAST transmission mechanism is shown in Fig. 2.4.

As seen in Fig. 2.4, when the transmitted data stream is received, the initial received vector will consist of $y_i(0)$ where $i = 1, 2, 3, 4$, thus being vertically received. The receivers, due to the effect of multipath, will receive the signals radiated from all n_t transmit antennas. In comparison to other multiple access techniques, such as Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), and Code Division Multiple Access (CDMA), V-BLAST ensures that the entire bandwidth of the system is used all the time by all transmitting antennas, each transmitted signal occupies the entire bandwidth, and the total bandwidth utilized is only a small fraction in excess of the symbol rate [5]. To achieve a sufficient level of de-correlation at the receiver, V-BLAST requires its

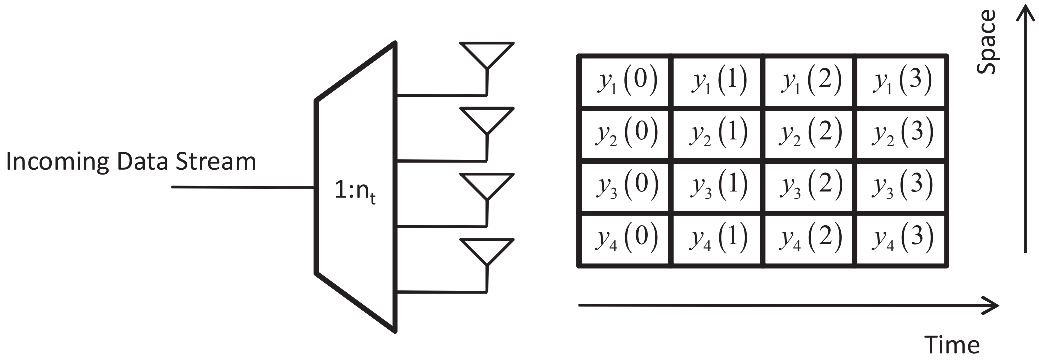

Incoming Data Stream
1:n$_t$
$y_1(0)$ $y_1(1)$ $y_1(2)$ $y_1(3)$
$y_2(0)$ $y_2(1)$ $y_2(2)$ $y_2(3)$
$y_3(0)$ $y_3(1)$ $y_3(2)$ $y_3(3)$
$y_4(0)$ $y_4(1)$ $y_4(2)$ $y_4(3)$
Space
Time

Figure 2.4: VBLAST Transmission Mechanism

operation to be conducted in a multipath environment. These conditions allow the receiver to distinguish signals occupying the same channel space. V-BLAST and OFDM can be combined in frequency selective fading channels to achieve a high rate of data transmission.

2.5 Decoding Methods

The research in detection algorithms for MIMO systems have focused on attempting to achieve optimal error performance while remaining practical for implementation purposes. Various decoding algorithms have been proposed for MIMO systems such as ZF, ZF-SIC, and MMSE, however, when OFDM is used as part of the architecture each algorithm must be repeatedly applied to each frequency tone. The general premise for each decoding method is to develop a means to counter the effects of the channel given that the received signal can contain delayed versions of the same signal due to multipath. Optimizing this detection process with real measurements of the CSI can greatly reduce the complexity of decoding which OFDM introduces into a MIMO system.

2.6 Related Research

A modified decoding algorithm was proposed in [11,17] that built on the ZF technique and set out to reduce the number of channel inversions performed in V-BLAST OFDM detection. Instead of performing a channel inversion on each tone of an OFDM symbol, the correlation between different subcarriers was explored to reduce complexity. A threshold correlation was defined in which subcarriers would be partitioned based on whether or not they were above such a boundary. It was suggested that in each group, conventional V-BLAST detection be performed on the center subcarrier and the detection order be recorded. Following the recorded order, other subcarriers in the same group would be detected using a QR decomposition technique defined in [17]. This study showed minimal performance degradation in terms of Bit Error Rate (BER), however, only provided simulations which made assumptions regarding the channel model to test their theory.

An alternative approach to reducing complexity at the receiver was presented in [7] in which the inversion calculation itself was modified. This study approached the inversion process by looking at the operation in terms of its adjoint and determinant. Separate interpolation schemes were used on each of these components in an effort to determine the CSI for different tones. This study merely presented the idea and did not provide any results in terms of BER performance. However, it was suggested that this modified operation would yield faster processing times.

BER performance analysis of MIMO OFDM systems were performed in [5,9,18-20]. These studies created their own MIMO channel model and conducted simulations on this model to develop presentable BER results. This study was limited by their inability to obtain real CSI in an effort to provide accurate BER performance measures.

All previous studies have modeled a MIMO channel and have only provided simulations in an attempt to reduce the computational complexity of decoding algorithms. Off air testing of a wireless platform has not been conducted to evaluate the true nature of a MIMO channel in order to verify the effectiveness of these results. Having the ability to evaluate a real channel will provide a means of conducting analysis not only in the frequency domain, but in the time domain to reduce decoding complexity.

Chapter 3

System Implementation

To fully understand the true nature of a wireless channel one must be able to view how the channel changes over time and frequency in real time. In previous work [5,9,11,17-20], simulations of a multipath channel are used to prove theories and develop a means of producing a measure by which improvements can be compared to, such as BER or capacity. Without transmitting data over a multipath environment and observing its effects, the true nature of such claims cannot be soundly established. In this dissertation, a platform was developed which transmits data using the V-BLAST MIMO OFDM architecture over a multipath channel. Real-time CSI is achieved at the receiver and bit error rates are determined from this data. Any portion of the baseband design can be viewed in real time with the framework that was developed for capturing data. The system consists of a series of Field Programmable Gate Arrays (FPGA) and Digital Signal Processors (DSP). This implementation is the result of efforts from several individuals within the Coding and Signal Transmission (CST) group.

3.1 Architecture Overview

A detailed overview of the overall hardware architecture is provided in Fig. A.1. This figure applies for both the transmitter and receiver. A simplified structure of the hardware is provided in Fig. 3.1.

VHS and SignalMaster Quad (SMQ) boards are connected together via a cPCI bus and the VHS board is connected to the Radio Frequency Front End (RFFE) by means of a

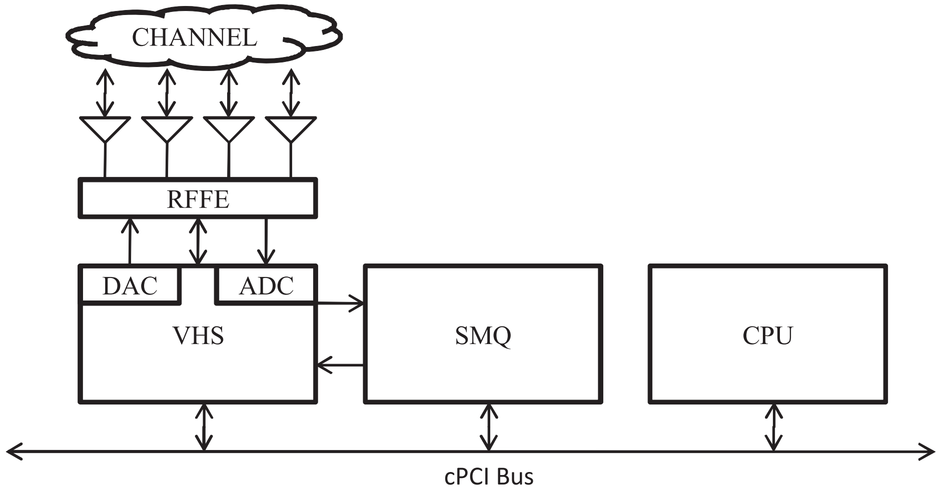


Figure 3.1: Hardware Architecture Layout

Digital to Analog Converter (DAC) and an Analog to Digital Converter (ADC). The VHS board houses a FPGA and the SMQ board contains a mixture of FPGA and DSP units. The RFFE connects to four antennas that are used to either transmit data or receive data. A CPU is connected to the cPCI Bus to have a means to set parameters associated with the VHS and SMQ boards.

3.2 Hardware Configuration

The VHS board contains a Virtex-4 XC4VLX160 FPGA, 128 MB SDRAM block, DAC, and an ADC. The Virtex-4 FPGA has some of the highest performance and density, some of the greatest memory capacity, and consumes only half the power needed by many other FPGA families [32]. The VHS ADC/DAC has two 64 MB SDRAM blocks for data buffering on the FPGA board. Any data that needs to be captured for post-processing is stored on these memory banks. The VHS ADC houses eight single-channel, 14 bit ADC input channels, with a maximum sample rate of 105 MSPS [32]. Eight inputs are required for a 4 x 4 MIMO system as each of the four receiving antennas require an in-phase and quadrature component. The number of input channels can be expanded to 16 to support eight receiver antennas by stacking two ADC boards on top of each other. The VHS DAC has four dual-channel, 14 bit DAC output channels with a maximum guaranteed sample rate of 480 MSPS [32]. Each channel supports one transmit antenna thus being sufficient for a 4 x 4 MIMO system. The number of channels can be expanded to eight by stacking two DAC boards together. The gain control of the output channel is fully independent with a dynamic range of 18.7 dB [32].

The SMQ board is constructed around two sets of Xilinx Virtex-4 LX FPGAs capable of 48 GMACS of processing power and four Texas Instruments C6416 DSP's each of which is capable of delivering up to 16000 MIPS/MMACS of processing power [31]. Each DSP block and FPGA has a 128 MB SDRAM memory module associated with it to buffer data for offline processing [31]. The two FPGA's on the SMQ board have RapidCHANNEL connections allowing for full duplex communication. There are four cPCI connectors on the SMQ board which allow for integration into the cPCI chasis. Additionally, there are Lyrtech Input/Output (LYRIO+) expansion connectors which allow for external modules to be connected to the SMQ board. Some of these external modules include high speed multichannel ADC/DAC platforms as well as additional memory storage of up to 2 GB. The FPGA contains registers which are used to interface the different chips of the SMQ. These registers must be accessed through the DSP.

RapidCHANNEL ports allow high bandwidth data transfers of up to 8 Gbps between FPGA components [32]. When these connections are used to interface FPGAs on different platforms (i.e. VHS to SMQ), one cable is necessary for the RapidCHANNEL port, however, to allow for full duplex transmission two cables are required. On the VHS ADC/DAC, this port is used as a communication channel between boards that are installed on the same cPCI bus [32]. As shown in Fig. A.1, the VHS and SMQ boards are connected together on the cPCI bus and require a RapidCHANNEL connection between the two.

The RFFE is comprised of the Comlab Dual Band Radio Frequency Transceiver and L-Com omnidirectional antennas. The Comblab Transceiver is based on the MAX2829 single chip RF transceiver Integrated Circuit (IC) which is designed for OFDM 802.11 WLAN applications. The transceiver covers the frequencies of 2.4 GHz to 2.5 GHz and 4.9 GHz to 5.875 GHz and supports 20 MHz and 40 MHz data bandwidths [28]. The primary purpose of this chip is to upconvert the baseband frequency to an appropriate transmission frequency (i.e. from 20 MHz baseband to 2.5 GHz on air). The MAX2829 integrates all the circuitry required to implement the RF transceiver function, providing a fully integrated

Zero IF receive path, transmit path, voltage controlled oscillator, frequency synthesizer, and baseband/control serial interface [29]. The power amplifiers, RF switches, RF diplexers/bandpass filters and baluns are external to the IC. The transceiver unit comes in a 2U rack mount case with all RF, Baseband, and Control connections located on the front panel [28]. Fig. 3.2, below, shows the rack mount unit of the Transceiver.

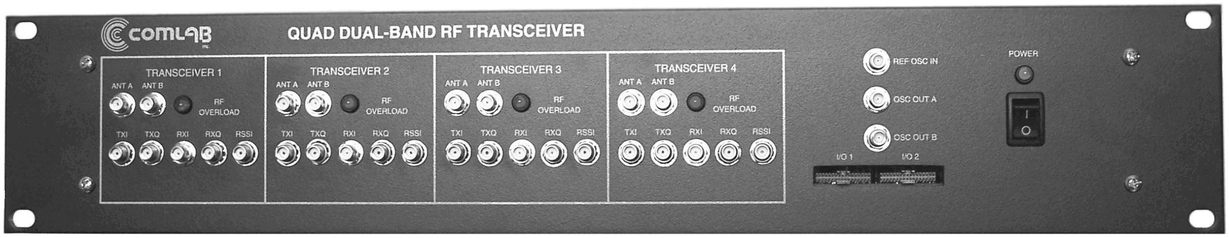


Figure 3.2: Quad Dual-Band RF Transeiver Front Panel Connections [28]

A 12 V/2.5 A power adapter is also powering each unit. The gain is set by the MAX2829 chip and is associated to a tap setting. Fig. 3.3, illustrates the correlation between the actual transmit gain and the tap setting associated with it. The linear relationship between these two properties indicates that it is synonymous to refer to the output power by the gain setting.

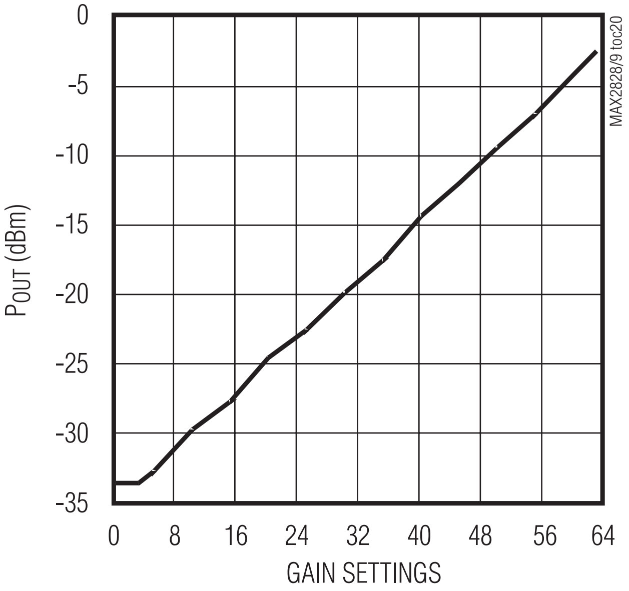


Figure 3.3: TX Output Power vs. Gain Settings [29]

These settings are important when taking measurements at different power levels. Due to uniformity of the random noise at the receiver, the SNR can be controlled by changing the transmit gain of the signal. There are a total of 64 possible TX gain settings that relate to 64 discrete output powers. Each transmit path can be set to a different gain setting based on what is defined by the user. This is a useful feature when utilizing the Automatic Gain Control (AGC) on the baseband design to adjust RFFE gains.

The L-Com omnidirectional antennas are used at both the transmitter and the receiver. This antenna provides a 7 dBi gain and operates at the 2.4 GHz ISM Band for use in 802.11 wireless LAN systems [30]. The radiation pattern of the antenna is given in [30] and is shown to be horizontally omnidirectional. Utilizing an omnidirectional antenna helps in creating more paths for the signal to travel which promotes a multipath environment. If the antenna were more directive, the transmitted signals would be more inclined to travel in a particular direction rather then in all directions to create additional reflections at the receiver. Fig. 3.4, illustrates the node encompassing all described components that were used to obtain measurements.

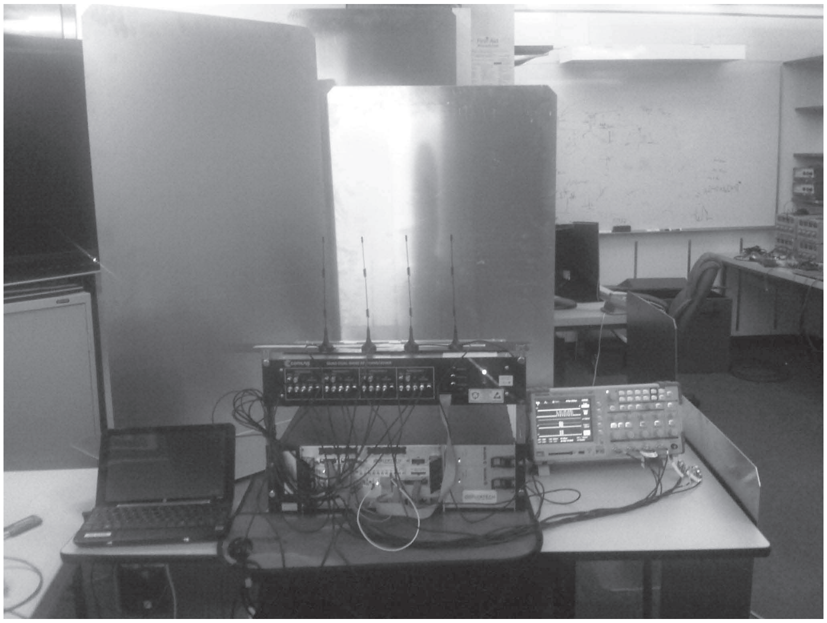


Figure 3.4: Transmission Node

3.3 Software Configuration

Development for both the FPGA and DSP units of the VHS and SMQ boards was conducted in MATLAB. A form of model based design was used to create the digital design of the baseband structure. Lyrtech, the parent company providing the VHS and SMQ boards, created a set of libraries for Simulink in MATLAB. These libraries contain digital design blocks that have Verilog code built into them. Simulink can synthesize any design with the use of a System Generator application provided by Xilinx and create a bit file that can be use to configure the FPGA boards. To operate the DSP, code can be written in the form of MATLAB m-code and this code can be stored and executed in the DSP by means of a System Generator specific to the DSP, again provided by Xilinx.

3.4 Baseband Design

The baseband design of the 4 x 4 V-BLAST MIMO OFDM system implemented in this study was a multi-platform endeavour which encompassed multiple FPGA and DSP systems. The transmitter (TX) design was developed entirely on the VHS FPGA platform. The receiver (RX) design was developed on a mixture of VHS and SMQ FPGA boards as well as a SMQ DSP chip. The baseband design does not encompass the frequency up-conversion required to transmit over the 2.5 GHz channel, it simply passes a 20 MHz time domain signal to an RF transceiver. More specifically, this process is performed by the Quad Band Dual RF Transceiver described in Section 3.2.

3.4.1 TX

A generalized overview of the transmitter baseband design is shown in Fig. 3.5. The main components of this design include a start pulse, data generator, IFFT block, Space Time Block Code (STBC) insertion, Short Training Sequence (STS) and Long Training Sequence (LTS) insertion, and a DAC inherent to the VHS FPGA board. The DAC sends its analog signals to the RFFE which up-converts the signal to 2.5 GHz.

For this study, pseudo-random data was constructed in the Data Generation portion of the TX design. Randomized data was modulated with Binary Phase Shift Keying (BPSK)

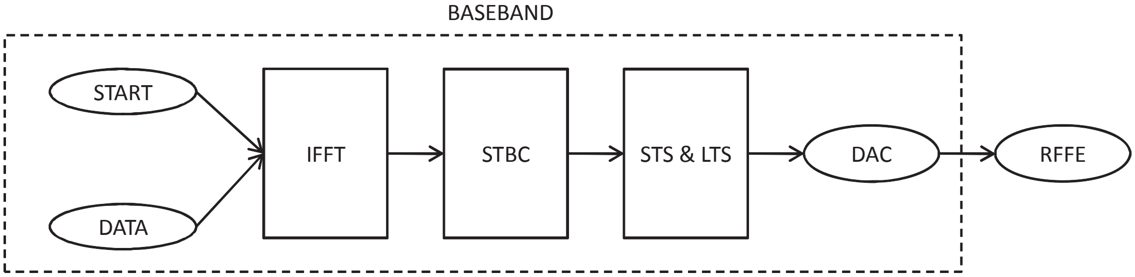


Figure 3.5: Simplified TX Baseband Design

and each bit created was assigned to a frequency tone to generate an OFDM symbol. Each OFDM symbol contains 52 tones of data. If utilizing Phase Tracking at the receiver, Pilot symbols can be inserted into the data sequence. In the developed implementation, it was decided to use four OFDM symbols in each frame of data. It was determined that using four OFDM symbols did not require the use of Phase Tracking and as a result, eliminated one additional source of outside interference. Prior to sending the data to the RFFE, cyclic prefixes are pre-pended in the time domain. In doing so, the convolution of the data with the channel becomes a cyclic convolution and frequency domain equalization is possible with only one multiplication per data symbol [26]. This adds enhanced robustness to non-linear distortion and phase noise.

The generated data is then passed to an IFFT block in which the OFDM symbols containing frequency domain data are converted to the time domain. The IFFT is a discrete fourier transform block implemented by Xilinx for use with the VHS FPGA board. The general operation performed by this block is given by:

$$x\left(j\right) = \left(\frac{1}{N}\right) \sum_{k=1}^{N} X\left(k\right) w_N^{-(j-1)(k-1)} \tag{3.1}$$

$$w_N = e^{\frac{(-2\pi i)}{N}} \tag{3.2}$$

where $X(k)$ is the frequency domain data, N is the number of frequency tones being used, and j is the sequence in time.

A layered STBC block was implemented to allow the use of trellis space time coding schemes. Space time codes are commonly used in slow fading environments, typical of indoor transmission, to achieve lower bounds on outage capacity [23]. Schemes such as those presented by Alamouti [24] can be defined for use within the STBC block. To reduce the complexity of the receiver design on the FPGA, this block, although implemented, was not used in the study performed in this research. However, within this block a rate change occurs to adjust incoming data to four parallel streams for transmission over the channel. This adjustment changes the baseband frequency from 80 MHz to 20 MHz.

Once through the IFFT and STBC blocks, the STS is pre-pended to one of the data streams and the LTS is pre-pended to all data streams. The general structure of a frame of data in the time domain is shown in Fig. 3.6. This structure is adopted from the 802.11a standard [25] and modified for a MIMO system.

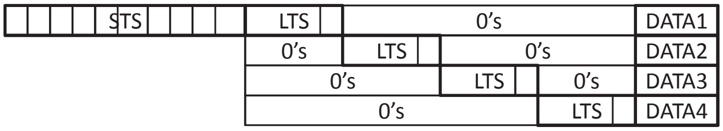


Figure 3.6: TX Frame Structure (Preamble and Data)

The STS consists of 10 frames of 15 samples of data. This length allows sufficient room for accurate time synchronization, diversity selection, and coarse frequency offset estimation at the receiver. There is no guard interval set up for the STS. The LTS consists of a 64 sample signal plus a 16 sample cyclic prefix. The LTS is sent diagonally in time such that no antennas are transmitting the LTS at the same time. This ensures that, due to multipath effects, the receiving antennas will obtain varied versions of the original LTS. At the receiver the LTS is used for fine frequency offset estimation and channel estimation. The entire preamble is 12 μs in length, 8 μs for the STS and 4 μs for the LTS.

Each OFDM symbol in the time domain is 4 μs in duration including a 0.8 μs guard interval. Each data symbol, in the frequency domain, consists of 52 subcarriers of data. If pilot symbols are used, there are 48 subcarriers of data with four pilot symbols inserted in place of data. Pilots are used for phase tracking at the receiver. Aside from the data and pilot symbols, 12 zeros are inserted at DC and in between data samples to help combat ICI.

The DAC connects the baseband design to the RFFE. The DAC Controller is implemented to have three configurations. The first configuration causes the output of the DAC to show the TX data being sent on air. The second configuration allows the DAC to output the RX off air data. The third configuration allows RX probe signals to be output. The latter two configurations are looped back from the RX baseband design.

3.4.2 RX

An overview of the RX baseband design is shown in Fig. 3.7. The primary components of this design include an ADC, Timing Synchronization, Rate Control, Frequency Offset Compensation (FOC), FFT, Channel Estimation, DSP Rate Control, and a ZF Decoding Block. The RX design uses two FPGA boards and a DSP chip in its implementation due to the complexity of the receiver.

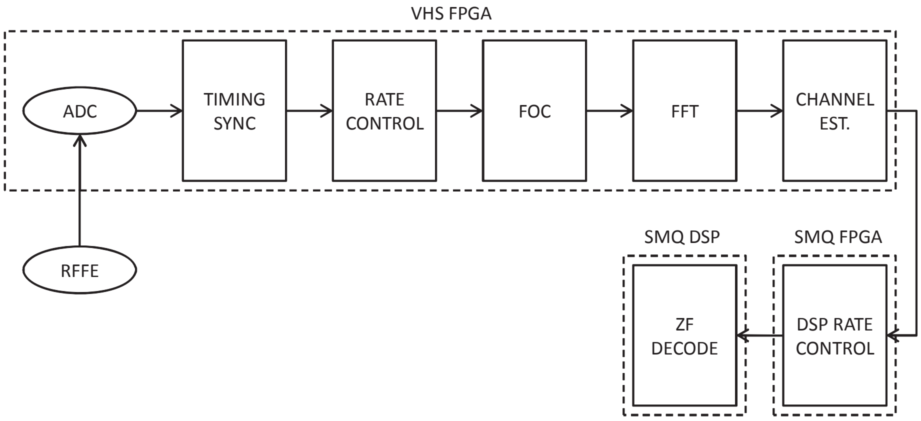


Figure 3.7: Simplified RX Baseband Design

The ADC converts the off air analog signals into digital signals that can be processed by the VHS FPGA board. The first component of the receiver is the Timing Synchronization block. For any digital communication system, synchronization is an essential task. The primary role of this block is to find an approximate estimate of the start of the preamble of an incoming data packet. Packet detection can be described as a binary hypothesis test. The test contains two complementary statements about a parameter of interest. These statements are called the null hypothesis, H_0, and the complementary hypothesis,H_1. For packet detection, the null hypothesis refers to a packet not being present and the complementary hypothesis refers to the packet being present. The actual decision, d_n, is a test on whether a decision variable surpasses a predefined threshold. This decision is given by the following:

$$H_0 : d_n < threshold \Rightarrow \text{Packet not present} \tag{3.3}$$

$$H_1 : d_n > threshold \Rightarrow \text{Packet present} \tag{3.4}$$

The delay and correlate algorithm [27] was used to determine the threshold used for acquisition. A sliding window which calculates the auto-correlation between the received signals STS and a delayed version of the received signals STS is used. The correlation calculation is given by [27]:

$$d_n = \sum_{k=0}^{L-1} r_{n+k} r^*_{n+k+D} \tag{3.5}$$

Fig. 3.8, illustrates how the auto-correlation described in Eq. 3.5 would develop and how an acquired signal is determined. I0 and Q0 represent the in-phase and quadrature components of the received signal containing the STS, LTS, and data on one antenna. Once the correlation operation produces a value above a threshold for 64 time instances, a start of frame (sof) pulse is sent to the rest of the system indicating a frame has been acquired. Coarse Frequency Offset (CFO) estimation is performed in the Time Synchronization portion of the design. The goal of CFO estimation is to roughly correct for the frequency offset introduced by receiver components from non-uniformity in frequency synchronization of the RFFE. This is achieved by using the STS.

Once a signal has been acquired, a rate change is required to successfully pass the data through the FFT block. The rate change converts the four parallel streams of incoming data into one serialized stream, effectively slowing the data down by a factor of four. To maintain the same timing, the rate must be decreased by a factor of four. The off air baseband data arrives at a rate of 80 MHz and is down converted to 20 MHz such that the FFT block only needs to handle one symbol at a given time instant. After the data has been serialized for suitable movement through the FFT block, Fine FOC (FFOC) is performed. FFOC attempts to improve the approximation of the CFO estimator by using the LTS to refine the frequency offset.

The generated data is then passed to an FFT block in which the time domain signals are converted to frequency domain OFDM symbols. The FFT is a discrete fourier transform block, much like the IFFT block on the TX design, that has been implemented by Xilinx

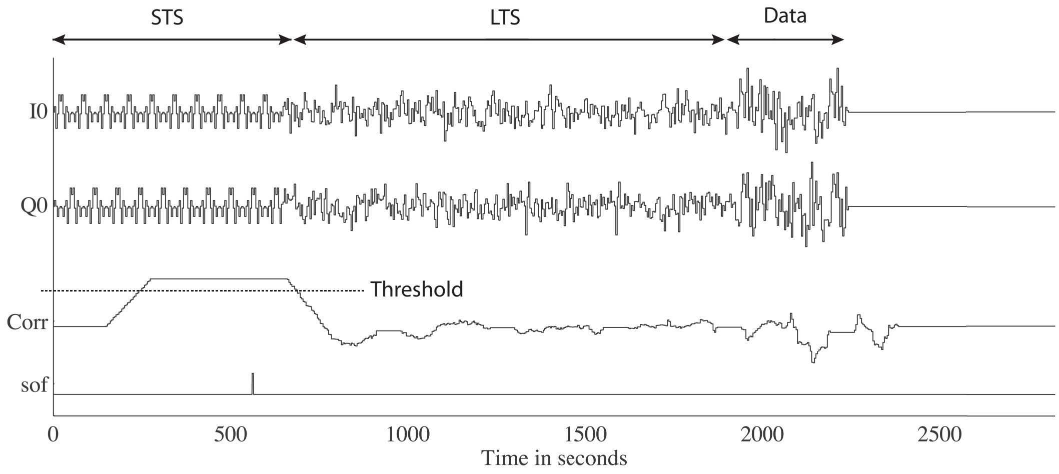


Figure 3.8: Time Synchronization

for use with the VHS FPGA board. The general operation performed by this block is given by:

$$X\left(k\right)=\sum_{j=1}^{N}x\left(j\right)w_{N}^{\left(j-1\right)\left(k-1\right)} \tag{3.6}$$

where w_N is given by Eq. 3.2.

After being converted into frequency domain OFDM symbols, channel estimation is conducted using the LTS to determine the coefficients for each path and each tone. For MIMO-OFDM systems, the received signal on each subcarrier is the superposition of multiple transmitted signals. The LTS portion of the preamble is included to facilitate accurate channel estimation. The frequency domain LTS is known at the receiver and consists of a series of zeroes, ones, and negative ones as presented in the IEEE 802.11a specification [25]. Multiplying the received LTS with a representation of what the LTS should be provides an accurate depiction of the channel for both the in phase and quadrature components [33]. The equations provided in [33] are summarized below in Eqs. 3.7-3.10.

$$\hat{y}_{i}^{j}=\begin{pmatrix} y_{i}^{j}\left(0\right) \\ \vdots \\ y_{i}^{j}\left(N_{c}-1\right) \end{pmatrix} \tag{3.7}$$

y_i^j represents the frequency domain version of the signal received on antenna j that corresponds to the training symbol sent from transmitter i and N_c represents the number of subcarriers. The received training symbol is then given by, Eq. 3.8, below:

$$\hat{y}_i^j = D \begin{pmatrix} h_{ij}(0) \\ \vdots \\ h_{ij}(N_c - 1) \end{pmatrix} + \hat{n}_i^j \tag{3.8}$$

where D is the diagonal matrix of training symbols on each of N_c subcarriers. The channel estimates can thus be given by, Eq. 3.9, below:

$$\hat{h}_{ij,est} = \begin{pmatrix} h_{ij,est}(0) \\ \vdots \\ h_{ij,est}(N_c - 1) \end{pmatrix} = D^{\dagger} \hat{y}_j^i \tag{3.9}$$

where $D^{\dagger}$ is the pseudo-inverse of D. Since D is diagonal and the training symbol specified by [25] contains amplitudes of 1, Eq. 3.9 can be represted as Eq. 3.10, below:

$$\hat{h}_{ij,est} = D^H \hat{y}_i^j \tag{3.10}$$

This indicates that by multiplying the received sequence by the known sequence the coefficients can be determined. Table 3.1, provides a simplified example for the formulation of channel coefficients. Note that only in-phase components are shown, but the same process is applied to quadrature components.

Table 3.1: Calculating Channel Coefficients

TONE	RX LTS	ORIGINAL LTS	COEFFICIENT
1	0.80	1	0.80
2	0.75	1	0.75
3	-0.79	-1	0.79
4	0.89	1	0.89
5	-0.99	-1	0.99
6	-0.95	-1	0.95

After the channel coefficients are known, the received data can be decoded and demodulated in an effort to regain the originally transmitted sequence. However, since the data moves from the VHS board to the SMQ board, as shown in Fig. 3.7, a buffer must be created on the SMQ FPGA board to ensure the DSP can process the data at an appropriate rate. Since the DSP is software based, it runs at a slower speed than the SMQ FPGA board and as a result the buffer located on the SMQ FPGA waits for the DSP to finish processing before sending data through from the VHS FPGA.

Once data has been received by the DSP block, it is run through the ZF-SIC algorithm. The purpose of this decoding block is to remove the effects of the MIMO channel in an efficient manner which takes advantage of the strongest signal detected by the four receiving antennas. More detail on this algorithm is provided in Chapter 4.

Alternatively, a secondary design was created in which data is not processed in real-time, rather it is stored and decoded offline. On the VHS and SMQ FPGA board there exists 128 MB SDRAM blocks as shown in Fig. A.1 in Appendix A. Each of these blocks can store data by tapping off any line on the receiver design. For instance, instead of sending data to the SMQ board, data can be sent to the SDRAM block off the VHS board and offline processing can be conducted. The advantage to using memory blocks to store data is that the data can be reused when testing different types of decoding methods.

Chapter 4

Study of Decoding Complexity

The primary focus for this study is on evaluating the performance of ZF-SIC and determining if knowing CSI at the receiver can reduce the number of inverse operations performed in MIMO OFDM systems. Reducing the overall number of inversions can significantly speed up processing time on the receiver and allow for quicker data transmission and recognition. Using the framework described in Chapter 3, captured data was used to see the effects of modifying the decoding algorithm based on observations made on the channel's behaviour.

4.1 Zero Forcing

ZF is a linear MIMO decoding technique in which the processing takes place at the receiver, so long as the channel matrix, **H**, is invertible. This technique requires **H** to be known only at the receiver to allow for coherent detection and not at the transmitter, which suffices for the V-BLAST open loop system implemented in this study. The transmitted vector is estimated by inverting the channel matrix and applying it to the received vector. This is given by:

$$y_{est} = \mathbf{H}^{-1}x \tag{4.1}$$

where x is the received signal $\mathbf{H}y + n$. The premise of the ZF technique utilizes linear combinatorial nulling [11]. Each stream is considered to be the desired signal and the remaining streams are considered as impeding the signal of interest. All interfering signals are nulled by linearly weighting the received signals in order to ensure that all obstructing terms are cancelled out. Weight vectors, in the case of ZF, can be chosen by selecting nulling vectors w^i, such that:

$$w^i h_j = \begin{cases} 0 & j \neq i \\ 1 & j = i \end{cases} \tag{4.2}$$

where h_j denotes the j-th column of $\mathbf{H}$. The linear processing in the receiver is represented by a matrix containing all n_t rows of nulling vectors which forces all impedances to zero such that y can be estimated. If $n_t \leq n_r$ this matrix is called the pseudo-inverse of H [2] and is represented by:

$$\mathbf{H}^{\dagger} = \left(\mathbf{H}^H \mathbf{H}\right)^{-1} \mathbf{H}^H \tag{4.3}$$

where $(\bullet)^H$ refers to complex conjugate transposition. If $n_t > n_r$, $\left(\mathbf{H}^H \mathbf{H}\right)^{-1}$ will have a determinant of 0, thus being singular, and the inverse will not exist. Additionally, ensuring this criteria is met will mean the channel matrix, $\mathbf{H}$, will have full column rank [9]. When the pseudo-inverse exists the estimated transmitted vector y_{est} can be given by:

$$y_{est} = \mathbf{H}^{\dagger} x = y + \left(\mathbf{H}^H \mathbf{H}\right)^{-1} \mathbf{H}^H n \tag{4.4}$$

As shown in Eq. 4.4, a large disadvantage of using ZF is that it has the potential for enhancing noise observed at the receiver. If it is assumed that the channel is static over one frame, the pseudo-inverse only needs to be calculated once per symbol in that frame. In the case of OFDM, the pseudo-inverse would need to be calculated for every tone in each symbol within the frame. The computational complexity is compounded when dealing with OFDM systems.

4.2 Zero Forcing with Successive Interference Cancellation

ZF-SIC is a non-linear MIMO decoding technique that has been proven in [12] to be superior to the ZF method of decoding. ZF-SIC exploits the benefit of timing synchronism [9] while building upon the premise of ZF. SIC attempts to decode the most reliable element of the frame first and then use this to further help decode the remaining elements of y and provide superior performance. SIC is based on the interference of already detected symbols of y from the transmitted vector x. The new received vector will now have fewer interfering symbols than it otherwise would without SIC.

SIC is dependent on the order in which the elements of y are detected. To determine an appropriate order of detection, the covariance matrix of estimation error defined by $y_{est} - y$ is employed. Using Eq. 4.4 the covariance matrix is given by the following:

$$\mathbf{C} = E\left[(y - y_{est})(y - y_{est})^H\right] = \sigma_n^2 \mathbf{H}^\dagger \left(\mathbf{H}^\dagger\right)^H = \sigma_n^2 Y \tag{4.5}$$

The best estimate is the one in which the diagonal element of $\mathbf{C}$ is the smallest as this would indicate the smallest error variance. Equivalently, the minimum squared length row would suffice as an optimal estimate.

Thus, the key components to ZF-SIC comprise of ordering, interference cancelling, and mapping. ZF-SIC for use with V-BLAST is given as a recursive function in [2]. A summary of this algorithm is shown below, in Alg. 1.

Algorithm 1 ZF-SIC Decoding

```
1:  Compute $\mathbf{H}^\dagger$
2:  Find minimum squared length row of $\mathbf{H}^\dagger$ and permute to last row (i.e., $n_t^{th}$ row)
3:  Estimate corresponding element of $y$ as $y_{est} = w^{n_t}x$, where $w^{n_t}$ is row $n_t$ of permuted $\mathbf{H}^\dagger$
4:  Demodulate to nearest constellation point $y_{est,dem}$
5:  if $n_t - 1 > 0$ then
6:      $\mathbf{H} \to \mathbf{H}^{n_t-1} = (h_1, h_2, \ldots, h_{n_t-1})$
7:      $x \to x - h_{n_t}(y_{est,dem})$
8:      $n_t \to n_t - 1$
9:      Go back to 1
10: else
11:     Accept new frame and go to 1
12: end if
```

For an OFDM system, this algorithm would need to be applied to each subcarrier that makes up the OFDM symbol as there would be a channel matrix associated to each tone. Therefore this algorithm would run a total of n times per symbol where n is the number of tones.

There are many different kinds of detectors that can be utilized in a V-BLAST MIMO OFDM system. In [12], it was shown that a MMSE detector is deemed the optimal choice, however, as shown in [5], the computational complexity of a MMSE detector increases drastically with the number of transmit and/or receive antennas used. For this study, a 4 x 4 MIMO system was designed and as a result the ZF-SIC detector was chosen in order to reduce known issues with computational complexity as indicated in [2].

4.3 Measurement Procedure

This section outlines the procedure that was taken to obtain the measurement results produced in the study of this thesis. Initially, the baseband design in Simulink must be configured to be recognizable by the FPGA boards. Xilinx provides a System Generator utility that embeds into Simulink. This utility provides the means to synthesize any design that has used Xilinx digital design block sets into a bit file that is created from all the Verilog code associated to that system. There is a separate System Generator utility for the VHS FPGA and SMQ FPGA boards. Once the bit files are generated, the environment must be set up correctly to ensure a multipath rich scattering environment.

Fig. 4.1, below, illustrates the setup used to obtain measurements. The receiver and transmitter nodes were set up at opposite ends of a room. Metal plates were placed around the receiver to ensure that signals would be reflected to create a rich scattering environment. The ZF-SIC decoding algorithm requires multipath to perform adequately.

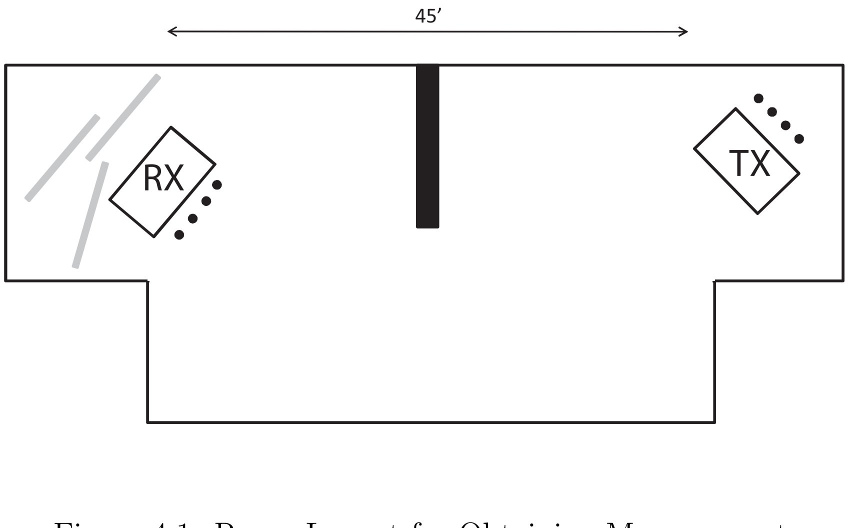


Figure 4.1: Room Layout for Obtaining Measurements

The TX and RX nodes contained four antennas at each end of the room. To ensure the radiation patterns were appropriately aligned, the TX and RX antennas were kept at the same height off the ground. Once the environment has been set up, both nodes can communicate. Note that for this study, different orientations of the metallic plates were used and measurements were taken for different multipath environments. It was observed that with altering the multipath environment the bit error rates were relatively consistent.

The clock for both nodes must be set to 80 MHz by means of an external signal source or through the internal ADACSync utility provided by Lyrtech. The 80 MHz clock ensures the rate of data arriving from the baseband design to the DAC/ADC is the same as the internal rate of the DAC/ADC. On the baseband design, registers were set up to allow for alternative modes of operation. These registers allow for switching between Single Input Single Output (SISO) and MIMO data transmission. Specific blocks not required for basic operation can be turned on or off through these registers. This includes FOC and Phase Tracking. A summary of the registers and their functionality is shown in Table B.1 in Appendix B. Prior to running the design for the first time, Register 0 should be set to SISO mode with Phase Tracking off and Register 2 should be set to TX Mode. Fig. B.1 displays the VHS Utility that is used to set the register values.

Once these registers have been set on both the TX and RX nodes, the TX and RX gain settings can be set using the RFFE utility. The RX gains should be set to six and the TX gains can be varied. Although the RX gain can be increased from zero to 32, it was seen that for RX gains larger than ten, noise and interference was amplified at the receiver. Fig. B.2, illustrates the RFFE utility used to set the RF gains at both the TX and RX nodes. For optimal performance a TX gain of 48 was used. The RFFE utility also allowed the up-converted frequency to vary between 2.4 GHz and 2.5 GHz. The RF Band setting at the TX and RX nodes should be the same to ensure that accurate acquisition occurs.

A tuning process must be conducted prior to switching over to MIMO transmission. The initial step in tuning is to view signals from the DAC with Register 2 set to TX Mode. In this mode the transmitted SISO signals can be viewed on the scope. This ensures that what is being sent over the air is as expected. Then, with Register 2 set to RX mode, the off air received signal can be viewed prior to any post-acquisition processing. The signal amplitude of the received signal should be large enough to be seen on the oscilloscope.

The final part of tuning is to change Register 2 to view probe signals. On channels 13 to 16 of the DAC, the start of frame, channel coefficients, expected received data, and number of errors can be viewed. The start of frame signal should be easily seen and used as the triggering signal. Under the optimal TX and RX gain settings the number of errors displayed should be close to zero.

Once the SISO design is ensured to be working, the tuning phase is complete and Register 0 can be switched to MIMO mode. In MIMO mode, the time synchronization block should still be producing a start of frame pulse. The number of errors will not be close to zero because decoding was not implemented for MIMO on the FPGA, whereas SISO channel equalization was included in FPGA development. For MIMO transmission, the DSP was used to develop the decoding algorithm. As a result, the DSP design must be opened and compiled in Simulink. Within the design file, a running BER will be displayed. Once the BER is verified, data can be saved using the SDRAM memory block for offline processing.

Adjusting the TX gain controls the SNR, and as a result, a BER vs SNR relationship can be determined. A single BER value can be obtained for a given SNR value by finding the total number of bit errors in a given sequence of frames and dividing by the total number of received bits of data.

4.4 Time Domain Analysis

In the time domain, reducing the overall number of channel inversions used in ZF-SIC can be assessed by looking at incoming symbols on a frame by frame basis. ZF-SIC requires that for each frame containing CSI, a channel inversion process must be conducted on the symbols within that frame. Channel variations are avoided by undergoing this inversion process for each frame received.

In the experimental set up, each frame consists of a series of training sequences used to determine CSI along with four OFDM symbols worth of data. The goal of this study is to see if the BER vs SNR characteristics can be maintained while reducing the overall number of pseudo-inverse operations performed in ZF-SIC. Reducing the number channel inversions will greatly reduce the complexity of the decoding process and as a result will reduce the time it takes to decode V-BLAST MIMO OFDM symbols.

4.4.1 Selecting Frames

Given an initial frame, ZF-SIC can be applied to that frame in an effort to decode the OFDM symbols that have been received. Instead of performing channel inversions on each incoming frame, the pseudo-inverse of the initial channel matrices can be stored and applied to incoming data so long as the CSI does not vary. Fig. 4.2, below, illustrates how the channel information can be held across multiple frames when utilizing ZF-SIC.

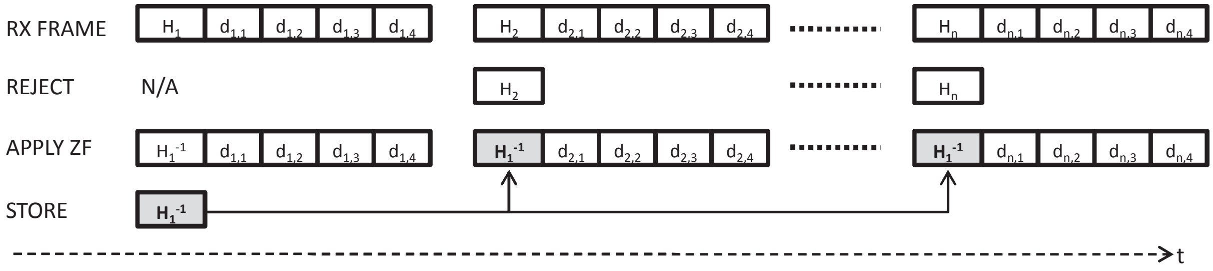

RX FRAME
H_1 $d_{1,1}$ $d_{1,2}$ $d_{1,3}$ $d_{1,4}$
H_2 $d_{2,1}$ $d_{2,2}$ $d_{2,3}$ $d_{2,4}$
H_n $d_{n,1}$ $d_{n,2}$ $d_{n,3}$ $d_{n,4}$
REJECT
N/A
H_2
H_n
APPLY ZF
H_1^{-1} $d_{1,1}$ $d_{1,2}$ $d_{1,3}$ $d_{1,4}$
H_1^{-1} $d_{2,1}$ $d_{2,2}$ $d_{2,3}$ $d_{2,4}$
H_1^{-1} $d_{n,1}$ $d_{n,2}$ $d_{n,3}$ $d_{n,4}$
STORE
H_1^{-1}
t

Figure 4.2: Holding Channel State Information Over n Frames

After the first frame has been processed through the ZF-SIC algorithm, the pseudo-inverses of the channel matrices can be stored. The channel matrices for the next $n-1$ frames can be rejected and the already calculated pseudo-inverses of the initial frame can be applied to decode the remaining $n-1$ frames. Alg. 2, below, summarizes how this detection process would work.

Using the experimental setup described in Chapter 3, measurements were taken to examine the effects of holding CSI over multiple frames. The number of frames for which the channel was held was varied and resulting BER vs. SNR characteristics were produced to analyse these effects. Fig. 4.3, compares these plots against standard ZF-SIC decoding which inverts every tone on every frame.

Algorithm 2 ZF-SIC Decoding - Selecting Frames

```
Receive initial frame in set of $n$ incoming frames
Compute and store $\mathbf{H}^{\dagger}$
Find minimum squared length row of $\mathbf{H}^{\dagger}$ and permute to last row (i.e. $n_t^{th}$ row)
Estimate corresponding element of $y$ as $y_{est} = w^{n_t}x$, where $w^{n_t}$ is row $n_t$ of permuted $\mathbf{H}^{\dagger}$
Demodulate to nearest constellation point $y_{est,dem}$
if $n_t - 1 > 0$ then
    $\mathbf{H} \rightarrow \mathbf{H}^{n_t-1} = (h_1, h_2, \ldots, h_{n_t-1})$
    $x \rightarrow x - h_{n_t}(y_{est,dem})$
    $n_t \rightarrow n_t - 1$
    Go back to 2
else
    Run through steps $3-5$ on the next $n-1$ frames with the $n_t$ stored pseudo inverses of $\mathbf{H}$
end if
```

For larger SNR values, the BER converged towards the characteristics of ZF-SIC so long as the number of frames the CSI was held for was less than or equal to ten. When the same CSI is used for ten frames, 52 x 4 x 4 x 9 pseudo-inverse operations are avoided resulting in significant reductions in decoding complexity. Fig. 4.4, illustrates varying channel coefficients over all frequency tones for high SNR. The data shown is for one of the 16 paths (i.e., h_{11}) for clarity. The minimal variation experienced by the channel coefficients explains why such characteristics were achieved in Fig. 4.3. The filtering effect experienced in the frequency response is due to various components in the platform chain comprised of antennas, and several hardware and software components. Explanation of this effect is beyond the scope of our research.

Frames 1, 11, and 21 represent the channel coefficients 10 frames apart in time. The variations in the coefficient values are minimal, however, over time the small variations in the channel no longer provide values that are applicable to frames that occurred 15 to 20 frames earlier. As a result, beyond 10 frames, the BER characteristics diverged from ZF-SIC and no longer produced a viable trend.

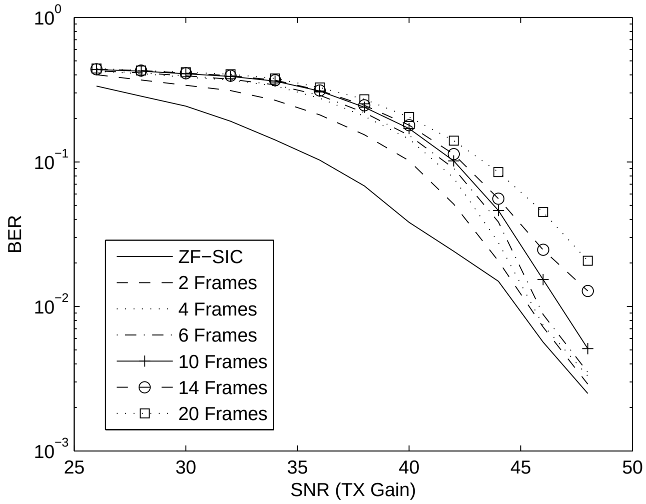


Figure 4.3: BER vs SNR - Selecting Frames

For lower SNR values, regardless of the number of frames the channel information was held for, it produced poorer results from ZF-SIC. Fig. 4.5, below, illustrates the variation in the frequency domain of the channel coefficients separated ten frames apart in the time domain for lower SNR values. The large variation across frames is due to the noisy nature of the received signal which suggests why results worsened.

At low SNR values, the training sequence used to determine the channel coefficients are severely distorted by interfering packets and noise. From frame to frame, the coefficients are no longer similar to one another. As a result, attempting to use one frames channel information ten frames later no longer provides results that are beneficial.

4.4.2 Averaging

An alternative approach to reducing the complexity of the decoding process for a V-BLAST MIMO OFDM system is to implement an averaging mechanism over the channel coefficients. From a set of consecutively received frames the channel information can be averaged and the pseudo-inverse of this averaged channel information can be used to decode this data. The premise is to use the average channel information in the ZF-SIC decoding of the first frame, store the inverse matrices, and then use these matrices on all frames involved in the averaging process. This will reduce the number of channel inversions and will aver-

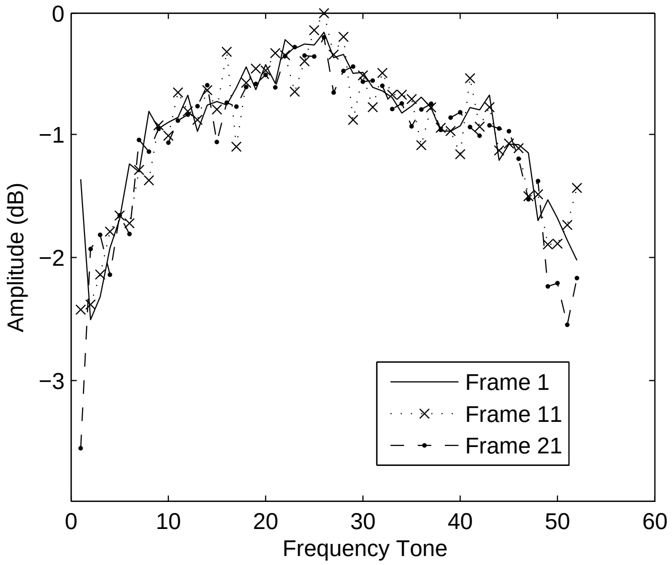


Figure 4.4: Channel Coefficients for High SNR (TX Gain = 48)

age the channels effects over all frames. Arithmetic and RMS methods of averaging were explored. Fig. 4.6, below, illustrates how channel coefficients are selected for averaging based on a certain number of incoming frames in the time domain.

Prior to applying the decoding algorithm, n incoming channel matrices must be stored. The average of these matrices, $\mathbf{H}_{av}$, is then applied to the first frame of this set. The inverse matrices from the decoding process can be stored. Alg. 3, below, summarizes how averaging would be implemented.

Arithmetic Mean

The arithmetic mean is simply the sum of the values of a random variable divided by the number of values. Eq. 4.6, below, provides an expression for the arithmetic mean. It should be noted that the arithmetic mean is an equally weighted form of averaging.

$$\mathbf{H}_{av} = \mathbf{H}_{am} = \frac{1}{n}\sum\nolimits_{i=1}^{n} \mathbf{H}_i \tag{4.6}$$

Fig. 4.7, provides the BER vs. SNR performance of applying arithmetic averaging to incoming channel information.

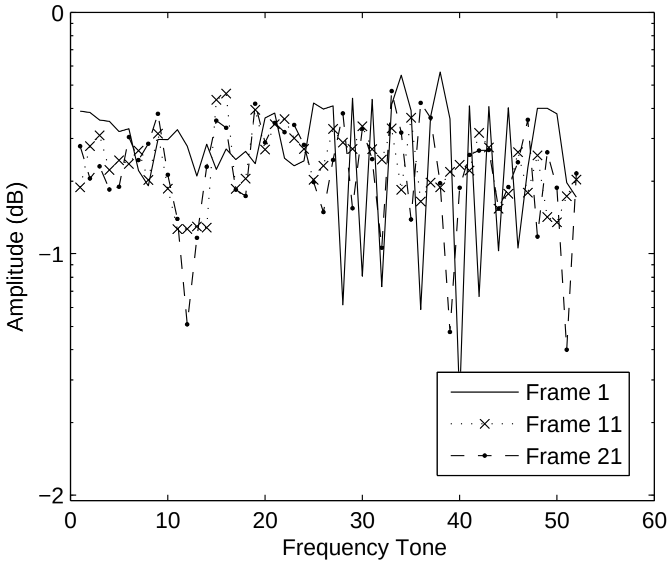


Figure 4.5: Channel Coefficients for Low SNR (TX Gain = 26)

Much like the method of frame selections, shown in Fig. 4.2, utilizing the arithmetic mean provided better results for higher SNR values than it did for lower SNR values. However, when averaging fewer than 6 frames, the BER stayed relatively close to ZF-SIC decoding for smaller SNR values. As the number of frames averaged increases the BER vs SNR characteristics no longer provide beneficial results. This is due to the arithmetic averaging removing all variations in the channel across those frames. Over 10 to 20 frames the channel begins to change enough to cause the decoding algorithm to incorrectly detect symbols.

RMS Mean

The RMS mean is a statistical measure of the magnitude of a varying quantity. In the case of this study it was used for a series of discrete values, namely the channel coefficients from incoming frames. Eq. 4.7, below, provides an expression for the RMS mean describing the square root of the mean of the squares of the coefficients for each frequency tone.

$$\mathbf{H}_{av} = \mathbf{H}_{rms} = \sqrt{\frac{\sum_{i=1}^{n} \mathbf{H}_i}{n}} \tag{4.7}$$

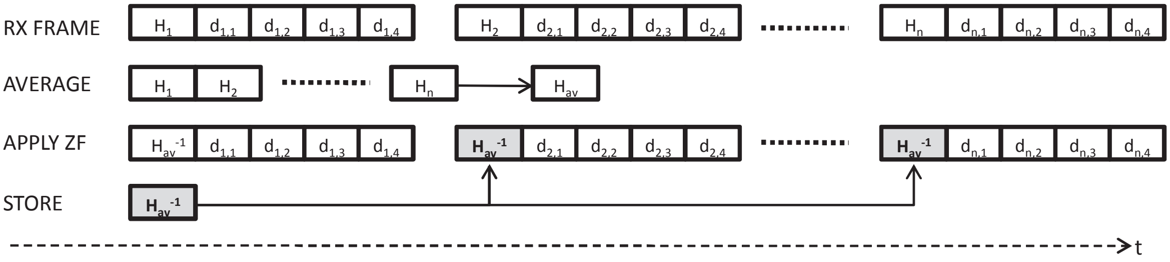


Figure 4.6: Averaging Channel Information over a set of Frames

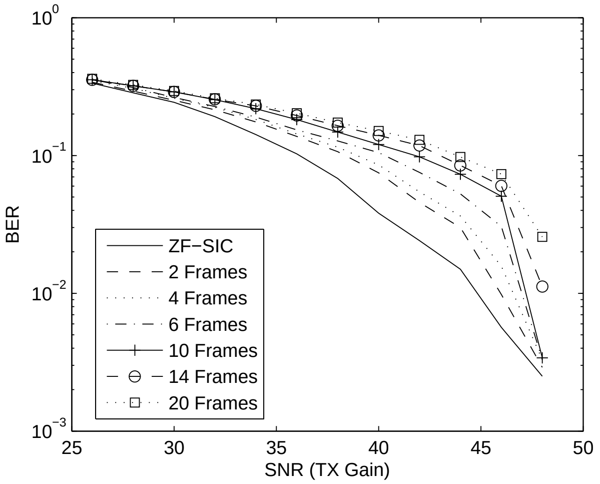


Figure 4.7: BER vs. SNR - Averaging Frames - Arithmetic Mean

Fig. 4.8, provides the BER vs. SNR performance of applying RMS averaging to incoming channel information.

Relative to Fig. 4.7, it is shown that the RMS mean removes any variation in the BER performance at low SNR values regardless of the number of frames used for averaging. At larger SNR values, as the number of frames averaged moves beyond ten, the BER started to degrade and no longer converged towards ZF-SIC. However, for ten or fewer frames, the BER does converge towards the performance of ZF-SIC decoding which suggests that for larger signal power the RMS mean will perform well and will provide a reasonable BER.

Algorithm 3 ZF-SIC Decoding - Averaging Frames

```
Receive set of $n$ incoming frames
Compute average of all $\mathbf{H}_i$, $\mathbf{H}_{av}$, where $i = 1, ..., n$
Compute and store $\mathbf{H}_{av}^{\dagger}$
Find minimum squared length row of $\mathbf{H}_{av}^{\dagger}$ and permute to last row (i.e. $n_t^{th}$ row)
Estimate corresponding element of $y$ as $y_{est} = w^{n_t}x$, where $w^{n_t}$ is row $n_t$ of permuted $\mathbf{H}^{\dagger}$
Demodulate to nearest constellation point $y_{est,dem}$
if $n_t - 1 > 0$ then
    $\mathbf{H} \to \mathbf{H}^{n_t - 1} = (h_1, h_2, \ldots, h_{n_t - 1})$
    $x \to x - h_{n_t}(y_{est,dem})$
    $n_t \to n_t - 1$
    Go back to 2
else
    Run through steps $4 - 6$ on the set of $n$ frames with the $n_t$ stored pseudo inverses of $\mathbf{H}$
end if
```

4.4.3 Interpolation

Interpolation is a method of constructing new data points within the range of a discrete set of data points. For this study, to reduce the number of inversions performed over time, specific received frames were selected and the pseudo-inverse of these matrices were interpolated to achieve inverse channel matrices for all frames not in the interpolating set. Fig. 4.9, illustrates interpolation across ten frames.

The first frames, which are a multiple of ten, are stored, and the ZF-SIC pseudo-inverse operation is performed on this subset of frames. The channel information provided by all frames not in this subset are rejected, thus not requiring inverse operations on each of these frames. The initial set of pseudo-inverse matrices are interpolated to obtain matrices for those frames not in the subset. Alg. 4, demonstrates how this method would be implemented.

The interpolated channel matrices are then used in the ZF-SIC algorithm to decode the corresponding data. There are linear and non linear methods of interpolation. Fig. 4.10,

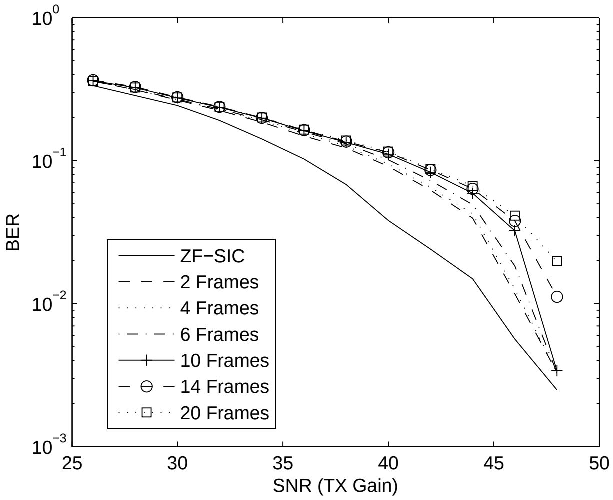


Figure 4.8: BER vs. SNR - Averaging Frames - RMS Mean

illustrates the varied effects of linear and non-linear forms of interpolation.

Linear

Linear interpolation is a method of curve fitting using linear polynomials. Eq. 4.8, below, describes the formulation for this type of interpolation.

$$y = y_0 + (x - x_0)\,\frac{y_1 - y_0}{x_1 - x_0} \tag{4.8}$$

If two points are known, the linear interpolant is the straight line between these two points. When given the pseudo-inverse channel matrix of two non sequential frames, the interpolated matrix of a frame between these two would be the straight line connecting them. Eq. 4.9, is based upon Eq. 4.8, above, expressed in terms of channel matrices.

$$\mathbf{H}_b^{-1} = \mathbf{H}_a^{-1} + (b - a)\,\tfrac{\mathbf{H}_c^{-1} - \mathbf{H}_a^{-1}}{c-a} \quad c < b < a \tag{4.9}$$

Eq. 4.9 is performed on each frequency tone such that there is a pseudo-inverse channel matrix for each bit of data in the OFDM symbol. Hence there will be p x 4 x 4 inverse operations that will be avoided, where p is the number of interpolated inverse channel matrices. This form of interpolation on the inverse channel matrices of incoming frames was explored and the BER vs. SNR performance is presented in Fig. 4.11.

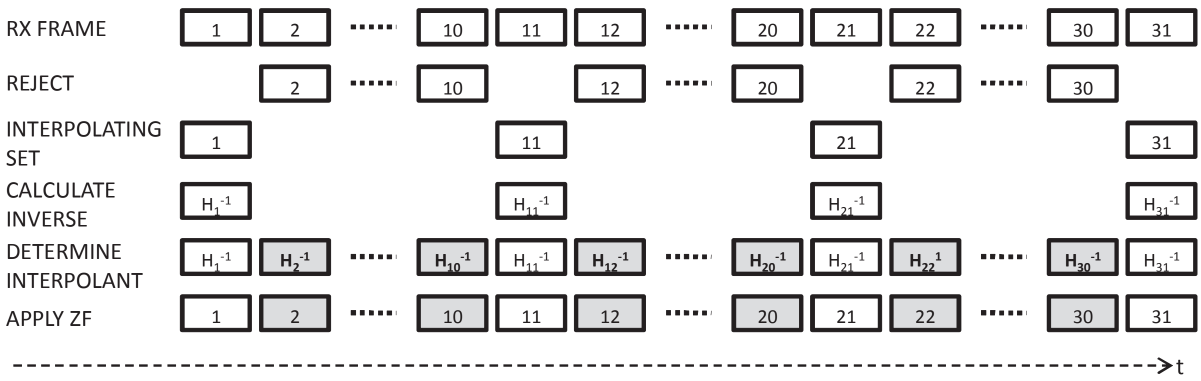

RX FRAME 1 2 ...... 10 11 12 ...... 20 21 22 ...... 30 31
REJECT 2 ...... 10 12 ...... 20 22 30
INTERPOLATING SET 1 11 21 31
CALCULATE INVERSE H_1^{-1} H_{11}^{-1} H_{21}^{-1} H_{31}^{-1}
DETERMINE INTERPOLANT H_1^{-1} H_2^{-1} ...... H_{10}^{-1} H_{11}^{-1} H_{12}^{-1} ...... H_{20}^{-1} H_{21}^{-1} H_{22}^{1} ...... H_{30}^{-1} H_{31}^{-1}
APPLY ZF 1 2 ...... 10 11 12 ...... 20 21 22 ...... 30 31
t

Figure 4.9: Interpolating Channel State Information

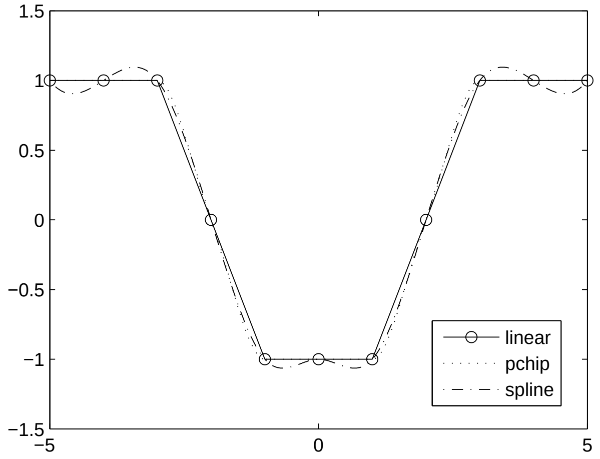


Figure 4.10: Interpolation Algorithm Comparison

As the number of frames being interpolated increased, the performance degraded. However, for the interpolation of six frames or fewer, the higher SNR values began to get closer to the performance of ZF-SIC but did not converge.

Non-Linear

Non linear forms of interpolation are generalizations of linear interpolation. Linear interpolation utilizes a linear interpolant function whereas non linear interpolation replaces this interpolant with a polynomial of a higher degree. To avoid Runges phenomenon [34] non

Algorithm 4 ZF-SIC Decoding - Interpolation

```
1:  Receive set of $n$ incoming frames
2:  Determine interpolating set containing $n - p$ frames
3:  Reject $p$ channel matrices not in interpolating set
4:  Compute and store $\mathbf{H}^{\dagger}$ for each frame in the interpolating set
5:  Deterine interpolants for remaining $p$ frames
6:  Find minimum squared length row of $\mathbf{H}^{\dagger}$ for all $n$ frames and permute to last row (i.e. $n_t^{th}$ row)
7:  Estimate corresponding element of $y$ as $y_{est} = w^{n_t}x$, where $w^{n_t}$ is row $n_t$ of permuted $\mathbf{H}^{\dagger}$
8:  Demodulate to nearest constellation point $y_{est,dem}$
9:  if $n_t - 1 > 0$ then
10:     $\mathbf{H} \rightarrow \mathbf{H}^{n_t - 1} = (h_1, h_2, \ldots, h_{n_t - 1})$
11:     $x \rightarrow x - h_{n_t}(y_{est,dem})$
12:     $n_t \rightarrow n_t - 1$
13:     Go back to 4
14: else
15:     Go back to 1 to accept next set of $n$ frames
16: end if
```

linear interpolation methods that utilize lower order polynomials were considered in the study, namely Natural Cubic Spline and PChip interpolation. Both of the studied methods of non linear interpolation do not use polynomials larger than the third degree.

Natural Cubic Spline

A spline is a special function defined piecewise by polynomials. The spline function is given by Eq. 4.10, below.

$$S(x) = \begin{cases} S_0(x) & x \in [x_0, x_1] \\ S_1(x) & x \in [x_1, x_2] \\ \vdots & \\ S_{n-1}(x) & x \in [x_{n-1}, x_n] \end{cases} \qquad x_0 < x_1 \ < \cdots < x_n \tag{4.10}$$

Each $S_i(x)$ is a polynomial of degree k. A spline of degree n which interpolates a data

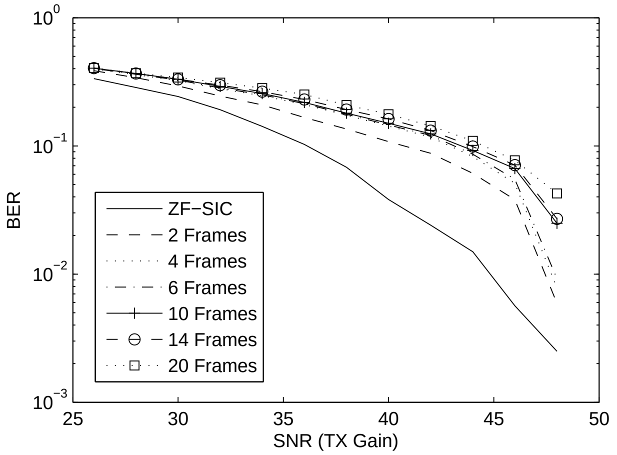


Figure 4.11: BER vs. SNR - Interpolation - Linear

set of discrete data points is not uniquely defined, as $n-1$ degrees of freedom need to be constructed to create a unique spline interpolant. For a cubic spline interpolant, each $S_i(x)$ is a polynomial of degree $k=3$. For the Natural Cubic Spline each $S_i(x)$ is defined by Eq. 4.11:

$$S_i\left(x\right) = \frac{z_{i+1}\left(x-x_i\right)^3 + z_i\left(x_{i+1}-x\right)^3}{3h_i} + \left(\frac{y_{i+1}}{h_i} - \frac{h_i}{3}z_{i+1}\right)\left(x-x_i\right) + \left(\frac{y_i}{h_i} - \frac{h_i}{3}z_i\right)\left(x_{i+1}-x\right) \tag{4.11}$$

Each h_i is defined by $x_{i+1}-x_i$ and the co-efficients z_i are found by solving a system of equations defined by Eq. 4.12, Eq. 4.13, and Eq. 4.14:

$$z_0 = 0 \tag{4.12}$$

$$h_{i-1}z_{i-1} + 2\left(h_{i-1}+h_i\right)z_i + h_i z_{i+1} = 6\left(\tfrac{y_{i+1}-y_i}{h_i} - \tfrac{y_i-y_{i-1}}{h_{i-1}}\right) \quad i = 1,\ldots,n-1 \tag{4.13}$$

$$z_n = 0 \tag{4.14}$$

For this study, each x_i is a frame being used for interpolation, and each $S_i(x)$ where $i=0,\ldots,n$ is an interpolant. For example, when $n=5$, there are five non-sequential frames which are interpolated by the functions $S_i(x)$, respectively. To obtain values for the frames in between the selected frames, a point on the function $S_i(x)$ is selected as the discrete value for the pseudo-inverse channel information. Cubic Spline interpolation was explored using measured data and the BER vs SNR performance is shown in Fig. 4.12.

In Fig. 4.12, the number of frames refers to the number of frames between samples being interpolated. The performance of Cubic Spline interpolation appeared to do poorly

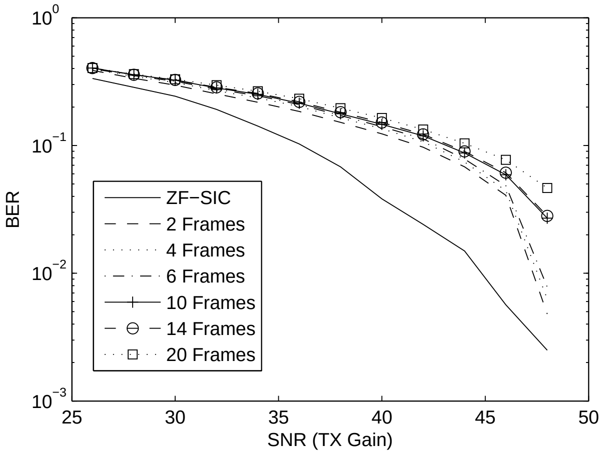


Figure 4.12: BER vs. SNR - Interpolation - Non Linear - Cubic Spline

at lower SNR values regardless of the number of frames being interpolated. However, at larger SNR values, for fewer than 10 frames, the BER tended to approach ZF-SIC performance levels but did not converge.

Piecewise Cubic Hermite Interpolation

Piecewise Cubic Hermite Interpolation (PChip) is a form of non-linear interpolation and is defined by the Piecewise Cubic Hermite Interpolating Polynomial. PChip finds the values of an interpolating polynomial function $C(x)$ at intermediate points. For the given values x in the subinterval $x_i < x < x_{i+1}$ and for certain slopes at the endpoints, $C(x)$ is the cubic hermite interpolant to those values. The slopes at the endpoints, x_i and x_{i+1}, are chosen such that $C(x)$ preserves the shape of the data. This means that when data is monotonically increasing or decreasing, so is $C(x)$. If the data set has a local minimum or maximum, so does $C(x)$. The expression for $C(x)$ can be given much like the expression for Cubic Spline interpolation shown in Eq. 4.10. As previously shown in Fig. 4.10, a comparison is provided of how PChip interpolation varies relative to Cubic Spline interpolation and linear interpolation.

PChip interpolation, as previously stated, once reaching a local maximum or minimum remains at that respective value. For the sake of this study, each of the points on the linear relationship in Fig. 4.10 represent the amplitude of a particular pseudo-inverse for a specific frequency tone and the line between the two points would represent the respective amplitudes of the frames between them corresponding to the PChip interpolant. PChip interpolation was explored using measured data and the BER vs SNR performance is shown in Fig. 4.13.

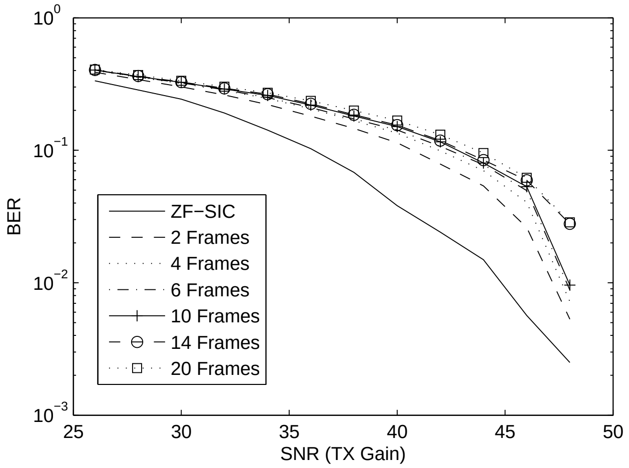


Figure 4.13: BER vs. SNR - Interpolation - Non Linear - PChip

In Fig. 4.13, the number of frames refers to the number of frames being interpolated between samples. The performance of PChip interpolation was very similar to that of Cubic Spline interpolation. The difference lies in the number of frames for which this method of interpolation was effective. When looking at 10 frames, PChip interpolation began to approach ZF-SIC performance levels whereas for Cubic Spline interpolation, the BER diverged away from the control sample set.

In general, it has been shown that in the time domain, the optimal number of frames for which BER performance still converges to the ZF-SIC algorithm is ten for specific methods of analysis. These methods include selecting frames, averaging frames, and PChip interpolation with the best BER performance coming from the RMS mean and selecting frames methods.

4.5 Frequency Domain Analysis

In the frequency domain, reducing the overall number of channel inversions performed during the decoding of a set of data can be obtained by looking at the frequency tones of the estimated channel.

In the experimental setup described in Chapter 3, each OFDM symbol contains 52 tones of data. As a result, each channel coefficient is the same size. Each tone contains a different piece of CSI that pertains solely to that particular frequency. For each tone in the OFDM symbol, the channel information must be inverted and applied to that tone in order to correct the channel effects on the received data. Since each set of channel coefficients is assumed to be constant over four OFDM symbols the total number of channel inversions performed is 52 x 4 x 4 on each frame. In Section 4.4, an attempt was made to reduce the total number of inversions that occur over time, this study will look at reducing the number of inversion that occur over the 52 frequency tones.

4.5.1 Selecting Tones

When an incoming frame is received, the preamble contains CSI for each tone. If the typical decoding process is applied to the frame, each tone undergoes the ZF-SIC algorithm. This means that if the OFDM symbol contains 52 tones, as in this experimental set up, it will perform 52 pseudo-inverse operations. An alternative approach to performing this operation on each tone is to see if the channel remains relatively constant over a series of tones and apply one inverse operation to that set of tones. Fig. 4.14, below, illustrates how this method of selecting tones can be applied to an incoming set of channel information.

The channel information arrives in four sets of 208 bits of data. Each set contains four of the sixteen possible channel states that would comprise the channel matrix. Fig. 4.14 shows one of the four received sets, consisting of $\mathbf{H}_{1,1}$ to $\mathbf{H}_{4,1}$, and illustrates holding CSI over two successive tones. Once ZF-SIC is applied to a selected tone, the pseudo-inverse information is stored and applied to any number of successive tones. In the case of Fig. 4.14, this is two successive tones. When this is held over two tones, the total number of inverse operations is reduced from 52 x 4 x 4 to 26 x 4 x 4. In other words, the number of inverse operations are halved. Alg. 5, demonstrates how this method of decoding is implemented.

In Fig. 4.15, measurements were taken to examine the effects of holding CSI over multiple tones. The number of tones which the channel matrices were held for were varied and resulting BER vs. SNR plots were produced to represent these effects. These plots were

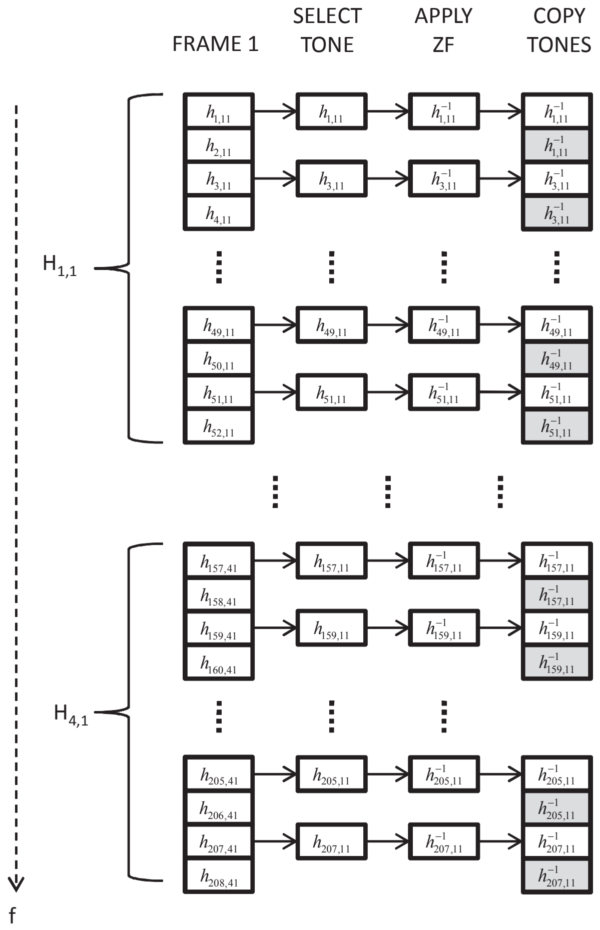

FRAME 1
SELECT TONE
APPLY ZF
COPY TONES
$h_{1,11}$ $h_{1,11}$ $h^{-1}_{1,11}$ $h^{-1}_{1,11}$
$h_{2,11}$ $h^{-1}_{1,11}$
$h_{3,11}$ $h_{3,11}$ $h^{-1}_{3,11}$ $h^{-1}_{3,11}$
$h_{4,11}$ $h^{-1}_{3,11}$
$H_{1,1}$
$h_{49,11}$ $h_{49,11}$ $h^{-1}_{49,11}$ $h^{-1}_{49,11}$
$h_{50,11}$ $h^{-1}_{49,11}$
$h_{51,11}$ $h_{51,11}$ $h^{-1}_{51,11}$ $h^{-1}_{51,11}$
$h_{52,11}$ $h^{-1}_{51,11}$
$h_{157,41}$ $h_{157,11}$ $h^{-1}_{157,11}$ $h^{-1}_{157,11}$
$h_{158,41}$ $h^{-1}_{157,11}$
$h_{159,41}$ $h_{159,11}$ $h^{-1}_{159,11}$ $h^{-1}_{159,11}$
$h_{160,41}$ $h^{-1}_{159,11}$
$H_{4,1}$
$h_{205,41}$ $h_{205,11}$ $h^{-1}_{205,11}$ $h^{-1}_{205,11}$
$h_{206,41}$ $h^{-1}_{205,11}$
$h_{207,41}$ $h_{207,11}$ $h^{-1}_{207,11}$ $h^{-1}_{207,11}$
$h_{208,41}$ $h^{-1}_{207,11}$
f

Figure 4.14: Selecting Tones

compared to ZF-SIC in which every tone is inverted and applied to the corresponding data.

For larger SNR values, as the number of tones for which the CSI was held increased, the BER performance significantly deteriorated. There was no need to test beyond four tones as the obtainable BER values were not beneficial. It was observed that although the channel information remained relatively flat, the amplitudes of the channel coefficients did vary, at times, by almost 20 percent, which was a significant enough factor to cause the BER to deteriorate. Fig. 4.16, illustrates the channel coefficient $h_{1,1}$ as it varies over frequency.

As shown in Fig. 4.16, there are regions where the amplitude of the coefficient from tone to tone will only vary by factors of 0.05, but across multiple tones these variances will

Algorithm 5 ZF-SIC Decoding - Selecting Tones

```
1: Select $n$ tones from incoming frames' co-efficients that are evenly spread throughout the symbol (e.g. every two tones)
2: Compute $\mathbf{H}^{\dagger}$ on $n$ tones
3: Copy $\mathbf{H}^{\dagger}$ to remaining tones
4: Find minimum squared length row of $\mathbf{H}^{\dagger}$ for all $n$ frames and permute to last row (i.e. $n_t^{th}$ row)
5: Estimate corresponding element of $y$ as $y_{est} = w^{n_t}x$, where $w^{n_t}$ is row $n_t$ of permuted $\mathbf{H}^{\dagger}$
6: Demodulate to nearest constellation point $y_{est,dem}$
7: if $n_t - 1 > 0$ then
8:     $\mathbf{H} \rightarrow \mathbf{H}^{n_t - 1} = (h_1, h_2, \ldots, h_{n_t - 1})$
9:     $x \rightarrow x - h_{n_t}(y_{est,dem})$
10:    $n_t \rightarrow n_t - 1$
11:    Go back to 2
12: else
13:    Go back to 1 to accept next frame
14: end if
```

increase to orders of 0.2. It can be seen that holding frequency tones does not prove to be an effective means of maintaining BER performance and reducing the number of channel inversions.

4.5.2 Averaging

An alternative approach to reducing the total number of channel inversions in the frequency domain is to perform averaging across frequency tones. Instead of performing an inversion on each tone to decode the received frame, averaging a set of successive tones and using the corresponding inverse to decode the data can reduce the overall number of inversions performed. Fig. 4.17, describes how the frequency tones would be averaged to reduce the overall number of channel inversions.

Prior to incoming frames undergoing ZF-SIC decoding, a successive sequence of frequency tones containing CSI are averaged. The pseudo-inverse of this average, utilized in

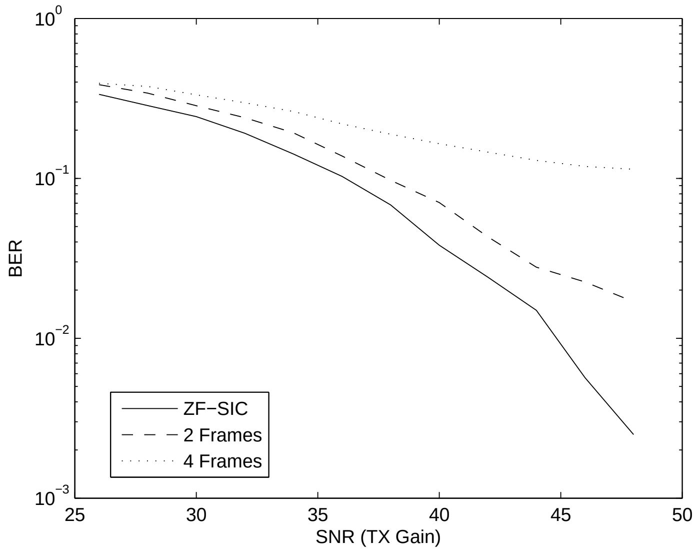


Figure 4.15: BER vs. SNR - Selecting Tones

ZF-SIC detection, is stored and applied across the same set of frequency tones. In the case of Fig. 4.17, four successive tones are averaged and the number of pseudo-inverse operations reduced to 13 x 4 x 4 from 52 x 4 x 4. Alg. 6, describes how this form of detection is implemented.

In Fig. 4.18, measurements were taken to examine the effects of averaging CSI over multiple tones. The number of tones which the channel matrices were held for were varied and resulting BER vs. SNR plots were produced to evaluate their performance. These plots were compared to ZF-SIC in which every tone is inverted and applied to the corresponding data.

For larger SNR values, as the number of tones for which the CSI was held increased, the BER performance significantly deteriorated. There was no need to test beyond averaging four tones as the obtainable BER values were not beneficial. It was observed that although the channel information remained relatively flat, the amplitudes of the channel coefficients did vary, at times, by almost 15 percent, which was a significant enough factor to cause the BER to deteriorate. Fig. 4.19, illustrates the channel coefficient $h_{1,1}$ as it varies over frequency.

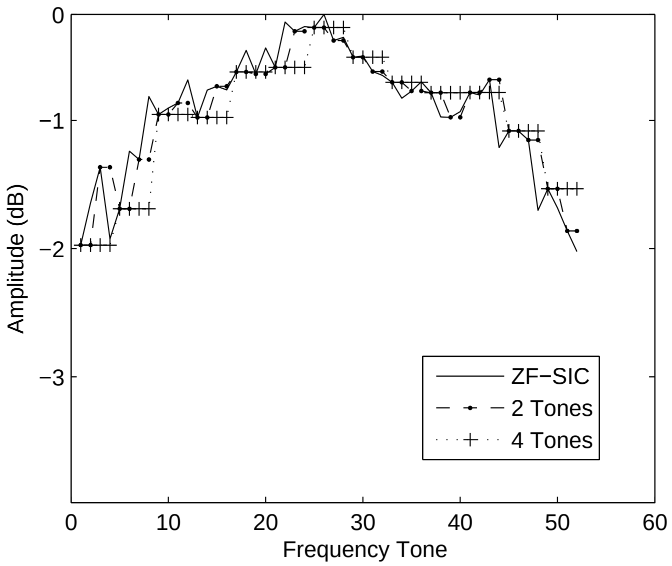


Figure 4.16: Channel Co-efficients for Selecting Tones

As shown in Fig. 4.19, there are regions where the amplitude of the coefficient from tone to tone will only vary by factors of 0.01. Due to the slightly varying nature of the channel, averaging tones which vary by an amplitude of 0.05 from one another will cause the averaged value to be significantly different from the actual value of some of the coefficients. It can be seen that averaging frequency tones does not prove to be an effective means of maintaining BER performance and reducing the number of channel inversions.

Analysis in the frequency domain has proven to be ineffective in both selecting tones and averaging. As a result, no further work was conducted in the frequency domain. In an effort to reduce complexity, the BER performance suffered and was deemed unworthy of further investigation.

4.6 Comparison

Fig. 4.20, compares all proposed algorithms for their applications over two frames. Analysis over two frames has shown that for lower SNR values, the frequency domain method of selecting tones reduces the complexity by a factor of two and provides a BER performance very close to the ZF-SIC results. Additionally, at low SNR values, the time domain method of averaging frames, both Arithmetic and RMS, provide good BER performance while cut-

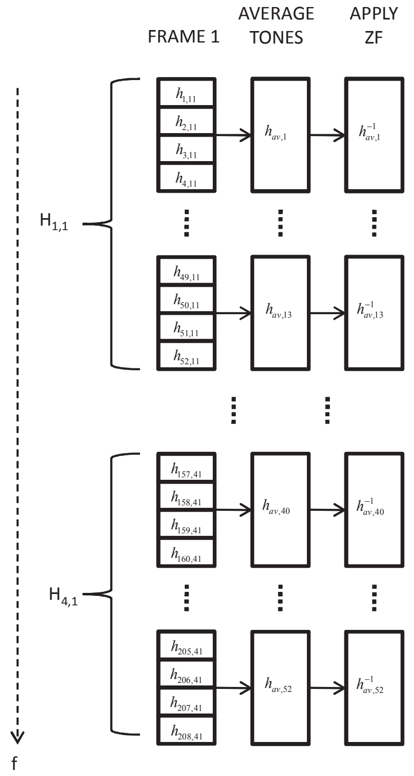


Figure 4.17: Averaging Tones

ting the number of pseudo-inverse operations by half. At high SNR, the frequency domain model does not provide good results, but the time domain analysis of selecting frames and averaging frames provide results similar to that of ZF-SIC.

Fig. 4.21, compares all proposed algorithms for their applications over four frames. Analysis over four frames has shown that for lower SNR values averaging in the time domain presents itself as the best means of maintaining reasonable BER values while reducing the complexity of decoding by a factor of four. The frequency domain methods prove themselves to be ineffective beyond two frames as their BER performances worsened when moving to four frame analysis. For larger SNR values, the time domain methods of selecting frames and averaging frames prove to provide BER performances very close to that of ZF-SIC. Although not as strong, the interpolating methods also provide reasonable BER performance for larger SNR's.

Fig. 4.22, compares all proposed algorithms for their applications over ten frames.

Algorithm 6 ZF-SIC Decoding - Averaging Tones

```
1: Select $p$ sets of $n$ sequential tones from incoming frame
2: Compute average of tones in each of $p$ sets
3: Compute $\mathbf{H}^{\dagger}$ on average of $n$ tones and apply to all $n$ tones in each set
4: Find minimum squared length row of $\mathbf{H}^{\dagger}$ for all $n$ frames and permute to last row (i.e. $n_t^{th}$ row)
5: Estimate corresponding element of $y$ as $y_{est} = w^{n_t}x$, where $w^{n_t}$ is row $n_t$ of permuted $\mathbf{H}^{\dagger}$
6: Demodulate to nearest constellation point $y_{est,dem}$
7: if $n_t - 1 > 0$ then
8:     $\mathbf{H} \rightarrow \mathbf{H}^{n_t-1} = (h_1, h_2, \ldots, h_{n_t-1})$
9:     $x \rightarrow x - h_{n_t}(y_{est,dem})$
10:    $n_t \rightarrow n_t - 1$
11:    Go back to 2
12: else
13:    Go back to 1 to accept next frame
14: end if
```

Analysis over ten frames has shown that for lower SNR values the time domain method of averaging frames provides the best results in terms of BER while reducing the complexity by a factor of ten. For larger SNR values, all of the time domain methods aside from the interpolation algorithms converge towards the ZF-SIC lower bound, with the best results given by selecting frames or averaging.

Fig. 4.23, compares all proposed algorithms for their applications over 14 frames. Analysis over 14 frames has shown that regardless of the algorithm used, none converge towards the ZF-SIC lower bound. This means that attempting to move past ten frames yields BER performances that are unacceptable.

As a result, the optimal number of frames that can be used are ten without compromising too much on BER performance. If the channel is assumed constant over this range and only one training sequence is sent over ten frames, the total number of OFDM symbols that can be sent per frame is 1932. If any of the averaging methods are used with the same frame size, each frame can contain 174 OFDM symbols since a training sequence must still

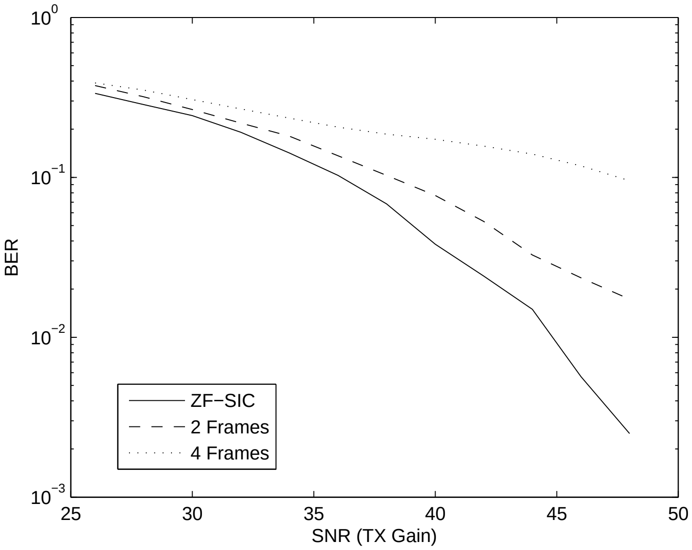


Figure 4.18: BER vs. SNR - Averaging Tones

be sent for each frame. To facilitate this many OFDM symbols, each frame would need to be sent one after the other.

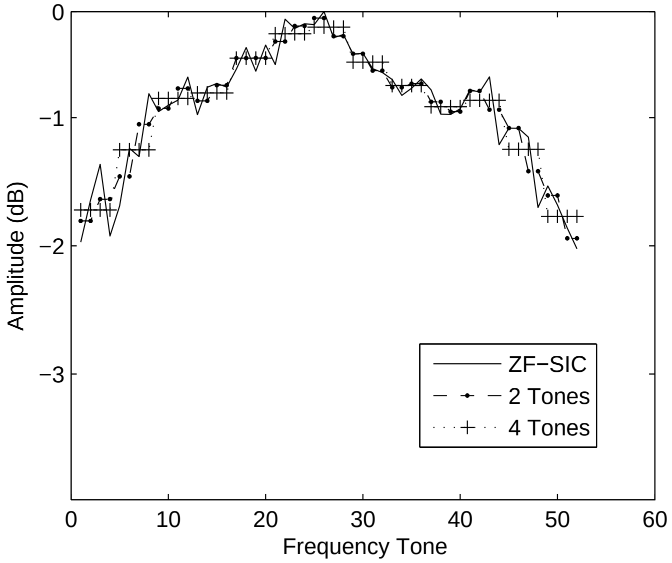


Figure 4.19: Channel Co-efficients for Averaging Tones

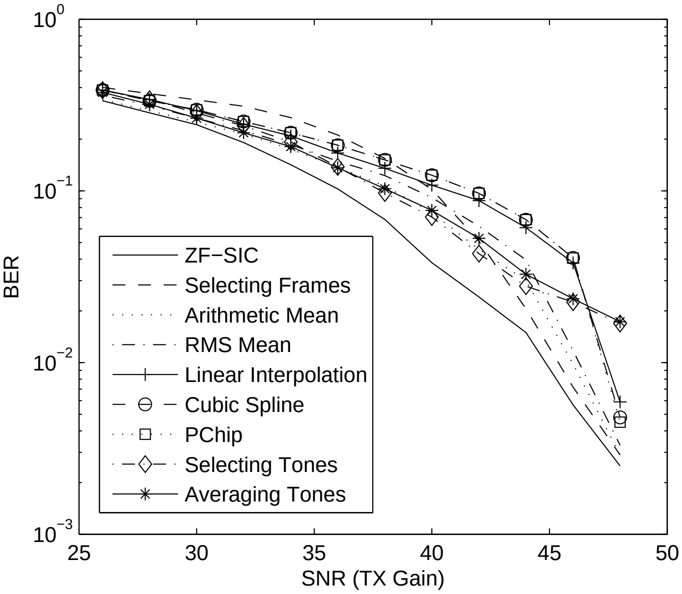


Figure 4.20: BER vs. SNR - Comparative Analysis - 2 Frames

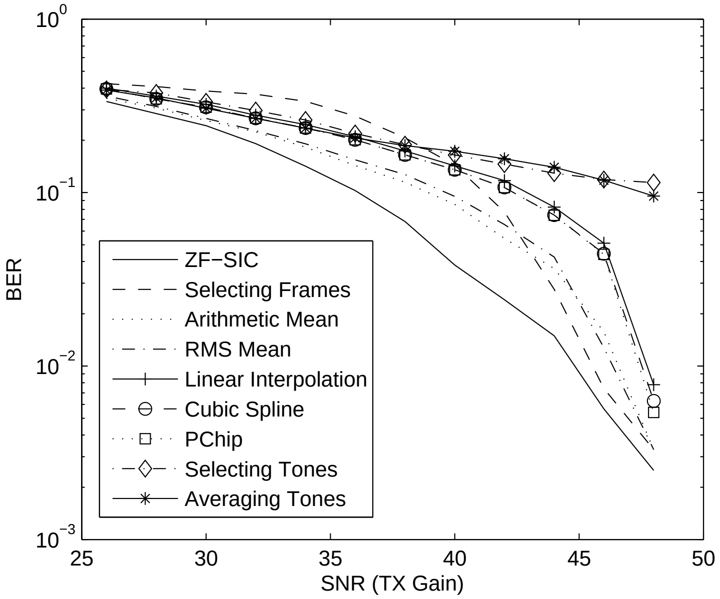


Figure 4.21: BER vs. SNR - Comparative Analysis - 4 Frames

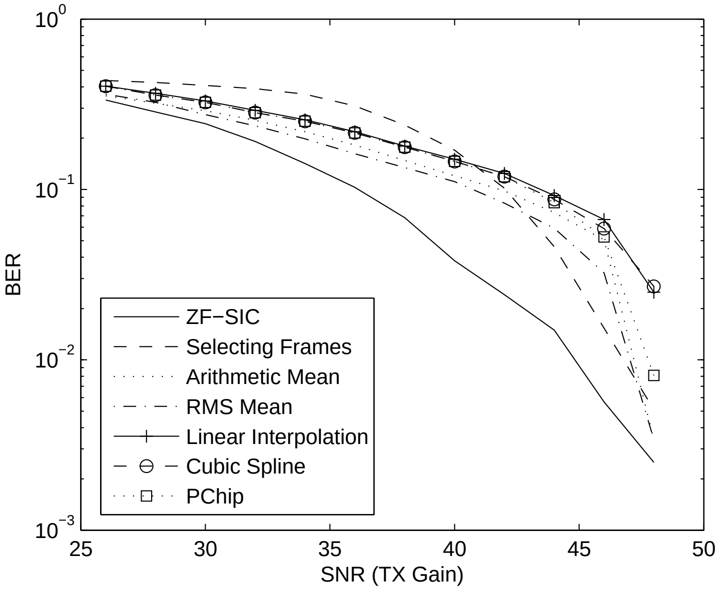


Figure 4.22: BER vs. SNR - Comparative Analysis - 10 Frames

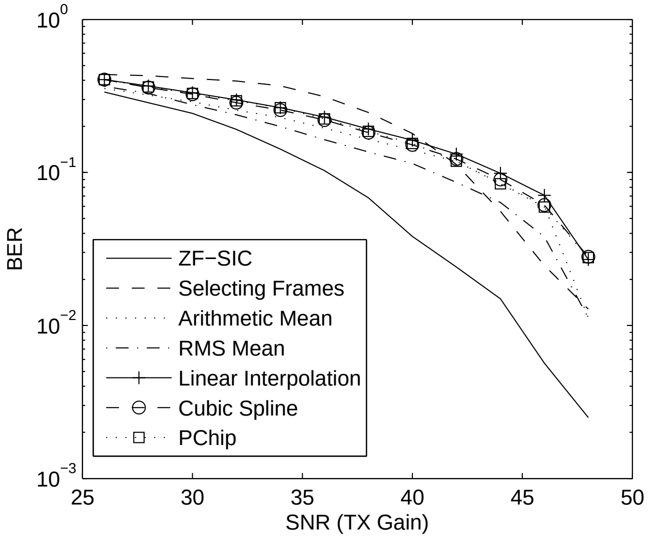


Figure 4.23: BER vs. SNR - Comparative Analysis - 14 Frames

Chapter 5

Concluding Remarks and Future Work

V-BLAST MIMO OFDM systems have proven to be an effective method of wireless transmission in the presence of a multi-path environment[2]. OFDM's ability to convert a frequency selective channel into a parallel collection of flat fading subchannels and MIMO's ability to take advantage of spatial diversity assists in increasing capacity. Due to the complexity of the receiver required to decode this data, the processing time has been an issue for realizing this technology. Additionally, MIMO communication has suffered from an inability to obtain accurate statistics of the CSI. Channel models have generally been used to obtain performance measures on multipath environments.

This thesis investigates methods in which to improve ZF-SIC by reducing the overall number of pseudo-inverse operations that are performed over a sample set, while maintaining an acceptable level of BER performance. Additionally, this analysis is performed over a multipath channel through a developed real-time wireless platform that supports V-BLAST MIMO OFDM communications. Based on the observed CSI, BER performance tests were conducted to provide a measure in which to compare proposed algorithms against ZF-SIC decoding.

Improvements to ZF-SIC decoding were proposed in both the frequency and time domain and were evaluated in terms of BER. Through this analysis, it was shown that reducing the number of pseudo-inverse operations in the frequency domain compromised the BER, however, reducing these operations in the time domain provided reduced complexity with acceptable BER performance. The algorithm with the most promise involved averaging CSI in the time domain across frames. It was shown that the number of frames that could be sent while still converging towards ZF-SIC performance was ten. From this statistic it was determined that the optimal number of OFDM symbols that can be associated with a single training sequence is 174, assuming Phase Tracking and frequency synchronization are taken into account. It was also shown that the method of selecting frames provides reduced complexity with acceptable levels of BER performance. The optimal number of OFDM symbols that can be transmitted using this method of decoding was determined to be 1932. Complexity reduction in the receiver serves as a tool in bringing MIMO technology into future wireless handhelds and communications technology.

The results concluded in this thesis have set the foundation for future research in technologies that involve MIMO communications. The experimental framework created to obtain BER performance measures can be used to tackle many other areas of research such as capacity, interference, and antenna formations. The study conducted within the context of this research can be further expanded by looking at the frequency and time domain simultaneously to reduce the number of pseudo-inverse operations performed in the decoding process. Alternatively, different methods for attempting to reduce the number of pseudo-inverse operations ZF-SIC performs can be explored. For instance, using the correlation between frequency tones or between sets of incoming frames to produce relational statistics can prove to be a beneficial measure. The BER performance measures provided in this study have only been presented for a 4 x 4 MIMO system. The wireless platform developed for obtaining these statistics can be further expanded to an 8 x 8 system or downsized to a 2 x 2 system with some changes to the baseband design. The same study can be conducted on alternate MIMO environments to observe if the results obtained for a 4 x 4 system are applicable to varying antenna quantities. This would prove useful in future wireless standards that require high throughput communications.

The data presented in this thesis can be used to benefit the implementation of 4G wireless communication systems. Reducing complexity in the receiver design can correspond with efficient processing on future wireless handhelds that take advantage of high throughput MIMO technologies.

Bibliography

[1] G. J. Foschini, “Layered Space-Time Architecture for Wireless Communication in a Fading Environment When Using Multiple Antennas”, *Bell Laboratories Technical Journal*, Vol. 1, No. 2, Autumn, 1996, pp. 41-59.

[2] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, “V-BLAST: An Architecture for Realizing Very High Data Rates Over the Rich-Scattering Wireless Channel,” *Proc. ISSSE Conference*, Pisa, Italy, Sept. 1998.

[3] G. J. Foschnini and M. J. Gans, “On the Limits of Wireless Communications in a Fading Environment when using Multiple Antennas,” *Wireless Personal Communications*, vol. 6, no.3, pp.311-335, 1998.

[4] E. Telatar, “Capacity of multi-antenna Gaussian Channels,” *European Trans. On Telecommun.*, vol. 6, no. 3, pp. 311-335, 1998.

[5] R. Zhang and J. M. Cioffi, “Approaching MIMO-OFDM Capacity With Zero-Forcing V-BLAST Decoding and Optimized Power, Rate, and Antenna-Mapping Feedback,” *IEEE Trans. Signal Process.*, vol. 56, pp. 5191-5203, Oct. 2008.

[6] G. Rui, L. Ji-Lin, and J. Yang, “Performance Analysis of BPSK for MIMO-OFDM in Nakagami-M Fading Channels,” *Proc. 2005 Int. Symp Intelligent Signal Processing and Comm. Systems*, pp. 109-112., Dec. 2005

[7] M. Borgmann and H. Bolcskei, “Interpolation-Based Efficient Matrix Inversion for MIMO-OFDM Receivers”

[8] E. U. Whu, “MIMO-OFDM Systems for High Data Rate Wireless Networks,” *Proj. Report for EE360 Advanced Wireless Networks: MIMO-OFDM Wireless Networks.*

[9] C. Shen, Y. Zhu, S. Zhou, and J. Jiang, “On the Performance of V-BLAST with Zero-Forcing Successive Interference Cancellation Receiver,” *IEEE Globecom*, 2004.

[10] S. Ohno and K. A. D. Teo, "Tradeoff between BER and Bandwidth Efficiency in MIMO Systems with ZF Equalization," *2006 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS2006)*, pp.518-521, 2006.

[11] W. Yan, S. Sun, and Z. Lei, "A Low Complexity VBLAST OFDM Detection Algorithm for Wireless LAN Systems," *IEEE Communications Letters*, Vol. 8, No. 6, pp. 374-376, June 2004.

[12] A. V. Zelst, "MIMO OFDM for Wireless LANs," *Dissertation*, Eindhoven University of Technology, April 2004.

[13] L. Litwin and M. Pugel, "The Principles of OFDM," *RF Signal Processing Magazine*, pp. 30-48, Jan. 2001

[14] A. Yasotharan, "Multirate Zero-Forcing Tx-Rx Design for MIMO Channel Under BER Constraints," *IEEE CCECE/CCGEI*, pp.126-130, May 2005.

[15] M. Torabi, S. Aissa, and M. Reza Soleymani, "MIMO-OFDM Systems with Imperfect Channel Information: Capacity, Outage, and BER Performance," *IEEE ICC*, pp. 5342-5347, 2006.

[16] G. D. Golden, G. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection Algorithm and Initial Laboratory Results Using V-BLAST Space-Time Communication Architecture," *Electronics Letters*, vol.35, No. 1, pp.14-16, Jan. 1999.

[17] W. K. Wai, C. Y. Tsui, and R. S. Cheng, "A Low Complexity Architecture of the V-BLAST System," *Proc. IEEE Wireless Communications and Networking Conf.*, vol.1, 2000, pp. 310-314.

[18] H. Abou Saleh and W. Hamouda, "BER Performance of MIMO-SM with Zero-Forcing in Spatially Correlated Ricean Fading," *IEEE*, 2009.

[19] C. Shen, H. Zhuang, L. Dai, and S. Zhou, "Detection Algorithm Improving V-BLAST Performance over Error Propagation," *Electronics Letters*, Vol. 39, No. 13, June 2003.

[20] A. Sadek, W. Su, and K. J. Ray Liu, "Diversity Analysis for Frequency-Selective MIMO-OFDM Systems with General Spatial and Temporal Correlation Model," *IEEE Transactions on Communications*, Vol. 54, No. 5, May 2006

[21] D. Seethaler, H. Artes, and F. Hlawatsch, “Detection Techniques for MIMO Spatial Multiplexing Systems,” *Elektrotechnik und Informationstechnik*, vol. 122, no. 3, pp. 91-96, March 2005.

[22] L. Zhou and M. Nakamura, “Channel Estimation of Multiple Transmit Antennas for OFDM Systems with Cyclic Delay Preamble,” *IEEE*, pp. 583-587, 2005.

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

[24] S. M. Alamouti, “A simple transmitter diversity scheme for wireless communications,” *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1451-1458, Oct. 1998.

[25] IEEE 802.11a, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer in the 5GHz Band,” *IEEE Std 802.11a*, 1999

[26] L. Deneire, B. Gyselinckx, and M. Engels, “Training Sequence vs. Cyclic PrefixA New Look on Single Carrier Communication,” *IEEE Communications Letters*, Vol. 5, No. 7, July 2001.

[27] T. M. Schidl, D. C. Cox, “Low-Overhead, Low-Complexity [Burst] Synchronization for OFDM,” *IEEE International Conference on Communications*, Vol. 3, 1996, pp. 1301-1306.

[28] Comlab, “Quad Dual Band RF Transceiver Technical Reference Guide,” Oct. 2007.

[29] Maxim, “Single-/Dual-Band 802.11a/b/g World-Band Transceiver ICs,” *19-3455*, Rev 0, Oct. 2004.

[30] L-Com Inc., “HyperLink Wireless brand HyperGain Range Extender”, Jan. 2009

[31] Lyrtech, “SignalMaster Quad Users Guide,” Jan. 2009

[32] Lyrtech, “VHS-ADC/DAC Users Guide,” Jan. 2009

[33] A. V. Zelst and T. C. W. Schenk, “Implementation of a MIMO OFDM-Based Wireless LAN System,” *IEEE Transactions on Signal Processing*, vol. 52, pp. 483-494, Feb. 2004

[34] B. Fornberg and J. Zuev, “The Runge Phenomenon and Spatially Variable Shape Parameters in RBF Interpolation,” *Comput. Math. Appl. Journal*, pp.379-398, 2006

Appendix A

Architecture

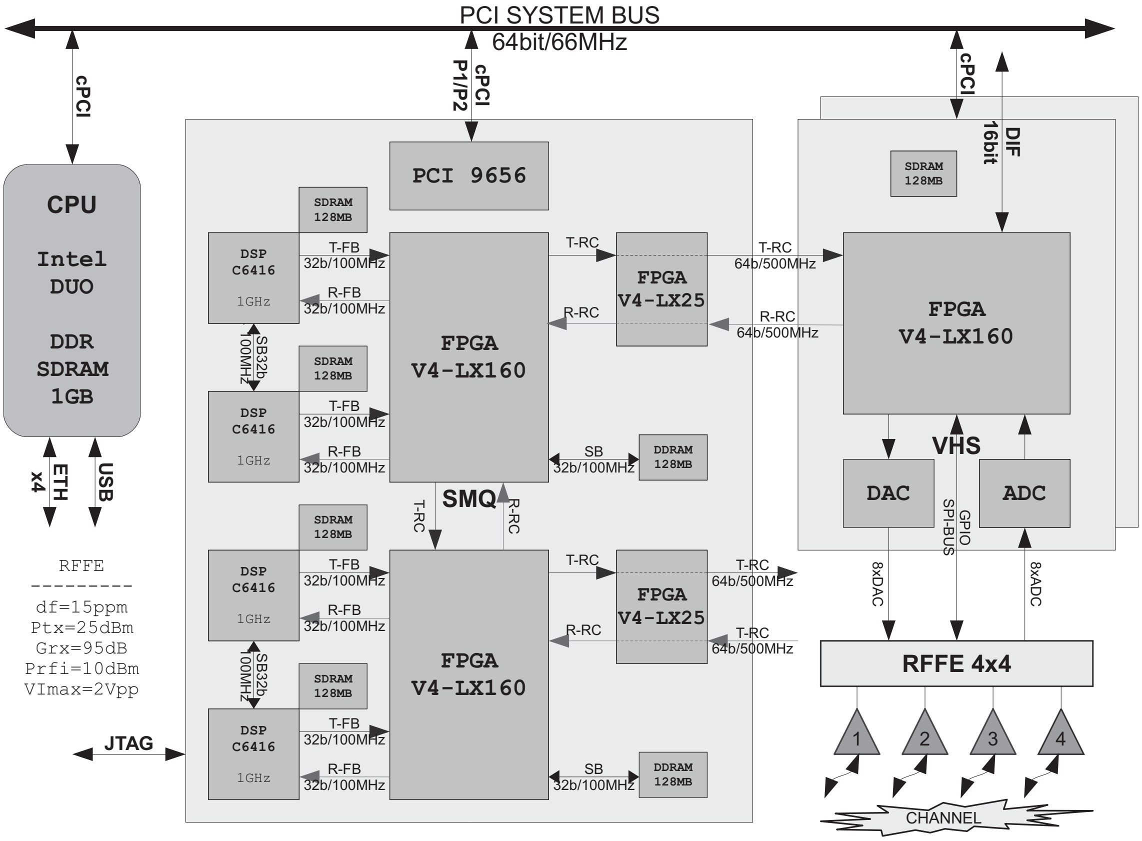


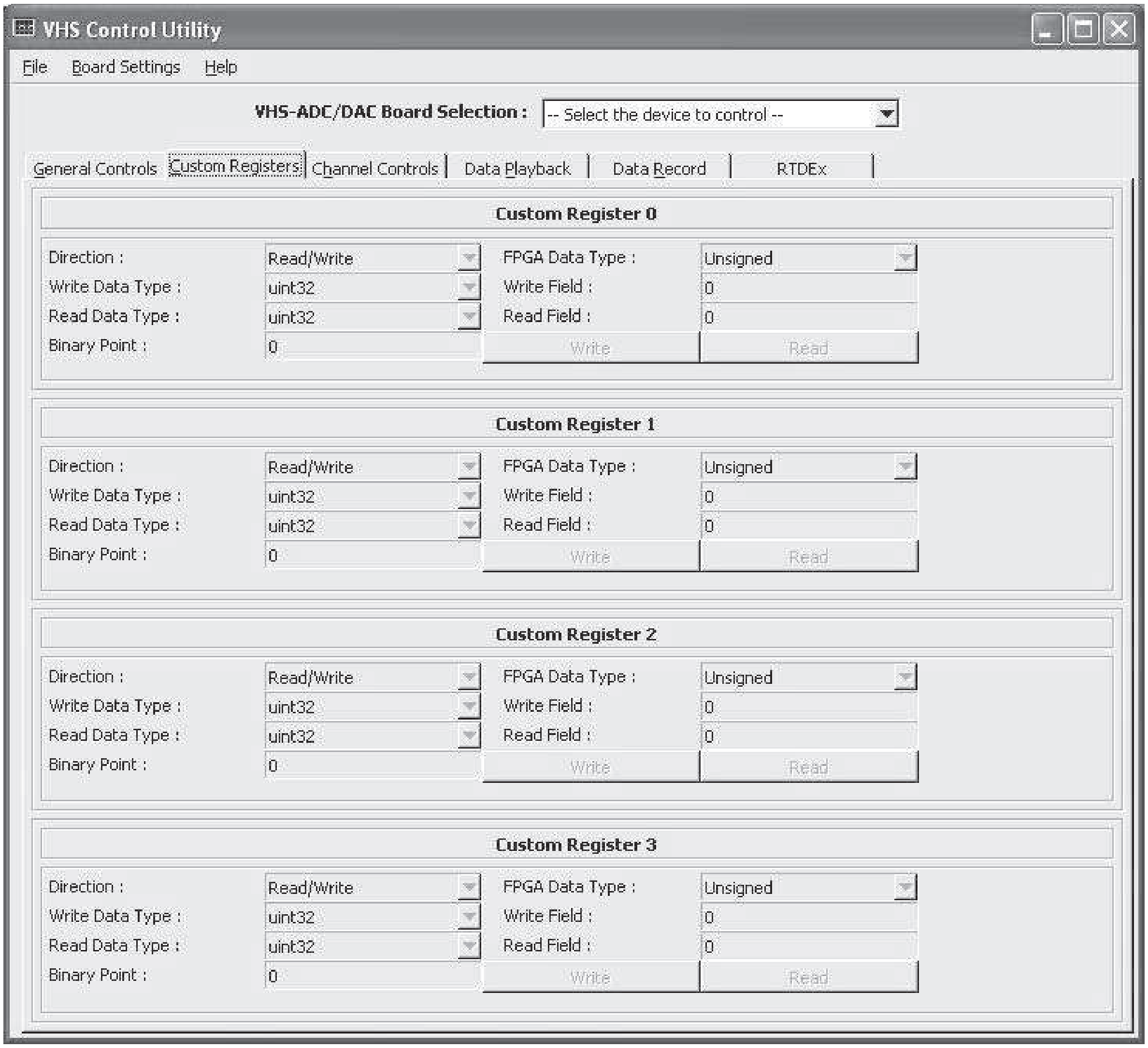
Figure A.1: Hardware Architecture

Appendix B

Measurement Utilities

Table B.1: Register Settings

REGISTER 1	0x87654321
8	MIMO Select (0 - SISO; 8 - MIMO)
7	FFT Start Delay (i.e.: 8) Rage: 0-15 (length of guard interval)
6	Phase Tracking Bypass and/or Frequency Offset Bypass
5	Bit 19-18: TX Antenna Select; Bit 17-16: RX Antenna Select
4	Not Used
3	Not Used
2	TX Output Baseband Gain
1	RX Output Baseband Gain
REGISTER 2	0x87654321
8-1	Number of Symbols for Transmission
REGISTER 3	0x87654321
8-6	Not Used
5	Baseband Loopback (Set to F)
4-3	Not Used
2-1	DAC Selector (0x00 - TX Mode; 0xAA - RX Mode; 0x55 - BB Probes)
REGISTER 4	0x87654321
8-7	Not Used
6-5	Gain Setting
4-1	Correlation Threshold Value for Time Synchronization



VHS-ADC/DAC Board Selection : -- Select the device to control --

Custom Register 0

Direction :	Read/Write	FPGA Data Type :	Unsigned
Write Data Type :	uint32	Write Field :	0
Read Data Type :	uint32	Read Field :	0
Binary Point :	0		

Custom Register 1

Direction :	Read/Write	FPGA Data Type :	Unsigned
Write Data Type :	uint32	Write Field :	0
Read Data Type :	uint32	Read Field :	0
Binary Point :	0		

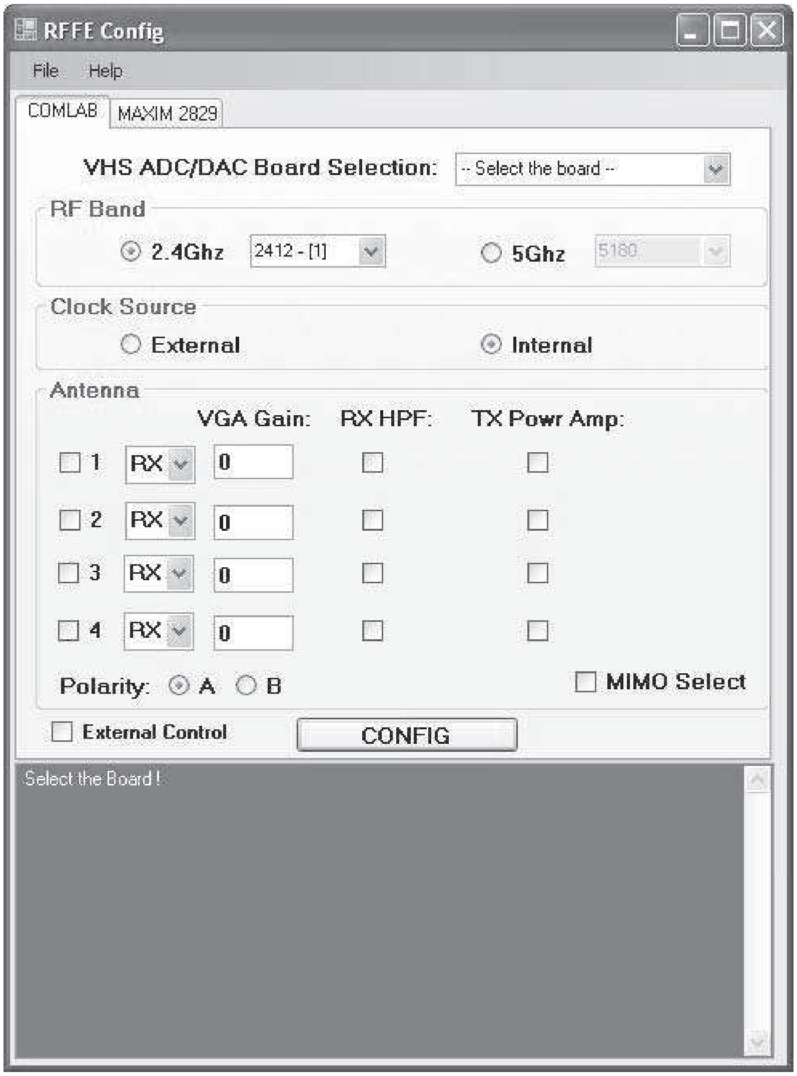
Custom Register 2

Direction :	Read/Write	FPGA Data Type :	Unsigned
Write Data Type :	uint32	Write Field :	0
Read Data Type :	uint32	Read Field :	0
Binary Point :	0		

Custom Register 3

Direction :	Read/Write	FPGA Data Type :	Unsigned
Write Data Type :	uint32	Write Field :	0
Read Data Type :	uint32	Read Field :	0
Binary Point :	0		

Figure B.1: Register Settings Interface on VHS Control Utility



VHS ADC/DAC Board Selection: -- Select the board --

RF Band

● 2.4Ghz 2412 - [1]   ○ 5Ghz 5180

Clock Source

○ External   ● Internal

Antenna

		VGA Gain:	RX HPF:	TX Powr Amp:
□ 1	RX	0	□	□
□ 2	RX	0	□	□
□ 3	RX	0	□	□
□ 4	RX	0	□	□

Polarity: ● A ○ B   □ MIMO Select

□ External Control

Select the Board !

Figure B.2: RFFE Gain Settings