

Design and Hardware Implementation Of a Cooperative Communication System

Binbin Jia

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science (Electrical Engineering) at
Concordia University
Montreal, Quebec, Canada

March 2007

© Binbin Jia, 2007



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-28917-4
Our file *Notre référence*
ISBN: 978-0-494-28917-4

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

ABSTRACT

Design and Hardware Implementation Of a Cooperative Communication System

Binbin Jia

Multiple Input, Multiple Output (MIMO) antenna systems have been widely studied. They play a key role in the next generation communication systems because of their capability to provide an extremely high capacity. However, the cost of using a large number of antennas should be considered when MIMO is put into practice. In order to reduce the cost of devices, there is another method called cooperative communication.

In a cooperative communication system, each single antenna (node) shares its information with its nearby antennas (nodes), and then those antennas transmit together their data towards the destination, therefore, they generate a virtual MIMO system. In this thesis, we present a new framework, which is a combination of the detect-forward cooperative method, channel coding, and space time coding methods. We assume that the cooperative system includes an inter-user channel (between nodes) and an uplink channel (from the nodes to the destination) that are subject to independently distributed, slowly-varying and flat Rayleigh fading. Due to the fact that the inter-user channel is less noisy than the uplink channel, we apply the 16 QAM modulation in the inter-user channel in order to acquire higher data rate. In the uplink channel, one user and its collaborator send bits together to the destination. Therefore, Alamouti space time code can be used. To

obtain a better performance and keep the same bandwidth, we utilize a Rate-compatible punctured convolutional code (RCPC). Simulation shows that improved performance can be achieved compared with that of a non-cooperative system. Based on this new system scheme, we implement the uplink receiver, which consists of a pair of parallel Square Root Raise Cosine (SRRC) filters, the Alamouti decoder, and the Viterbi decoder for decoding of the RCPC codes. In order to save the area, a parallel sequence of Alamouti decoder is controlled by the Moore state machine; a simplified method of the Branch Metric Unit (BMU) is introduced; the in-place scheduling is used in the Path Metric Unit (PMU).

The design is modeled in Very high speed integrated circuit Hardware Description Language (VHDL) and synthesized on a single chip FPGA (Xilinx Virtex 2 Pro). According to the RTL level and the gate level simulation results, the receiver can work at a speed of 12 Mbps with Virtex 2 pro FPGA.

ACKNOWLEDGEMENTS

Foremost, I am deeply indebted to my supervisor, Dr. M. Reza. Soleymani, for giving me the opportunity to work with him. His continuous support and valuable suggestions helped me all the time during my research. I also would like to express my gratitude to him for creating a convenient research environment and providing all advanced development tools and devices for my research works.

My parents and my grandmother, though not with me in Canada, deserve special acknowledgements. They have supported me with their love, care and encouragement. Owing them my success, I will never be able to thank them enough. I also want to thank my boyfriend Yan Liu, who always encourages me to strive for my objective.

Special thanks would go to all specialist, administrators and technical staff for their great assistance in using design tools. Finally, I want to express my thanks to all the students in our Wireless Satellite and Communication Lab. I benefited a lot from their friendly help.

TABLE OF CONTENTS

LIST OF FIGURES	VIII
LIST OF TABLES	X
LIST OF ABBREVIATIONS	XI
CHAPTER ONE INTRODUCTION	1
1.1 Development of Cooperative Communication	2
1.2 Development of FPGA	4
1.3 Contribution of the Thesis	6
1.4 Outline of the Thesis.....	7
CHAPTER TWO BACKGROUND AND SOME EXTENSIONS.....	9
2.1 Channel Capacity.....	10
2.2 Fading Channel Capacity.....	10
2.3 Several issues of the Multi-Antennas Channel	11
2.3.1 <i>Spatial multiplexing (SM)</i>	12
2.3.2 <i>Space-Time Coding</i>	14
2.3.3 <i>Beamforming</i>	14
2.4 Cooperative communication System	16
2.4.1 <i>Relay Channels</i>	17
2.4.2 <i>Cooperative Communication</i>	20
2.4.3 <i>Proposed Scheme of cooperative communication</i>	23
2.5 Extension summarization of Ad-Hoc networks.....	30
CHAPTER THREE VIRTUAL SPACE TIME CODINGFOR COOPERATIVE SYSTEM.....	32
3.1 Proposed System Model	33
3.2 Inter-user channel transmission	34
3.2.1 <i>LLR for 16 QAM in a Flat Fading Channel</i>	34
3.3 Uplink channels transmission	38
3.3.1 <i>Rate-compatible punctured convolutional codes</i>	39

3.3.2 Alamouti's 2X1 Space Time Coding	40
3.4 Energy Distribution.....	41
3.5 Error Performance Simulation	42
CHAPTER FOUR UPLINK RECEIVER HARDWARE IMPLEMENTATION ..	44
4.1 Finite Impulse Response(FIR) filter Implementation	46
4.1.1 Square Root Raised Cosine Filter.....	46
4.1.2 Finite Impulse Response Filter	48
4.1.3 Implementation of the FIR filter	49
4.1.4 Input/Output Interface Description.....	51
4.1.5 Multiply Accumulate Unit	52
4.1.6 Registers and Memory	52
4.1.7 Implementation Result	53
4.2 Alamouti Decoder Implementation.....	55
4.2.1 Alamouti Decoder 2 X 1 Implementation.....	55
4.2.2 Input/ Output Interface	56
4.2.3 Top Level Finite State Machine (FSM) for Alamouti Decoder.....	57
4.2.4 Arithmetic Logic Units Implementation.....	59
4.2.5 Extension to Alamouti Decoder 2 X 2	61
4.2.6 Implementation Result	62
4.3 Viterbi Decoder Implementation	63
4.3.1 Description of the Viterbi Algorithm	63
4.3.2 Design Specification	65
4.3.3 Arithmetic Logic Units Specification	65
4.3.4 Memory Unit Specification	74
4.3.5 Input/ Output Interface	76
4.3.6 RTL Level Simulation.....	77
4.3.7 Overall Architecture of the Receiver	80
CHAPTER FIVE CONCLUSION	83
5.1 Suggestions for Future Research	85
REFERENCES.....	87

LIST OF FIGURES

<i>Figure 2-1 Spatial Multiplexing Method</i>	13
<i>Figure 2-2 Beamforming Method in MIMO system</i>	15
<i>Figure 2-3 Beamformer Realization</i>	16
<i>Figure 2-4 Various relaying in wireless networks</i>	18
<i>Figure 2-5 Cooperative Communication</i>	22
<i>Figure 2-6 Amplify and forward scheme</i>	25
<i>Figure 2-7 Detected and Forward scheme</i>	26
<i>Figure 2-8 Channel mode for the detected and forward scheme</i>	26
<i>Figure 2-9 Code cooperation scheme</i>	28
<i>Figure 2-10 Transmission sequence in unlink channel of Coded cooperative system</i>	29
<i>Figure 3-1 Virtual MISO cooperation system scheme</i>	33
<i>Figure 3-2 Sixteen-QAM Constellation</i>	37
<i>Figure 3-3 The Comparison of Cooperative and Non-Cooperative Simulation Result</i>	43
<i>Figure 4-1 Receiver block diagram</i>	44
<i>Figure 4-2 Basic elements of a transmission system</i>	47
<i>Figure 4-3 Non-symmetric FIR filter</i>	48
<i>Figure 4-4 Sampling the SRRC pulse</i>	49
<i>Figure 4-5 I/O interface of 24 taps symmetric FIR</i>	51
<i>Figure 4-6 The MAC architecture</i>	52
<i>Figure 4-7 Even Symmetric FIR</i>	53
<i>Figure 4-8 Simulation result for 24 symmetric FIR</i>	54
<i>Figure 4-9 I/O layout of Alamouti Decoder</i>	56
<i>Figure 4-10 Top Level State Diagram of FSM for Alamouti Decoder</i>	58

<i>Figure 4-11 Architecture of 8X8 Multiplier.....</i>	<i>59</i>
<i>Figure 4-12 Structure of ADD/SUB Unit.....</i>	<i>60</i>
<i>Figure 4-13 Overall Architecture of Alamouti Decoder.....</i>	<i>60</i>
<i>Figure 4-14 Alamouti Decoder RTL Level Simulation Result.....</i>	<i>62</i>
<i>Figure 4-15 Trellis diagram for a (2, 1, 2) code with L = 15.....</i>	<i>63</i>
<i>Figure 4-16 Butterfly module.....</i>	<i>66</i>
<i>Figure 4-17 Block Diagram of BMU for soft decision</i>	<i>69</i>
<i>Figure 4-18 The ACS module</i>	<i>70</i>
<i>Figure 4-19 The PMM with ACS for Butterfly Unit</i>	<i>72</i>
<i>Figure 4-20 Register Exchange(RE) Method</i>	<i>73</i>
<i>Figure 4-21 I/O interface for Vitebi Decoder.....</i>	<i>76</i>
<i>Figure 4-22Viterbi Decoder K=7 simulation wave form</i>	<i>77</i>
<i>Figure 4-23Viterbi Decoder with punctuation ratio 4/7.....</i>	<i>78</i>
<i>Figure 4-24 The Architecture of the Uplink Receiver</i>	<i>79</i>
<i>Figure 4-25 Uplink Receiver RTL level simulation Result</i>	<i>81</i>
<i>Figure 4-26 Hardware Design Summary</i>	<i>82</i>

LIST OF TABLES

<i>Table 4-1: Values of 2's complement for the 24-taps symmetric coefficients</i>	<i>50</i>
<i>Table 4-2: Specification of 24 taps symmetric FIR</i>	<i>51</i>
<i>Table 4-3: The usage and control sequence of Paths in Each State</i>	<i>56</i>
<i>Table 4-4: The particular specifications for I/O Layout of Alamouti decoder.....</i>	<i>57</i>
<i>Table 4-5: The usage of Paths in Each State for 2X2 Alamouti Decoder</i>	<i>62</i>
<i>Table 4-6: Viterbi Decoder Parameters</i>	<i>65</i>
<i>Table 4-7: Simplification Process in the calculation of the Euclidean distance</i>	<i>69</i>
<i>Table 4-8: Viterbi Decoder Interface parameter</i>	<i>76</i>

LIST OF ABBREVIATIONS

GSM	Global System for Mobile Communications
CDMA	Code division multiple access
2G	Second Generation
3G	Third Generation
4G	Fourth Generation
W-CDMA	Wideband code division multiple access
OFDM	Orthogonal Frequency Division Multiplexing
VLSI	Very Large Scale Integration
SNR	Signal noise ratio
CRC	Cyclic Redundancy Check
ASIC	Applicant Specific Integrated Circuit
FPGA	Field Programmable Gate Array
PLDs	Programmable Logic Devices
PLA	Programmable Logic Array
PAL	Programmable Array Logic
JTAG	Joint Test Action Group
CLB	Configurable Logic Blocks
LUT	Lookup Table
GAL	Generic PAL
PCI	Peripheral Component Interconnect
QAM	Quadrature amplitude modulation
BPSK	Binary phase-shift keying

QPSK	Quadrature phase-shift keying
8PSK	Eight phase-shift keying
VHDL	VHSIC Hardware Description Language
RTL	Register Transfer Language
MIMO	Multiple Input and Multiple Output
RCPC	Rate Compatible Punctured Convolutional Codes
BER	Bit Error Rate
SM	Spatial multiplexing
STC	Space Time Code
LST	Layered Space-Time Codes
STBC	Space-Time Block Codes
STTC	Space-time trellis codes
BLAST	Layer space time coding
LAN	Local Area Network
BS	Base Station
Wi-Fi	Wireless Fidelity
LLR	Log-Likelihood Ratio
MISO	Multiple Input and Single Output
VA	Viterbi algorithm
AWGN	Additive White Gaussian Noise
ADC	Digital Converter
DAC	Digital to Analogy Converter
LNA	Lower Noise Amplifier

DSP	Digital Signal Processing
FIR	Finite Impulse Response
SRRC	Square Root Raised Cosine
ISI	InterSymbol Interference
UMTS	Universal Mobile Telecommunications System
MAC	Multiply-Accumulate
FSM	Finite State Machine
ADD/SUB	Addition/Subtract
BMU	Branch Metric Unit
ACS	Add Compare Select Unit
SMU	Survivor Memory Unit
PM	Path Metric
BM	Branch Metric
MSB	Most Significant Bit
PMM	Path Metric Memory Unit
RE	Register Exchange
TB	Trace Back
OFDM	Orthogonal Frequency-Division Multiplexing

CHAPTER ONE

INTRODUCTION

Digital communication plays an important role in our daily life. The combination of advanced wireless communication algorithms and integrated circuit technology makes the realization of many new services feasible.

The wireless communication has gone through its first and second generation systems. While still a few first generation systems exist, most services are now second generation, e.g., Global System for Mobile Communications (GSM). Code division multiple access (CDMA) is an advanced technique used in some 2G systems and accepted for almost all third generation systems. Third generation (3G) is posed to provide enhanced broadband voice, video, data access. More specifically, W-CDMA is used as a wideband spread-spectrum mobile air interface. It utilizes the CDMA signaling method to achieve higher rates and to support more users. Beyond 3G, there are two candidates enabling technologies to be used in the fourth generation systems. These are Orthogonal Frequency Division Multiplexing (OFDM) and Multiple Input, Multiple Output (MIMO) systems. OFDM reduces the impact of fading by spreading out signals over an interval of time. MIMO depends on multipath to send multiple versions of data streams that are transmitted from several antennas. By doing this, the spectral efficiency is greatly

increased. Meanwhile, the rapid development of Very Large Scale Integration (VLSI) and computer technology allows increasingly complex and powerful wireless devices to be implemented. To meet the requirement of multimedia communication, new schemes have been proposed in order to gain diversity. Cooperative communication is one of the new approaches used for single- antenna mobiles sharing their own antennas in a multi-user environment. In this thesis, our focus is on cooperative communication system design and the implementation of the receiver for this system.

1.1 Development of Cooperative Communication

Cooperation is a generalization of the relay channel; multiple sources with information to transmit also serve as relays for each other. Combinations of relaying and cooperation are also possible and are often referred to as “cooperative communications.”

The classical relay channel model is comprised of three terminals: a source that transmits information, a destination that receives the information, and a relay that both receives and transmits information to enhance communication between the source and destination. Problems in general relay channel appeared early in the area of information theory. Van der Meulen originally introduced three terminal communication in [1], [2], Cover and El Gamal provided the upper and lower bounds on the capacity for the relay channel in [3]. More recently, models with multiple relays have been examined. In order to make the relay channel symmetric, a parallel relay channel mode with coding strategies was introduced by Schein and Gallager [4], [5]. More sophisticated channels were also studied: the cooperative scheme with two transmitters and receivers was analyzed by Cover and El Gamal [3]. Willem and others extended the above model to the

multiple-access channels. All of these models fall within various cooperation with generalized feedback [6], [7], [8], [9]; Kramer, Wilingaarden [10] and Sendonaris et al [11], [12], [13] examined a channel model where the mobiles share the same relay between themselves and the base station.

Although we have an increased understanding of the benefits of multi-antenna systems in wireless channels [14], transmitting diversity may be impractical due to the cost, size, power and hardware limitations. Many have come to the realization that multiple relays can emulate the strategies designed for multiple transmit antenna systems and offer significant network performance enhancements in terms of various metrics, including: increased capacity (or larger capacity region), improved reliability because of diversity gain, diversity-multiplexing tradeoffs, and reduced frame or symbol error probabilities.

Therefore, cooperative communication is introduced in order to provide spatial diversity in a fading environment, as well as envision a collaborative scheme where both terminals help one another to communicate by acting as relays for each other. Various proposed schemes have been provoked by the potential advantage of cooperative communication. Three frequently used methods are called amplify-and-forward, detect-and-forward, and code-cooperative, respectively.

The amplify-and-forward method was first introduced by Lanemen et al [15]. As the name implies, the idea of this method is that one user simply amplifies noisy signal versions it received from the other user and retransmits these amplified signals to the destination. Although this scheme makes it challenging to sample, amplify and retransmit signals, it leads more opportunities for the further research. The detect and forward method is one of the closest to the traditional relay. It was developed by Sendonaris et al

[16], [17], [18]. Each mobile receives a noisy version of the partners' transmitted signal, and combine it with its own information bits to transmit towards the base station. The full diversity can be achieved by introducing CDMA implementation. The application of the spreading codes creates two separate channels, while, time diversity between the mobiles has been gained in [18]. Moreover, Laneman et al [15], [19] proposed a hybrid detect-and-forward method to overcome the problems of the original scheme. The method is that when the base station has no idea about the optimal detection for the inter-user channel, the partner may forward an erroneous version of the user's bits. It is shown that the user only applies the detect-and-forward method when the channel has high SNR but communicates under non-cooperative mode when the channel has low SNR . The coded cooperation [20], [21], [22] is a scheme to utilize channel coding. It is assumed that each user has M information bits per frame, which is encoded into N bits per frame that is divided into $N1$ and $N2$. The transmission of the N coded bits takes two steps. First, the users broadcast the $N1$ bits both to its partner and the base station. And during the second step, if a user can decode a partner's message, determined by the CRC code, the user will compute and transmit $N2$ bits for the partner, whereas, it will send the rest of $N2$ bits. In our work, we proposed a new scheme in cooperative communication with a combination of the detect-forward method, channel coding, and space time coding.

1.2 Development of FPGA

An Application Specific Integrated Circuit (ASIC) is a circuit which performs a specific function in a particular application. It is often optimized for the area and the performance. Almost all ASICs and programmable logic chips today use CMOS

technology. To reduce the time it takes to produce for market, Field Programmable Gate Arrays (FPGAs) can be used instead of ASICs. ASICs have the fastest speed, more power efficiency and optimal resource saving; however, they require a longer design cycle and are more costly. Therefore, they are not suited for prototype products and products with small volumes of production.

FPGAs are the continuation of the trend starting with the introduction of Programmable Logic Devices (PLDs). PLDs were introduced thirty years ago. Different from microprocessors, the idea was to construct a combinational logic circuit that was programmable. In other words, a PLD is a general purpose chip and its hardware can be reconfigured to meet particular specifications.

The evolution of PLDs can be classified into three stages. The first PLDs, namely Programmable Array Logic (PAL) or Programmable Logic Array (PLA) used only logic gates but without flip-flops, so they only implemented combination circuits. Then, registered PLDs were developed with one flip-flop at each circuit's output to get the sequential function. In the beginning of the 1980s, Macrocells were introduced. Besides the flip-flop, Macrocells provide logic gates, multiplexers, feedback signal that create greater flexibility. They were also programmable. All the chips mentioned above as well as generic PAL (GAL) are included in Simple PLDs (SPLD). The next wave, was the introduction of the popular approach known as Complex PLD (CPLD) is derived by using JTAG support and logic standard interface. Finally, FPGAs were introduced by Actel, Xilinx, and others in the mid 1980s. They focused on the implementation of large and high performance circuits.

An FPGA contains a matrix of Configurable Logic Blocks (CLB), which are interconnected by an array of switch matrices. Its operation is based on a Lookup Table (LUT). A number of flip-flops support more sophisticated sequential architecture. Different from the CPLDs' non-volatile feature, most FPGAs are volatile because they use SRAM instead of EEPROM, and a similar Flash that is used by CPLDs. Moreover, some FPGA chips also include clock multiplication (PLL or DLL), PCI interface, multipliers, DSPs, etc. Several companies manufacture FPGAs, such as Xilinx, Actel, Altera, Quicklogic, Atmel and so on.

Many researchers focus on the issues of hardware implementation for various elements of the receivers. These include filters [23], [24], [25] decoders [26], [27], and so on. Our design architecture is associated with the requirement of our proposed cooperative communication scheme. We chose Xilinx Virtex 2 pro series FPGA for our implementation. The clock frequency of this FPGA can reach as high as 100 MHz.

1.3 Contribution of the Thesis

The contribution of this thesis is to construct a virtual space time coding for a cooperative system that is developed from the traditional detect-forward approach and combined with well known space time block coding and channel coding. The corresponding hardware implementation is also introduced.

By assuming that the cooperative system contains both an inter-user channel (between users) and an uplink channel (from users to the destination), we applied 16 *QAM* modulations for inter-user channel in order to achieve higher data rates. The log likelihood ratio approach is used for detecting signals in the inter-user channel. Then, for

the uplink channel, one user and its partner, i.e, the node with which it cooperates, will send bits together to the destination. Therefore, the application of Alamouti space time coding in the uplink channel is justified. For obtaining a better performance and keeping the same bandwidth, we utilized a rate-compatible punctured convolutional code for both the source and its partner (the relay). Improved better performance can be achieved compared to the non-cooperative system.

For the receiver hardware implementation, we present the complete VLSI design for the uplink receiver, which consists of a pair of parallel Square Root Raise Cosine Filters, the Alamouti decoder controlled by the Moore state machine, and the Viterbi decoder for RCPC code. The design is modeled with VHDL (VHSIC Hardware Description Language) in RTL (Register Transfer Language) level and the simulation is performed by Modelsim. After synthesizing with Xilinx tools, placing and routing has been verified by gate level simulation.

1.4 Outline of the Thesis

In Chapter 2, the background of cooperative communication and related issues is presented. First, we discuss different issues relate to the multiple input multiple output (MIMO) system. Then, the relay channel is reviewed as an example of a cooperative communication scheme. Finally, the proposed approaches for cooperative communication are summarized. In addition, the ad-hoc networking is introduced at the end.

In Chapter 3, we present our new framework where the virtual space time coding for cooperative communication is employed. Starting with the overall system model illustration, the protocol and algorithm used in our scheme will be introduced. Moreover,

at the end of the chapter, simulation results showing the performance of the proposed scheme is presented and conclusion is provided.

In Chapter 4, the detail hardware architecture of the uplink receiver is addressed and the implementation details of differences including the components, including the Square Root Raised Cosine Filter, Alamouti Decoder, and Viterbi Decoder for RCPC coding, are specified clearly.

Then, an efficient architecture that combines the different components is given. Meanwhile, the result of RTL level simulation is presented. The total hardware design summary shows that the total area, consumption on the test FPGA board and the estimated achievable receiving speed.

In Chapter 5, the conclusion and future work are presented.

CHAPTER TWO

BACKGROUND AND SOME EXTENSIONS

The purpose of this chapter is to provide an overview of the background for this thesis. In the first three sections, we give an introduction of the channel capacity and some further work on the capacity of fading channels. Then, the development of Multiple Input and Multiple Output antenna system is discussed. Furthermore, we provide a systematic overview of in the Cooperative Communication; the pros and cons of various schemes are summarized. Then, some issues about Ad-hoc networking are mentioned in the end of this chapter.

2.1 Channel Capacity

One of the most famous results of information theory is Shannon's theorem [35]. It states that given a noisy channel with channel capacity C and information transmitted at a rate R , if

$$R < C$$

There exists a code that allows the probability of error at the receiver to be made arbitrarily small. This means that theoretically, it is possible to transmit information without error at any rate below a limiting rate, C . The converse is also important. If

$$R > C$$

an arbitrarily small probability of error is not achievable. All codes will have a probability of error greater than a certain positive minimal level, and this level increases as the rate increases. So, information cannot be guaranteed to be transmitted reliably across a channel at rates beyond the channel capacity.

2.2 Fading Channel Capacity

Since there are a lot of conditional constraints in wireless communication, several different characteristics for the Shannon capacity extend depending on different channels conditions. A lot of research has been done on different fading channels. There are several cases studied when the fading process is ergodic. An interleaved fading channel whose state is known to the receiver is analyzed by Thomas H.E .Ericson [30]. If the receiver can share channel state information with the transmitter, by means of a separate feedback channel, then the transmitter can adapt transmitted signals to the channel states. The capacity of this fading channel with channel side information is studied by Andrea

J. Goldsmith and Pravin P. Varajya.[31]. Abou-Faycal et. al [28], [29] were the first to examine the case where the channel state information is available neither to the transmitter nor to the receiver. However, when the fading process turns to non ergodic, the wireless channel is appropriately modeled as a family of channels which is called a compound channel [32]. The Shannon capacity will be arbitrarily small or zero since the effects of multipath fading may be small or zero. Thus, Shannon capacity is not a useful tool for system design in such scenarios.

2.3 Several issues of the Multi-Antennas Channel

Since the available radio spectrum is limited, the communication capacity requirements cannot be achieved without increasing spectral efficiency. A number of methods are utilized in the single antenna channel, for example, turbo coding [33] and parity check coding [34], to approach the Shannon capacity limit [35]. However, in recent years, there is a new trend for the communication which involves multi-antenna physical arrays at the transmitters and/or receivers in a wireless system. These multi-antenna systems are theoretically able to provide increased throughput, and better error performance than traditional systems.

The particular aspect that is used by MIMO systems is called Multi-Path propagation . It is the propagation phenomenon that occurs when the radio signals sent from the transmitter bounce off some intermediate objects before reaching the receiver. As a result, the signals may reach the receiving antenna by several paths, and at different times. But in traditional single antenna systems, multi-path propagation degrades the performance of the system. The multiple propagation paths can cause multiple "copies"

of a signal to arrive at the receiver at different times. These time delayed signals then become interference when trying to recover the signal of interest. However MIMO systems aim to use this multiple path propagation to obtain performance improvement.

Wireless MIMO communication can be sub-divided into three main categories, spatial multiplexing [36] for enhancing the peak data transmission rate, transmitting diversity methods such as space-time coding [37] for enhancing the robustness of the transmission, and beamforming [38] technologies for improving received signal gain and reducing interference to other users. Thus, it is possible to construct a MIMO system with both spatial multiplexing and diversity benefits.

2.3.1 Spatial multiplexing (SM)

It is an advanced transmission technology that increases the bit rate in a wireless radio link without additional power or bandwidth consumption. In addition, spatial multiplexing provides linear increase of the performance with the number of antennas.

Assuming that N antennas are used at the transmitter and receiver respectively; at the transmitter, the streams of information symbols are split into N substreams (see in Figure 2-1) and allocated to N transmit antennas. Since there are different obstructions in the environment, each signal suffers multi-path propagation. Eventually, receiver antennas receive noisy signals with random phase and amplitude. For every substream, the set of N received phases and N received amplitudes constitutes its spatial signature.

At the receiving array, the spatial signature of each of the N signals is estimated. Based on this information, a signal processing technique is then applied to separate the signals, Linear or non-linear receivers can be used providing a range of performance

trade-off. For example, a linear SM receiver can be viewed as a bank of superposed spatial weighting filters where every filter aims at extracting one of the multiplexed substreams by spatially nulling the remaining ones. This assumes that the substreams have different signatures.

When there are different numbers of the antennas in transmitter and receivers (i.e., M antennas in the transmitter while N antennas in the receiver), the rate improvement factor allowed by SM is the minimum of these two numbers. Additional antennas on the transmitter or on the receiver are then used for diversity purposes and improve further the link reliability.

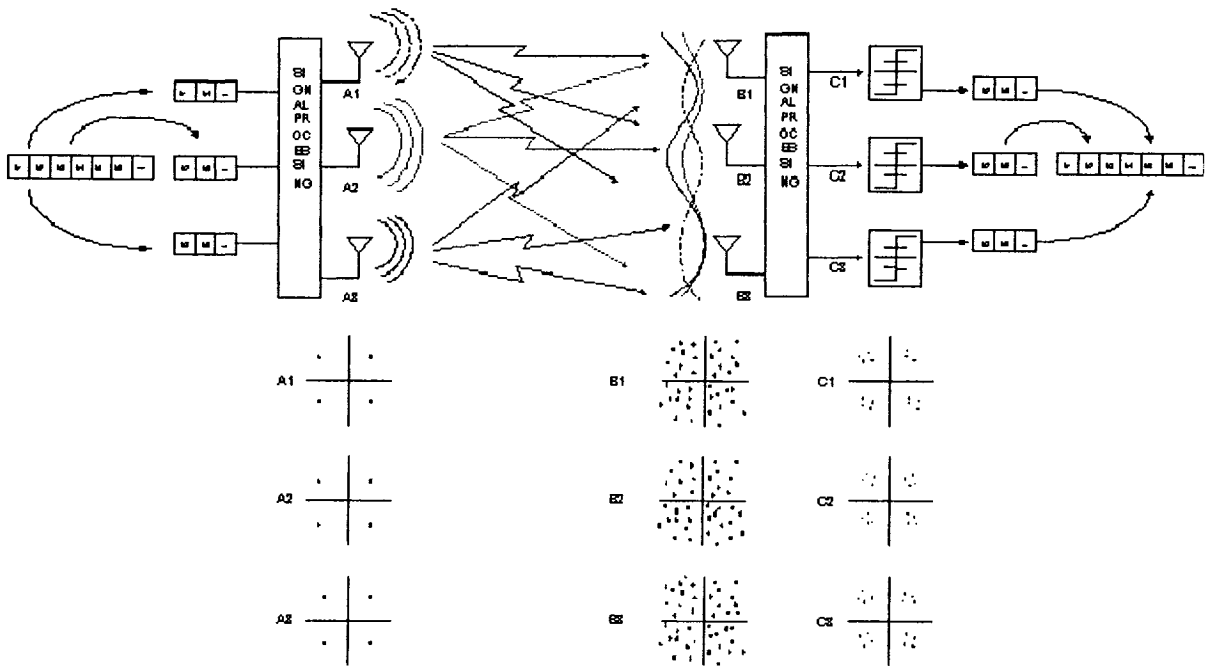


Figure 2-1 Spatial Multiplexing Method

2.3.2 Space-Time Coding

A space-time code (STC) is a method employed to improve the reliability of data transmission by using multiple transmitting antennas. Basically, STCs work by transmitting multiple, redundant copies of information bits to the receiver, in order to provide the possibility that at least some information bits can be received in a good condition and decoded reliably. There are three main categories of STCs: Layered Space-Time Codes (LST), Space-Time Block Codes (STBC) and Space-time trellis codes (STTC).

The distinguishing feature of LST is that it allows processing of multidimensional signals in the space domain in dimension of space. The method relies on powerful signal processing techniques at the receiver and conventional one dimension channel codes. The work of Foschini and Gans [39], [40] on the BLAST system is a notable achievement; Space-time block codes (STBCs) [41], [42] act on a block of data at once (similarly to block codes) and provide diversity gain. The work of Alamouti [41] on simple block codes that achieve full diversity and have particularly simple, but efficient linear decoding algorithms; Space-time trellis codes (STTCs) distribute a trellis code over multiple antennas and multiple time-slots to provide both coding gain and diversity gain. We can refer to the work of Tarokh et. al [43] on space-time trellis codes.

2.3.3 Beamforming

Beamforming in MIMO systems is to extend the range of existing data rates by transmitting and receiving beamforming. Using an array of antennas, one can receive a number of signals, which are broadcasted from different locations at same time. By linear

combining the antenna outputs, desired signals can be restored from the interference of the other signals and noise. The task of the adaptive beamformer is to compute and derive the proper weight vectors using adaptive beamforming algorithms.

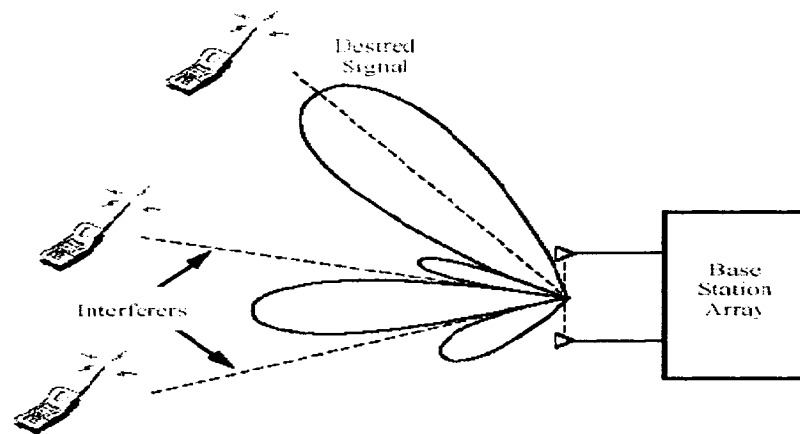


Figure 2-2 Beamforming Method in MIMO system

The beamforming method steers the gain pattern of an adaptive array according to a desired direction through either beam steering or null steering signal processing algorithms. Adaptive beamforming can provide substantial gains (of the order of $10\log(M)$ dB, where M is number of array elements) as compared to omni-directional antenna system.

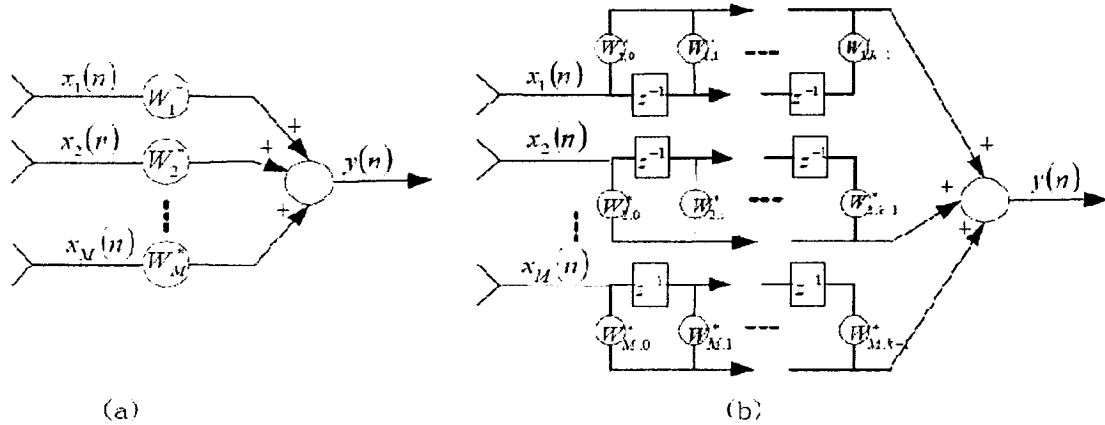


Figure 2-3 Beamformer Realization
 (a) For narrow band signal
 (b) For broad band signal

2.4 Cooperative communication System

Next generation wireless communication (third generation and beyond) will bear little similarity in mainly voice cellular systems to first- and second- generation systems. In order to meet the demands of multimedia communication, next generation cellular system must employ advanced algorithms and techniques that not only increase the data rate, but also enable the system to guarantee the quality of service desired by various media classes. The techniques currently being investigated for meeting these goals include advanced signal processing, modulation, detection, source and channel coding, specifically for the wireless environment, and using various forms of diversity [44], [45], [46], [55], [56]. Among these techniques, diversity is principal importance due to the nature of the wireless environment.

In wireless communication, the signal attenuation from mobiles varies from one to another since the mobile radio channel suffers from fading. By transmitting or processing independent copies of the signal, applying diversity is a great method that can effectively combat the deleterious effects of fading. There are some well known forms of diversity,

which are called spatial diversity, temporal diversity, and frequency diversity. In particular, spatial diversity relies on transmitting signals from different locations. Therefore, it allows independently faded versions of the signal at the receiver.

We begin by reviewing work on the relay channel since it is the origin of the cooperative communication. Then, we draw an outline for cooperative communication, and the fundamental construction is presented in subsection 2.5.2. Next, in subsection 2.5.3, we cite several various architecture schemes which have been further studied

2.4.1 Relay Channels

Relay channels and their multi-terminal extensions play a leading role as the root of cooperative diversity. Much of the work on these channel models to date has focused on discrete or additive white gaussian noise channels, and examined performance in terms of the well-known Shannon capacity (or capacity region) [47]. Only the more recent work has considered the issue of multi-path fading.

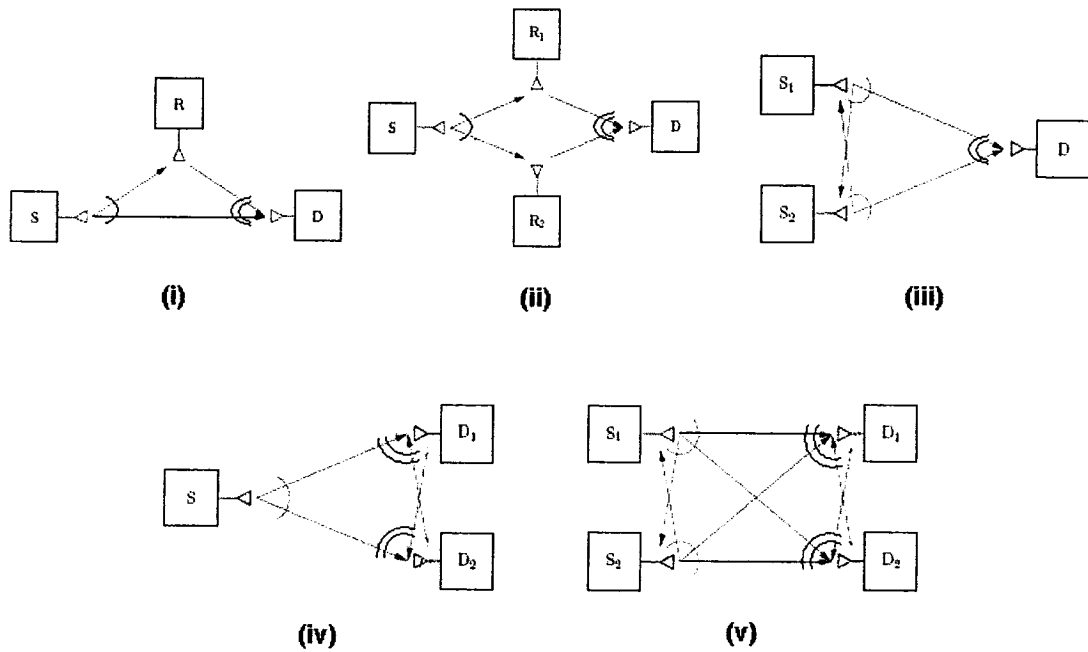


Figure 2-4: Various relaying in wireless networks:

- (i) classical relay channel
- (ii) parallel relay channel
- (iii) multiple-access channel with relaying
- (iv) broadcast channel with relaying
- (v) Interference channel with relaying.

Here, single arc means a broadcast channel, while double arcs mean a multiple access channel.

The classical relay channel models a class of three terminal communication channels originally introduced and examined by van der Meulen [1], [2]. This relay channel model, containing source, destination and relay, is the root of multi-hop communication's information theory. The distinctive property of relay channels is that certain terminals called "relays", receive, process, and re-transmit some information signal(s) of interest in order to improve performance of the system. As we illustrate in Fig. 2-4, in some cases, additional terminals in the network serve as relays without transmitting or receiving any information, while in other cases transmitting and/or receiving terminals can cooperate by serving as relays for one another. Cover and El Gamal provided a number of relaying

strategies, found achievable regions and developed lower and upper bounds on the capacity of the degraded relay channel, in which the communication channel between the source and the relay is physically better than the source-destination link. Generally these lower and upper bounds are unrealistic, but they are achievable in the class of degraded relay channels [3]. Although the class of degraded relay channels is mathematically convenient, we emphasize that none of the wireless channels in this class exist in practice.

The lower bounds on capacity, named achievable rates, are obtained from three different random coding schemes, referred to in [3] as facilitation, cooperation, and observation, respectively. The facilitation scheme is nothing special: the relay does not help the source actively, but rather, supports the source transmission by inducing as little interference as possible. However, the cooperation and observation schemes are more positive. In the cooperation scheme, the relay completely decodes the source message, and retransmits some information bits about that signal to the destination. Practical implementations of this cooperation scheme can be obtained with suitable configurations of multi-level codes [48]. Cover and El Gamal proposed the observation scheme, in which the relay encodes a quantized version of its received signal. On the receiving side, the destination combines information about the relay received signal with its own in order to create a better estimate of the source message. It is shown that the destination can essentially average to two observations of the source message in Gaussian noise channels.

Parallel relay channels and multiple-access channels with relaying, refer to Figure 2-4, are the most popular studied in this area. In order to make the relay channel symmetric,

Schein and Gallager [4] introduced the parallel relay channel model which includes many coding techniques in various regimes [5], and improved tighter converse results for certain discrete alphabet channels based on using distributed source coding [49]. Willems and others [6], [7], [8], [9] have studied the multiple-access channel with various degrees of cooperation and created feedback between the transmitting terminals. Also in a multiple-access channel model [10], the mobiles share a common relay between themselves and the base station as examined by Kramer and Wijnngaarden. Sendonaris et. al [11], [12], [13]. They have taken count of the cooperative diversity for a multiple-access channel with relaying and fading. The cooperation scheme where each transmitter and receiver employs two users is issued in [3]. If the signal-to-noise ratio (SNR) between the transmitting terminals is high, cooperative diversity of this case is to increase the achievable sum rate when transmitters and receivers have knowledge of the fading in ergodic fading, and to improve outage performance for non-ergodic fading [12], [13]. In addition, a number of results for extensions to multiple relays have developed in the work of Gupta, Gastpar, and others [50], [51], [52], [53].

2.4.2 Cooperative Communication

It is quite obvious that the advantage of multiple-antenna systems have been widely acknowledged. Specifically, some transmit diversity methods, such as Alamouti Coding have been utilized into wireless communication standards. However, the advantage of transmit diversity requires more than one antenna at the transmitter to implement. Therefore, many devices that are limited to one antenna could not make use of this feature. However a new class of approach, called cooperative communication, has been

widely studied in recent years. It makes single- antenna mobiles in a multi-user environment to share their antennas and generate a virtual multiple-antenna transmitter that allows them to achieve transmit diversity. In this section, we present the developments of this method in this emerging field.

In wireless communication, the signal attenuation from mobiles is varying from one to another since the mobile radio channel suffers from fading. By transmitting or processing independent copies of the signal, one can use diversity to effectively combat the effects of fading. There are some well known forms of diversity, which are called spatial diversity, temporal diversity, and frequency diversity. In particular, spatial diversity relies on transmitting signal from different locations, thus allowing independently faded versions of the signal at the receiver.

Although transmit diversity is clearly advantageous on a cellular base station, it may be impractical for any other scenarios. For example, in the uplink of a cellular system by the limitation of cost, size, power, and hardware, a wireless user may not be able to support multiple transmit antennas. In order to overcome this shortcoming, cooperative communication generates diversity in a new way by imitating transmit antenna diversity.

In the following context, we are going to take a systemic overview of cooperative communication. The basic idea dates back to multi-hop communications, which is also the original idea of ad-hoc wireless networks. Without a fixed infrastructure, the ad-hoc networks work by the relaying operation that overcomes the path loss incurred over large distances. By the same idea, multi-hop is also widely utilized in cellular and wireless LAN systems to provide higher quality of service, power saving and extended coverage.

In most case of multi-hop systems, the destination only processes the signal coming from the relay despite that the information theoretic model allows for the destination to listen to both the source and the relay. When the source is much further away from the destination than the relay, the received signal at the destination due to the source would be much weaker than the relay signal. Moreover, if the fading is taken into account, a considerable loss, especially in diversity, will be caused by this scheme. But on the other hand, the destination processes both signals from source and relay may not only overcome path loss, but also to provide diversity.

Therefore, cooperative communication is motivated by two principle ideas. First of all, it uses relays (or multi-hop) to provide spatial diversity in a fading environment. Second, envision a collaborative scheme where the relay also has its own information to send so both terminals help one another to communicate by acting as relays for each other (called “partners”).

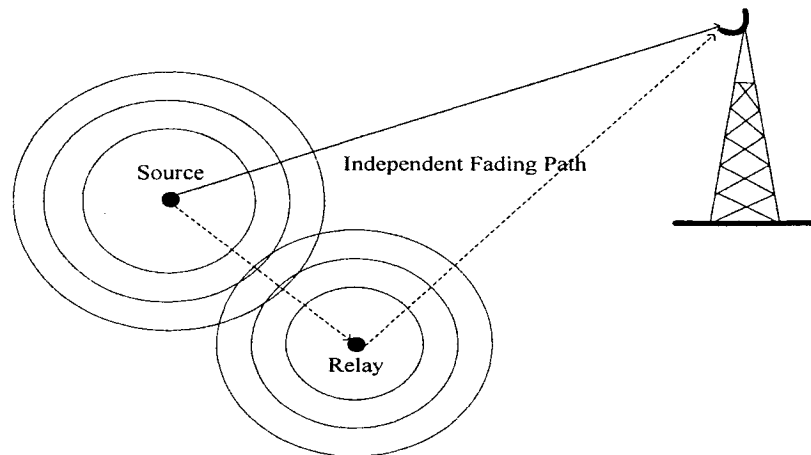


Figure2-5: Cooperative Communication

We first refer to Figure 2-5, there are two mobile nodes communicating with the same destination; each mobile node has only one antenna and cannot individually generate spatial diversity. While it is possible for one mobile to receive the signals from the other, in which case it can forward some version of “overhead” information along with its own data. Since the fading paths from two mobile nodes are statistically independent, intuitively, cooperation communication refers to processing this overhead information at the surrounding nodes and to retransmission towards the destination to create spatial diversity. This provides extra observations of the source signals at the destinations, one antenna mobile in the multiple users’ environment can share their antennas by creating a virtual MIMO system. Although the elements of this array are not co-located and are connected via noisy, fading links, some of the researches have already shown the benefits it gains, in terms of significant error performance, diversity and achievable data rate.

There are a lot of proposed schemes aimed at using the potential advantages of cooperative communication, where the higher throughput and reliability can be obtained. Several significant milestones have been achieved, and meanwhile, research and development of the concerned aspects is on the upswing.

2.4.3 Proposed Scheme of cooperative communication

In a cooperative communication system, mobile users may increase their effective quality of service that includes bit error rates, frame error rates, or outage probability. One wireless user is assumed to both transmit data and act as a cooperative agent for another mobile user. Maybe it is not true at every given point, but it is valid considering a stable statistical environment.

At this point, we try to briefly address the feasibility of cooperative communication. In cooperative communication, each user transmits both its own bits, as well as some information bits for its partner. Therefore, each user requires more bandwidth, but the spectral efficiency for each mobile node improves and the channel code rates can be increased due to cooperation diversity. On the other hand, it may seem that more power is needed because of power consumption used on transmitting for both itself and its partner. However, the point to be made is that the gain in diversity from cooperation allows the users to reduce their transmitting power and maintain the same performance. Therefore, the tradeoffs in the code rate and transmission power are observed.

Several studies have proved the usefulness of the cooperative approach. We now go over several of the main cooperative signaling methods. For the purpose of illustration, we consider two mobile nodes helping each other, but generally, it is quite possible that there are more than two nodes, which is another popular topic in ongoing research called partner assignment. There are three frequently used methods in the cooperative signaling. They are called amplify and forward, detect and forward and code cooperative, respectively.

A. Amplify- and- Forward Methods

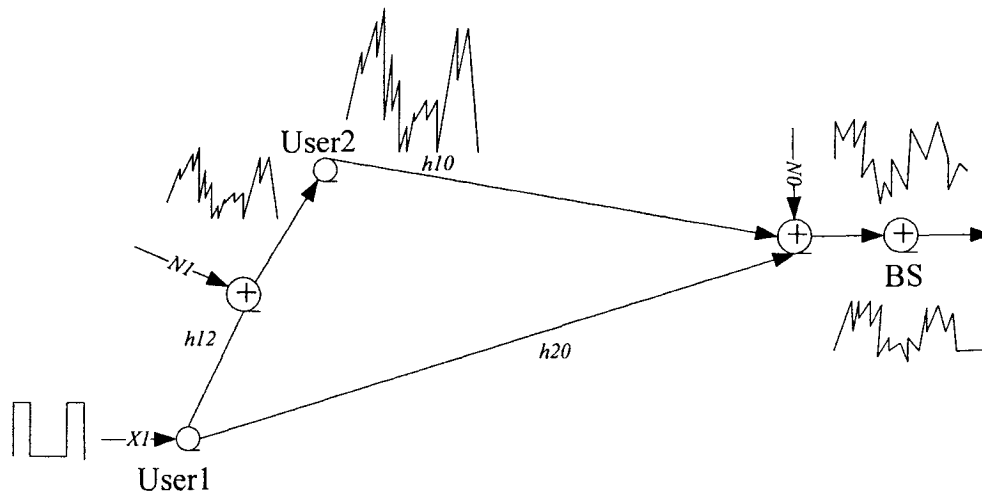


Figure 2-6: Amplify and forward scheme

Within the diversified approaches of the cooperative communication, the most ordinary method first developed by Lanemen et al [15], [19]. In this condition, for example, each user receives a noisy version of the signal transmitted by its partner (Figure 2-6). Then, the user simply amplifies its received signal, as an amplifying repeater [4], and retransmits these adjusted versions to the destination. The base station will combine the information sent by the user and partner and will make a final decision on the transmitted bits.

This method has both advantages and drawbacks. Although the noise of the partner is amplified in this scheme, on the receiver side, the base station is still able to make a better decision for the transmitted bits, because the information bits, which got in the receiver, are coming from two independently faded versions. On the other hand, this scheme brings technology challenges in sampling, amplifying, and retransmitting analog values. But this simple method makes a good example for further research in cooperative communication system.

B. Detect-and-Forward Methods

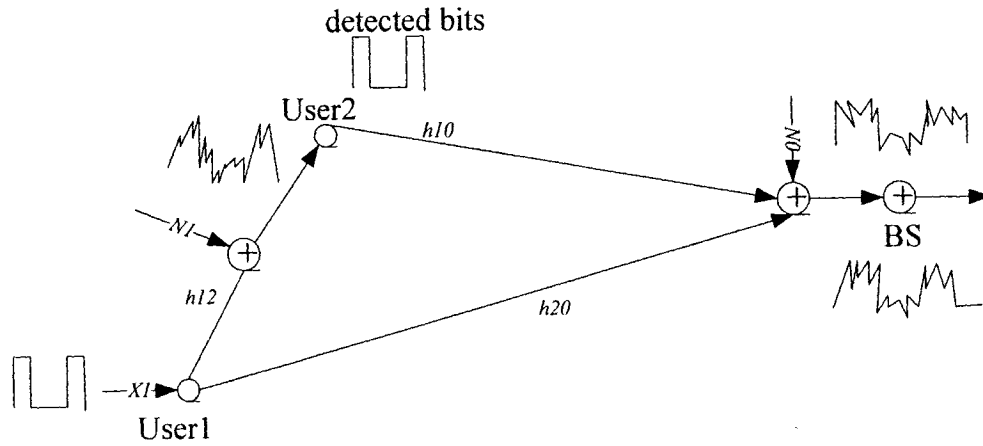


Figure 2-7: Detected and Forward scheme

The channel model can be illustrated in Figure 2-7. This is an example of detect-and-forward signaling introduced by Sendonaris et al [16], [17], [18].

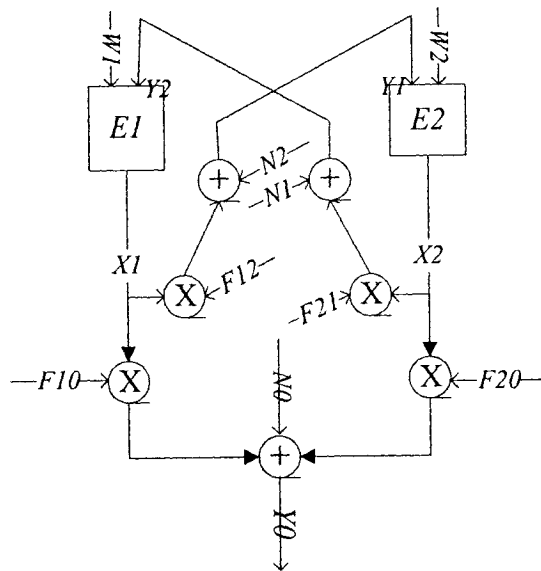


Figure2-8: Channel mode for the detected and forward scheme

As we can see, each mobile receives a noisy version of the partner's transmitted signal, and then it combines with its own signal to transmit towards the destination called the

base station. This approach is one of the closest to the traditional idea of relay. The whole process can be represented by the following formulas:

$$\begin{aligned} Y_0(t) &= F_{10}X_1(t) + F_{20}X_2(t) + N_0(t) \\ Y_1(t) &= F_{21}X_2(t) + N_1(t) \\ Y_2(t) &= F_{12}X_1(t) + N_2(t) \end{aligned} \quad (2-3)$$

where $Y_0(t)$, $Y_1(t)$, and $Y_2(t)$ are the base band models of the received signals at the BS. By employing two users, $X_i(t)$ is the signal transmitted by user i , for $i=1,2$, and $N_i(t)$ are the noisy version at the BS, user 1 and user 2 respectively, for $i=0,1,2$. The fading coefficient, $\{F_{ij}\}$, keep constant over at least one symbol period.

The CDMA implementation is also introduced by using spreading codes to create two separate channels, also time diversity among the mobiles was investigated in [18], then full diversity can be provided. Each user has its own spreading code, which denoted as $C_1(t)$ and $C_2(t)$. And $P_{i,j}$ indicates power allocation on various signaling. The two user's data bits are denoted as $Q_i^{(n)}$, where $i=1,2$ are the user indices; and n denotes the time index of information bits. In addition, $\hat{Q}_i^{(n)}$ represents the partner's hard-detected estimation of user i 's bit. By defining $X_1(t)$, $X_2(t)$ as the signals of user1 and user2 respectively, we introduce the two users' transmit signals

$$\begin{aligned} X_1(t) &= P_{11}Q_1^{(1)}C_1(t), \quad P_{12}Q_1^{(2)}C_1(t), \quad P_{13}Q_1^{(2)}C_1(t) + P_{14}\hat{Q}_2^{(2)}C_2(t) \\ X_2(t) &= \underbrace{P_{21}Q_2^{(1)}C_2(t)}_{\text{PeriodI}}, \quad \underbrace{P_{22}Q_2^{(2)}C_2(t)}_{\text{PeriodII}}, \quad \underbrace{P_{23}\hat{Q}_1^{(2)}C_1(t) + P_{24}Q_2^{(2)}C_2(t)}_{\text{PeriodIII}} \end{aligned} \quad (2-4)$$

Sending data to the BS is processing in *Period I*. While, *Period II* is in charge of both sending data to the BS and to each user's partner. After this data is detected by each user's partner, each user creates a cooperative signal to send to the BS during *Period III*.

The transmitted power is an inconstant value for three different periods. This approach contributes a power adaptability to channel condition. In other words, when the inter-user channel is in good condition, more power may be allocated to cooperative communication, however, the inter-user channel is not in good condition, more power may be used in it.

In this scheme, the base station needs to know the optimal detection for the inter-user channel. Otherwise, the partner forwards an erroneous version of the user's bit that may bring a detrimental problem. Then, Laneman et al [15], [19] proposed a hybrid detect-and-forward method. It is indicated that users only have to detect and forward their partner's data when the fading channel has high SNR, whereas users recover to a non-cooperative mode when that fading channel has low SNR.

C. Coded-Cooperation Methods

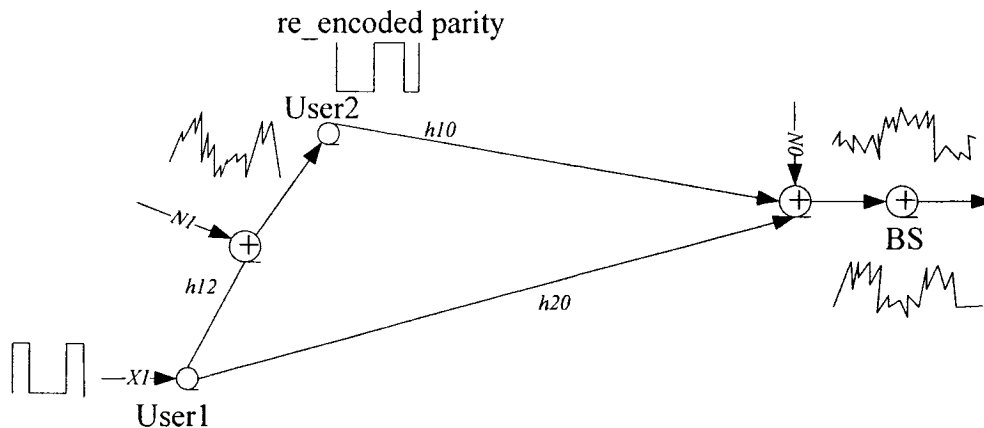


Figure 2-9: Code cooperation scheme

As the name implies, coded cooperation [20], [21], [22] is an approach to utilize channel coding. The idea of coded cooperation is to use the same overall rate for coding

and transmission; the redundancy for each partner is transmitted by the corresponding user.

It is assumed that each user has M information bits per block, and the information bits of each user are encoded into a code word with N bits per block, so that the code rate $R = M/N$. After being punctured, each codeword of length N is divided into two segments of lengths N_1 and N_2 , which also satisfy $N_1 + N_2 = N$. Then, the transmission of the N coded bits is divided into two successive time periods. During the first time period, the sub-codeword of rate $R_1 = K/N_1$ is broadcast by the user; and a noisy version of the coded message is received by the base station and the corresponding partner as well.

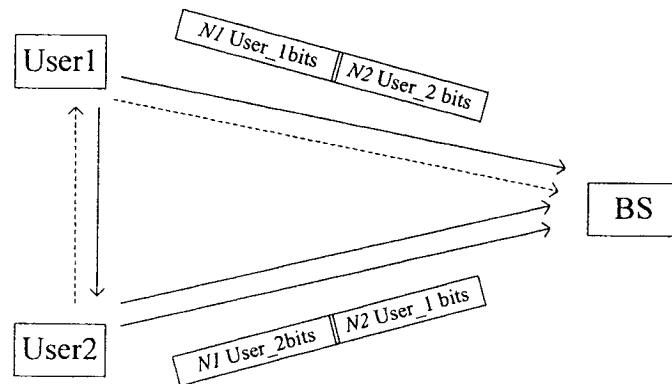


Figure 2-10: Transmission sequence in unlink channel of Coded Cooperative system

During the second time period, if a user can decode a partner's message, determined by the CRC code, the user will compute and transmit the N_2 bits for the partner. Whereas if a user cannot correctly decode a partner, N_2 additional parity bits for the user's own data will be transmitted. By the end of these two periods, there are four possible cases [22], [54] for the result of the second time period: In Case 1, both users successfully decode each other, therefore they each transmit for their partner in the second frame, resulting in the fully cooperative scenario. In Case 2, none of the user successfully

decodes their partner's first frame; the system reverts to the non-cooperative case automatically. In Case 3 and 4, first user successfully decodes its partner, but the second user does not successfully decode its partner. As a result, both the first and second users transmit N_2 bits for the second user. These two independent copies of the second user's bits are optimally combined.

There is an additional condition that the destination must know which of these four cases has occurred in order to correctly decode the received bits. Two methods have been utilized on this issue [60]. One is that the base station decodes according to the assumption of all the case until CRCs indicate successive decoding. Probabilistic analysis shows that it brings tiny average increase in computational complexity. The other way is by adding one additional bit which represents each user to indicate its state to the base station.

2.5 Extension summarization of Ad-Hoc networks

As we have emphasized, the development of cooperative diversity arise primarily in networks of radio terminals. We now summarize important results obtained in ad-hoc networks.

A wireless ad-hoc network is a computer network in which the communication links are wireless. Different from previous network technologies in which some designated nodes, usually with custom hardware to perform the task of forwarding the data, the network is ad-hoc because each node is willing to forward data for other nodes, and so the determination of which nodes forward data is made dynamically based on the network connectivity. Minimal configuration and quick deployment make ad hoc

networks suitable for emergency situations. A mobile ad-hoc network is a kind of wireless ad-hoc network, and the routers are free to move randomly and organize themselves arbitrarily; therefore, the network's wireless topology may change rapidly and unpredictably. Such a network may operate alone, or may be connected to the larger Internet.

Ad-hoc networks became a popular subject for research as laptops and 802.11/Wi-Fi wireless networking became widespread. It is a self-configuring network of mobile routers and associated hosts connected by wireless arbitrary topology. It was first introduced in [63], [64] as a wireless extension of packet switching in wire-line networks. Many of the academic papers evaluate protocols and abilities assuming varying degrees of mobility within a bounded space; and recent work in the information theory community has contributed some fundamental performance and scaling laws.

Gupta and Kumar, in a paper [65], prove that certain fixed ad-hoc networks containing M stationary terminals have total throughputs per terminal that decay to zero with increasing M in a constant area. The fixed protocol maximizes transport capacity (bit-meters/second) by having terminals transmit to their nearest neighbors. Shepard [66] draws essentially the same conclusion that terminals should transmit only to their nearest neighbors, by examining the asymptotic behavior of the interference from non-nearest neighbors. A mobile ad-hoc network is examined by Grossglauser and Tse [67]. They prove that a suitable cascade transmission policy which is a so-called multi-user diversity effect: each time the destination terminal receives that it will very likely be near either the original source terminal or an intermediate terminal carrying packet for the source.

CHAPTER THREE

VIRTUAL SPACE TIME CODING FOR COOPERATIVE SYSTEM

In this chapter, we present a new framework, which is a combination of detect-forward, channel coding, as well as space-time coding methods. First, we describe the overall system model. Then, the protocol and algorithms used in our scheme will be illustrated. Finally, we provide the analysis and simulation of this system's performance.

3.1 Proposed System Model

For simplicity, we consider a system consisting of a source, a relay and a destination. The distance between the source (relay) and destination is several times larger than the distance between the source and relay. In cooperative wireless communication, each node can act either as a transmitter or a relay. It is usually hard to distinguish between its spontaneous and supportive transmission. In order to avoid time conflict, we assume that the nearby node can only listen when one node is sending the information bits.

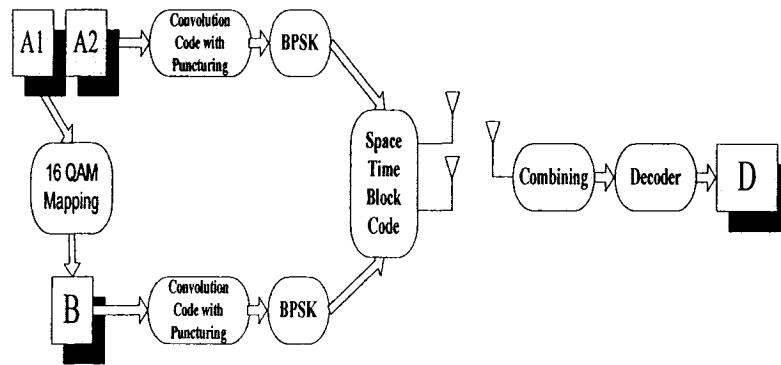


Figure 3-1: Virtual MISO cooperation system scheme

Figure 3-1 shows the entire communication divided into two steps. In the first time slot, source (A) transmits the information bits (A1) towards the nearby relay (B). The relay detects bits by coherent detection. During the second time slot, while source transmits the information bits (A2) to the destination, relay (B) transmits the bits it has detected to the destination simultaneously.

We assume that the cooperative system includes an inter-user channel (between nodes) and uplink channels (from nodes to destination), that are subject to independent

distributed, slowly-varying and flat Rayleigh fading. In our simulations, we assume that the channel state information is known at the receivers. The path gains between source-relay, source-destination and relay-destination are h_{AB} , h_{AD} and h_{BD} , respectively.

3.2 Inter-user channel transmission

In the first time slot, we focus on the channel between the source and the nearby node. To acquire higher data rate, we choose 16QAM modulation [58]. Therefore, we use $\log_2 M = \log_2 16 = 4$ bits that can be represented by (r_1, r_2, r_3, r_4) to represent a symbol. For instance, the symbol with coordinates $(-d, 3d)$ maps the 4-bit combination $r_1 = 1, r_2 = 0, r_3 = 0, r_4 = 1$. As we can see from the above, the channel coefficient between the source and relay is h_{AB} , the received signal Y corresponding to the transmitted symbol S_{A_1} can be written as

$$Y = h_{AB}S_{A_1} + N \quad (3-1)$$

Here h_{AB} is the complex fading channel coefficient with $E\{\|h\|^2\} = 1$ and $r.v$'s h for different symbols are assume to be *i.i.d* Raleigh distributed, and $N = N_I + jN_Q$ is a complex Gaussian $r.v$ of zero mean and variance $\sigma^2/2$ per dimension.

Before going further, let us take a look at the log-likelihood ratio (LLR) for the 16 QAM modulation.

3.2.1 LLR for 16 QAM in a Flat Fading Channel

The received symbol y corresponding to the transmitted symbol a can be expressed as

$$y = ha + n \quad (3-2)$$

The log likelihood ratio can be expressed as

$$LLR(r_i) = \log \left(\frac{\Pr\{r_i = 1 | y, h\}}{\Pr\{r_i = 0 | y, h\}} \right) \quad (3-3)$$

$$r_i, i = 1, 2, 3, 4$$

The optimum decision is $\hat{r}_i = 1$ if $LLR(r_i) \geq 0$, and 0 otherwise. We define two sets, $S_i^{(1)}$ and $S_i^{(0)}$, where $S_i^{(1)}$ comprises symbols with $r_i = 1$ and otherwise $S_i^{(0)}$ comprises symbols with $r_i = 0$ in the constellation, we have

$$LLR(r_i) = \log \left(\frac{\sum_{\alpha \in S_i^{(1)}} \Pr\{a = \alpha | y, h\}}{\sum_{\beta \in S_i^{(0)}} \Pr\{a = \beta | y, h\}} \right) \quad (3-4)$$

In our system, we assume that all the symbols are equally likely and that fading is independent of the transmitted symbols. Bayes' rule is used, and then we have

$$LLR(r_i) = \log \left(\frac{\sum_{\alpha \in S_i^{(1)}} f_{y|h,a}\{y | h, a = \alpha\}}{\sum_{\beta \in S_i^{(0)}} f_{y|h,a}\{y | h, a = \beta\}} \right) \quad (3-5)$$

Since $f_{y|h,a}\{y | h, a = \alpha\} = \frac{1}{\sigma\sqrt{\pi}} \exp\left(-\frac{\|y - h\alpha\|^2}{\sigma^2}\right)$, the function (3-5) can be written as

$$LLR(r_i) = \log \left(\frac{\sum_{\alpha \in S_i^{(1)}} \exp\left(-\frac{\|y - h\alpha\|^2}{\sigma^2}\right)}{\sum_{\beta \in S_i^{(0)}} \exp\left(-\frac{\|y - h\beta\|^2}{\sigma^2}\right)} \right) \quad (3-6)$$

After approximating by $\log(\sum_j \exp(-X_j)) \approx -\min_j(X_j)$, Equation (3-6) can be written as

$$LLR(r_i) = \frac{\left\{ \min_{\beta \in S_i^{(0)}} \|y - h\beta\|^2 - \min_{\alpha \in S_i^{(1)}} \|y - h\alpha\|^2 \right\}}{\sigma^2} \quad (3-7)$$

We define that $x \triangleq y/h = a + n/h = a + \hat{n}$, where \hat{n} is also a complex Gaussian R.V. with variance $\sigma^2/\|h\|^2$. Substituting x into Equation (3-7) and normalizing $LLR(r_i)$ by $4/\sigma^2$,

$$\begin{aligned} LLR(r_i) &= \frac{\|h\|^2}{4} \left\{ \min_{\beta \in S_i^{(0)}} \|x - \beta\|^2 - \min_{\alpha \in S_i^{(1)}} \|x - \alpha\|^2 \right\} \\ &= \frac{\|h\|^2}{4} \left\{ \min_{\beta \in S_i^{(0)}} (\|\beta\|^2 - 2x_I \beta_I - 2x_Q \beta_Q) - \min_{\alpha \in S_i^{(1)}} (\|\alpha\|^2 - 2x_I \alpha_I - 2x_Q \alpha_Q) \right\} \end{aligned} \quad (3-8)$$

Since the sets $S_i^{(1)}$ and $S_i^{(0)}$ are partitioned by either vertical or horizontal boundaries, two symbols in different sets closet to the received symbol lie on the same row when the boundaries are vertical, while, lie on the same column when the boundaries are horizontal.

As a consequence, for the bit r_1 , the two constellation symbols in $S_1^{(1)}$ and $S_1^{(0)}$ having closest distances to the received symbol satisfy the condition $\alpha_Q = \beta_Q$.

Therefore, for bit r_1

$$LLR(r_1) = \begin{cases} -\|h\|^2 x_1 d & |x_1| \leq 2 \cdot d \\ 2\|h\|^2 d(d - x_1) & x_1 > 2 \cdot d \\ -2\|h\|^2 d(d + x_1) & x_1 < -2 \cdot d \end{cases} \quad (3-9)$$

where $2d$ is the minimum distance between pairs of signal points.

Following the same rule, for bit r_2 ,

$$LLR(r_2) = \begin{cases} -\|h\|^2 x_Q d & |x_Q| \leq 2 \cdot d \\ 2\|h\|^2 d(d - x_Q) & x_Q > 2 \cdot d \\ -2\|h\|^2 d(d + x_Q) & x_Q < -2 \cdot d \end{cases} \quad (3-10)$$

For bit r_3 , we have

$$LLR(r_3) = \|h\|^2 d \{ |x_I| - 2d \} \quad (3-11)$$

And for bit r_4

$$LLR(r_4) = \|h\|^2 d \{ |x_Q| - 2d \} \quad (3-12)$$

Finding (r_1, r_2, r_3, r_4) from the $LLR(r_i)$, $i = 1, 2, 3, 4$, we can find the symbol from the constellation as shown in Figure 3-2.

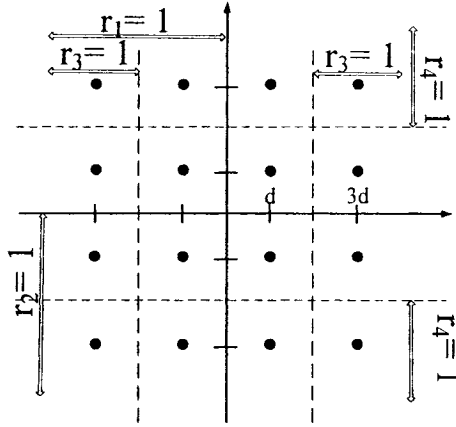


Figure 3-2: 16-QAM Constellation

Defining $z \triangleq Y/h_{AB} = S_{A1} + N/h_{AB} = S_{A1} + \hat{N}$, where \hat{N} is a complex Gaussian R.V with

variance $\frac{\sigma^2}{\|h_{AB}\|^2}$, the LLR equations for each bit of the symbol in our inter-channel can

be presented by:

$$LLR(r_i) = \begin{cases} -\|h_{AB}\|^2 dz_j & |z_j| \leq 2d \\ 2\|h_{AB}\|^2 d(d - z_j) & |z_j| > 2d \\ -2\|h_{AB}\|^2 d(d + z_j) & |z_j| < -2d \end{cases} \quad (3-13)$$

$$j = \begin{cases} I(\text{when } i = 1) \\ Q(\text{when } i = 2) \end{cases}$$

And

$$LLR(r_i) = \|h_{AB}\|^2 d \{ |z_j| - 2d \} \quad (3-14)$$

$$j = \begin{cases} I(\text{when } i = 3) \\ Q(\text{when } i = 4) \end{cases}$$

Following the LLR approach, we can easily detect and make a decision at node B. When the relay is close to the source, most of the bits can be detected by the relay node. This provides an ideal condition for the next stage of data transmission.

3.3 Uplink channels transmission

After detecting the bits transmitted by the source at the relay node, in the collaborative transmission phase, source and relay will send bits together to the destination. If the channel between the source and relay is good enough; then relay can detect a vast majority of original bits with no error. From the situation described above, it is easy to associate communication in the second time slot with a virtual MISO channel; therefore, the application of Alamouti space-time coding [59] is intuitively justified.

In order to get a better performance and keep the same bandwidth, we use a punctured convolutional code which is called Rate-compatible punctured convolutional codes in both transmission parts (i.e., source and relay to destination).

3.3.1 Rate-compatible punctured convolutional codes

Puncturing is the process of removing certain symbols from the code-word, therefore, reducing the codeword length and increasing the overall code rate. Rate-compatible punctured convolutional (RCPC) codes were first introduced by Hagenauer [60]. RCPC is a special case of punctured convolutional codes that has flexible rates and requires adaptive decoder.

RCPC codes can be obtained by adding a rate-compatibility restriction which implies that a high rate code is embedded in the lower rate codes. Mathematically, a family of RCPC codes can be described by a mother code and a sequence of puncturing matrices. We assume that the generator matrix is $G = (g_{i,j})_{S \times (M+1)}$ with rate $R = 1/S$ and memory order M . We also assume that the puncturing matrices are $a(l) = (a_{i,j}(l))_{S \times P}$ for $l = 1, \dots, (S-1)P$, with the puncturing period P , and $a_{i,j}(l) \in \{0,1\}$ where 0 implies puncturing.

The rate-compatibility restriction implies

$$\text{If } a_{i,j}(l_0) = 1, \text{ then } a_{i,j}(l) = 1 \text{ for all } 1 \leq l_0 \leq l.$$

Meanwhile, the rate of a RCPC code is $R(l) = P/(P+l)$. Therefore, a code with a large value of l has more powerful error correction capability.

As shown in [61], the example of family of RCPC code with rate 1/2 convolutional code and $M = 2$ is punctured periodically with $P = 4$. The generator polynomial of the mother code is $G(D) = \{D^2 + D + 1, D^2 + 1\}$, and a sequence of puncturing tables is

$$a(1) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} \text{ with code rate } 4/5$$

$$\begin{aligned}
a(2) &= \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix} \quad \text{with code rate } 2/3 \\
a(3) &= \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix} \quad \text{with code rate } 4/7 \\
a(4) &= \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \quad \text{with code rate } 1/2
\end{aligned} \tag{3-15}$$

Their code rates are shown on the right hand side.

On the received side, the decoder can use the Viterbi algorithm (VA) with a trellis modified by the current puncturing matrix $a(l)$. Suppose X is sent and Y is received. For binary transmission over an additive white Gaussian noise (AWGN) channel, the VA will find the path \hat{X}^m which satisfies

$$\max_m \left(\sum_{j=1}^J \sum_{i=1}^S a_{i,j} \hat{X}_{i,j}^m Y_{i,j} \right) \tag{3-16}$$

Where $a_{i,(j+p)} = a_{i,j}$ is the (i, j) th entry of $a(l)$, and J is the trellis length.

3.3.2 Alamouti's 2X1 Space Time Coding

Following PSK modulation, the Alamouti's 2×1 STC codeword is employed.

The transmitted 2×1 virtual space time codeword X can be demonstrated as below:

$$X = \begin{bmatrix} x_{A_20} & -x_{A_21}^* \\ x_{B1} & x_{B0}^* \end{bmatrix} \tag{3-17}$$

The signals in each column represent the symbols transmitted in each adjacent period.

The received signal vector at the destination at adjacent times t_0 and t_1 is

$$\begin{aligned}
r_D &= [r_0, r_1] = [h_{AD} \quad h_{BD}] \cdot X + n \\
&= [h_{AD} \cdot x_{A20} + h_{BD} \cdot x_{B1}, \quad -h_{AD} \cdot x_{A21} + h_{BD} \cdot x_{B0}] + [n_{AD}, n_{BD}]
\end{aligned} \tag{3-18}$$

Then linear combiner calculates

$$\begin{aligned}
\begin{bmatrix} \tilde{x}_0 & \tilde{x}_1 \end{bmatrix} &= \begin{bmatrix} h_{AD}^* & -h_{BD} \\ h_{BD}^* & h_{AD} \end{bmatrix} \begin{bmatrix} r_0 \\ r_1 \end{bmatrix} \\
\tilde{x}_0 &= |h_{AD}|^2 x_{A20} + |h_{BD}|^2 x_{B0} + h_{AD}^* h_{BD} (x_{B1} - x_{A21}) + h_{AD}^* n_{AD} + h_{BD}^* n_{BD} \\
\tilde{x}_1 &= |h_{AD}|^2 x_{A21} + |h_{BD}|^2 x_{B1} + h_{BD}^* h_{AD} (x_{A20} - x_{B0}) - h_{AD}^* n_{BD} + h_{BD}^* n_{AD}
\end{aligned} \tag{3-19}$$

Eventually, at the receiving side, received signal will be detected after the linear combination and Viterbi decoding.

3.4 Energy Distribution

It has been shown that MIMO system can support higher data rates under the same transmit power budget and bit-error-rate performance requirements as a SISO system. In order to comparison, we assume that the total energy consumed in our cooperative virtual space time coding scheme is as the same as a SISO system. For the sake of simplicity, we do not consider the circuit energy consumption of Analog to Digital Converter (ADC), Digital to Analogy Converter (DAC), Mixer, Lower Noise Amplifier (LNA), Frequency Synthesizer, Power Amplifier, and base band DSP.

We consider BPSK-based between the SISO channel where the number of bits in each frame is n , the distance from the source to the destination is defined as d , and the distance between the source and relay is ad ($a < 1$). We assume that the energy spent on

the inter user channel is $E_s(QAM)$ per bit; and the energy used in the uplink channel is $E_s(BPSK)$ per bit. Meanwhile, $E_s'(BPSK)$ represents the energy of each bit in the SISO non-cooperative channel. Then, the energy allocation can be expressed by the following:

$$E_s(QAM) \cdot n / 4 + E_s(BPSK) \cdot n \cdot 2 = E_s'(BPSK) \cdot n \quad (3-20)$$

The left side of equation (3-20) is the total energy consumption in the whole cooperative system. We assume error free transmission in both inter-channel (*16 QAM*) and SISO channel (*BPSK*); when $BER=10^{-5}$. By normalizing the energy of SISO channel $E_{siso}(BPSK)$ into unity, we can calculate the approximate value of energy that should be distributed on both in inter-user channel and uplink channels.

3.5 Error Performance Simulation

For comparison, we present the simulation result with 100,000 frames, where each frame contains 160 bits. Frame error rate of our scheme is studied and compared to the performance of the convolutional code in both SISO and MIMO systems. We assume that the bits we detected from Node B are error free ($BER=10^{-5}$). Energy allocation has been done through estimation before sending. In order to retain the same bandwidth as that of the convolutional code (rate 1/2), we employ a simple but effective RCPC (Rate-compatible punctured convolutional) code with code rate 4/7 [61]. It can be observed that there is around 4 dB improvement using our scheme compared to a non-cooperative system. Moreover, the effect of the distance between the source and relay on the achieved

improvement is also shown. It is clear that the scheme's performance improves with the reduction of this distance. In other words, instead of spending energy on inter-node communication, we consume more energy to ensure a better performance on the uplink channel.

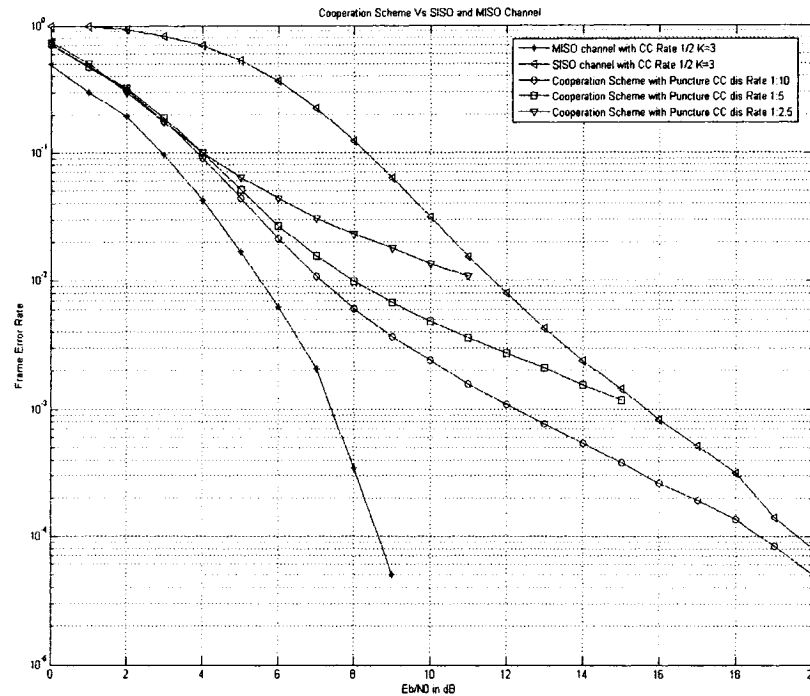


Figure 3-3: the Comparison of Cooperative and Non-Cooperative Simulation Result

CHAPTER FOUR

UPLINK RECEIVER HARDWARE IMPLEMENTATION

The algorithm of the receiver part introduced in the previous section can be described using the block diagram shown in Figure 4-1. First of all, the received signal is passed through a Matched Filter that contains two RRC filters for both in-phase and quadrature-phase channels. After being transferred into an Alamouti Decoder, 4-bit quantization is used. Finally, we utilize a Viterbi decoder for decoding of the RCPC code.

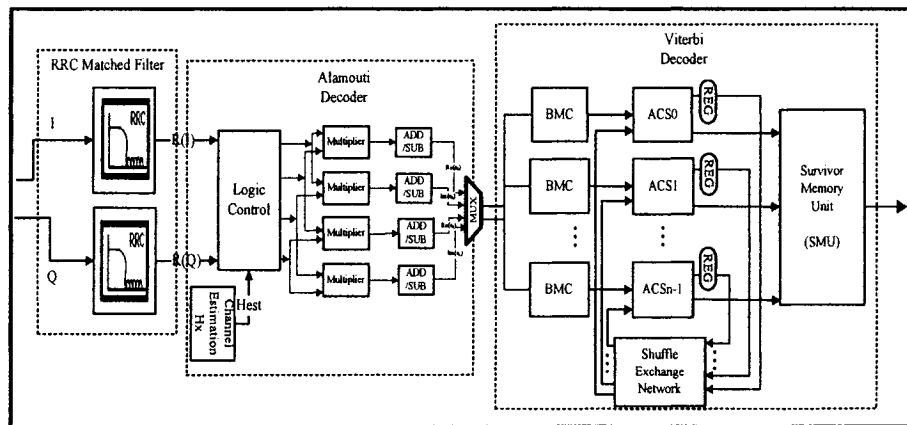


Figure 4-1: Receiver block diagram

This chapter is organized in the follow way. Section 4.1 presents the detailed process of the Square Root Raised Cosine Filter implementation; then in the next section, we introduce a parallel approach that utilizes the Moore state machine to construct a logic control unit. Moreover, the architecture of the Viterbi decoder for the the RCPC code is illustrated in Section 4.3. Finally, the architecture of the whole receiver is given in the last section. In addition, the results of the RTL level simulation are given in each section, and the overall hardware design summary will be shown in the last section.

4.1 Finite Impulse Response(FIR) filter Implementation

On the receiver side, the principal task for recovering the received symbols is to demodulate. We choose the Square Root Raised Cosine (SRRC) matched filter for our design. The section is organized as follows: Firstly, The Square Root Raised Cosine Filter is introduced. Then, the detailed implementation is given in Subsection 4.1.3. Finally, Subsection 4.1.7 presents the simulation result of the RTL level simulation.

4.1.1 Square Root Raised Cosine Filter

The Square Root Raised Cosine matched filter is used in almost all modern digital modems. It produces a frequency response with unity gain at low frequencies and complete at higher frequencies.

The root raised cosine filter is generally used in pairs, where the transmitter first applies a root raised cosine filter, and the receiver then applies a matched filter. So the total filtering effect is that of a raised cosine filter. The advantage is that if the transmit side filter is stimulated by an impulse, then the receive side filter is forced to filter an input pulse shape that is identical to its own impulse response, thereby setting up a matched filter and maximizing signal to noise ratio while minimizing intersymbol interference (ISI).

The cosine roll-off transfer function can be achieved by using identical square root raised cosine filter $\sqrt{X_{rc}(f)}$ at the transmitter and receiver. The root raised cosine filter can be generically defined by its frequency response $X_{srrc}(f)$. If t is the time variable, T the length of the baseband chip and α the excess bandwidth for the Universal Mobile Telecommunications System (UMTS) then

$$X_{srrc}(f) = \begin{cases} 1 & 0 \leq |f| \leq \frac{1-\alpha}{2 \cdot T} \\ \cos \left[\frac{\pi \cdot T}{2 \cdot \alpha} \cdot \left(|f| - \frac{1-\alpha}{2 \cdot T} \right) \right] & \frac{1-\alpha}{2 \cdot T} \leq |f| \leq \frac{1+\alpha}{2 \cdot T} \\ 0 & |f| > \frac{1+\alpha}{2 \cdot T} \end{cases} \quad (4-1)$$

An FIR Root Raised cosine filter may be synthesized directly from the impulse response, which is

$$SRRC_2(-t) = SRRC_1(t) = \frac{\sin(\pi \frac{t}{T} (1-\alpha)) + 4\alpha t \cos[\pi \frac{t}{T} (1+\alpha)]}{\pi \frac{t}{T} [1 - (4\alpha \frac{t}{T})^2]} \quad (4-2)$$

In the transmission system shown in Figure 4-2, information bits that need to be transmitted are multiplexed onto in phase and quadrature phase components. Therefore, the transmitter filter block has a pair of parallel SRRC filters, one for the I channel, and another one for Q channel. After modulating onto orthogonal carriers, the signal will be transmitted through a channel; after demodulation, the receiver base-band pulse is sampled and applied to a pair of SRRC filters prior to diversity combining.

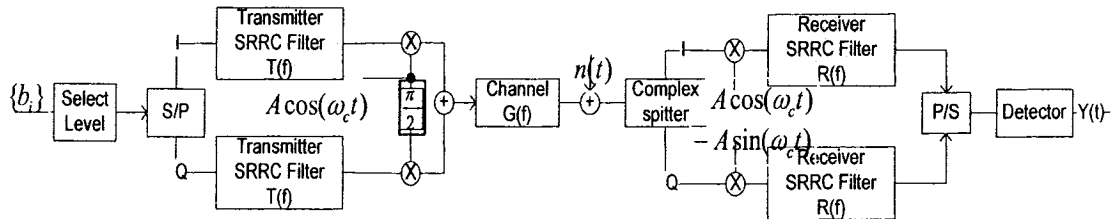


Figure 4-2 Basic elements of a transmission system

4.1.2 Finite Impulse Response Filter

The output of the FIR is calculated as the accumulation of the inner products between the predetermined set of coefficient C_k , and the input data words x_k with the expression: (see Figure 4-3)

$$y_t = \sum_{k=0}^{K-1} C_k x_{t-k} \quad (4-3)$$

$$y_t = C_0 x_t + C_1 x_{t-1} + \dots + C_{k-2} x_{t-(k-2)} + C_{k-1} x_{t-(k-1)} \quad (4-4)$$

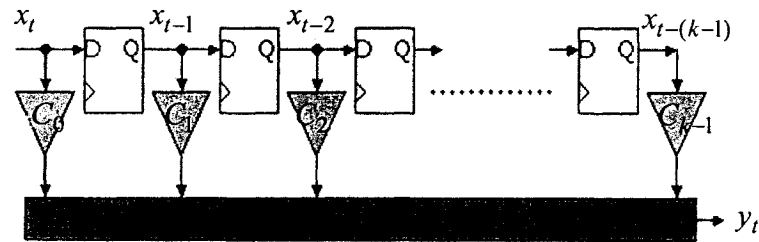


Figure 4-3: Non-symmetric FIR filter

If the coefficient exhibits the following symmetry that

$$C_0 = C_{k-1}, C_1 = C_{k-2}, C_2 = C_{k-3} \dots etc \quad (4-5)$$

then the equation becomes

$$y_t = C_0(x_t + x_{t-(k-1)}) + C_1(x_{t-1} + x_{t-(k-2)}) + \dots etc \quad (4-6)$$

The filter represented by the equation above is called an even-symmetric FIR filter due to the even number of coefficients.

If there are an odd number of the coefficients, then we have the following symmetry

$$C_0 = C_{k-1}, C_1 = C_{k-2}, C_2 = C_{k-3}, \dots etc. \quad \text{and} \quad C_{k/2} \text{ is not a paired coefficient.}$$

Then the equation changes to

$$y_t = C_0(x_t + x_{t-(k-1)}) + C_1(x_{t-1} + x_{t-(k-2)}) + \dots + C_{k/2}(x_{k/2}) \quad (4-7)$$

The above expression is an odd-symmetric FIR filter since all of the coefficients are symmetric and have a matching coefficient apart from the middle coefficient.

4.1.3 Implementation of the FIR filter

From the comparison between the non-symmetric and symmetric FIR filters, we can find that some resources can be saved in the architecture of symmetric FIR. Since some taps share the same coefficient, the multiplier structure can be further reduced to take advantage of this property.

In order to reduce the hardware complexity and latency, our project starts with even-symmetric FIR implementation.

The receiver RRC structure was implemented as a 24 tap symmetrical FIR filter. By sampling the Square Root Raised Cosine pulse 48 times during the period $-3T$ and $3T$. Accordingly, the sample rate is $8/T$. Figure 4-4 shows the sampling results of the impulse response.

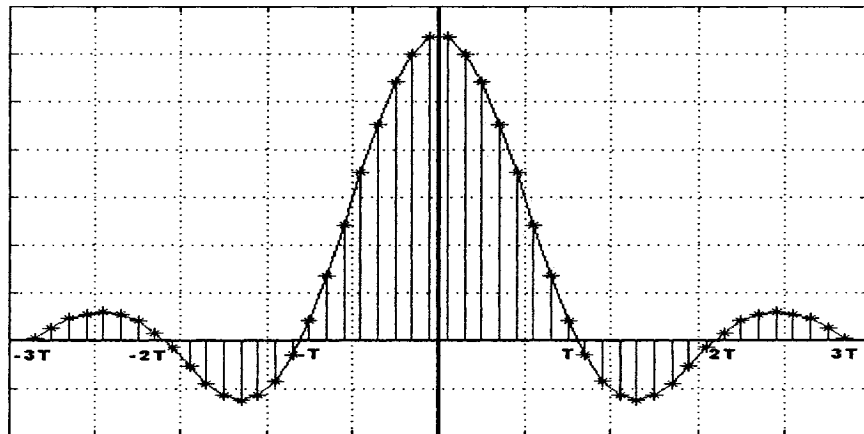


Figure 4-4: Sampling the SRRC pulse

As we can see from the above plot, we can get exactly 24 symmetrical coefficients during $6T$ period. After normalizing their values to fit in the -128 to 127 range; the values of 2's complement value can be found in the following table.

Coefficient Index	Coefficient in Dec	Coefficient in Bin
C(23)	127	01111111
C(22)	120	01111000
C(21)	108	01101100
C(20)	90	01011010
C(19)	70	01000110
C(18)	48	00110000
C(17)	27	00011011
C(16)	8	00001000
C(15)	-6	11111010
C(14)	-17	11101111
C(13)	-23	11101001
C(12)	-25	11100111
C(11)	-23	11101001
C(10)	-18	11101110
C(9)	-11	11110101
C(8)	-3	11111101
C(7)	3	00000011
C(6)	8	00001000
C(5)	11	00001011
C(4)	12	00001100
C(3)	11	00001011
C(2)	9	00001001
C(1)	5	00000101
C(0)	1	00000001

Table 4-1: Values of 2's complement for the 24-taps symmetric coefficients

Due to the symmetrical structure, only half of these values are taken and ranged from 23 down to 0 on the right side of x axis. Table 4-1: shows these values both in decimal and binary form. The entire fixed coefficients can be saved in a section of the fixed length memory.

4.1.4 Input/Output Interface Description

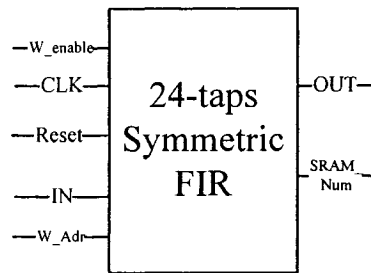


Figure 4-5: I/O interface of 24 taps symmetric FIR

In addition to the clock (CLK), input (IN), reset (RST), and output (OUT) interfaces, as shown in Figure 4-5, there are 4 pins in the I/O interface. Refer to table 4-2, they are called Write_enable, Write_address, and SRAM_Num. Functionally, they are reserved for the next steps in the receiver.

Signal	I/O	Size	Description
CLK	Input	1bit	Clock-rising edge active
RST	Input	1bit	Asynchronous Reset
W_enable	Output	1bit	Sign for writing into the memory
SRAM_Num	Output	2 bits	Number of memory section that is written
Write_add	Output	2bits	Address in one memory section that is written
IN	Input	8bits	Input samples for the FIR filter.(two's complement format)
OUT	Output	16bits	Output data for the FIR filter. (two's complement format)

Table 4-2: Specification of 24 taps symmetric FIR

4.1.5 Multiply Accumulate Unit

From the expression of FIR filter, we can observe that multiplication followed by accumulation is a common operation.

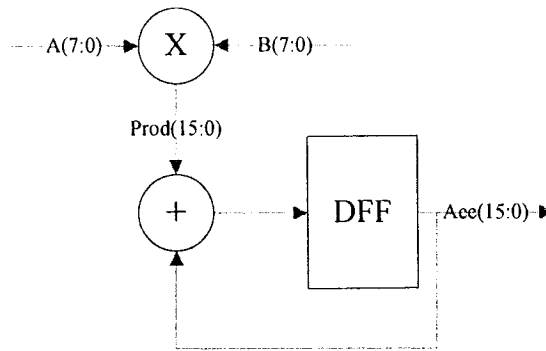


Figure 4-6: the MAC architecture

One typical MAC (multiply-accumulate) architecture is illustrated in Figure 4-6. In our implementation, we use 8 bits both for the Coefficient and the inputs. Therefore, a and b will be 8-bit values. After multiplying a and b , we add the result to the previously accumulated value. It is necessary to create a register for the accumulated value which is re-stored and refreshed in the register for the future accumulation. Another feature of a MAC circuit is that it must check for overflow, which might happen when the number of MAC operations is large. We check the overflow at the end of each accumulation. In this way, we get the approximate value of the 16 bits in the final output.

4.1.6 Registers and Memory

For the 24-tap symmetric FIR filter, shown in Figure 4-7, each rising clock, 8 bits x will come, and after shifting 48 times from one register to another, it finally exits from the system. Hence, there are 48 registers used in the FIR filter.

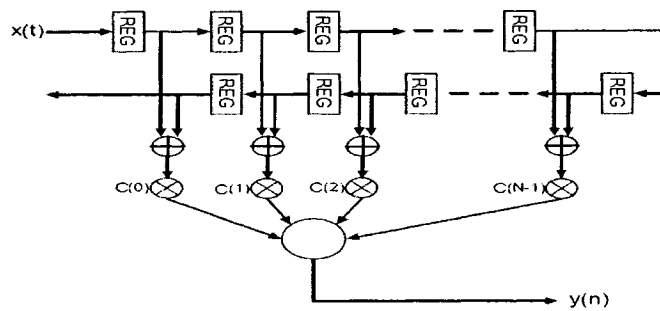


Figure 4-7: Even Symmetric FIR

From the last section, we have the idea that one accumulation register should be utilized during the 24 times accumulations in a clock cycle. Since we use fixed values for all the coefficients, it is convenient to use a memory section with 24 addresses, which are 5 bits memory addresses to store all of them. Then, we fetch one corresponding value of them 24 addresses when applied to the MAC units.

4.1.7 Implementation Result

After simulating the Register Transfer Language (RTL) level by Modelsim, the wave plot is shown in the Figure 4-8. We can observe that the values in the registers are shifted all the way to the right, and after each clock cycle, a 16-bit output can be obtained.

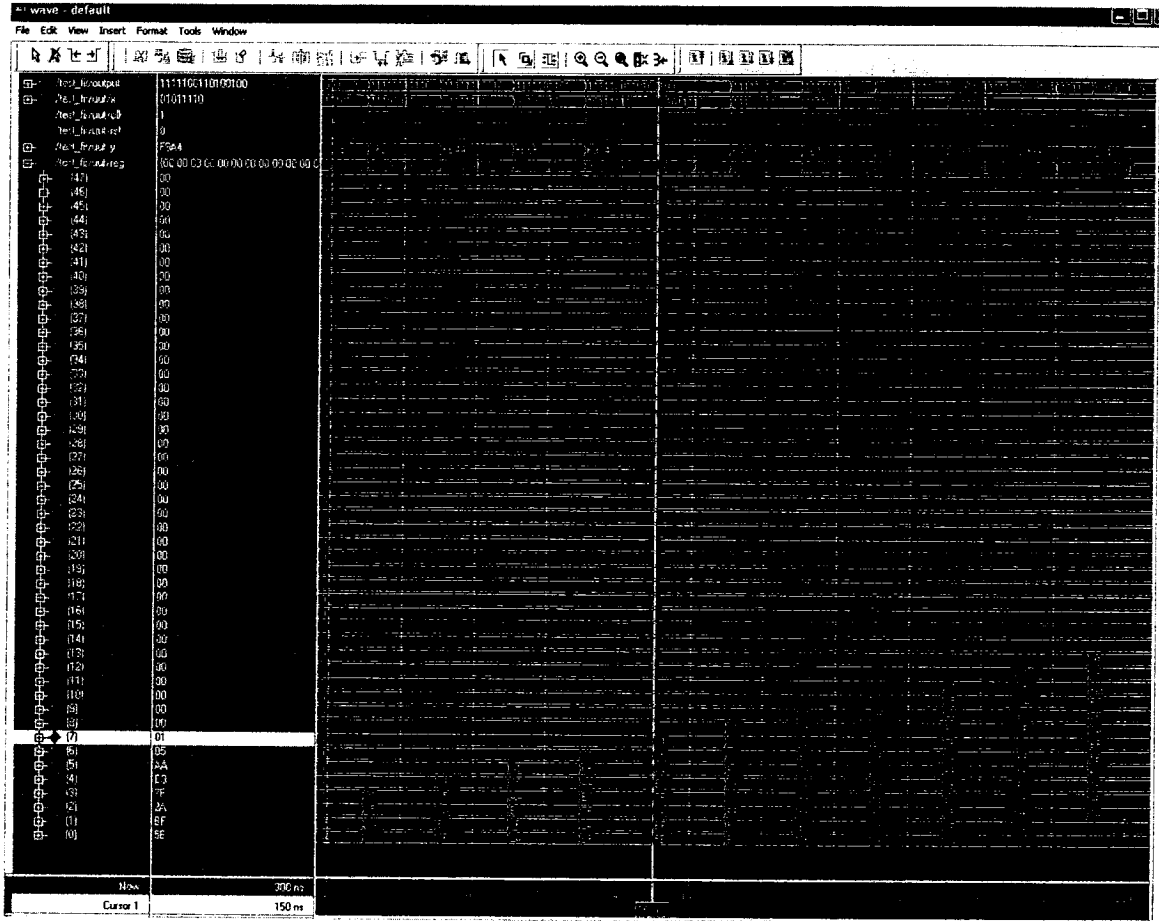


Figure 4-8: Simulation result for 24 symmetric FIR

4.2 Alamouti Decoder Implementation

As we can see from the whole structure which is introduced in Figure 4-1, the next step after the FIR matched filter demodulating, the current step is to execute the space-time decoder. In our cooperative system, we assume that there are two antennas cooperating in the transmitter and one antenna in the receiver. Therefore, Alamouti (2×1) decoding can be applied. And the hardware implementation of Alamouti coding will be depicted in the following section.

4.2.1 Alamouti Decoder 2×1 Implementation

The next step is to implement the Alamouti decoder. We can extend the formulas for the received signal in two adjacent clock periods shown in Section 3.3.2 into four equations for real and imaginary parts.

$$\begin{aligned} s_{0_{re}} &= re\{h_0\} * re\{y_0\} + im\{h_0\} * im\{y_0\} + re\{h_1\} * re\{y_1\} + im\{h_1\} * im\{y_1\}; \\ s_{0_{im}} &= re\{h_0\} * im\{y_0\} - im\{h_0\} * re\{y_0\} - re\{h_1\} * im\{y_1\} + im\{h_1\} * re\{y_1\}; \\ s_{1_{re}} &= re\{h_1\} * re\{y_0\} + im\{h_1\} * im\{y_0\} - re\{h_0\} * re\{y_1\} - im\{h_0\} * im\{y_1\}; \\ s_{1_{im}} &= re\{h_1\} * im\{y_0\} - im\{h_1\} * re\{y_0\} + re\{h_0\} * im\{y_1\} - im\{h_0\} * re\{y_1\}; \end{aligned} \quad (4-8)$$

It is obvious that we need to implement only two operation (multiplication and add/subtraction units) in each equation. A multi-cycle design can be considered for calculating all equations in parallel. There are four multiplier functional units, and four associated add/subtract units with registers to accumulate the results. By observing the above equations, we find that each equation can be divided into four steps, and in each step, there are two parts we need to determine which are operands of the multiplier and the sign of the current polynomial. Practically, the usage of different paths in each state after the logic control is listed in the following Table 4-3.

State	Path				Sign			
	$\alpha(op_a)$	$\beta(op_b)$	$\gamma(op_c)$	$\delta(op_d)$	for S_{re}	for S_{im}	for S_{re}	for S_{im}
State1	Re(h_0)	Re(y_0)	Im(y_0)	Re(h_1)	“+”	“+”	“+”	“+”
State2	Im(h_0)	Im(y_0)	Re(y_0)	Im(h_1)	“+”	“-”	“+”	“-”
State3	Re(h_1)	Re(y_1)	Im(y_1)	Re(h_0)	“+”	“-”	“-”	“+”
State4	Im(h_1)	Im(y_1)	Re(y_1)	Im(h_0)	“+”	“+”	“-”	“-”

Table 4-3: The usage and control sequence of Paths in Each State

4.2.2 Input/ Output Interface

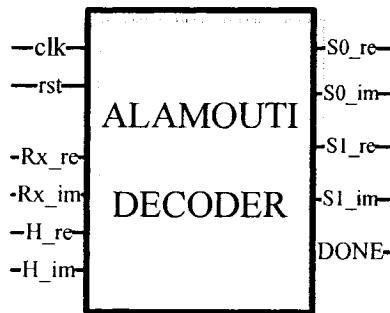


Figure 4-9: I/O layout of Alamouti Decoder

After presenting the whole idea, we now provide the I/O interface layout of the Alamouti decoder for our design in Figure 4-9. The inputs of this component are placed on the left hand side, while the outputs are listed on the right hand side. The specifications for all of these are shown in Table 4-4.

Signal	I/O	Size	Description
Clk	Input	1	Clock-rising edge active
Rst	Input	1	Asynchronous Reset
Rx_re	Input	2x1_matrix_8	Array(0 to 1)of std_logic_vector(7 downto 0) Array(0) which include 8 bits represents $re\{y_0\}$ Array(1)which include 8 bits represents $re\{y_1\}$
Rx_im	Input	2x1_matrix_8	Array(0 to 1)of std_logic_vector(7 downto 0) Array(0) which includes 8 bits represents $im\{y_0\}$ Array(1) which includes 8 bits represents $im\{y_1\}$
H_re	Input	2x1_matrix_8	Array(0 to 1)of std_logic_vector(7 downto 0) Array(0) which includes 8 bits represents $re\{h_0\}$ Array(1) which includes 8 bits represents $re\{h_1\}$
H_im	Input	2x1_matrix_8	Array(0 to 1)of std_logic_vector(7 downto 0) Array(0) which includes 8 bits represents $im\{h_0\}$ Array(1) which includes 8 bits represents $im\{h_1\}$
S0_re	Output	std_logic_vector (17 downto 0)	Output data for S0 in the real value.
S0_im	Output	std_logic_vector (17 downto 0)	Output data for S0 in the imagine value
S1_re	Output	std_logic_vector (17 downto 0)	Output date for S1 in the real value
S1_im	Output	std_logic_vector (17 downto 0)	Output data for S1 in the imagine value
Done	Output	1	If Done bit ='1' It indicates that each received 2X1 metrics is decoded.

Table 4-4: The particular specifications for I/O Layout of Alamouti decoder

4.2.3 Top Level Finite State Machine (FSM) for Alamouti Decoder

The Logic Control Unit is the core unit of the Alamouti decoder. It controls the way the “multiplier” and “adder / subtracter” units work together and in what sequence. In addition, it works as a finite state machine in this parallel sequence. In Figure 4-10, we present the detailed functional diagram of the FSM.

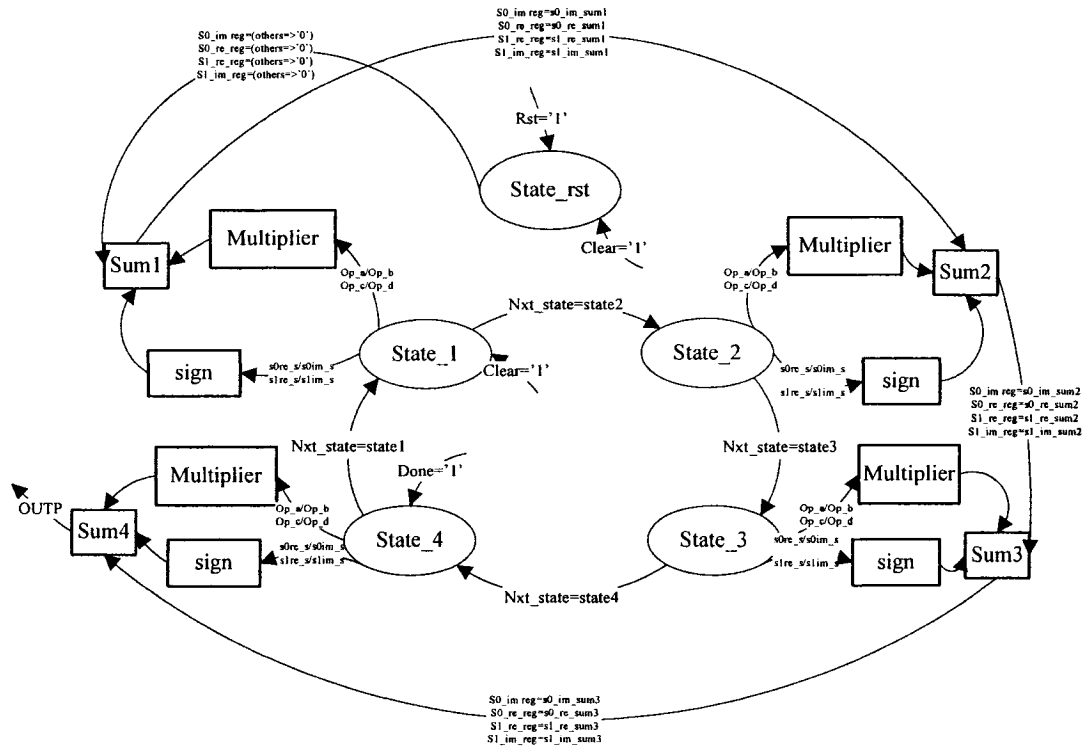


Figure 4-10: Top Level State Diagram of FSM for Alamouti Decoder

This state machine is a Moore machine where the outputs are determined by the current state alone. “State_rst” will be activated when initialized or whenever the asynchronous “rst” signal is inserted. In “State_1”, the operands for multiplier are loaded, and the register containing the sum, Sum_reg is updated by accumulating the summation with the given sign in each parallel equation. This process will be continued until the end of “State_3”. In “State_4”, the initial processes are the same (e.g. the multiplier operands loading, parallel equations summation). But at the end, instead of saving the accumulated value into the Sum_reg, 4 parallel outputs will be generated when “Done” signal turns to “1”. That is the final result for the Alamouti decoder. When the next clock cycle comes, the system will deal with the coming received bits. We can see from the above process

that each coming received $2XI$ matrix takes 4 clock cycles to finish its Alamouti decoding.

4.2.4 Arithmetic Logic Units Implementation

In the accumulation of the final result (Equation 4-8), there are two arithmetic units being used. These are the multiplier and adder/subtractor, respectively. We can start by describing the multiplier first.

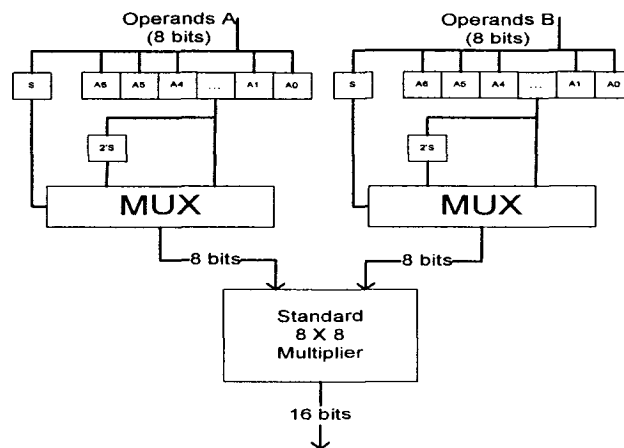


Figure 4-11: Architecture of 8X8 Multiplier

Instead of using 16 multipliers at the same time, we only used 4 multipliers in this Alamouti decoder. Both multiplier and multiplicand are represented by 8 bits as seen in Figure 4-11. After each clock cycle, we need to change the value of multiplier and multiplicand by fetching the corresponding values from Table 4-3. Then, after multiplication, the result of the multiplication should be used as one input for the add/sub units.

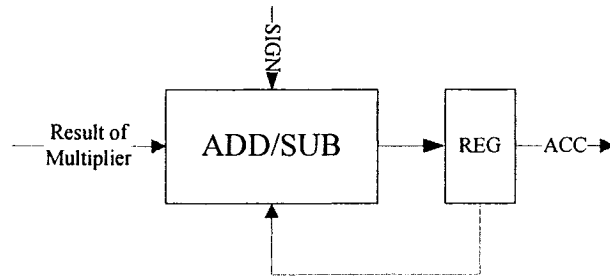


Figure 4-12: Structure of ADD/SUB Unit

Also, four add/sub units are involved in this design; we can see from Figure 4-12 that each of them combines two operand inputs and one sign input. One of them represents the decision to add or sub. When this input bit is 1, the two inputs are added; when this input bit is 0, they are subtracted. The two operand inputs add and subtract. Among these two inputs, one is connected with the output of the multiplier, and the other is connected to the feedback output register of this add/sub unit. It means that we need to accumulate 4 times the result of the multiplication in order to get the final result.

Therefore, the total construction can be illustrated by the following block diagram (Figure 4-13)

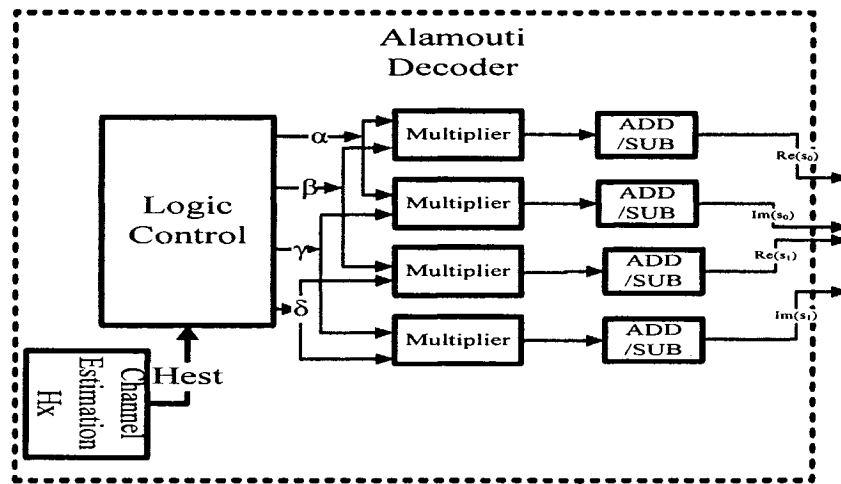


Figure 4-13: Overall Architecture of Alamouti Decoder

4.2.5 Extension to Alamouti Decoder 2 X 2

The same idea which was introduced in the previous section can be utilized if the number of the received antennas is increased. The received signals of the Alamouti decoder can then be expressed as follows.

$$\begin{aligned}
 s0_{re} &= re\{h_{0-0}\} * re\{y_{0-0}\} + im\{h_{0-0}\} * im\{y_{0-0}\} + re\{h_{0-1}\} * re\{y_{0-1}\} + im\{h_{0-1}\} * im\{y_{0-1}\} \\
 &\quad + re\{h_{1-0}\} * re\{y_{1-0}\} + im\{h_{1-0}\} * im\{y_{1-0}\} + re\{h_{1-1}\} * re\{y_{1-1}\} + im\{h_{1-1}\} * im\{y_{1-1}\} \\
 s0_{im} &= re\{h_{0-0}\} * im\{y_{0-0}\} - im\{h_{0-0}\} * re\{y_{0-0}\} - re\{h_{0-1}\} * im\{y_{0-1}\} + im\{h_{0-1}\} * re\{y_{0-1}\} \\
 &\quad + re\{h_{1-0}\} * im\{y_{1-0}\} - im\{h_{1-0}\} * re\{y_{1-0}\} - re\{h_{1-1}\} * im\{y_{1-1}\} + im\{h_{1-1}\} * re\{y_{1-1}\} \\
 s1_{re} &= re\{h_{0-1}\} * re\{y_{0-0}\} + im\{h_{0-1}\} * im\{y_{0-0}\} - re\{h_{0-0}\} * re\{y_{0-1}\} - im\{h_{0-0}\} * im\{y_{0-1}\} \\
 &\quad + re\{h_{1-1}\} * re\{y_{1-0}\} + im\{h_{1-1}\} * im\{y_{1-0}\} - re\{h_{1-0}\} * re\{y_{1-1}\} - im\{h_{1-0}\} * im\{y_{1-1}\} \\
 s1_{im} &= re\{h_{0-1}\} * im\{y_{0-0}\} - im\{h_{0-1}\} * re\{y_{0-0}\} + re\{h_{0-0}\} * im\{y_{0-1}\} - im\{h_{0-0}\} * re\{y_{0-1}\} \\
 &\quad + re\{h_{1-1}\} * im\{y_{1-0}\} - im\{h_{1-1}\} * re\{y_{1-0}\} + re\{h_{1-0}\} * im\{y_{1-1}\} - im\{h_{1-0}\} * re\{y_{1-1}\}
 \end{aligned}
 \tag{4-9}$$

Compared to 2 X 1 Alamouti decoder, instead of having four multiplier functional units, and four associated add/subtract units with registers to accumulate the totals, we need to increase the number of multipliers and add/sub units into eight. But, from the same approach, the application of different path in each state after the logic control is listed in the following Table 4-5.

Although this idea can be extended to more complex MIMO systems, the time duration for processing each coming signal will be longer. We can take a look at the Alamouti (2 X 1) and Alamouti (2 X 2) decoder, for Alamouti (2 X 1) Decoder, it takes 4 clock cycles to process the incoming signal, and for the Alamouti (2 X 2) decoder, using a paralleled structure, it takes 8 clock cycles to process the coming signal. Although it saves the number of multipliers and the adders, it is not necessary to pay the price on the time for the decoder.

State	Path				Sign			
	$\alpha(op_a)$	$\beta(op_b)$	$\gamma(op_c)$	$\delta(op_d)$	for S_{ve}	for S_{im}	for S_{ve}	for S_{im}
S1	$re\{h_{0-0}\}$	$re\{y_{0-0}\}$	$im\{y_{0-0}\}$	$re\{h_{0-1}\}$	“+”	“+”	“+”	“+”
S2	$im\{h_{0-0}\}$	$im\{y_{0-0}\}$	$re\{y_{0-0}\}$	$im\{h_{0-1}\}$	“+”	“-”	“+”	“-”
S3	$re\{h_{0-1}\}$	$re\{y_{0-1}\}$	$im\{y_{0-1}\}$	$re\{h_{0-0}\}$	“+”	“-”	“-”	“+”
S4	$im\{h_{0-1}\}$	$im\{y_{0-1}\}$	$re\{y_{0-1}\}$	$im\{h_{0-0}\}$	“+”	“+”	“-”	“-”
S5	$re\{h_{1-0}\}$	$re\{y_{1-0}\}$	$im\{y_{1-0}\}$	$re\{h_{1-1}\}$	“+”	“+”	“+”	“+”
S6	$im\{h_{1-0}\}$	$im\{y_{1-0}\}$	$re\{y_{1-0}\}$	$im\{h_{1-1}\}$	“+”	“-”	“+”	“-”
S7	$re\{h_{1-1}\}$	$re\{y_{1-1}\}$	$im\{y_{1-1}\}$	$re\{h_{1-0}\}$	“+”	“-”	“-”	“+”
S8	$im\{h_{1-1}\}$	$im\{y_{1-1}\}$	$re\{y_{1-1}\}$	$im\{h_{1-0}\}$	“+”	“+”	“-”	“-”

Table 4-5: The usage of Paths in Each State for 2X2 Alamouti Decoder

4.2.6 Implementation Result

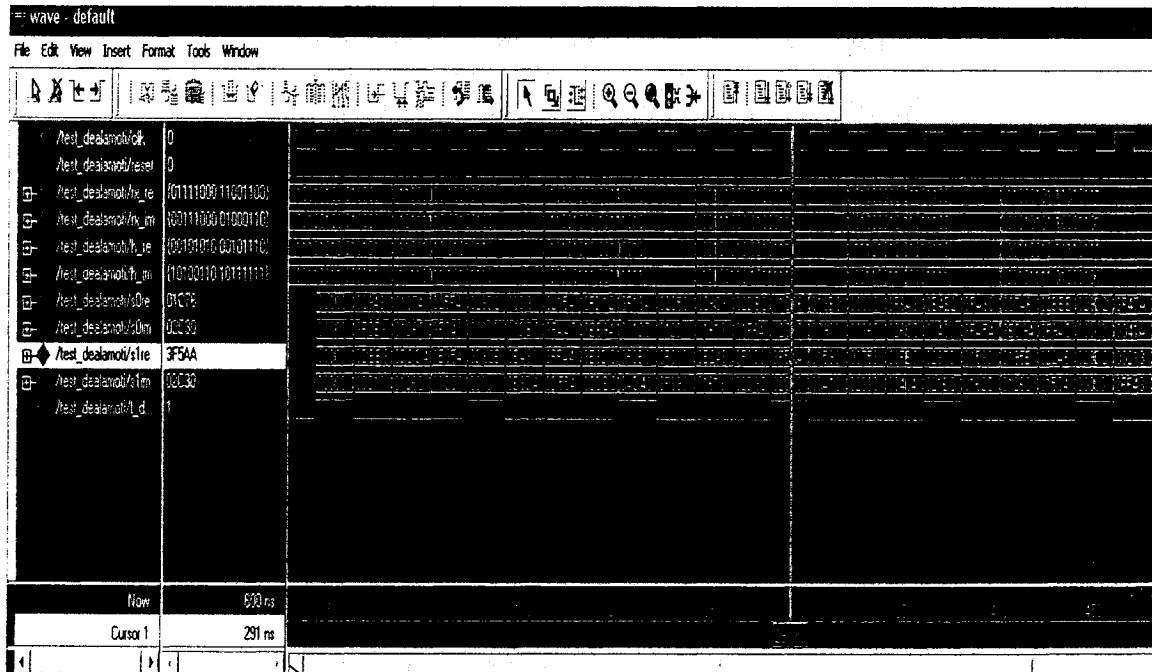


Figure 4-14: Alamouti Decoder RTL Level Simulation Result

4.3 Viterbi Decoder Implementation

This section presents a programmable Viterbi Decoder implementation that is suitable for rate-compatible punctured convolutional codes (RCPC). The architecture is based on the popular Viterbi Decoder implementation which involves several special arithmetic blocks and read/write memories. In addition, some puncturing patterns are stored in the memory in order to apply zero insertion. The different decoding schemes can be implemented by selecting different parameters from a look up table (LUT).

4.3.1 Description of the Viterbi Algorithm

In order to explain the implementation of the Viterbi algorithm, it is convenient to expand the state diagram of the encoder in time (i.e., to represent each time unit with a separate state diagram). If we assume an information sequence of length L , and an encoder memory order of m , then the trellis diagram contains $L + m + 1$ stages, labeled from 0 to $L + m$. Figure 4-15 below shows the trellis diagram for the example of rate $1/2$, $K = m + 1 = 3$ convolutional encoder for a 15-bit message:

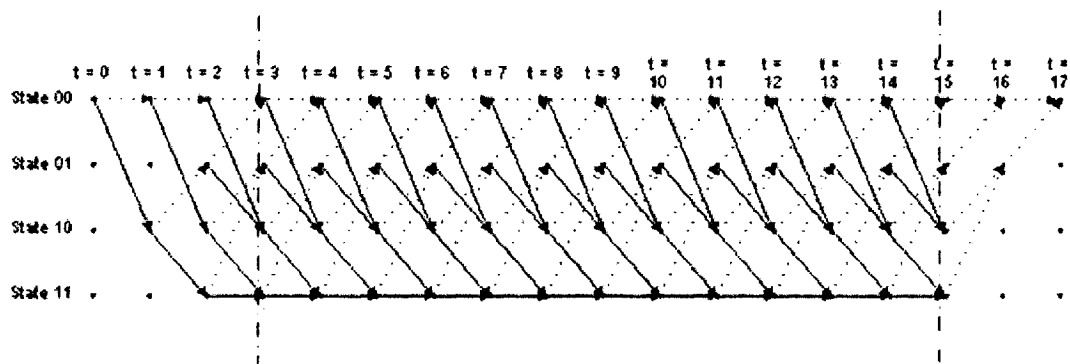


Figure 4-15: Trellis diagram for a (2, 1, 2) code with $L = 15$.

Assuming that the encoder always starts in state S_0 (the beginning of the trellis) and returns to state S_0 (the end of the trellis), the first m time units correspond to encoder's departure from S_0 , and the last m time units correspond to the encoder's return to state S_0 . It follows that not all states can be reached in the first m or the last m time units. However, in the center portion of the trellis, all states are possible, thus each time unit contains a replica of the state diagram. There are two branches leaving and entering each state. The solid-line branch leaving each state at time unit i represents the input $u_i = 1$, while the dashed-line branch represents $u_i = 0$. Furthermore, each branch is labeled with the n corresponding outputs v_i . There are 3 steps per loop in the Viterbi Algorithm. In general, a convolutional code can be represented by its rate R and constraint length $K = m + 1$, where m is the number of memory units.

First, assuming we begin at the time unit $j = m$, we can compute the partial metric for the signal path entering each state. We then store the path (the survivor) and its metric for each state. Following this, the second step, where after increasing j by 1, we can compute the partial metric for all the paths entering a state by adding the branch metric entering that state to the metric of the connecting survivor at the preceding time unit. For each state, we store the path with the largest metric (the survivor), together with its accumulated metric, and eliminate all other paths. The final step is to compare the value of j with $L + m$; if j is still less than $L + m$ we will return to the second step to continue the loop. Otherwise, we can simply finish this loop.

4.3.2 Design Specification

The parameters and specifications of our design are listed as follows (Table 4-6):

Constraint Length	3	5	7
Code Word	111 101	10111 11001	1001111 1101101
Memory Order	2	4	6
Number of states in decoding trellis	4	16	64
Code rate	1/2		
Receiver Quantization	4 bits Quantization		
Puncturing Tables for $G(D) = \{D^2 + D + 1, D^2 + 1\}$	$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix}$	
Punctuated Rate	4/5	4/7	

Table 4-6: Viterbi Decoder Parameters

4.3.3 Arithmetic Logic Units Specification

There are three fundamental arithmetic units related to Viterbi Decoder, namely the branch metric unit (BMU), the add compare select unit (ACS), and the survivor memory unit (SMU).[68] The branch metric unit is utilized to compute the branch metrics at each time stage based on the received data and the branch codeword. After collecting all these branch metric values, the add compare select unit will be applied to update the path metric of each survivor path by accumulation and comparison. Eventually, the decoded bits can be obtained by the survivor memory unit, which stores the survivor sequence for each state.

4.3.3.1 Branch Metric Unit

Before giving more information, we need to review the butterfly module, which is widely used in Viterbi Decoder design. From the trellis diagram of the convolutional code, the decoding function can be efficiently performed by breaking up the trellis into a number of identical elements. For example, the trellis diagram of the rate 1/n convolutional code can be broken up into elements containing a pair of origin and destination states and four interconnecting branches. A classic example of a butterfly module for the convolutional code which is the basic processing unit in our design is shown in Figure 4-16.

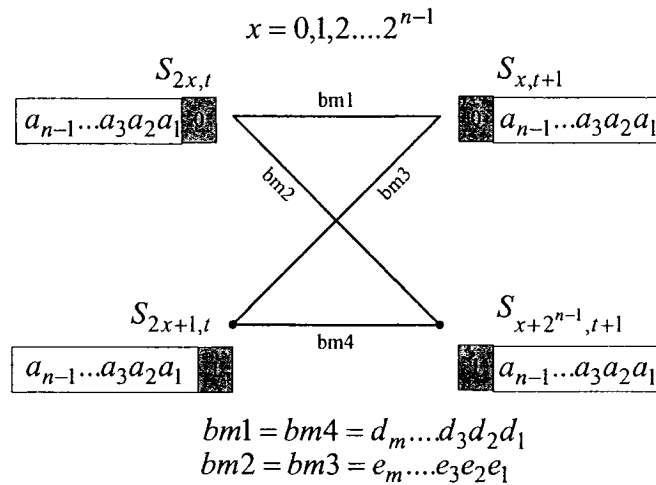


Figure 4-16: Butterfly module.

The butterfly module contains two initial states: $S_{2x,t}$ and $S_{2x+1,t}$; two final states $S_{x,t+1}$ and $S_{x+2^{n-1},t+1}$. The only difference between these four states is shown in the shaded region of the labels. We assume that the current state is $S_{2x,t}$ at the t^{th} time stage. If the input bit is 0, then at the $(t+1)^{th}$ time stage, the next state is $S_{x,t+1}$ and the output

branch symbol is $bm_1 = (d_m \dots d_3 d_2 d_1)$. On the other hand, if the input bit is 1, then the next state will be $S_{x+2^{n-1}, t+1}$ with output branch symbol $bm_2 = (e_m \dots e_3 e_2 e_1)$. Similarly, the transition from state $S_{2x+1, t}$ can be interpreted in an identical way.

Based on the butterfly module, we can calculate the branch metric between the received data and the branch code word. With the path metric $PM_t(S_{2x, t})$ and $PM_t(S_{2x+1, t})$, which is associated with state $S_{2x, t}$ and $S_{2x+1, t}$, the updated path metric, $PM_{t+1}(S_{x, t+1})$ and $PM_{t+1}(S_{x+2^{n-1}, t+1})$ with state $S_{x, t+1}$ and $S_{x+2^{n-1}, t+1}$, can then be retrieved. Through the same approach, all the states at each time stage are processed and the new path metric associated with each state is updated. The process is then repeated from one time stage to another.

The branch metric unit is used to generate the branch metrics. Since in our design the maximum memory constraint length is up to 7, we have to use a maximum of 32 butterfly modules for the branch metrics from the current stage to the next stage.

After figuring out the butterfly module, we observe that there are several characteristics involved in it. Firstly, the relationship between the branch metric bm_1 (bm_4) and bm_2 (bm_3) can be expressed in the following formula:

$$bm_2 = bm_3 = (e_m \dots e_3 e_2 e_1) = (\bar{d}_m \dots \bar{d}_3 \bar{d}_2 \bar{d}_1) \quad (4-10)$$

This can be implemented by simply applying an inverter after the output of the bm_1 (bm_4). Secondly, the most significant bit (MSB) of a final state could be used as a decision bit on the path history.

The branch metrics are calculated based on the differences between the received symbol and the corresponding label from the encoder trellis. These encoder branch labels are the code symbols that would be expected to come from the encoder output as a result of the state transitions. There are two decision methods: hard-decision and soft-decision. For the hard-decision decoding, the metric is called Hamming distance. The Hamming distance $d(A, B)$ between A and B is defined as the number of the differing elements. In the soft-decision decoding, the Euclidean distance is used. If we assume that the received symbol is A , and the encoder symbol is B , the Euclidean distance can be obtained from the formula $(A - B)^2$. We can simplify this calculation in the following way:

Since our design utilized the code rate $1/2$, A can be extended to $A_1 \cdot A_2$, and B can be written as $B_1 \cdot B_2$. Then

$$(A - B)^2 = (A_1 - B_1)^2 + (A_2 - B_2)^2 = A_1^2 + B_1^2 - 2A_1 \cdot B_1 + A_2^2 + B_2^2 - 2A_2 \cdot B_2 \quad (4-11)$$

$A_1^2, B_1^2, A_2^2, B_2^2$ are all the same for all possible branches, therefore, these terms can be omitted from this calculation which leave us only with $-2A_1 \cdot B_1 - 2A_2 \cdot B_2$. As mentioned before, B is the encoder symbol that only has 4 possible values when code rate $1/2$. Further simplification is shown in Table 4-7.

B_2B_1	divide by 2	add $A_1 + A_2$ and then divide by 2
00	$-A_1 - A_2$	0
01	$-A_1 + A_2$	A_2
10	$A_1 - A_2$	A_1
11	$A_1 + A_2$	$A_1 + A_2$

Table 4-7: Simplification Process in the calculation of the Euclidean distance

If the data is punctured before transmission, the null bits have to be added at the beginning of the decoding. Thus, we can construct the block diagram of BMU (Figure 4-17) in this way. The puncturing pattern table is stored in the memory section, and it controls the zero insertion on the input sequence. There is a *zero_insert* signal connected with the BMU; if *zero_insert* = 0, it means that nothing has to be done. Otherwise, some null bits should be added.

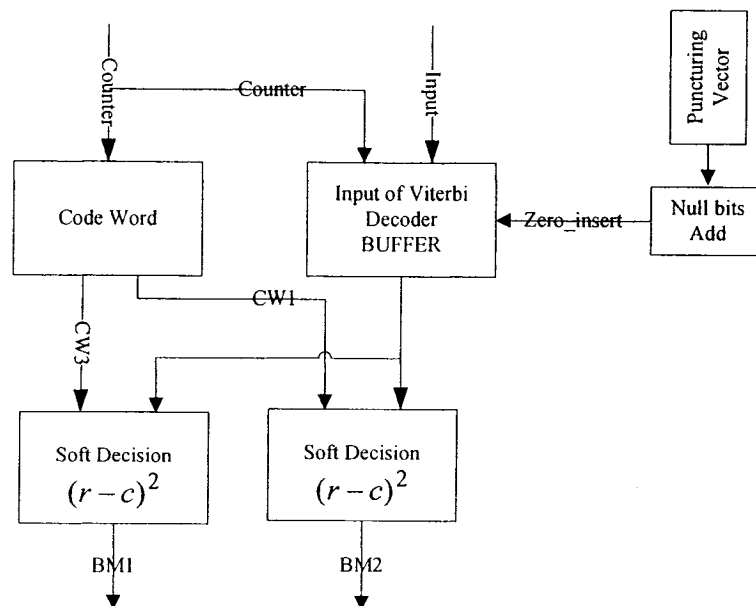


Figure 4-17: Block Diagram of BMU for soft decision

4.3.3.2 Add Compare Select Unit (ACS)

The logic executes the special purpose computation called add-compare-select (ACS). Three tasks are included in the unit. First, it obtains the corresponding path metric of the survivor path for each state by accumulating the branch metrics. Second, it gets the decision bit which will be used in the survivor memory unit. Third, the proper value of the current Path Metric (PM) can be saved in the register, and the decision bit is based on the MSB of the states to which branches merges. Figure 4-18 illustrates the ACS unit. There are 32 ACSs used when the constraint length is 7. Two 12-bits summations result from the Adder for the Comparator.

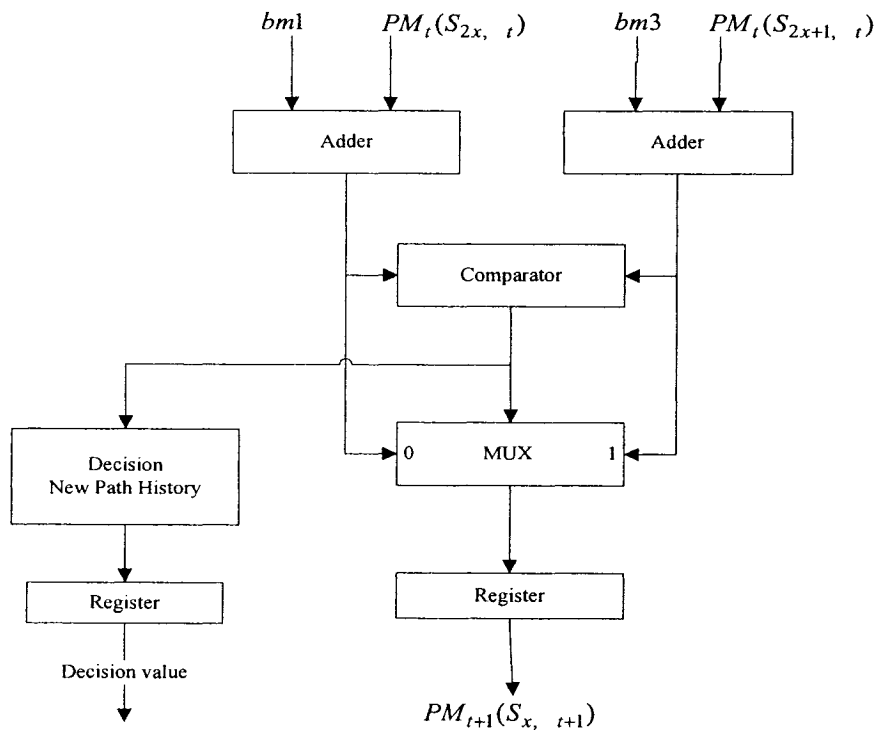


Figure 4-18: The ACS module

4.3.3.3 Path Metric Memory Unit (PMM)

General speaking, there are two approaches for storing the path metrics, namely, the ping pong mode and in-place scheduling. The traditional ping-pong mode doubles the size of RAMs for the path metric memory because a pair of memories is used for the operation of ACS units. One memory space provides the previous path metrics as the input of ACS, and the other one is responsible for saving the new survivor path metrics. This method increase the area, however the control logic is easy to implement. In-place scheduling use only half of the ping pong mode memory to renew path metrics by recursively overwriting the old path metrics, but the control logic complexity is increased.

In order to get a smaller area, we select the in-place scheduling. A dual port memory is employed here, which can either read two old path metrics or update two new path metrics at the same time. From Figure 4-19, we will find that a pair of ACS modules works in parallel within a single clock cycle.

The memory section is divided into 2 parts; both of them are of equal size. As shown in the figure above, the $2^K \times 12$ SRAM is separated into two $2^{K-1} \times 12$ areas, and $K = 7$. Meanwhile, Mux2 is used to read and distribute the value of the previous path metrics to the ACS modules, and Mux1 is used to gather the new values of the current path metrics and write them to refresh the constant in the SRAM; therefore, the overall memory bandwidth is increased such that two ACS modules are able to fetch the corresponding path metric without any conflict.

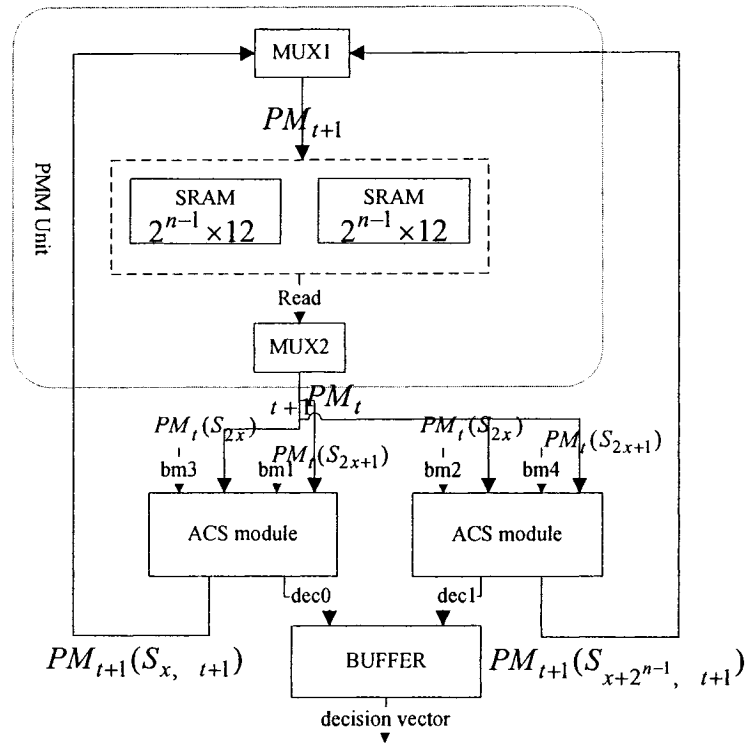


Figure 4-19: The PMM with ACS for Butterfly Unit

4.3.3.4 Survivor Memory Unit

Basically, there are two techniques widely used for the Viterbi Decoder (VD). They are called Register Exchange and Trace Back separately. Conceptually, the Register Exchange (RE) method is simpler and faster than the Trace Back (TB) method, in the RE method, each bit in the memory must be read and rewritten for each bit of information that is decoded. Therefore, the RE method is not appropriate for decoders with long constraint lengths. Although researchers have focused on implementing and optimizing the TB method, the RE method is still used in our design. Because the constraint length in our design is only up to 7 and the number of the messages sent from transmitter is limited to 100 bit/ frame.

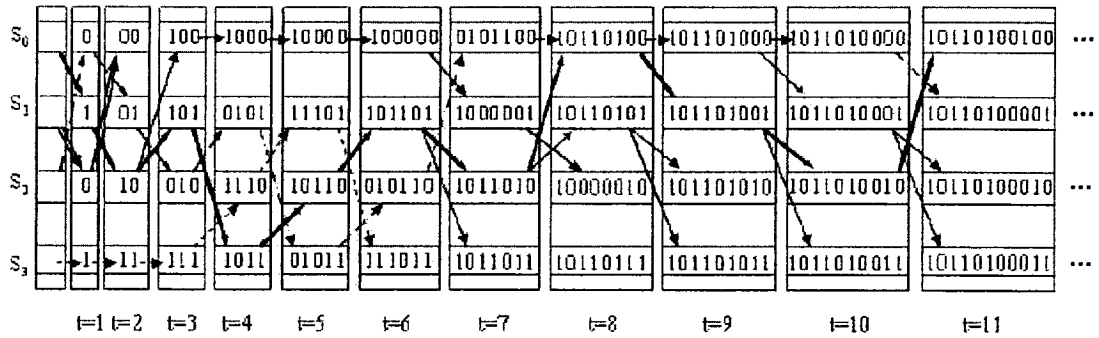


Figure 4-20: Register Exchange (RE) Method

In the RE approach, a register is assigned to each state. The register records the decoded output sequence along the path from the initial state to the final state. This is depicted in Figure 4-20, here we only show the case when constraint length is 3. At the last stage, the decoded output sequence is the one that is stored in the survivor path register, the register assigned to the state with the minimum PM. Since the RE method does not need tracing back, it is faster. However, the RE method does require the copying of all the registers at each stage.

Let us review the trellis which is shown in Figure 4-15; there is a point we need to emphasize. Actually, we do not need to calculate the full 2^{K-1} butterfly module all the time. For the $K-1$ stages in the beginning, because only one branch forwards into next states until process $K-1$ times in each state, just half of butterfly module need to be calculated. On the other hand, during the end of $K-1$ stages, half of butterfly module is calculated as well, but it is very obvious that decision vector is predicted to be zeros since the track of the trellis turns back to all zero state. The task for the ACS is reduced to only compare the value of PMs. Finally, at the end of the trellis, there is only one state, the all zero state, and hence only one survivor, and the algorithm terminates.

4.3.4 Memory Unit Specification

From the calculation units introduced above, we realize that it is necessary to reserve several memory sections for receiving buffers, PMs, BMs, and survivor path as well. In the following part, we explicate the size of each memory.

Although it is possible to unify the system address bus to the same bit width, we introduce four categories of memory address format for different bit widths in our project. Their sizes and functions are 256×4 for Receiver Buffer, $128 \times 5 / 32 \times 5 / 8 \times 5$ for BMs, $64 \times 12 / 16 \times 12 / 4 \times 12$ for PMs, and 128×1 for Survivor paths individually.

I. 256×4 SRAM

8 bit Memory Address is used for the input buffer. This address type is reserved for memories which could store 4 bits of data at each time instant. Here, we assume that transmitter sends 100 bits per frame, for example, memory $K=7$ convolutional component code is used, and then the size of the "Received Buffer" is decided to be 212 bits. Therefore, 8 bits is employed to represent the index of these 212bits ($2^7 < 212 < 2^8$). And follow the same arithmetic, by keeping send the same bits per frame, 8 bits of memory address also can be used for $K=3$ and $K=5$ convolutional component codes as well.

II. $128 \times 5 / 32 \times 5 / 8 \times 5$ SRAM

We use 7-bit/5-bit/3-bit Memory Address for BMs. It is because that these metric have different paths due to different states at each time instant. When using memory $K=7$

convolutional component code, for all conjoined steps in each stage, we will need at most 7-bits to address to represent 128 different values of BMs, and in the case of memory $K=3$ and $K=5$, 3-bit memory address and 5-bit address memory can be utilized, respectively.

III. 64 X 12 / 16 X 12 / 4 X 12 SRAM

Since there are two trellis branches merge into each state at each time point, we accumulate and compare value at each step, feed back and refresh the updated accumulation of PMs into memory on the clock's rising edge. If the architecture of the PMM in the previous part is used, the requirement of the memory reduces to 6-bit address for 64 states' accumulation. Similarly, 4-bit memory address can be used for 16 states' accumulation; 2-bit memory bus can be used for a 4 state' accumulation.

IV. 128 X 1 SRAM

7-bit memory address is used for all the survivor path contained all the decision bit. Because we assume that the transmitter sends 100 bits per frame, the survivor path definitely has the same length no matter what the memory order is.

4.3.5 Input/ Output Interface

We present the I/O interface layout of the entire Viterbi decoder for our design in Figure 4-21.

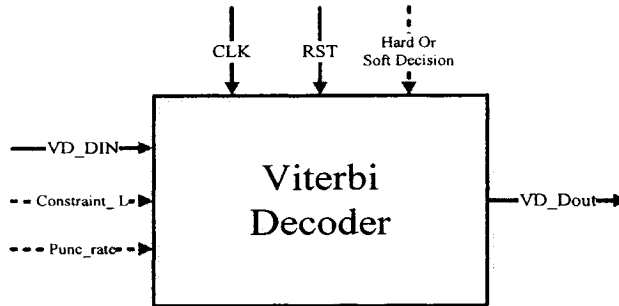


Figure 4-21: I/O interface for Viterbi Decoder

The user-configurable parameters are depicted in Table 4-8 below.

Term	Name	Definition	Range
Constraint_L	Constraint Length	Memory order of Convolutional Code	2,4,6
VD_DIN	Data input	Since each message frame contains n bits, and 4 bits quantization is used, data input contains $(2n+L) \times 4$ bits	$(20+L) \times 4 / (200+L) \times 4$
CLK	Clock	Clock-rising edge active	0/1
RST	Reset	Asynchronous Reset	0/1
Soft/Hard Decision	Decoder Decision	'0' represents the hard decision '1' represents the soft decision	0/1
Punc_rate	Puncture Rate	Ratio of output to input bits for convolutional encoder using the puncture process	See Table 4.1
VD_DOUT	Data output	The output contains n bits as the same as message bits.	10/100

Table 4-8: Viterbi Decoder Interface parameter

4.3.6 RTL Level Simulation

The simulation is used for checking the functionality of the circuit. Some of the simulation results are listed below. For the Viterbi Decoder of the K=7 convolutional code, ten message bits are considered in the test bench, while for the same Viterbi Decode with the punctuation ratio 4/7, sixteen message bits are considered in the test bench. And the number of the message bits equal to 100 bits as shown in the next section.

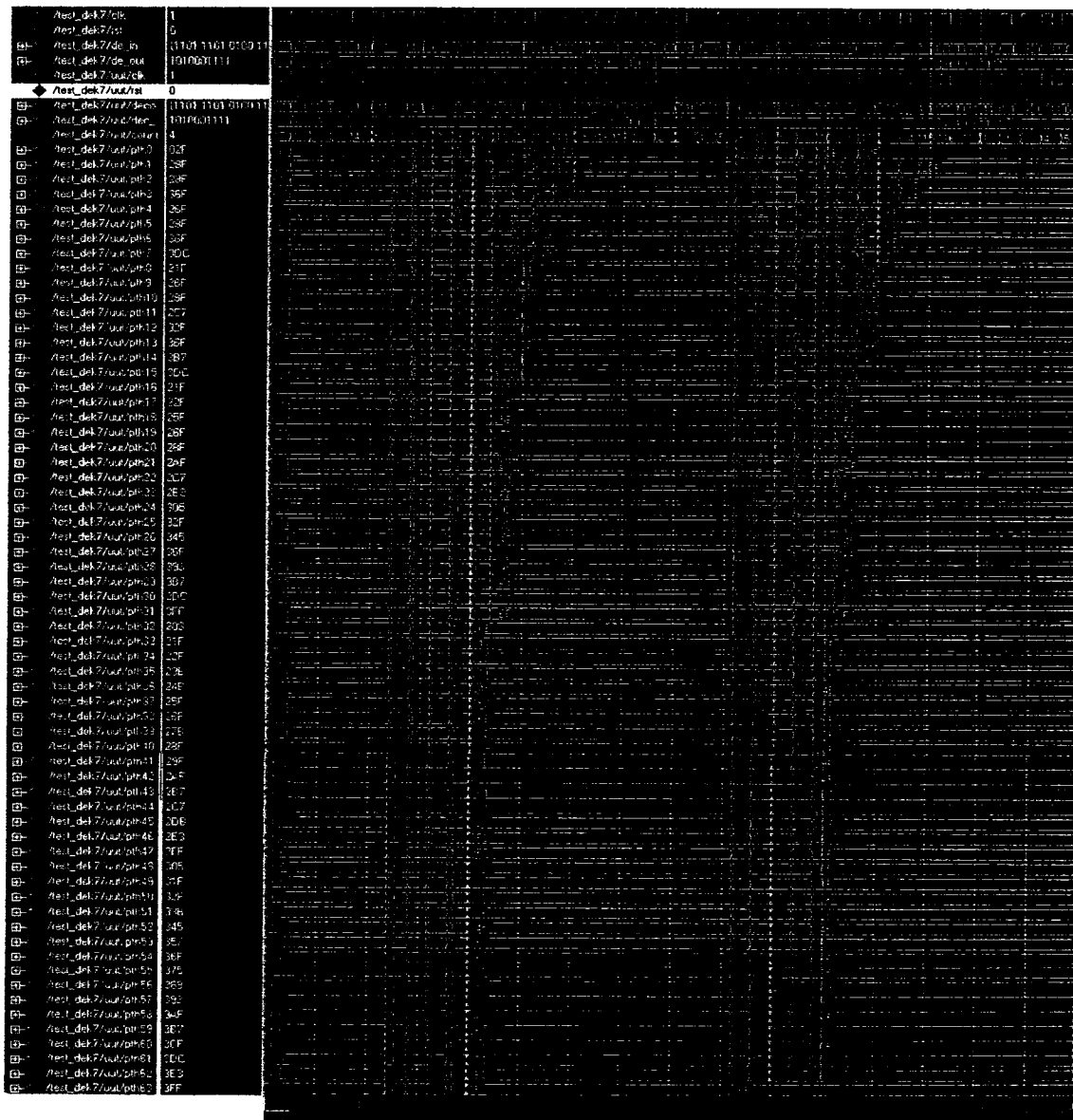
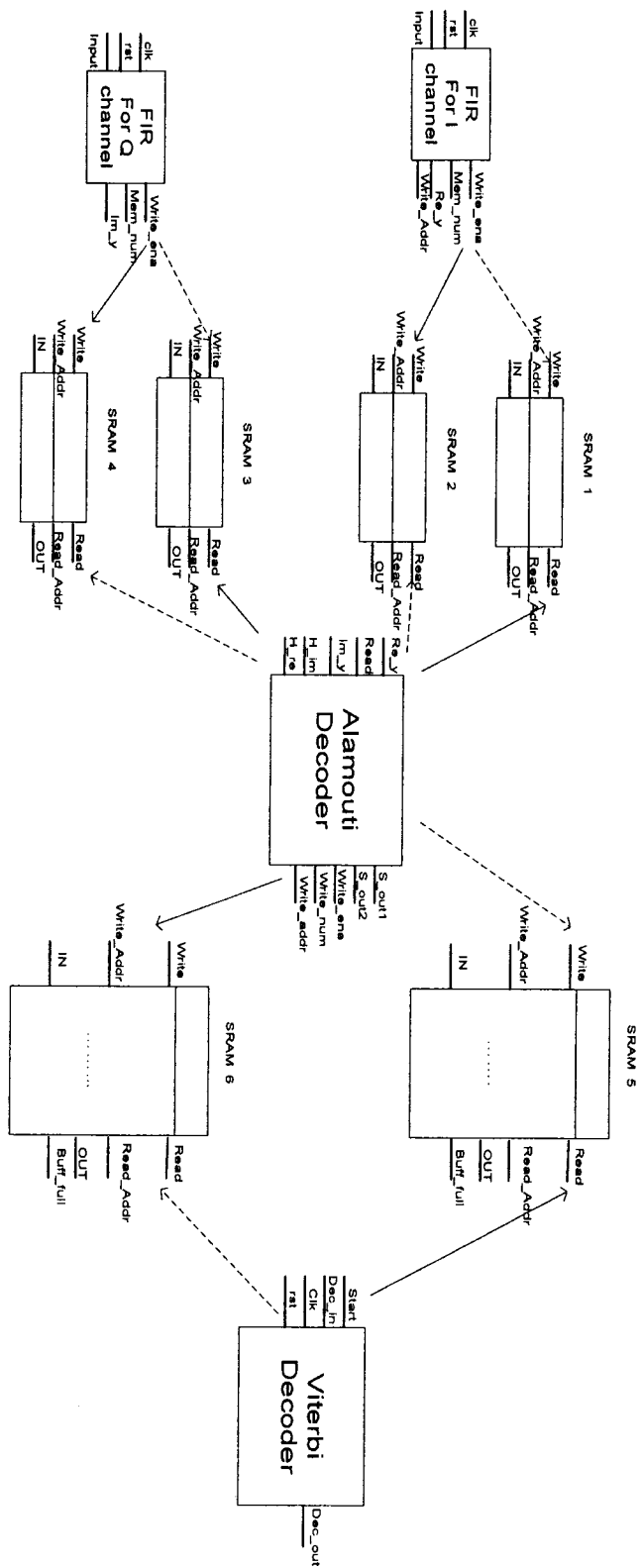


Figure 4-22: Viterbi Decode K=7 simulation wave form

Act_dec00k	1	
Act_dec00t	0	
Act_dec00g	00011010100000	
Act_dec00p	00000000111111	
Act_dec00b00	00	
Act_dec00p01	00010000111111	
Act_dec00p01	00010000111111	
Act_dec00p02	00000000111111	
Act_dec00p03	00111000111111	
Act_dec00p04	011101000	
Act_dec00p05	011101000	
Act_dec00p06	011101000	
Act_dec00p07	011101000	
Act_dec00p08	011101000	
Act_dec00p09	011101000	
Row	1600ns	
Cursor 1	740ns	

Figure 4-23: Viterbi Decoder with punctuation ratio 4/7

Figure 4-24: The Architecture of the Uplink Receiver



4.3.7 Overall Architecture of the Receiver

In the receiver, we use two SRAMs to store the results of the previous output in two adjacent time periods. This is especially used in the case where the next element processor is slower than the previous element processor. When the previous unit's output is writing into one of the memory banks, the next unit input reads the content of the other memory. For example, referring to Figure 4-24, SRAM1, SRAM2, SRAM3, and SRAM4 are the connections of the pair of the SRRC filters and Alamouti decoder. SRAM1 and SRAM2 are in charge of storing the outputs from the *I channel* filter, and providing the inputs for the Alamouti Decoder. At each rising clock edge, the *I channel* filter write to SRAM1(SRAM2), while, the Alamouti decoder read the contents from the SRAM2(SRAM1). Since the SRRC filter works 4 times faster than the Alamouti Decoder, the Alamouti decoder fetches the adjacent two inputs from the output of the I/Q channel, we created two units in each SRAM. The same approach can be applied in the connection of Alamouti decoder and punctured convolutional decoder, but these SRAMs should be configurable because of the different constraint lengths, rates, and frame lengths. For instance, if we consider the convolutional code with rate 1/2, and constraint length $K=7$, by decoding frames with 100 bits one by one, the size of one of these SRAMs should be 212 units.

The Overall RTL level simulation is shown in Figure 4-25. It takes 8.3 μ s to decode a 100-bit frame. Therefore, the receiver can work at the speed of 12 Mbps with the target device of Xilinx Virtex 2 Pro (xc2vp40). The design summary is specified by the Xilinx ISE tools in Figure 4-26.

Release 7.1i Map H.38

Xilinx Mapping Report File for Design 'Receiver'

Design Information

Command Line : /CMC/tools/xilinx_7.1i/bin/sol/map -ise
/nfs/home/b/b_jia/Synopsys/Xilinx/Receiver/Receiver.ise -intstyle ise -
p

xc2vp40-ff1148-7 -cm area -pr b -k 4 -c 100 -tx off -o Nov_14_map.ncd
Nov_14.ngd
Nov_14.pcf
Target Device : xc2vp40
Target Package : ff1148
Target Speed : -7
Stepping Level : 0
Mapper Version : virtex2p -- \$Revision: 1.26.6.3 \$
Mapped Date : Thu Nov 23 09:58:53 2006

Design Summary

Logic Utilization:

Total Number Slice Registers:	7,982 out of	38,784	20%
Number used as Flip Flops:		116	
Number used as Latches:		7,866	
Number of 4 input LUTs:	11,864 out of	38,784	30%
Logic Distribution:			
Number of occupied Slices:	7,929 out of	19,392	40%
Number of Slices containing only related logic:	7,929 out of		
7,929	100%		
Number of Slices containing unrelated logic:		0 out of	
7,929	0%		
Total Number 4 input LUTs:	12,035 out of	38,784	31%
Number used as logic:		11,864	
Number used as a route-thru:		171	
Number of bonded IOBs:	152 out of	804	18%
IOB Flip Flops:		32	
IOB Latches:		216	
Number of PPC405s:	0 out of	2	0%
Number of MULT18X18s:	2 out of	192	1%
Number of GCLKs:	14 out of	16	87%
Number of GTs:	0 out of	12	0%
Number of GT10s:	0 out of	0	0%

Total equivalent gate count for design: 138,595

Additional JTAG gate count for IOBs: 7,296

Peak Memory Usage: 353 MB

Figure 4-26: Hardware Design Summary

CHAPTER FIVE

CONCLUSION

The scheme investigated in this thesis is based on cooperative detect-forward approach. Employing the ideas from the relay channel, the development of the new communication system is on the upswing. The three most critical approaches are the amplify-forward, detect and forward, and code cooperation. For example, the amplify-and-forward method, [15], [19] where each user simply amplifies the noisy versions received from its partner and then retransmits these to the destination is the simplest method. Then, the detect-and-forward method [18] came from the traditional idea of relay channel. The CDMA implementation applies spreading codes to create two separate channels, thus, full diversity can be achieved. Finally, the coded cooperation method [20], [21], [22] is to use the same overall rate for coding and transmission; the redundancy for each partner is transmitted by the corresponding user.

The objective of this thesis is to propose a new cooperative communication system, analyze its performance, and introduce its hardware implementation. We construct a virtual space time coding for a cooperative system that was developed based on the concepts of the traditional detect-forward approach and combined with the well known space time block coding and channel coding using RCPC code. Due to the fact that in the

cooperative system the inter-user channel (between users) is less noisy than the uplink channel (from users to the destination), we applied 16 *QAM* modulations in the inter-user channel in order to acquire higher data rate. The log likelihood ratio approach was used for the inter-user channel. Then, for the uplink channel, one user and its partner will send bits together to the destination. Therefore, the application of Alamouti space time coding in the uplink channel is intuitively justified. For getting a better performance and keeping the same bandwidth, we utilized a rate-compatible punctured convolutional code for both user and its partner. We present the simulation result with 100,000 frames, where each frame contains 160 bits. Frame error rate of our scheme is studied and compared to the performance of the convolutional code in both SISO and MIMO systems. We assume that the bits detected by the partner in the inter-user communication phase are error free. Energy allocation has been done through estimation before sending. We employ an effective RCPC code to keep the same bandwidth. If we compare our scheme with a non-cooperative system, there are 4 dB improvement obtained. Moreover, the effect of the distance between the source and relay on the achieved improvement is also shown. It is evident that the scheme's performance improves with the reduction of this distance. In other words, Rather than spending energy on inter-node communication, we consume more energy to ensure a better performance on the uplink channel.

Furthermore, the corresponding hardware implementation is presented. We present the complete VLSI design for the uplink receiver, which consists of a pair of parallel Square Root Raise Cosine Filters, the Alamouti decoder controlled by the Moore state machine, and Viterbi decoder for RCPC code. Later on, the design is modeled with VHDL in RTL level and then the simulation is performed by Modelsim, after synthesizing by Xilinx

tools, placing and routing has been verified with gate level simulation. The total area consumption for the test FPGA board is shown. Finally, The estimated achievable transmission speed for the series of Xilinx Virtex 2 Pro family was obtained.

5.1 Suggestions for Future Research

In this thesis, we describe the software design and hardware implementation of a space-time coded virtual MISO cooperative system. Unlike previous relay channel models, we employ the full transmission diversity and achieve better performance. However, there is a potential for the application of some techniques such as different channel models, advanced channel codes, more complicated MIMO system approaches and custom design implementation.

Channel Models and System Characteristics: We focused throughout the thesis on the Rayleigh fading channel model. Moreover, we have considered scenarios in which channel state information is available only to the appropriate receivers. We expect our results to be readily extendable to more general fading distributions. More general communication strategies than the ones considered in preceding chapters arise under different system characteristics and using alternative performance measures. If the transmitters obtain channel state information, power control becomes possible. It would be interesting to determine whether cooperative diversity with power control improves substantially over the case of power control alone, and how they relate to the basic algorithms developed in this thesis.

Practical Coding and Decoding Algorithms: Throughout the thesis we have employed RCPC coding to evaluate performance of our cooperative scheme. We could obtain some possible improvement by altering the coding scheme. There is no doubt that employing a

stronger code, like turbo code, can provide better results. Moreover, instead of using BPSK, QPSK/8PSK can be employed for the virtual MISO channel. The OFDM technique may be applied to gain higher spectral efficiency, and to reduce multi-path distortion. More generally, designing effective algorithms, evaluating performance, and selecting codes are necessary for practical implementation of cooperative diversity. In addition, multi user cooperative communication (more than two cooperative nodes) should be considered in future research.

Some improvements in implementation: There exist some aspects of the hardware implementation that can be further improved. First, there are many details in this design that can be refined to improve the performance in terms of the area consumption and power consumption. Second, some of the algorithms can be implemented by a DSP processor. For example, the Viterbi engine can be included in the core for some modern digital signal processors (DSPs). The wireless multimedia DSP chip in [62] supports the packed ACS byte instruction for Viterbi Decoding. Third, although this design is implemented in the Xilinx Virtex 2 pro FPGA, it is necessary to map this design into an advanced ASIC technology for more accurate implementation.

REFERENCES

- [1] Edward C. van der Meulen. "*Transmission of Information in a T-Terminal Discrete Memoryless Channel.*" Department of Statistics, University of California, Berkeley, CA, 1968.
- [2] Edward C. van der Meulen. "*Three-terminal communication channels.*" Adv. Appl. Prob., 3:120{154, 1971.
- [3] Thomas M. Cover and Abbas A. El Gamal. "*Capacity theorems for the relay channel.*" IEEE Trans. Inform. Theory, 25(5):572{584, September 1979.
- [4] Brett Schein and Robert G. Gallager. "*The Gaussian parallel relay network.*" In Proc. IEEE Int. Symp. Information Theory (ISIT), page 22, Sorrento, Italy, June 2000.
- [5] Brett Schein. "*Distributed Coordination in Network Information Theory.*" PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, August 2001.
- [6] Frans M.J. Willems. "*Information theoretical Results for the Discrete Memoryless Multiple Access Channel.*" PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, October 1982.
- [7] Frans M.J. Willems. "*The discrete memoryless multiple access channel with partially cooperating encoders.*" IEEE Trans. Inform. Theory, 29(3):441 {445, May 1983.
- [8] Frans M.J. Willems and Edward C. van der Meulen. "*The discrete memoryless multiple access channel with cribbing encoders.*" IEEE Trans. Inform. Theory, 31(3):313 {327, May 1985.
- [9] Frans M.J. Willems, Edward C. van der Meulen, and J. Pieter M. Schalkwijk. "*An achievable rate region for the multiple access channel with generalized feedback.*" In Proc. Allerton Conf. Communications, Control, and Computing, pages 284 {292, Monticello, IL, October 1983.
- [10] Gerhard Kramer and Adriaan J. van Wijngaarden. "*On the white Gaussian multiple access relay channel.*" In Proc. IEEE Int. Symp. Information Theory (ISIT), page 40, Sorrento, Italy, June 2000.
- [11] Andrew Sendonaris, Elza Erkip, and Behnaam Aazhang. "*Increasing uplink capacity via user cooperation diversity.*" In Proc. IEEE Int. Symp. Information

Theory (ISIT), Cambridge, MA, August 1998.

- [12] Andrew Sendonaris, Elza Erkip, and Behnaam Aazhang. "*User cooperation Diversity Part I: System description.*" Submitted to IEEE Trans. Commun., 1999.
- [13] Andrew Sendonaris, Elza Erkip, and Behnaam Aazhang. "*User cooperation diversity, Part II: Implementation aspects and performance analysis.*" Submitted to IEEE Trans. Commun., 1999.
- [14] V. Tarokh, H. Jafarkhani, and A. Calderbank, "*Space-time block codes from orthogonal designs,*" IEEE Transactions on Information Theory, vol. 45, issue 5, pp. 1456-1467, July 1999.
- [15] J.N.Laneman, G.W.Wornell, and D.N.C.Tse, "*An efficient protocol for realizing cooperative diversity in wireless networks*" in Proc IEEE ISIT Washington D.C., June 2001, p 294.
- [16] A. Sendonaris, E.Erkip, and B.Aazhang , "*Increasing uplink capacity via user cooperation diversity,*" in Proc. IEEE ISIT, Cambridge, MA, August 1998
- [17] A. Sendonaris, E.Erkip, and B.Aazhang , "*User cooperation diversity Part I: System description,*" IEEE Trans. Commun., 2002
- [18] A. Sendonaris, E.Erkip, and B.Aazhang , "*User cooperation diversity Part II: Implementation aspects and performance analysis,*" IEEE Trans. Commun, 2002
- [19] J.N.Laneman, "*Cooperative diversity in wireless networks: Algorithms and Architectures*" Ph.D. dissertation, Massachusetts Institute of Technology, August 2002
- [20] T.E.Hunter and A.Nosratinia, "*Cooperative diversity through coding,*" in Proc IEEE ISIT, Laussane, Switzerland, July 2002, p220
- [21] T.E.Hunter and A.Nosratinia, "*Coded Cooperative under slow fading, fast fading, And power control,*" in Proc. Asilomar Conference on signals, systems and computer, Pacific Grove, CA, November 2002.
- [22] T.E.Hunter and A.Nosratinia, "*Performance analysis of coded cooperation diversity,*" in Proc.IEEE ICC, Anchorage, AK, May 2003.
- [23] J.Chandran, R.Kaluri, J.Singh, V.Owall and R.Velijanovski, "*Xilinx Virtex II Pro Implementation of a Reconfigurable UMTS Digital Channel Filter*" IEEE Computer Society 2004
- [24] Ahmed Elhossini, Shawki Areibi, Robert Dony, "*An FPGA Implementation of the*

LMS Adaptive Filter of Audio Processing”

- [25] Ken Chapman, Paul Hardy, Andy Miller, and Maria George “*CDMA Matched Filter Implementation in Virtex Devices*” January 10, 2001
- [26] Marko Savolainen and Jari Nurmi, “*Reusable Viterbi Decoder Implementation*”
- [27] Peter J. Green and Desmond P. Taylor, “*Implementation of a High Speed Four Transmitter Space-Time Encoder using Field Programmable Array and Parallel Digital Signal Processors*” IEEE Computer Society 2005
- [28] Ibrahim Abou-Faycal, Mitchell D. Trott, and Shlomo Shamai (Shitz). “*The capacity of discrete-time Rayleigh fading channels.*” In Proc. IEEE Int. Symp. Information Theory (ISIT), page 473, 29 June { 4 July 1997.
- [29] Ibrahim C. Abou-Faycal, Mitchell D. Trott, and Shlomo Shamai (Shitz). “*The capacity of discrete-time memoryless rayleigh-fading channels.*” IEEE Trans. Inform. Theory, 47(4):1290{1301, May 2001.
- [30] Thomas H. E. Ericson. “*A Gaussian channel with slow fading.*” IEEE Trans. Inform Theory, 16(3):353{355, May 1970.
- [31] Andrea J. Goldsmith and Pravin P. Varaiya. “*Capacity of fading channels with Channel side information.*” IEEE Trans. Inform. Theory, 43(6):1986{1992, November 1995.
- [32] Amos Lapidoth and Prakash Narayan. “*Reliable communications under channel Uncertainty*” IEEE Trans. Inform. Theory, 44(6):2148{2177, October 1998.
- [33] Berrou, Glavieux, and Thitimajshima “*Near Shannon Limit error-correcting coding and decoding: Turbo-codes*” published in the Proceedings of IEEE International Communications Conference 1993
- [34] Shulin, Daniel J. Costello, Jr. “*Error Control Coding: Fundamentals and Applications*” ISBN 0-13-283796-X
- [35] C. E. Shannon, “*The Mathematical Theory of Communication.*” Urbana, IL:University of Illinois Press, 1949 (reprinted 1998).
- [36] D. Gesbert “*Smart antennas and spatial multiplexing*” June 1999
- [37] Vahid Tarokh, Nambi Seshadri, and A. R. Calderbank (March 1998). “*Space-time codes for high data rate wireless communication: Performance analysis and code construction*”. *IEEE Transactions on Information Theory* **44** (2): 744–765.

- [38] Toby Haynes, "*A Primer on Digital Beamforming*" March 26, 1998
- [39] Gerard J. Foschini. "*Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas.*" Bell Syst. Tech. J., pages 41 {59, Autumn 1996.
- [40] Gerard J. Foschini and Michael J. Gans. "*On limits of wireless communications in a fading environment when using multiple antennas.*" Wireless Personal Communications, 6(3):311 {335, March 1998.
- [41] S.M. Alamouti (October 1998). "*A simple transmit diversity technique for wireless communications*". IEEE Journal on Selected Areas in Communications
- [42] Vahid Tarokh, Hamid Jafarkhani, and A. R. Calderbank (July 1999). "*Space-time block codes from orthogonal designs*". IEEE Transactions on Information Theory
- [43] Vahid Tarokh, Nambi Seshadri, and A. R. Calderbank (March 1998). "*Space-time codes for high data rate wireless communication: Performance analysis and code construction*". IEEE Transactions on Information Theory
- [44] A.R.Calderbank, "*The art of signaling: Fifty years of coding theory*" IEEE Trans. Inform. Theory, vol. 44, pp. 2561-2595 Oct 1998
- [45] S.Verdu, "*Wireless bandwidth in the making,* " IEEE Commun Mag., pp 53-58, July 2000
- [46] A. Naguib, N. Seshadri, and A. R. Calderbank "*Increasing data rate over wireless channels*" IEEE Signal Processing Mag., pp 76-92 May 2000.
- [47] Thomas M. Cover and Joy A. Thomas. "*Elements of Information Theory.*" John Wiley& Sons, Inc., New York, 1991.
- [48] Udo Wachsmann, Robert F.H. Fischer, and Johannes B. Huber. "*Multilevel codes: Theoretical concepts and practical design rules.*" IEEE Trans. Inform. Theory, 45(5):1361 {1391, July 1999.
- [49] Toby Berger, Zhen Zhang, and Harish Viswanathan. "*The CEO problem: Multiterminal source coding.*" IEEE Trans. Inform. Theory, 42(3):887 {902, May 1996.
- [50] Michael Gastpar, Gerhard Kramer, and Piyush Gupta. "*The multiple-relay channel: Coding and antenna-clustering capacity.*" In Proc. IEEE Int. Symp. Information Theory (ISIT), page 136, Lausanne, Switzerland, July 2002.
- [51] Michael Gastpar and Martin Vetterli. "*On the asymptotic capacity of Gaussian relay networks.*" In Proc. IEEE Int. Symp. Information Theory (ISIT), page 195,

Lausanne, Switzerland, July 1 {5 2002.

- [52] Michael Gastpar and Martin Vetterli. "*On the capacity of wireless networks: The relay case.*" In Proc. IEEE INFOCOM, New York, NY, June 2002.
- [53] Piyush Gupta and P.R. Kumar. "*Towards and information theory of large networks: An achievable rate region.*" In Proc. IEEE Int. Symp. Information Theory (ISIT), Page 150, Washington DC, June 2001.
- [54] T.E.Hunter and A. Nosratinia, "*Diversity through coded cooperation,*" in IEEE J.select.Areas Commun., 2003, submitted for publication.
- [55] J.Hageanauer and T. srockhammer, "*Channel coding and transmission aspects of wireless multimedia,*" Proc IEEE vol 87, pp 1764-1777, Oct 1999
- [56] S.S. Hemami, "*Robust image communication over wireless channels*" IEEE commun Nov 2001
- [57] M.Janani, A. Hedayat, T.E. Hunter, and A. Nosratina, "*Code Cooperation in wireless communication: Space-time transmission and iterative decoding,*" IEEE Trans. Signal Processing 2003
- [58] M.Surendra Raju, A.Remesh and A. Chochalingam "*BER Analysis of QAM with Transmit Diversity in Rayleigh Fading Channels*" Globecom 2003
- [59] S. Alamouti, "*A simple transmit diversity technique for wireless communications*" IEEE Journ. On Select. Areas in Comm.
- [60] J. Hagenauer, "*Rate-compatible punctured convolutional codes and their Application,*" IEEE Trans Commun. April 1988
- [61] Y.Shen and P.C Cosman "*weight distribution of a class of binary linear block codes formed from RCPC codes.*"
- [62] K. L. Heo, M. H. Sunwoo, and S. K. Oh, "*Implementation of a wireless multimedia dsp chip for mobile applications,*" IEEE Workshop on Signal Processing Systems, pp. 51-56, August 2003.
- [63] R.E.Kahn. "*The organization of computer resources into a packet radio network*" IEEE Trans. Commun January 1977.
- [64] R.E.Kahn, S.A.Gronemeyer, J.Burchfiel, and R.C.Kunzelman. "*Advance in packet radio technology*" November 1978.
- [65] Piyush Gupta and P.R. Kumar "*The capacity of wireless networks*" IEEE Trans Inform March 2000.

[66] Timothy J. Shepard. "*Decentralized Channel Management in Scalable Multihop Spread Spectrum Packet Radio Network.*" July 1995

[67] Matthias Grossglauser and David N.C. Tse. "*Mobility increase the capacity of ad-Hoc wireless networks*" IEEE/ACM Trans Networking March 2001