

SPACE CODING APPLIED TO HIGH-SPEED CHIP-TO-CHIP INTERCONNECTS

by

Kamran Farzan



A thesis submitted in conformity with the requirements
for the Degree of Doctor of Philosophy,
The Edward S. Rogers Sr. Department
of Electrical and Computer Engineering,
at the University of Toronto

© Copyright by K. Farzan, 2004

SPACE CODING APPLIED TO HIGH-SPEED CHIP-TO-CHIP INTERCONNECTS

Kamran Farzan

Doctor of Philosophy

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering

University of Toronto, 2004

Abstract

This dissertation presents new signaling schemes and circuit architectures for reducing the power and cost of high-speed chip-to-chip links. After an overview on chip-to-chip interconnects and its building blocks, a new signaling scheme is proposed that can provide many advantages of a fully-differential signaling scheme while employing as few as $N + 1$ signal paths for communicating N differential signals. Next, power-efficient signaling schemes that use channel coding to achieve appreciable coding gain are proposed. One discussed method is to use 6-PAM signaling instead of 4-PAM to transmit two bits of information per channel. The proposed low-complexity architecture of this scheme makes its high-speed implementation feasible. A realistic model for a typical interconnect channel is used for simulations. We then introduce a coding scheme which employs a convolutional encoder in space to achieve 3-5 dB coding gain without expanding the modulation from 4-PAM to 6-PAM. The functionality and performance of the proposed scheme is verified by experimental results obtained from a fabricated chip based on this method. Finally, a novel power-efficient architecture for multi-level PAM drivers is presented. In addition, a data-look-ahead technique, used for high-speed implementation of this method, eliminates the need for a pre-driver to further reduce the driver power. Based on this architecture, a 4-PAM transmitter is designed in $0.18\mu\text{m}$ digital CMOS technology. The transmitter achieves 10 Gb/s with a 2-V supply and it occupies an area of 0.16 mm^2 . The output driver and the entire transmitter consume only 20 mW and 121 mW at 10 Gb/s, respectively, which are the lowest reported powers at this speed.

Acknowledgements

During the course of this research and the writing of the dissertation, I have been very fortunate to be surrounded by teachers, family, friends, and colleagues who have continually offered me support, encouragement, and aid.

I would like to express my deepest gratitude to my advisor, Professor David Johns, for his continual support, guidance, and friendship. This work would not have been possible without his excellent teaching and great insight. I really feel that I could not have asked and hoped for a better advisor.

I would also like to thank Professor Kschischang, Professor Loeliger, Professor Konrad, Professor Ng, and Professor Genov for taking the time to serve on my committee and for reviewing my thesis. Also, I like to express my appreciation to Professor Kschischang for his great comments throughout this research.

I feel very proud to be a part of such a talented research group: Bahram Zand, Amir Hadji-Abdolhamid, Prof. Tony Chan-Carusone, Samira Naraghi, Sherif Abdalla. I would like to thank them for their friendship and for offering unconditional assistance throughout this project. I am also grateful to my colleagues, Prof. Shahriar Mirabbasi, Prof. Anas Hamoui, Afshin Rezaee, Mohammad Hadji-Rostam, Afshin Haft-baradaran, Farshid Rezaee, Kasra Ardalan, Mehrdad Ramezani, Sotoudeh Hamedi-Hagh, Saman Sadr, Jorge Verona, Masoud Ardakani, Mehrdad Shamsi, Mehrdad Eslami, Yadi Eslami, Takis Zourntos, Sebastian Magierowski, Tooraj Esmailian, Reza Jafari, and many others for being there for me and for all the good time that we had together. Going through the difficulties of the graduate years would have not been possible without the joy, fun and happiness that have been brought to my life by great friends such as Kayvan, Farzad, Amir, Samira, AfshinHF, Afshin, Farshid, and Bahram.

It is certainly impossible to find the appropriate words to thank my wife, Arezoo, for her love, support, and patience over the years. Her presence gave me the will and strength to go through the difficulties in my graduate years. Also, I would not have been able to carry myself all the way through the Ph.D. degree without my parents, sister, and brother.

Growing up, I never had to worry about anything but my education. Thanks to my parents for all their sacrifices and all they have done for me. Thanks to my sister, Nasrin, and my brother, Kayvan, who cheerfully took over all my responsibilities at home during the term of this research. Finally, I would like to dedicate this thesis to my wife, who endured this long process with me, and to my mom and dad, for being the best parents in the world.

To my wife Arezoo and to my parents

Table of Contents

Abstract	i
1 Introduction	1
1.1 Motivation and Overview	1
1.1.1 Cost-Efficient Signaling Schemes	2
1.1.2 Power-Efficient Signaling Schemes	3
1.1.3 Power-Efficient Circuit Architectures	4
1.2 Organization	4
2 Chip-to-Chip Interconnect: Basic Architecture and Non-Ideal Issues	6
2.1 Channel	7
2.1.1 Packages	7
2.1.2 Transmission Lines	9
2.1.3 Noise in Digital Systems	10
2.2 Transmitter	12
2.3 Receiver	13
2.4 Timing in a Chip-to-Chip Link	14
2.4.1 Timing Recovery and Clock Generation	15
2.4.2 Skew	16
2.4.3 Jitter	16
3 Differential Signaling with a Reduced Number of Signal Paths	17
3.1 Incremental Signals	18
3.2 Peak Detection	20

3.3	Maximum Likelihood Sequence Detection with Viterbi Algorithm	23
3.4	The Viterbi Algorithm with Noise Cancellation	25
3.5	Balanced Codes	28
3.6	Analog Implementation of Incremental Signaling	30
3.7	Summary	31
4	Coding Schemes for Chip-to-Chip Interconnect Applications	32
4.1	Coding Schemes for Chip-to-Chip Communication	33
4.1.1	Coding Schemes for Two-Level Signaling	33
4.1.2	Coding Schemes for Multi-Level Signaling	35
4.2	A Realistic Channel Model for Chip-to-Chip Applications	41
4.3	Simulation Results	45
4.3.1	Simulation Results for the 3LINE-PAM2 Scheme	45
4.3.2	Simulation Results for the 4LINE-PAM6 Scheme	46
4.4	Analog Implementation	50
4.4.1	Analog Implementation of the 3LINE-PAM2 Scheme	50
4.4.2	Analog Implementation of the 4LINE-PAM6 Scheme	50
4.4.3	Circuit-Level Simulations	57
4.5	Summary	60
5	A Power-Efficient 4-PAM Signaling Scheme for Inter-Chip Links Using Coding in Space	61
5.1	Proposed Coding Scheme for Inter-Chip Communication	62
5.2	Simulation Results	66
5.2.1	Simulation Results with an AWGN Channel Model	66
5.2.2	Simulation Results with a More Realistic Channel Model	67
5.3	Different Implementations for the Proposed Method	67
5.4	Analog Implementation	72
5.5	Experimental Results	77
5.6	Summary	83

6	A CMOS 10-Gb/s Power-Efficient 4-PAM Transmitter	84
6.1	Driver Architectures	84
6.2	Power-Efficient Driver Topology	86
6.3	Simulation Results	88
6.4	Transmitter Architecture	90
6.5	Experimental Results	93
6.6	Summary	97
7	Conclusion and Future Work	98
7.1	Contributions	98
7.2	Future Work	100
	Appendices	108
A	MLSD Probability of Error Derivation	109
B	A Power-Efficient Architecture for High-Speed D/A Converters	112
B.1	A Power-Efficient Topology for DACs	112
B.2	6-bit Power-Efficient DAC Architecture	114
B.3	Simulation Results	116
B.4	Summary	117

List of Figures

2.1	Basic architecture of a chip-to-chip interconnect.	7
2.2	A typical package model.	8
2.3	Equivalent circuit model of a differential section of a transmission line (RLCG).	10
2.4	Reflection of incidental signal from an unmatched load.	11
2.5	Transmitter architecture for parallelism implementation.	12
2.6	Receiver architecture for parallelism implementation.	13
2.7	Receiver architecture for a system with 4:1 demultiplexer.	14
2.8	The basic architecture of (a) PLL and (b) DLL.	15
3.1	A practical single-ended signaling system or “pseudo-differential” signaling with additional reference lines added after every four signal paths.	19
3.2	Block diagram of a fully-differential signaling system for binary data.	19
3.3	A general incremental signaling system.	20
3.4	Block diagram of a possible incremental signaling system for binary data using peak detection.	21
3.5	Bit error rates for the fully-differential system in Fig. 3.2 and the peak detection system in Fig. 3.4. Lines are theoretical results and markers indicate simulation results.	23
3.6	Block diagram of an incremental signaling system using the Viterbi algorithm for MLSD at the receiver.	24
3.7	Bit Error Rate of the MLSD system in Fig. 3.6 with the regular Viterbi algorithm and the Viterbi-NC algorithm. Lines are theoretical results and markers are simulation results.	25

3.8	BER versus bus width, N , using the Viterbi-NC algorithm for MLSD at $SNR = 14$ dB.	28
3.9	Simulation results for the serial and parallel Viterbi-NC algorithms.	29
3.10	A system combining incremental signaling with balanced codes.	29
3.11	Termination of a balanced bus at the receiver with common-mode tap, y_{cm}	30
4.1	(a) General block diagram of a 2-PAM signaling scheme; (b) General Block diagram of the proposed coding scheme (3LINE-PAM2)	33
4.2	Simulation results for 3LINE-PAM2 and regular 2-PAM in the case of AWGN	34
4.3	Modified channel model to take into account the effect of crosstalk	35
4.4	Performance of the proposed method in the presence of crosstalk.	35
4.5	Subset partitioning in 4D space with 5-PAM in each dimension	37
4.6	Simulation Result for 4-PAM, 5-PAM, 4LINE-PAM6, and Coded-Modulation-PAM5 schemes	40
4.7	Simulation Result for 4-PAM, 4LINE-PAM6 in peak-power-limited case	42
4.8	Performance of the 4LINE-PAM6 and 4-PAM methods in the presence of crosstalk in peak-power-limited case.	42
4.9	A general block diagram for a chip-to-chip communication system.	43
4.10	Per-unit-length model of a transmission line.	43
4.11	Eyediagram for a 10 Gb/s link at the receiver (channel: package, 300-mm microstrip, package) when $Z_s = 80$ and $Z_l = 120$	45
4.12	Channel frequency response for the case of microstrip and $Z_L = 120\Omega$, $Z_S = 80\Omega$, $d = 0.3\text{m}$	46
4.13	3LINE-PAM2 simulation results.	47
4.14	Channel frequency response for the case of stripline $Z_L = 110\Omega$, $Z_S = 90\Omega$, $d = 0.2\text{m}$	48
4.15	Case I: $Z_L = 110\Omega$, $Z_S = 90\Omega$, 0.2-m stripline.	49
4.16	Channel frequency response for the case of microstrip with high-speed package and $Z_L = 105\Omega$, $Z_S = 95\Omega$, $d = 50\text{mm}$	49
4.17	Case II: $Z_S = 95\Omega$ $Z_L = 105$, 50-mm microstrip at 20Gb/s.	49
4.18	Transceiver architecture for the 3LINE-PAM2 method	51

4.19	Main idea of an analog implementation of the 4LINE-PAM6 scheme.	52
4.20	Block diagram of an analog implementation of 4LINE-PAM6 scheme.	53
4.21	The detail of the receiver block (L1-L4) for each line.	54
4.22	The detail of the 4×3 decoder.	55
4.23	Transmitter structure	57
4.24	Simulation results for different implementations of the 4LINE-PAM6 scheme.	58
4.25	Schematic of the transconductance amplifier.	58
4.26	Detail of circuit implementation for receiver blocks (L1-L4 in Fig. 4.20). . . .	59
5.1	Set partitioning for a 4-point constellation	62
5.2	A possible coding scheme for inter-chip applications	62
5.3	a.A trellis for the scheme of Fig. 5.2 b.A minimum-distance error event c. A set of minimum distance events. There are infinite number of minimum- distance events	63
5.4	Modified trellis to force the state to return to the zero state every fourth symbol	64
5.5	a.Trellis for one time-step, b.The branch metrics corresponding to each branch, c.Simplified branch metrics, d.General form of the trellis for one time-step .	65
5.6	a. Block diagram of transmitter and the receiver using Viterbi decoder in space, b. Digital implementation of the Viterbi detector	65
5.7	Performance comparison of different schemes.	66
5.8	Case I: $Z_L = Z_S = 55\Omega$, $d = 0.3$ m.	68
5.9	Case II: $Z_S = 55\Omega$ $Z_L = \infty$, $d = 0.2$ m.	69
5.10	Using a tree instead of trellis for decoding the proposed scheme	71
5.11	Block diagram of the receiver	71
5.12	The current maximum selector circuitry	71
5.13	Using a tree instead of trellis for decoding the proposed scheme with rate=5/6	72
5.14	Performance comparison for 4-PAM, 3LINE-PAM4 and the 4LINE-PAM4 scheme	73
5.15	Block diagram of the transceiver.	74
5.16	a. Block diagram of the receiver, b. Detail of the state-metric calculator unit.	74
5.17	Detail of the 12X5 digital decoder.	75

5.18	Branch metric calculator schematic.	76
5.19	Detail of the transconductance amplifier.	76
5.20	Simplified schematic of a comparator.	77
5.21	Chip micrograph.	78
5.22	Layout of the active area of the chip.	78
5.23	PCB test fixture used for obtaining the experimental results.	79
5.24	Virtual instrument developed in LabVIEW.	80
5.25	Measured BER versus SNR for two methods.	81
5.26	Test setup for high-speed measurement.	82
5.27	Output of the ParBERT at 340 MS/s.	82
5.28	Output of the ParBERT at 1 GS/s.	83
6.1	Driver architectures: (a) Unipolar, (b) Bipolar	85
6.2	A common 4-PAM driver architecture	85
6.3	Basic architecture of the power-efficient 4-PAM Driver	86
6.4	Detail of each driver basic unit	87
6.5	The output of a 2-level bipolar driver: (a) With pre-switching, (b) Without pre-switching	88
6.6	Termination structure and its simulation results	89
6.7	Transmitter block diagram	91
6.8	Encoder circuitry	91
6.9	Multiplexer architecture	92
6.10	The circuitry for generating high-frequency clock	92
6.11	On-chip termination circuit	92
6.12	Printed circuit board test fixture used for experimental measurement	93
6.13	Transmitter chip micrograph	94
6.14	Eye diagram at 7Gb/s over 0.8m cable and 30mm printed circuit board channel	95
6.15	Eye diagram at 8 Gb/s and 10Gb/s over 0.8-m cable and 30-mm printed circuit board channel	96
6.16	Eye-diagram with and without power-supply noise	96

A.1	Trellis diagram for a binary dicode system showing the correct path (bold line) and an adversary (dashed line) of length $l = 3$.	109
B.1	Basic architecture of the power-efficient 2-bit DAC	113
B.2	Detail of each DAC basic-unit	114
B.3	The general block diagram of the 6-bit power-efficient DAC	115
B.4	Binary segment encoders	116
B.5	Unary segment encoders	116
B.6	The simulated DNL and INL profiles versus DAC input code	117
B.7	DAC output and its spectrum for a 5 MHz sine wave input	117

List of Tables

3.1	Signal values for the peak detection system in Fig. 3.4 with random binary data of width $N = 6$ and no noise.	22
3.2	Comparison of various transmitter and receiver architectures for a 32-bit binary bus.	31
4.1	Bit-rate comparison for different constellations with $MSED=4$ for 5-PAM. . .	38
4.2	Bit-rate comparison for different constellations with $MSED = 4$ for 6-PAM.	39
4.3	Performance comparison for different schemes.	39
4.4	Mapping design for 4×3 decoder.	55
6.1	Test result summary	95
B.1	Result summary	118

Chapter 1

Introduction

1.1 Motivation and Overview

Advances in IC fabrication technology, coupled with aggressive circuit design, have led to an exponential growth in speed and integration levels. To improve the overall system performance, the communication speed between systems and ICs must increase accordingly. Currently, communication bus links in various applications approach Gb/s data rates. Such links are often an important part of multi-processor interconnection [1], processor-to-memory interfaces [2], and SONET/Fibre channels [3], high-speed network switching, and local area networks [4]. It is also likely that many high-speed digital signals will be transmitted between analog and digital chips. Traditionally, system designers have addressed the need for high-speed chip-to-chip links by increasing the number of high-speed signals, which leads to an increase in the cost and complexity of the system. Therefore, the per-pin interconnection bandwidth should be increased.

Improving the performance of both parallel and serial interconnects has been an important research area over the last decade [5–7]. Although each type of interconnect has some advantages and disadvantages, the general trend has been toward serial links. However at the same time, significant amount of research has been performed to improve the performance of popular, general purpose parallel buses [8]. One of the main challenges in this area is to reduce the power and increase the bandwidth of the peripheral component interconnect (PCI) bus. The PCI bus, which has been widely used in the last decade, cannot be

easily scaled up in frequency or down in voltage. Its data transfer is also skew limited [9]. Almost all approaches to push these limits to create a higher bandwidth, general purpose bus result in a large cost increase for little performance gain [9]. Recent advances in high-speed, low-pin-count, point-to-point technologies offer a new solution for major bandwidth improvements, which is called PCI Express.

The fundamental PCI Express link consists of two, low-voltage, differentially driven pairs of signals: a transmit pair and a receive pair. The bandwidth of a PCI Express link may be linearly scaled by adding signal pairs to form multiple lanes [10]. A data clock is embedded using the 8b/10b encoding scheme to achieve high data rates [9]. Although this system is capable of providing higher bandwidth compared to the conventional PCI bus, it suffers from the need for a time recovery block for each channel, which in turn increases the power and area. Therefore, increasing the performance and bandwidth of the conventional parallel bus interconnects is still an ongoing research. Another high-performance low-pin-count packet switched system level interconnect architecture is RapidIO that could be another application for this research [11].

In general, the main focus of high-speed chip-to-chip communication research is to increase the per-pin speed of high-speed links [6]. This dissertation not only uses low-voltage differential signaling to increase the performance of the conventional parallel bus interfaces [12], but also explores various communication techniques and circuit architectures to reduce the cost and power-consumption of high-speed links. These techniques can be partitioned to three different areas: cost-efficient signaling schemes, power-efficient signaling schemes, and power-efficient circuit architectures.

1.1.1 Cost-Efficient Signaling Schemes

Recently, the noise margin on digital chip-to-chip interconnects has been decreasing for two main reasons. One reason is that supply voltages in digital complementary metal oxide semiconductor (CMOS) processes are decreasing thereby reducing the voltage available for driving I/Os. A second reason is that small signal swings are being used to reduce dynamic power dissipation on high-speed busses. It has long been known that fully-differential signals effectively reject common-mode noise and even-order distortion terms. Since common-mode

noise is prevalent on matched printed circuit board (PCB) traces, differential signaling is effective for both voltage [13], [14] and current mode [15] digital chip-to-chip interfaces. Fully differential signals are now used in the Scalable Coherent Interface [16], [12] and RamLink [17] standards. Unfortunately, a practical problem with their implementation is that two signal paths are required for each signal. For example, using fully-differential signals for a 64-bit data bus would require 128 pins on each IC package and 128 PCB traces routed between ICs. These additional costs are often prohibitive. Therefore, one important approach is to reduce the required number of pins of an interconnect. A signaling scheme that has most of the advantages of fully differential signaling scheme with reduced number of signal paths is proposed in Chapter 3 to help alleviate this problem.

1.1.2 Power-Efficient Signaling Schemes

Multi-level signaling, such as 4-level pulse amplitude modulation (4-PAM), can be used to reduce the required number of signal paths in a link or to increase the data rate of a link [18]. However, circuit and system improvements are needed to compensate the impact of this signaling scheme on bit error rate (BER). In other words, to obtain the same BER in multi-level schemes, the transmitter power should be increased. In addition, due to the large number of digital signals in the interconnect, the power of a typical chip-to-chip link is often a significant part of the total power of the chip [19].

Therefore, reducing the power consumed by interconnect circuitry is extremely important. Previous research has focused on using coding schemes for minimizing transitions to reduce the power dissipation of a digital bus [20, 21]. However, these coding schemes are mainly effective in high-capacitance busses. Channel coding can be used to reduce the power consumption of a high-speed inter-chip link by introducing some redundancy at the transmitter [22], [23], [24]. There is still a significant gap between the Shannon limit, the theoretical limit for channel capacity, and the data rates of the current state-of-the-art designs [25].

To find a low-power scheme, channel coding can be employed as an attempt to approach the Shannon limit [26]. Finding codes that can approach Shannon limit is not a complicated task. Indeed, randomly generated codes with a large block size can be used to approach this limit. The problem lies in the fact that while encoding is always a rather simple task, the

decoding complexity increases exponentially with the block size, and thus quickly becomes unmanageable [27]. Therefore, instead of making the codes more and more complex, the search should focus on finding low-complexity codes with good coding gain. In chip-to-chip communication applications, where high-speed implementation is the main concern, this becomes even more important.

On the other hand, to maintain high system performance, not only high-speed circuits but also low-loss matched transmission lines for interconnect are necessary to ensure good propagation properties such as minimum crosstalk, delay, reflection, and dispersion. Achieving a highly dense system by bringing the chips closer together is only a partial solution since denser systems require denser interconnects, which in turn cause more crosstalk [28]. Indeed, crosstalk is the dominant noise in most microstrip interconnects. This dissertation proposes several coding schemes in Chapters 4 and 5 that are suitable for high-speed chip-to-chip applications.

1.1.3 Power-Efficient Circuit Architectures

Employing circuit techniques for designing the building blocks of a high-speed link is another efficient method to reduce the power and cost of high-speed links. The potential benefits of 4-PAM signaling for increasing data rates in physical short-bus systems have been shown in [29–32]. Since there are several drivers in a parallel bus signaling system, the power dissipation of each driver is extremely important. Therefore, power-efficient drivers are desirable. The reported high-speed multi-level drivers have used power-inefficient unipolar architectures [29–33]. Chapter 6 proposes a power-efficient architecture for high-speed multi-level PAM transmitters.

1.2 Organization

This dissertation is partitioned into seven chapters. Chapter 2 provides an overview on different components of a high-speed chip-to-chip interconnect along with some useful background information. This chapter also describes the limitations and design concerns of high-speed links.

As mentioned before, reducing the cost of high-speed interconnect is one of our goals in this research. Chapter 3 describes a general technique for obtaining many of the advantages of fully-differential signals while using a reduced number of signal paths. Specifically, N differential signals are communicated over as few as $N + 1$ signal paths.

Using coding schemes to improve the performance of high-speed interconnects is another important part of this research. Chapter 4 proposes different coding schemes suitable for high-speed chip-to-chip applications. Interestingly, the proposed signaling schemes in this chapter are also less sensitive to crosstalk, intersymbol interference and reflections. In addition, low-complexity architectures for high-speed implementation of the proposed schemes are proposed.

The proposed method in Chapter 4 suffers from small coding gains in peak-power-limited applications. As an alternative, Chapter 5 introduces a coding scheme that employs a convolutional encoder in space. A low-complexity analog implementation of this method is proposed, which makes high-speed implementation of this method feasible. The receiver for this signaling scheme has been implemented and fabricated to verify the functionality and performance of this signaling scheme.

Using circuit techniques to reduce the power consumption of high-speed links is also an ongoing research topic. Chapter 6 addresses this issue by proposing a power-efficient 4-PAM driver. The proposed driver in this chapter exploits a power-efficient bipolar architecture. A data-look-ahead technique is used for high-speed implementation of this 4-PAM driver. A 4-PAM transmitter based on this architecture is implemented and fabricated in a $0.18\mu\text{m}$ standard CMOS technology. This transmitter shows significant power reduction compared to the conventional 4-PAM implementations.

Finally, Chapter 7 summarizes the contributions of this research and provides some guidance for the future work in this research area.

Chapter 2

Chip-to-Chip Interconnect: Basic Architecture and Non-Ideal Issues

Since chip-to-chip communication is the main application for this research, a brief introduction to inter-chip interconnects is necessary. Fig. 2.1 shows the basic architecture of a signaling system. As shown in this figure, a general chip-to-chip interconnect has three components: transmitter, channel, and receiver. The transmitter converts digital information to a signal on the transmission medium (communication channel). This channel is commonly a board trace, coaxial cable, or twisted-pair wire. The receiver on the other end of the channel restores the signal, by sampling, filtering, and quantizing it to the original digital information.

This chapter provides not only a brief overview for each component, but also addresses the non-ideal interconnect issues. A typical interconnect channel and its limitations are explained in Section 2.1. Transmitter and receiver blocks are addressed in Sections 2.2 and 2.3, respectively. Clock generation and timing recovery are tightly coupled to signal transmission and reception. The timing recovery, often embedded in the receiving side, adjusts the phase of the clock that samples the received signal [4]. Section 2.4 briefly explains the important timing considerations for high-speed interconnects.

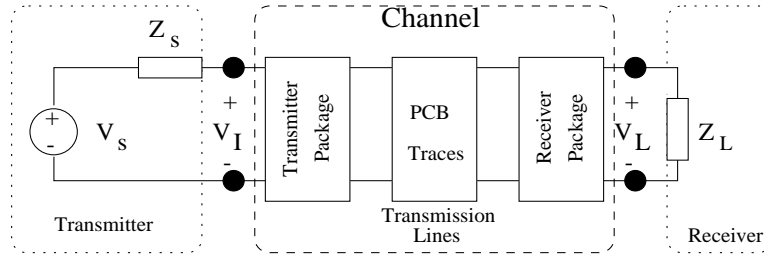


Figure 2.1: Basic architecture of a chip-to-chip interconnect.

2.1 Channel

As shown in Fig. 2.1, a typical interconnect channel comprises three components: a transmitter package, transmission lines (medium), and a receiver package.

2.1.1 Packages

Most integrated circuits are bonded to small ceramic or plastic packages. Although it is possible to attach chips directly to boards, placing the chips in packages makes independent testing of package parts possible, simplifies reworking of boards, and eases the requirements on board line pitch by spreading out the pins. The most common technique for electrically attaching chips to packages is wire-bonding. Wire-bonding is inexpensive and has the advantage of allowing unequal thermal expansion of the chip and the package. However, wire bonds have significant inductance, about 1 nH/mm, that is not desirable in high frequency applications.

An array of solder balls can also be used to attach a chip to a package. This style of packaging is often called “C4” (i.e., controlled collapse chip connection) or “flip-chip”. This technique has the advantages of very low inductance and the ability to form contacts with pads located over the entire area of the chip. Area bonding is advantageous for power distribution and placing large number of pads in a small area. However, because area bonding requires expensive multi-layer packages and special processing to apply the solder, currently its use is restricted for expensive high-performance chips [34].

Packaging has been evolved significantly during the last three decades, starting from dual-in line package (DIP) and wire-bond in the 1970’s, quad flat pack (QFP) in the 1980’s,

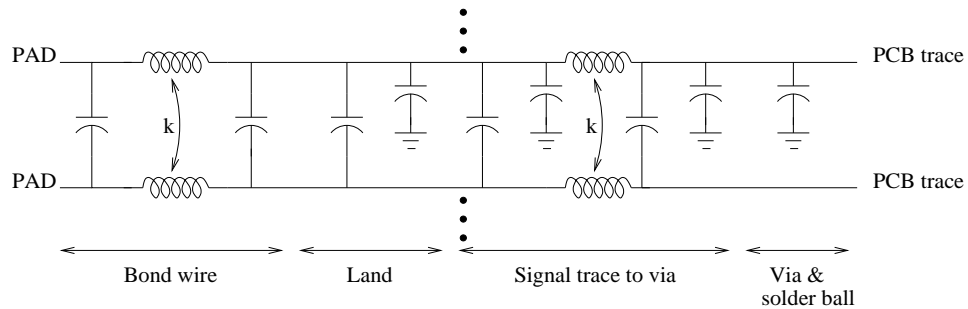


Figure 2.2: A typical package model.

and ball-grid array (BGA) in the 1990's. However, due to the increasing need for suitable packages for high frequency applications, more sophisticated packages are expected in the near future [35].

Package Electrical Model

The important parts of a typical package model are: the bond wire between chip and package, the package land to which the bond wire is attached, the signal trace between land and via, and the via and solder ball [34]. In principle, an electrical model of a package would have to take into account all of the currents and charges on a complex 3-dimensional arrangement of conductors and dielectrics. In practice however, a lumped-circuit-element equivalent model that will serve well for most high-performance signaling designs is often used.

Bond wires are usually modelled by an inductor. A value of 1 nH per millimeter is a good rule of thumb for the inductance of bond wires [36]. The signals on neighboring bond wires interact via coupling capacitance and mutual inductance. The effect of land and package trace can be characterized using a 2-dimensional field solver. For simplification, however, the land can be modelled by a lump capacitance. The package traces have capacitances to the neighboring planes, series inductance, and mutual inductance to the neighboring traces. The via and solder ball have very low inductance compared with the signal traces on the package and board and can be accurately modelled by a single lumped capacitance. Typical values for the capacitances and inductances in the model can be obtained from [34], [36]. A typical package model is shown in Fig. 2.2.

Several simulations have been performed to measure the performance of two different

packages for high-speed interconnect based on the models provided by Canadian Microelectronic Corporation (CMC). HSPICE simulation results show roughly a bandwidth of 4.5 GHz for an 80-pin ceramic flat package (CFP) and a QFP package. Therefore, although these packages can be used in high-speed application, their effect on signal integrity of the link have to be considered in the design of interconnects with more than 10Gb/s data rate.

2.1.2 Transmission Lines

Due to the high data rate of the current interconnect links, it is necessary to treat the PCB and multi-chip module as transmission line. Transmission line structures seen on a typical PCB consist of conductive traces buried in or attached to a dielectric with one or more reference planes. The metal in a typical PCB is usually copper and the dielectric is FR4, which is a type of fiberglass [37].

The two most common types of transmission lines used in high-speed links are microstrips and striplines. A microstrip is usually routed on the outer layer of the PCB and has only one reference plane whereas the stripline is routed on an inside layer and has two reference planes.

Transmission lines are often modelled by the RLGC model of their differential section, as shown in Fig. 2.3 [37]. Components R and G in this figure model the finite conductivity of the transmission line and conductivity of the dielectric. In addition, since R and G are frequency dependent, they can accurately model the skin effect and dielectric loss of transmission lines. Characteristic impedance and propagation velocity are the most important electrical characteristics of a transmission line. The characteristic impedance (Z_0) is defined by the ratio of the voltage and current waves at any point of the line and is equal to:

$$Z_0 = \frac{V_0^+}{I_0^+} = \frac{V_0^-}{I_0^-} = \sqrt{\frac{R + j\omega L}{G + j\omega C}} \quad (2.1)$$

Propagation velocity is the propagation speed of the wave in the medium and is equal to the speed of light in vacuum over the square root of dielectric constant. Transmission line theory along with a typical package model will be used in Section 4.2 to provide a general channel model for high-speed interconnect applications.

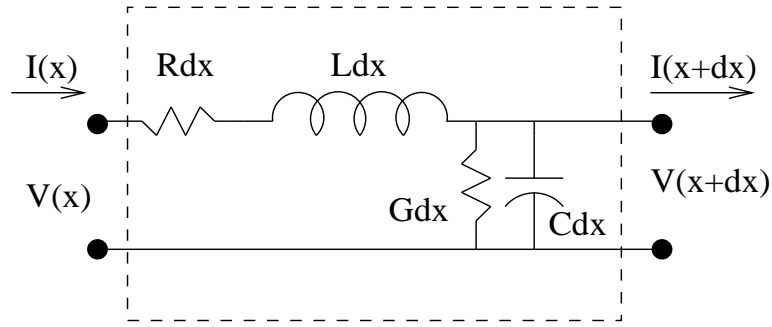


Figure 2.3: Equivalent circuit model of a differential section of a transmission line (RLCG).

2.1.3 Noise in Digital Systems

Most noise in digital systems is created by the system itself [34]. Typical digital systems operate at such high energy levels that thermal noise, shot noise, and electromagnetic interference from external sources are not significant. A major part of the system-created noise, which scales with the signal, is induced by the transmission of signals. This type of noise cannot be overpowered since increasing signal also increases this component of noise. Crosstalk and intersymbol interference due to the channel attenuation and residual reflection belong to this group. Power-supply noise, timing noise, and transmitter/receiver offsets are important noise sources that are not proportional to the signal level. Here, some of the important noise sources in inter-chip applications are briefly described.

Transmission Line Reflections

The characteristics of the driving circuitry and the transmission line greatly affect the integrity of a signal being transmitted from one chip to another. The relation between magnitude of V_i in Fig. 2.4 and the source voltage, V_s , is given by

$$V_i = V_s \frac{Z_0}{Z_0 + Z_s} , \quad (2.2)$$

where Z_s , Z_0 are the source termination and characteristic impedance of the line, respectively. As shown in Fig. 2.4, the incident wave gets reflected at the far end of the line if the load is not matched with the characteristic impedance of the line. Although interconnect links are often designed to have matched termination, it is very difficult to have termination

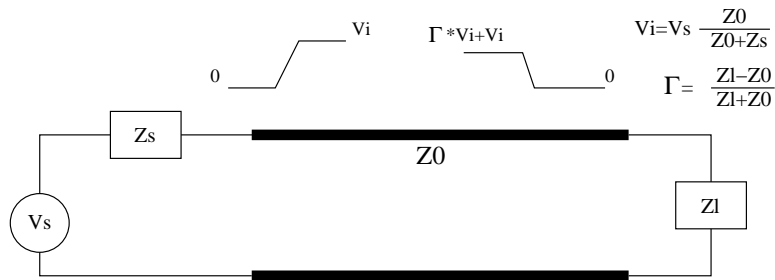


Figure 2.4: Reflection of incidental signal from an unmatched load.

resistors with less than 5% mismatch due to the process variation. Therefore, circuit and system techniques should be used to reduce these residual reflections.

Intersymbol Interference (ISI)

The channel attenuation due to the skin effect and dielectric loss causes intersymbol interference. Although this ISI is not significant for low-speed links over a short distance, it is one of the most important limitations of the backplane applications at high speed. Channel equalization usually is employed to compensate the channel attenuation and alleviate this effect [18]. Although equalization can be performed in the receiver and transmitter (or even both), transmit equalization is more common. The conventional method is pre-emphasis/de-emphasis in the transmitter because of its low-complexity architecture for a high-speed implementation [7], [32].

Since our application target is short distance intra-board chip-to-chip communication, we have not exploited the equalization techniques in this dissertation. Nevertheless, most of the proposed signaling schemes and circuit techniques in this dissertation are general and can be employed along with equalization.

Crosstalk

Crosstalk, which is the coupling of energy from one line to another, occurs whenever the electromagnetic fields from different structures interact. Coupling capacitances and mutual inductance between adjacent lines in the package and the PCB traces are the main sources of crosstalk in high-speed links. Simulation results for a parallel bus interface show that crosstalk can inject 140 mV (p-p) errors into the victim line for an 800 mV aggressor step,

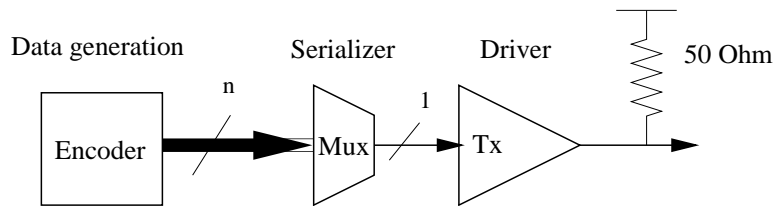


Figure 2.5: Transmitter architecture for parallelism implementation.

which translates to a crosstalk as large as 20% of the aggressor step amplitude [30].

The dominant noise in most microstrip interconnects is crosstalk. Striplines, on the other hand, constrain the wave to travel entirely in the dielectric and thus the capacitive and inductive coupling cancel, resulting in no far-end crosstalk [34]. However, even in the case of stripline, significant crosstalk can be generated by package. Indeed, experimental results in [7] have shown that the crosstalk due to the package can be as large as four times the crosstalk due to the PCB traces.

2.2 Transmitter

A transmitter circuit encodes a symbol into a current or voltage for transmission over the medium. In general, different pulse shapes, such as Nyquist pulses [24], can be exploited for transmitting information bits over the channel. However, due to the need for a low-complexity transceiver architecture, a simple rectangle pulse shape is often used in almost all of the high-speed interconnect applications.

Conventional high-speed transmitters often multiplex the low-speed data into a single serial bitstream. Fig 2.5 shows the block diagram of a transmitter implementing parallelism by multiplexing on-chip parallel data. The primary bandwidth limitation here stems from either the multiplexer or the clocks [4]. Rather than relying on on-chip multiplexer's bandwidth, we can use the high bandwidth available at the chip output pins since the chip output often drives a low ($25 - 50 \Omega$) impedance. However, if the multiplexer's bandwidth is not a limitation in the design, using a multiplexer prior to the chip output pin can reduce the overshoot/undershoot and glitch at the output [29]. Chapter 6 explains different transmitter building blocks in more detail.

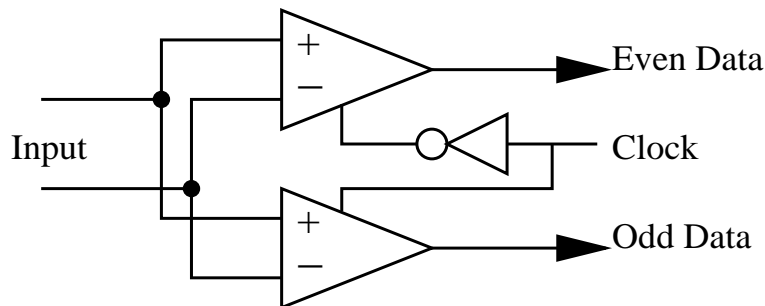


Figure 2.6: Receiver architecture for parallelism implementation.

2.3 Receiver

A receiver detects an electrical quantity, current or voltage, to recover a symbol from a transmission medium. A receiver should have good resolution and low offset in both the voltage and time dimensions. These voltage and time resolutions determine the size of the required eye-opening for reliable detection of the signal.

In most applications, a receiver with a minimum aperture time is preferable because sensing the input voltage in a narrow time window gives the largest possible timing and voltage margins. It is well known that a matched filter is the ideal receiver architecture for any interconnect in general [24]. Assuming a flat frequency response for the interconnect channel and a rectangle pulse shape at the transmitter, the matched filter would be an integrating receiver. In practical high-speed interconnect channels, however, the channel is often similar to a low-pass filter and the integrating receiver is not the ideal receiver. Nevertheless, integrating receivers have some advantages and disadvantages. Integrating receivers filter out high-frequency noise at the expense of increased sensitivity to jitter and low-frequency noise [34]. These problem can be reduced by shortening the integration window or aperture, in effect averaging the signal over the aperture time of the receiver. However, this is what a normal receiver already does [34]. Therefore, ideal solution is to build a receiver with an impulse response matched to the received waveform, data eye. Unfortunately, the complexity of such a receiver precludes its use in most high-speed interconnect applications.

If a conventional amplifier restores the signal, its bandwidth is the limiting factor for the achievable data rate. Again, the use of parallelism relaxes the bandwidth requirement of receiver components. Fig. 2.6 shows a simple parallel receiver which uses two receiver blocks

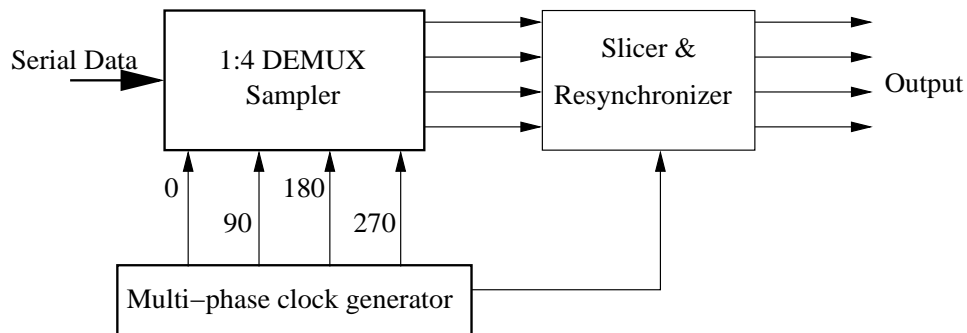


Figure 2.7: Receiver architecture for a system with 4:1 demultiplexer.

to detect the received signal on both rising edge and falling edge of the clock. In this way, both receiver blocks have one full cycle to sample and latch the signal. This method can be extended to receivers with higher degrees of parallelism. However, increasing the degree of parallelism increases the input capacitance of the receiver. Although the comparator cycle-time is no longer an issue, the minimum data bit-time which can be sampled by the receiver is an inherent limitation. In addition, for the parallel structure, the timing recovery becomes more challenging since the clock edge in each channel must be aligned to the middle of the data bit in that channel. Practical parallelism degrees are often between four and eight [31, 38–40].

Fig. 2.7 shows the basic block diagram of a receiver which employs a 4:1 demultiplexer. As shown in this figure, a resynchronizer block is necessary to synchronize the output of the parallel branches in the receiver. Moreover, a multi-phase clock generator is necessary to generate the required four different clock signals, which are 90 degrees apart, for the demultiplexer block. Timing recovery and clock generation for high-speed interconnect is extremely important and is addressed in the next section.

2.4 Timing in a Chip-to-Chip Link

As shown in Fig. 2.7, a multi-phase clock generator block is an essential part of high-speed receivers. Phase spacing errors and phase noise (jitter) in the generated clocks all affect system performance. In addition, sampling clock phases for the receiver must be aligned with respect to the input data stream to maximize timing margins.

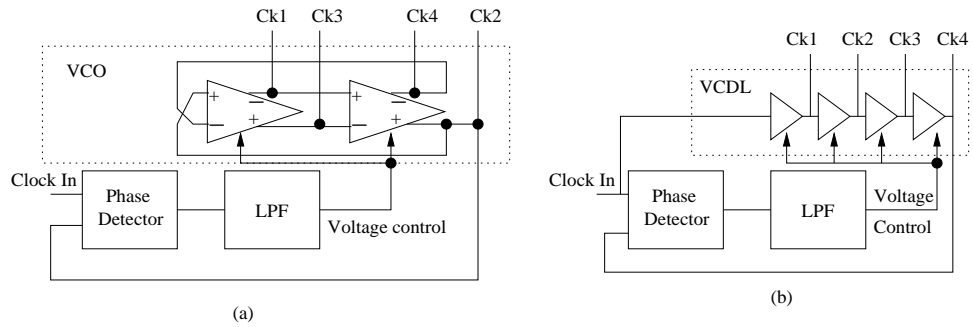


Figure 2.8: The basic architecture of (a) PLL and (b) DLL.

2.4.1 Timing Recovery and Clock Generation

Usually, a phase-locked loop (PLL) is used for timing recovery of high-speed links. PLL clock sources also provide internal clock phases aligned with the input data. They are also useful for generating a high-speed clock from a low-frequency reference clock in the transmitter. If a reference clock is available in the receiver, a delay-locked-loop (DLL) clock generator can also be used for aligning the clock with data.

Fig. 2.8 shows the basic architecture for a PLL and a DLL. A phase detector (PD) and a low-pass filter (LPF) are used to generate the control voltage for the voltage-controlled oscillator (VCO) in a PLL and the voltage-controlled delay line (VCDL) in a DLL. DLLs are usually less susceptible to noise than PLLs because the corrupted zero crossings of a waveform disappear at the end of a delay line whereas they are recirculated in an oscillator. Moreover, in the VCDL of Fig. 2.8b, a change in the control voltage immediately changes the delay. Thus, the feedback system of a DLL has the same order as the LPF and its stability and settling issues are more relaxed than those of a PLL. However, the principal drawback of DLLs is that they cannot generate a variable output frequency [41]. Since timing errors shift the transition edges of the received data signals relative to the transition of reference clock, the PLL/DLL should be designed to reduce these timing errors and to increase the timing margins of the link. The timing errors can be decomposed into a dc phase offset (skew) and the dynamic phase noise (jitter).

2.4.2 Skew

Any static phase offset in clock recovery shifts the sampling point away from the optimum center and further narrows the noise margins. Skew appears in a system due to path-length and rise-time mismatches. Single-ended systems suffer more from skew since it is much easier to control the path length of only a pair of lines than that of an entire bus.

Skew can deteriorate the performance of a link significantly. This effect can be as small as generating data-dependent common-mode variations to as large as total loss of data and false detection [7]. At high data rates in parallel links, such as 10 Gb/s, skew between the lines becomes a serious issue. This imposes a challenge in synchronizing the incoming data with the sampling clock. Fortunately, skew is a static error and can be compensated. Per-pin skew compensation has been incorporated in many designs; see for example [13]. On start-up, a calibration engine estimates each bit's skew relative to a timing reference. This information is then used to control the delay of an adjustable delay line. A per-pin skew compensation, using phase interpolation to enable full-range compensation, is introduced in [39].

2.4.3 Jitter

A strictly periodic waveform contains zero crossings that are evenly spaced in time. Now consider the nearly periodic signal whose period experiences small changes, deviating the zero crossing from their ideal points. We say the second waveform suffers from jitter. VCOs usually suffer from jitter accumulation. This is why jitter performance of DLLs are usually better than jitter performance of PLLs [41]. A multiplying delay-locked loop (MDLL) for high-speed on-chip clock generation, which overcomes the drawbacks of PLLs such as jitter accumulation, high sensitivity to supply and substrate noise, is recently proposed in [42]. In general, jitter is an important non-ideal phenomenon that should be taken into account for designing high-speed interconnect circuits.

Chapter 3

Differential Signaling with a Reduced Number of Signal Paths

Differential signaling is often used for digital chip-to-chip interconnects because it provides common-mode noise rejection. Unfortunately, differential signals generally require $2N$ signal paths to communicate N signals. In this Chapter, a method for differential signaling is described which requires as few as $N + 1$ signal paths for N signals. This technique achieves many of the advantages of fully differential signals while using a reduced number of signal paths ¹.

In Section 3.1, the basic idea is described. The technique is similar to partial response signaling except that the encoded sequence is transmitted in parallel over a set of wires rather than sequentially in time over a serial connection. Possible implementations are then discussed. As in partial response systems, the simplest approach is peak detection, which requires only $N + 1$ signal paths, and is discussed in Section 3.2. Maximum likelihood sequence detection (MLSD) is another popular technique for partial response systems. It uses $N + 2$ signal paths and is discussed in Section 3.3. Modifications to the Viterbi algorithm are described in Section 3.4 which bring the performance of MLSD closer to that of fully differential signaling, while using only $N + 2$ signal paths. Theoretical and simulated bit error rate results are presented for each approach on a digital data bus. As in partial response systems, the approaches are general and multilevel signaling is possible. However,

¹The original idea was proposed by Tony Chan Carusone and David A. Johns.

in this work results are only presented for a binary data bus. A system which combines this approach with constant-weight digital encoding is proposed in Section 3.5. Finally, Section 3.6 discusses the analog implementation of the modified Viterbi algorithm proposed in this chapter.

3.1 Incremental Signals

In general, the problem is to find an efficient signaling scheme for communicating N signals, x_k , $1 \leq k \leq N$, over M signal paths, y_l , $1 \leq l \leq M$. In a simple single-ended scheme $M = N$ and the receiver operates by comparing the signal on each path to reference threshold levels. Of course, the receiver is susceptible to common-mode noise on the bus. To combat this problem, practical single-ended systems often include reference signals transmitted along the bus to provide some common-mode noise rejection. This approach has been referred to as “pseudo-differential” signaling. For instance, a system with an extra reference line after every 4th active signal path is shown in Fig. 3.1.

Of course, this increases the pin count by 25%. Furthermore, there will always be some finite common-mode to differential conversion due to mismatches between signal paths along the bus. These mismatches can be minimized by comparing only *neighboring* signal paths.

Using fully-differential signals requires $M = 2N$. The signals appear as the difference between neighboring matched signal paths. As shown in Fig. 3.2, the received signal is given by

$$x_k = y'_{2k} - y'_{2k-1}. \quad (3.1)$$

The primes in Fig. 3.2 and (3.1) denote noisy signals at the receiver, $y'_k = y_k + n_k$. Note that (3.1) implies we have complete freedom to arbitrarily select one-half of the signals levels². Clearly, there is some redundancy inherent in transmitting N signals over $2N$ signal paths. This redundancy is eliminated by having the signals appear incrementally as the difference between adjacent signal paths:

²To minimize power consumption, differential signals are usually driven in a balanced fashion so that $y_{2k} = -y_{2k-1}$.

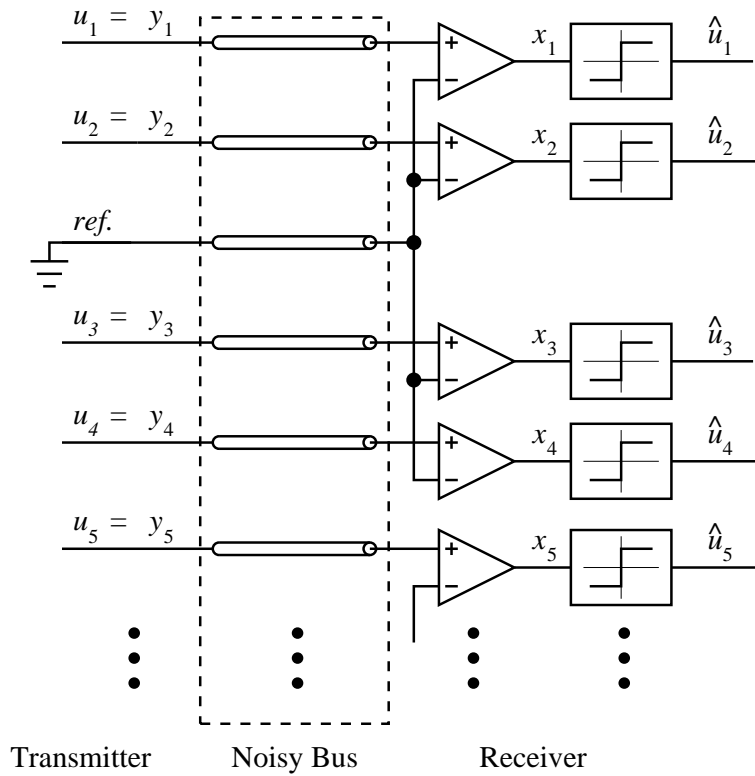


Figure 3.1: A practical single-ended signaling system or “pseudo-differential” signaling with additional reference lines added after every four signal paths.

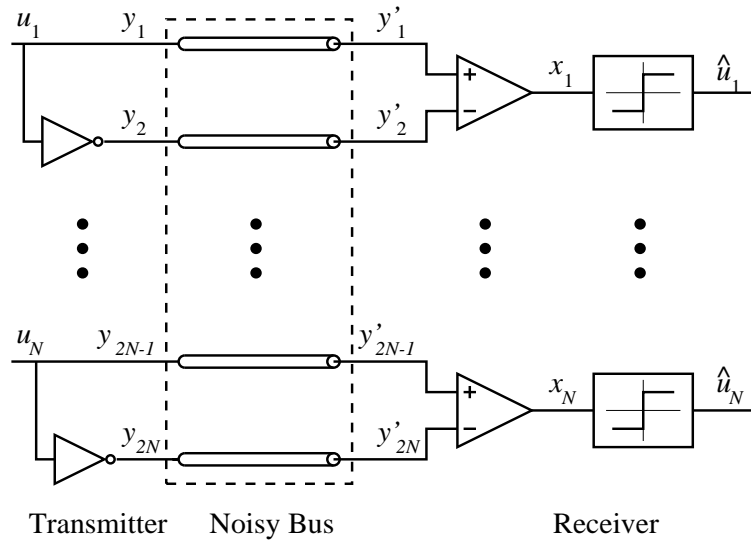


Figure 3.2: Block diagram of a fully-differential signaling system for binary data.

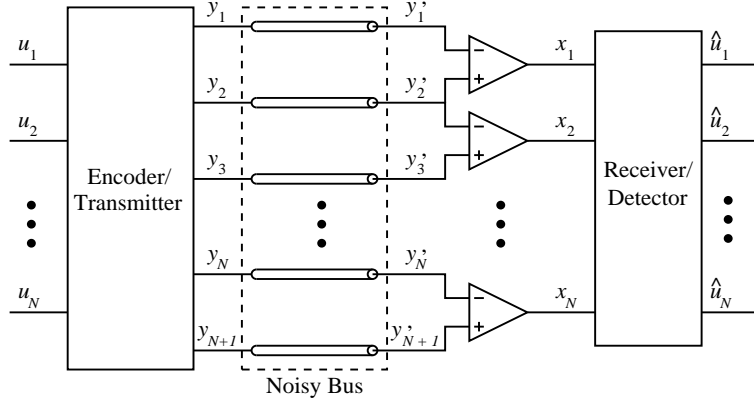


Figure 3.3: A general incremental signaling system.

$$x_k = y'_{k+1} - y'_k . \quad (3.2)$$

Using this scheme, hereafter called incremental signaling, the signals still appear differentially between two adjacent wires so all of the noise rejection advantages of fully-differential signals are obtained using only $M = N + 1$ signal paths. However, in (3.2) there is only freedom to fix one signal path value, namely y_1 .

A completely general incremental signaling system is shown in Fig. 3.3. Several possible encoder/transmitter and receiver/decoder combinations are possible, each offering a different compromise between complexity and performance. Interpreting the received signals as the difference between adjacent signal path values as in (3.2) is analogous to applying the decode $(1 - D)$ partial response operator to a time series:

$$x(k) = y(k) - y(k - 1) . \quad (3.3)$$

Therefore, popular approaches to encoding/transmitting and receiving/decoding partial response signals are also applicable here.

3.2 Peak Detection

To keep the receiver hardware as simple as possible, the information bits can be precoded prior to transmission as described in [43]. Specifically, the signal path values are encoded by

$$y_{k+1} = (y_k + u_k) \bmod L , \quad (3.4)$$

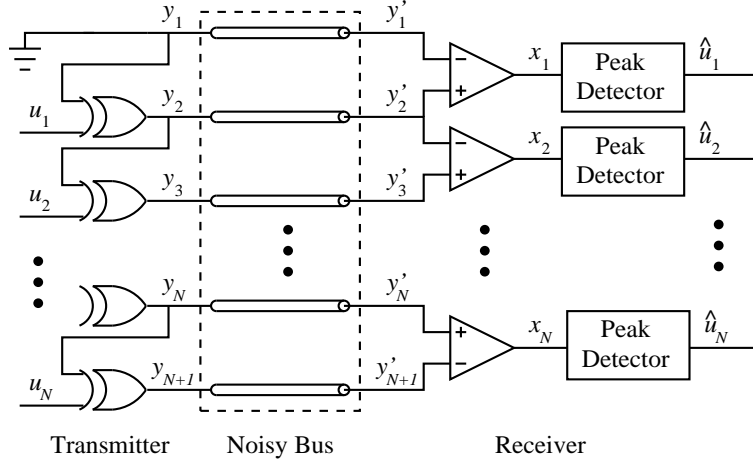


Figure 3.4: Block diagram of a possible incremental signaling system for binary data using peak detection.

where u_k is the (possibly multilevel) information symbol being encoded and L is the number of signal levels to be transmitted on the bus. The receiver must then interpret the received signals by

$$\hat{u}_k = x_k \bmod L . \quad (3.5)$$

The modular arithmetic in (3.4) and (3.5) has a particularly straightforward hardware implementation when the u_k are binary signals. The modular addition in (3.4) can be performed by exclusive-OR gates:

$$y_{k+1} = y_k \oplus u_k . \quad (3.6)$$

The modulo- L receiver can be just two differential comparators operating as a peak detector. A system block diagram of this approach is shown in Fig. 3.4. Note that the precoder includes a cascade of N exclusive-OR gates. However, it is possible to shorten the logic's critical path if it is limiting speed through the use of carry-look-ahead or pipelining techniques. Table 3.1 shows all of the signal values for a sample binary sequence of length $N = 6$ in the absence of noise.

The receiver in Fig. 3.4 sees the noisy signals, $y'_k = y_k + n_k$. A bit error will occur when independent noise on y'_k and y'_{k+1} causes the differential signal $x_k = y'_{k+1} - y'_k$ to cross one of the slicer thresholds. If the bus signals (y_1 to y_{N+1}) are members of a binary alphabet with spacing $2A$, σ^2 is the variance of independent Gaussian noise on each y_k , and $\eta = A/\sigma$,

Table 3.1: Signal values for the peak detection system in Fig. 3.4 with random binary data of width $N = 6$ and no noise.

k	1	2	3	4	5	6	7
Input Data: u_k	1	1	0	1	0	0	
Transmit Data: $y_k = y_{k-1} \oplus u_{k-1}$ ($y_1 = 0$)	0	1	0	0	1	1	1
Received Signal: $x_k = y_{k+1} - y_k$	1	-1	0	1	0	0	
Recovered Data: $\hat{u}_k = x_k \bmod 2 = x_k $	1	1	0	1	0	0	

then it can be shown that the probability of error is given by

$$P_{e_{\text{peak detection}}} = \frac{3}{2}Q\left(\frac{1}{\sqrt{2}}\eta\right), \quad (3.7)$$

where $Q(\cdot)$ is defined as

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\alpha^2/2} d\alpha. \quad (3.8)$$

For comparison, the probability of error of the fully-differential system shown in Fig. 3.2 is given by

$$P_{e_{\text{fully differential}}} = Q(\sqrt{2}\eta). \quad (3.9)$$

Of course, since there are approximately twice as many lines being driven in a fully differential system, the power consumption is doubled. In order to take this into account, a normalized signal-to-noise ratio is defined as follows:

$$SNR = \frac{(\text{number of lines being driven})}{(\text{number of information bits})} \times \eta^2. \quad (3.10)$$

Equations (3.7) and (3.9) can be rewritten in terms of SNR as

$$P_{e_{\text{peak detection}}} = \frac{3}{2}Q\left(\sqrt{\frac{SNR}{2}}\right), \quad (3.11)$$

$$P_{e_{\text{fully differential}}} = Q(\sqrt{SNR}). \quad (3.12)$$

Equations (3.11) and (3.12) are plotted along with simulation results in Fig. 3.5. The theoretical and simulation results indicate approximately a 3 dB decrease in the performance of a digital transmission system using a peak-detection-based incremental signaling

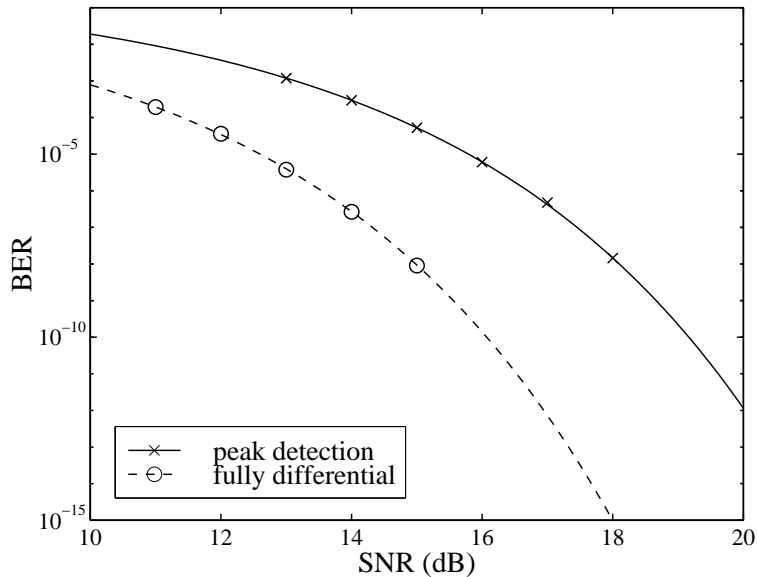


Figure 3.5: Bit error rates for the fully-differential system in Fig. 3.2 and the peak detection system in Fig. 3.4. Lines are theoretical results and markers indicate simulation results.

scheme compared with fully differential signaling. In the following sections, we shall see how it is possible to recover most of that 3 dB using somewhat more complicated receiver hardware. However, if speed is the main concern, peak detection is preferable because of the low-complexity architecture of its high-speed implementation. An implementation of peak-detection at 7.2Gb/s is presented in [7].

3.3 Maximum Likelihood Sequence Detection with Viterbi Algorithm

Received signals in the bus can be represented by a vector and an optimum detector such as ML detector or maximum a-posteriori probability detector (MAP detector) can be exploited to detect the transmitted information [24]. However, the complexity of these schemes precludes their high-speed implementation for this application. As in magnetic storage systems, which use partial response signals, it is possible to use a Viterbi detector for incremental-signaling receivers. An incremental signaling system utilizing MLSD at the receiver is shown in Fig. 3.6. Notice that no precoding logic is necessary. However, it is necessary to utilize

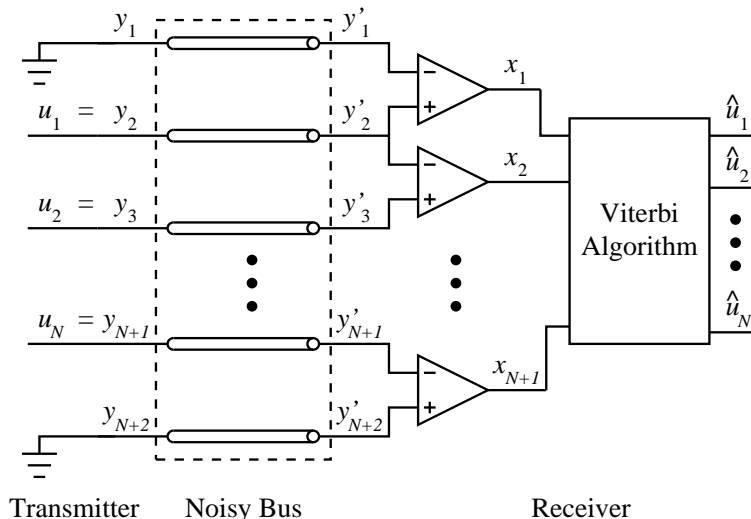


Figure 3.6: Block diagram of an incremental signaling system using the Viterbi algorithm for MLSD at the receiver.

$N + 2$ rather than $N + 1$ signal paths. The extra signal path appears at the end of the bus to terminate the Viterbi algorithm's trellis. Without this extra signal path, the last few bits on the bus would be decoded with approximately the same probability of error as a regular peak detection-based system.

It can be shown (Appendix A) that the probability of error using MLSD is bounded by

$$P_{e_{\text{MLSD}}} < Q\left(\frac{\sqrt{6 \cdot \text{SNR}}}{3}\right) + 3Q(\sqrt{\text{SNR}}). \quad (3.13)$$

This expression, along with simulation results, are plotted in Fig. 3.7. The theoretical expressions from equations (3.11) and (3.12) are also shown for comparison. At high SNRs, the bound in (3.13) provides an accurate estimate of the bit error rate. MLSD provides approximately a 1.25 dB performance improvement over peak detection (1.75 dB worse than a fully differential system). Of course, high-speed hardware implementation of the Viterbi algorithm is an area of ongoing research. A digital implementation of the algorithm would require an A/D converter operating on each signal path and considerable digital signal processing (DSP). Analog implementations such as those developed for magnetic storage applications [44] are more realistic.

In some partial response systems, the use of a MLSD receiver provides a full 3 dB improvement in BER versus SNR . However, to achieve this 3 dB improvement, the Viterbi

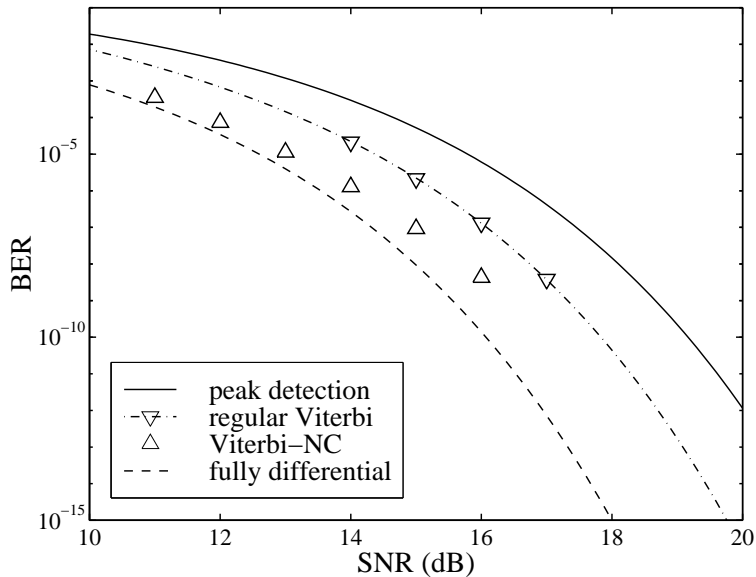


Figure 3.7: Bit Error Rate of the MLSD system in Fig. 3.6 with the regular Viterbi algorithm and the Viterbi-NC algorithm. Lines are theoretical results and markers are simulation results.

algorithm requires independent noise at its inputs [45]. In incremental systems such as Fig. 3.3, the receiver takes the difference $(1 - D)$ between signal path values which are already noisy. Therefore, any independent noise which appears on a given signal path, y'_k , will, according to (3.2), appear on both x_k and x_{k+1} . As a result, the noise on x_k and x_{k+1} is correlated. This causes suboptimal performance of the Viterbi algorithm and a 1.75 dB performance penalty.

3.4 The Viterbi Algorithm with Noise Cancellation

As described above, the Viterbi algorithm provides only a 1.25 dB performance improvement instead of 3 dB because consecutive values in the sequence, x_k , are subject to correlated noise introduced by the $(1 - D)$ operator. A similar problem has been encountered in magnetic storage systems where correlated noise is introduced by the magnetic media and by the receive equalizer. Several solutions have been proposed for magnetic storage systems [46–48]. Generally, the approach taken is to cancel the correlated noise on each sample using a linear noise-prediction filter. These techniques, sometimes called noise-predictive

maximum likelihood detection (NPML), are excellent when the noise fits an autoregressive model. However, they are not as well suited to systems where the noise includes a common-mode term. In this section, a modification to the Viterbi algorithm is described which cancels both the correlated noise (using noise prediction) and the common-mode noise on x_k resulting in bit error rates comparable to fully differential systems. First, noise prediction will be described in the absence of common-mode noise, similar to the approaches described in [46–48]. Then, a further modification is introduced to handle common-mode noise.

If there is no common-mode noise on the bus, the noisy received signals, y'_k , will be made up of the transmitted value, y_k , and independent noise, n_k :

$$y'_k = y_k + n_k . \quad (3.14)$$

Therefore, the differential signals at the receiver, x_k , will include the desired received signal ($y_{k+1} - y_k$) plus the noise terms n_k and n_{k+1} . The equation (3.2) can be rewritten as

$$\begin{aligned} x_k &= y'_{k+1} - y'_k \\ &= (y_{k+1} + n_{k+1}) - (y_k + n_k) \\ &= (y_{k+1} - y_k) + n_{k+1} - n_k . \end{aligned} \quad (3.15)$$

Before calculating each branch metric in the Viterbi algorithm, estimates of n_k (called \tilde{n}_k) are obtained iteratively by

$$\tilde{n}_k = (x_{k-1} - \hat{x}_{k-1}) + \tilde{n}_{k-1} . \quad (3.16)$$

In (3.16), \hat{x}_{k-1} refers to the expected value of x_{k-1} corresponding to the survivor path which terminates in the branch under consideration. So, a different estimate \tilde{n}_k must be calculated for each branch in the trellis. For the correct path, $\hat{x}_k = (y_{k+1} - y_k)$. Furthermore, assuming $\tilde{n}_{k-1} = n_{k-1}$, we have from (3.16) $\tilde{n}_k = n_k$. The branch metric is then computed using $(x_k + \tilde{n}_k)$ instead of x_k :

$$x_k + \tilde{n}_k = (y_{k+1} - y_k) + n_{k+1} - n_k + \tilde{n}_k . \quad (3.17)$$

The terms $(-n_k + \tilde{n}_k)$ in (3.17) cancel resulting in a 3 dB performance improvement.

The problem with this approach is how to initialize \tilde{n}_1 for the iterative computation in (3.16). In the absence of common-mode noise, the following definition works well:

$$\tilde{n}_1 = y_1 + A . \quad (3.18)$$

However, since (3.18) relies upon the single-ended signal y_1 , any common-mode noise will appear on all \tilde{n}_k 's and hinder the system's noise performance. To solve this problem, an estimate of the common-mode noise is calculated for each state in the trellis based upon the surviving path for that state. This estimate can be written as

$$\tilde{n}_{\text{CM}} = \frac{1}{k} \sum_{t=1}^k (y'_t - \hat{y}_t) , \quad (3.19)$$

$$= n_{\text{CM}} + \frac{1}{k} \sum_{t=1}^k n_t . \quad (3.20)$$

The term \hat{y}_t in (3.19) is the expected value of y'_t corresponding to the survivor path for the state under consideration.

So, the procedure for computing branch metrics is:

1. Obtain an estimate of the correlated noise term, \tilde{n}_k , using equation (3.16).
2. Obtain an estimate of the common-mode noise, \tilde{n}_{CM} , using equation (3.19).
3. Compute the branch metric as usual using the value $x_k + \tilde{n}_k - \tilde{n}_{\text{CM}}$ instead of x_k .

The combination of noise prediction and common-mode noise estimation will be called the Viterbi algorithm with noise cancellation (Viterbi-NC). The second term in (3.20), $\frac{1}{k} \sum_{t=1}^k n_t$, represents the error in the common-mode noise estimate, \tilde{n}_{CM} . This error term limits the noise performance of the Viterbi-NC algorithm to somewhat less than that of a fully differential system (Fig. 3.7). As k increases, $\frac{1}{k} \sum_{t=1}^k n_t \rightarrow 0$ and $\tilde{n}_{\text{CM}} \rightarrow n_{\text{CM}}$. Therefore, one would expect most of the errors to occur at the start of the bus when the common-mode noise estimate is still poor. Furthermore, the overall bit error rate should improve as the bus gets wider. As a verification, Fig. 3.8 shows simulation results for the Viterbi-NC algorithm at different bus widths.

Since \tilde{n}_{CM} is an accurate estimate of n_{CM} at the end of the bus, the following improvements are possible:

1. Apply the algorithm once from beginning to end, y'_1 to y'_N , and keep only half of the decoded sequence, $u_{(N/2)+1}$ to u_N . At the same time the algorithm can be applied from the end to the beginning, y'_N to y'_1 , and keep the first half of the decoded sequence, u_1 to $u_{N/2}$. This will be referred as the "parallel Viterbi-NC" algorithm.

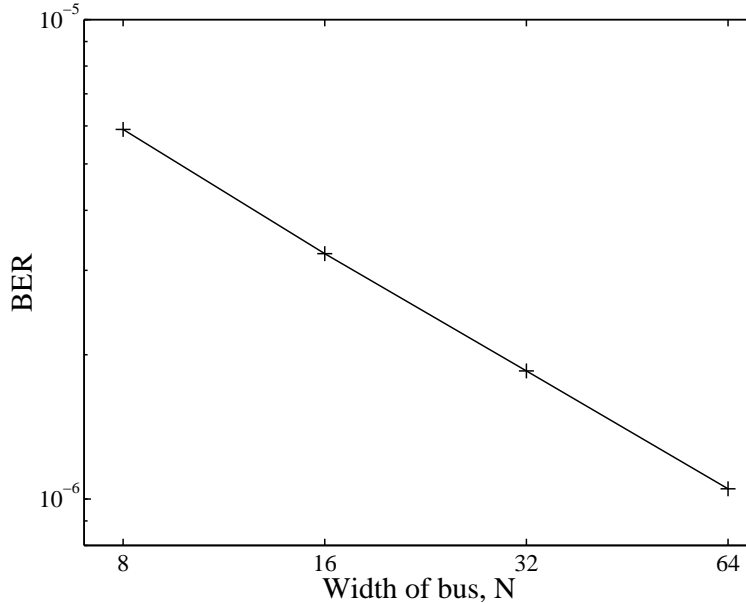


Figure 3.8: BER versus bus width, N , using the Viterbi-NC algorithm for MLSD at $SNR = 14$ dB.

2. Run the Viterbi-NC algorithm once. Then apply the algorithm again using the final (accurate) estimate \tilde{n}_{CM} for common-mode noise. This will be referred as the “serial Viterbi-NC” algorithm.

Simulation results for the parallel and serial Viterbi-NC algorithms are presented in Fig. 3.9 for $N = 32$. Both algorithms show roughly the same performance as fully differential signaling. Also, these results were found to be relatively insensitive to the bus width, N . Of course, the complexity of these algorithms is significant and efficient high-speed hardware implementations are an open issue.

3.5 Balanced Codes

Another advantage of differential signaling over single-ended schemes is that switching noise both on-chip and radiated on the PCB are, to a first-order approximation, cancelled since there are always an equal number of high and low signals on the bus. It is well known that an N -bit binary bus can be coded on $(N + \log_2 N)$ bits or less to have an equal number of 1’s and 0’s at all times (see, for instance [49, 50]). As long as precoding is not used at

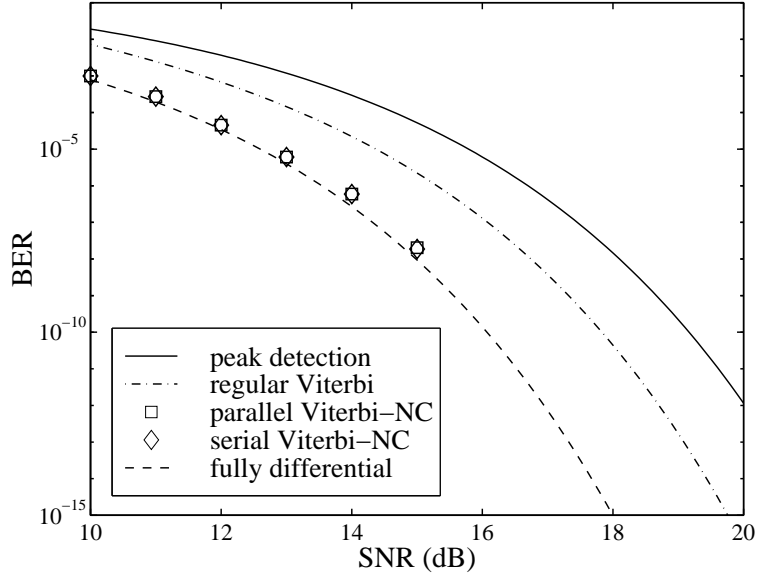


Figure 3.9: Simulation results for the serial and parallel Viterbi-NC algorithms.

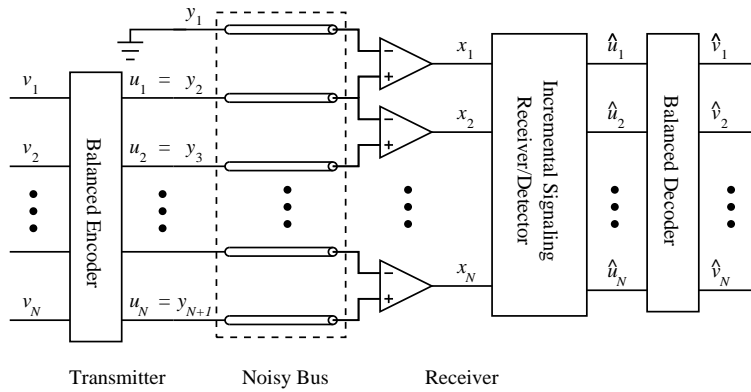


Figure 3.10: A system combining incremental signaling with balanced codes.

the transmitter, these codes can be combined with incremental signaling as shown in Fig. 3.10. The resulting system would reject common-mode interference and minimize switching noise similar to a fully differential system, but with far fewer IC pins and PCB traces. For instance, a 32-bit bus could be implemented with just 38 interconnects instead of 64.

Interestingly, if a balanced code is used, all received signals can be terminated at a common point in the receiver as shown in Fig. 3.11. The resulting signal at y_{cm} provides an estimate of the common-mode noise on the bus, which simplifies hardware implementations of the modified Viterbi algorithms described in Section 3.4 by taking $\widetilde{n}_{CM} = y_{CM}$.

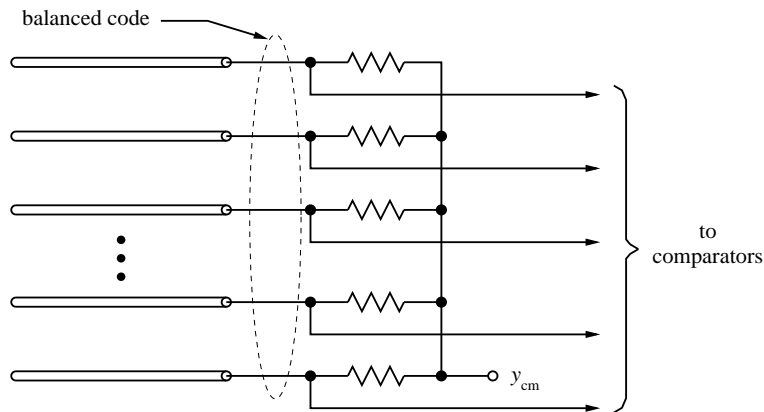


Figure 3.11: Termination of a balanced bus at the receiver with common-mode tap, y_{cm} .

3.6 Analog Implementation of Incremental Signaling

The main challenge in implementation of the proposed methods is the implementation of the Viterbi-NC algorithm. The Viterbi algorithm is the maximum likelihood means for decoding convolutional codes. If the signal processing prior to the Viterbi detection is relatively simple and can be done in analog as well, analog implementation would be much more useful due to eliminating the need for the analog to digital converter (ADC) and can substantially decrease the power consumption of the decoder. The Viterbi detector proposed by Shakiba in 1993 was a breakthrough in analog realization of the Viterbi algorithm [51], [52]. Instead of storing and updating the state metric of each state, the difference between these state metrics can be stored and updated equivalently [52].

Unfortunately, the implementation of the Viterbi-NC algorithm with this method is not straightforward due to the increased complexity of branch-metric calculation. This is mainly because of the estimated noise terms in the calculation of these metrics. Therefore, regular Viterbi implementation is more straightforward in this case. Regular implementation of the Viterbi-NC requires adders, comparators, multiplexers and multipliers. Simulation results show that there would be no substantial degradation in performance if the absolute value, instead of the square value, is used in the calculation of the branch metrics. This eliminates the need for multipliers which, in turn, significantly reduces the complexity of this implementation because high-performance implementation of a multiplier block is a challenging task by itself. This method for implementing Viterbi-NC method has been

Table 3.2: Comparison of various transmitter and receiver architectures for a 32-bit binary bus.

	Fully Differential	Peak Detection	MLSD w/ Viterbi	MLSD w/ Viterbi-NC	Parallel Viterbi-NC	Serial Viterbi-NC
Relative SNR @ BER = 10^{-8}:	0.0 dB	3.0 dB	1.25 dB	0.75 dB	0.15 dB	0.15 dB
Total no. of signal paths	64	33	34	34	34	34

verified by simulation results obtained from Simulink [53].

3.7 Summary

A technique called incremental signaling has been discussed which allows for N differential signals to be communicated via as few as $N + 1$ signal paths. It is possible to implement the technique using just two differential comparators per bit. Common-mode noise and even-order distortion terms would be completely rejected by such a system. However, the BER performance would be 3 dB worse than a fully differential system with respect to independent noise sources. One possible approach for regaining the 3 dB of lost performance is MLSD. Several algorithms for MLSD were presented and their relative performance is summarized in Table 3.2. Although the exact numbers depend upon the noise model used for the data bus simulations, general trends can be identified. Using the modified Viterbi algorithms, it is possible to obtain practically the same noise performance as fully differential signaling using just $N + 2$ signal paths. The proposed techniques are all very general. They are compatible with either voltage or current mode drivers and the results can be extended to multi-level signals. Finally, incremental signaling can be combined with balanced-bus encoding schemes described elsewhere to reduce switching noise and obtain an estimate of the common-mode noise on the bus.

Chapter 4

Coding Schemes for Chip-to-Chip Interconnect Applications

Increasing demand for high-speed inter-chip interconnects requires faster links that consume less power. Since the Shannon limit for these links is at least an order of magnitude higher than the data rate of the current state-of-the-art designs, channel coding can be used to approach the theoretical Shannon limit. Although there are numerous capacity-approaching codes in the literature, the complexity of these codes prohibits their use in high-speed inter-chip applications. This chapter studies several suitable coding schemes for chip-to-chip communication and backplane applications.

Section 4.1 introduces a simple coding scheme for binary (2-PAM) signaling that is significantly less sensitive to crosstalk, jitter, ISI and residual reflections than the regular 2-PAM scheme. This section also proposes several multi-level coding schemes that are motivated by the Gigabit-Ethernet scheme [22]. The approach is to transmit information in 5-PAM or 6-PAM instead of 4-PAM and use some techniques, such as coded modulation [23], to achieve a moderate coding gain. Section 4.2 provides a realistic model for the channel. This model is used for three typical channels. Simulation results are shown in Section 4.3 for one binary signaling scheme and one multi-level signaling scheme using these channel models. Finally, Section 4.4 explains low-complexity architectures for analog implementation of these coding schemes. It should be noted that although these schemes can be applied to both single-ended and fully-differential architectures, most figures show the single-ended architecture only to

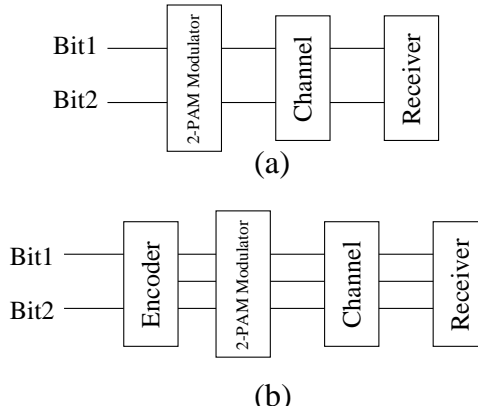


Figure 4.1: (a) General block diagram of a 2-PAM signaling scheme; (b) General Block diagram of the proposed coding scheme (3LINE-PAM2)

simplify the illustration.

4.1 Coding Schemes for Chip-to-Chip Communication

Although the 50-year-old edifice of coding theory has resulted in numerous capacity-approaching codes, the search for low-complexity coding schemes for practical implementation is still an active research topic [27]. In chip-to-chip communication applications, the main challenge is to come up with low-complexity coding schemes that can be implemented at high-speed. This section investigates several suitable coding schemes for chip-to-chip communication applications. The use of coding in inter-chip applications can be categorized into two subsections: two-level signaling and multi-level signaling.

4.1.1 Coding Schemes for Two-Level Signaling

As shown in Fig. 4.1a, in a 2-PAM (binary) signaling scheme, two lines are required for transmitting two bits of information. Symbols $(-1, -1)$, $(-1, 1)$, $(1, -1)$, and $(1, 1)$ can be used to send the information bits 00, 01, 10, and 11. The minimum squared Euclidean distance (MSED) in this constellation is 4. To achieve an appreciable coding gain, a signaling scheme with more than two lines could be used. A simple scheme, 3LINE-PAM2, is to use codewords $(-1, -1, -1)$, $(-1, 1, 1)$, $(1, -1, 1)$, and $(1, 1, -1)$ for transmitting 00, 01, 10, and 11,

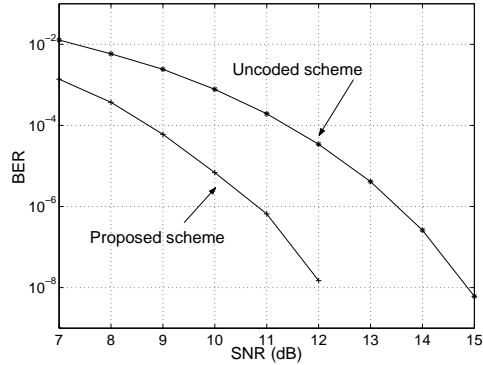


Figure 4.2: Simulation results for 3LINE-PAM2 and regular 2-PAM in the case of AWGN respectively. The MSED of these codewords (MSED=8) is twice of the one in the uncoded 2-PAM signaling scheme (MSED=4) and therefore it provides 3 dB coding gain. Obviously, this gain is achieved at the cost of adding one more line to the interconnect link as shown in Fig. 4.1b.

For decoding of the received signal, the Euclidean distance of the received signal to each of the transmitted codewords $((-1,-1,-1), (-1,1,1), (1,-1,1), (1,1,-1))$ should be calculated and the one that has the smallest distance is decoded as the output. For example, the decoder output would be 00 if the codeword $(-1, -1, -1)$ has the smallest distance to the received signal. Although this seems to significantly increase the complexity of the receiver, a low-complexity method, which needs only 6 comparators and several logic gates, is proposed in Section 4.4.

A Simulink model is used here to verify the coding gain of 3LINE-PAM2 in the case of additive white Gaussian noise (AWGN) channel [53]. As shown in Fig. 4.2, the proposed coding scheme provides roughly 2.8 dB coding gain at a BER of approximately 10^{-6} . The two curves in this figure diverge slightly and the full 3 dB gain is expected to be obtained at higher SNRs. However, the extra line in this signaling scheme needs an extra 1.7 dB power, which reduces the overall gain. Nevertheless, this method can significantly reduce the required SNR in the presence of crosstalk. The AWGN model for the channel is modified to the one in Fig. 4.3, which takes into account the effect of crosstalk. As shown in this figure, we model the crosstalk by taking the derivative of the transmitted signal in the discrete time domain. Crosstalk amplitude can be adjusted by a gain factor g .

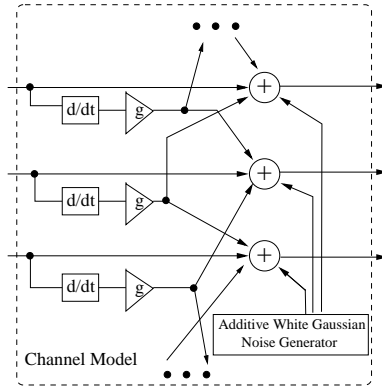


Figure 4.3: Modified channel model to take into account the effect of crosstalk

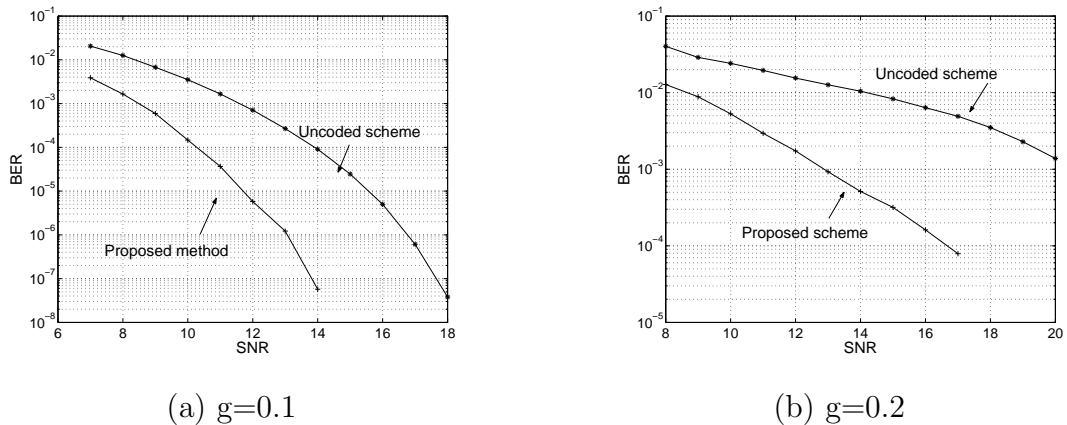


Figure 4.4: Performance of the proposed method in the presence of crosstalk.

Two sets of simulations have been performed to determine the performance of the proposed method in the presence of crosstalk. Fig. 4.4 shows the simulation results for two different values of the crosstalk gain ($g = 0.1$ in Fig. 4.4a and $g = 0.2$ in Fig. 4.4b). As shown in this figure, the proposed method achieves 4 dB gain at BER of 10^{-7} over the ordinary 2-PAM signaling when $g = 0.1$. Interestingly, the performance improvement for the case of $g = 0.2$ is roughly 8 dB at $BER = 10^{-3}$. These results show that the proposed method is significantly less sensitive to crosstalk than the regular binary signaling scheme, which in turn justifies the use of one more line.

4.1.2 Coding Schemes for Multi-Level Signaling

Although the proposed coding scheme in section 4.1.1 provides a significant gain, especially in the presence of crosstalk, the overhead of the 3LINE-PAM2 method precludes its use

in many applications where the total number of signal traces between two chips is limited. Multi-level signaling such as 4-PAM can be used to reduce the number of required signal traces in a bus. In this section several coding schemes for multi-level signaling are proposed that are based on the Gigabit-Ethernet coding scheme. Therefore, a brief explanation of Gigabit-Ethernet and coded-modulation is necessary.

Coded Modulation and Gigabit-Ethernet Coding

In Gigabit Ethernet, 1 Gb/s throughput is achieved with four pairs of twisted pair cables. The IEEE 802.3ab standard settled on a base-band 5-level PAM (5-PAM) combined with trellis coding [24], [54]. This scheme makes use of 5-level PAM ($\{-2, -1, 0, 1, 2\}$) on each pair of wires to code two bits of information. Transmitting two bits of information needs only four levels and therefore the extra level in the 5-PAM scheme provides a code redundancy that can be used for improving the performance. The four pairs of cables form a four-dimensional constellation (each pair represents one dimension). The total number of the points in the constellation is $5^4 = 625$, but only 256 points are necessary for transmitting eight bits. This redundancy can be used to achieve a 1.5 dB coding gain with symbol-to-symbol detection and 4.5 dB with a sequence detector, such as a Viterbi decoder, over the uncoded ordinary 4-PAM signaling [54].

Partitioning the set of points in each dimension into two subsets $A : \{-1, 1\}$ and $B : \{-2, 0, 2\}$ is the basic idea behind this method. As shown in Fig. 4.5, eight four-dimensional subsets (S_0 to S_7) can be formed by means of this partitioning. The intra-subset MSED for each subset is four. Notice that the set M ($M = \{S_0, S_2, S_4, S_6\}$), which has 313 points, can be used to construct a constellation for transmitting eight bits. The MSED of this constellation is two, which is twice the MSED of ordinary 4-PAM. The scaled version of this constellation, which has the same MSED as 4-PAM, has roughly the same BER as 4-PAM. However, it can be shown that it needs 1.5 dB less transmitted power than the transmitted power of 4-PAM. This idea can be used to construct a signaling scheme, hereafter referred to as Coded-Modulation-PAM5 scheme, which provides 1.5 dB gain over the ordinary 4-PAM signaling. Moreover, a trellis encoder in the transmitter and a Viterbi decoder in the receiver can be used to achieve an overall 4.5 dB coding gain over the uncoded 4-PAM scheme [22].

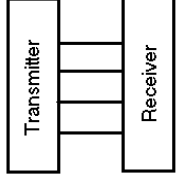
PAM5 Partitioning		4D Subsets	# of Points	MSED
B	■ +2	S0: AAAA & BBBB	97	4
A	● +1	S1: AAAB & BBBA	78	4
B	■ 0	S2: AABB & BBAA	72	4
A	● -1	S3: AABA & BBAB	78	4
B	■ -2	S4: ABBA & BAAB	72	4
Gigabit Ethernet Structure 		S5: ABBB & BAAA	78	4
		S6: ABAB & BABA	72	4
		S7: ABAA & BABB	78	4
		$M=\{S0,S2,S4,S6\}$	313	2
		$N=\{S1,S3,S5,S7\}$	312	2
		$\{M,N\}$	625	1

Figure 4.5: Subset partitioning in 4D space with 5-PAM in each dimension

Multi-Level Coding Schemes for Chip-to-Chip Communication

As mentioned before, using a Viterbi decoder in the receiver leads to a 4.5 dB gain over 4-PAM. Unfortunately, the complexity of the Viterbi decoder prevents its use for high-speed chip-to-chip interconnects. One possible solution is to use the Coded-Modulation-PAM5 scheme. However, the 1.5 dB gain of this scheme over ordinary 4-PAM is only a modest gain.

As shown in Fig. 4.5, the intra-subset MSED of each subset ($S0$ to $S7$) is four, which is twice the MSED of set M . Therefore, using only one subset provides a better coding gain. However, the number of points in subset $S0$ (97), which has the maximum number of points, is far from the required number for transmitting eight bits (256).

Adding one dimension, i.e. one extra line, to the constellation increases the redundancy, so different schemes or constellations with $MSED=4$ could be found by adding more dimensions to the constellation. Table 4.1 provides some information that could be used to find such a scheme. This table shows the logarithm of the number of codewords with $MSED = 4$ (second column) for different number of dimensions (first column). The third column shows the number of bits per symbol per line. The number of bits per symbol for each scheme is divided by the number of bits per symbol for 4-PAM to get the normalized bits/symbol, $(bits/symbol)_n$, for each constellation. This parameter is shown in the fourth column.

Table 4.1: Bit-rate comparison for different constellations with MSED=4 for 5-PAM.

# dimensions	$\log_2(\# \text{ points with MSED}=4)$	bits/symbol/(# of lines)	$(bits/Symbol)_n$	scheme name
4	6.5	1.5	.75	4LINE-PAM5
5	8.18	1.6	0.8	5LINE-PAM5
6	10.18	1.666	0.833	6LINE-PAM5
7	12.26	1.714	0.857	
8	14.6	1.75	0.875	
16	32.15	2	1	

Each row in this table represents a constellation or, equivalently, a signaling scheme for chip-to-chip communication. The names of some of the more useful schemes for this application are shown in the far-right column of Table 4.1. For example, the third row of table 4.1 shows that a six-dimensional constellation can be used for transmitting 10 bits. This method is called 6LINE-PAM5 scheme throughout this chapter. The same idea can be applied to the first and the second row of this table. This introduces two new schemes: 4LINE-PAM5 (transmitting six bits over four lines) and 5LINE-PAM5 (transmitting eight bits over five lines).

On account of the structure of these constellations (schemes), the minimum distance of the codewords in these constellations is more than the minimum distance of the codewords in 4-PAM for a given transmitted power. This means that for the same BER, this scheme requires less power compared to the conventional 4-PAM scheme. Actually, it can be shown that these constellations can provide roughly 3 dB gain over the uncoded 4-PAM signaling scheme. As shown in this table, there is a trade off between complexity and data rate; reducing the number of dimensions in the constellation results in lower data rate and lower complexity.

An alternative approach is to use 6-PAM modulation. Here, the set of points in each dimension can be partitioned into two subsets $A : \{-1.5, .5, 2.5\}$ and $B : \{-2.5, -.5, 1.5\}$. Table 4.2, which is similar to Table 4.1, summarizes the possible schemes in this case. Again each row in this table is a possible scheme for this application. For instance, the first row introduces a method, hereafter referred to as 4LINE-PAM6, for transmitting seven bits over

Table 4.2: Bit-rate comparison for different constellations with $MSED = 4$ for 6-PAM.

# dimensions	$\log_2(\# \text{ points with } MSED=4)$	bits/symbol/(# of lines)	$(bits/Symbol)_n$	scheme name
4	7.34	1.75	.875	4LINE-PAM6
5	8.92	1.6	0.8	
6	11.51	1.83	0.92	
7	14.09	2	1	7LINE-PAM6

Table 4.3: Performance comparison for different schemes.

scheme	gain over 4-PAM	$(bits/symbol)_n$	Complexity
PAM3	2 dB	0.75	low
4LINE-PAM6	3.02 dB	0.875	low
4LINE-PAM5	3.1 dB	0.75	low
5LINE-PAM5	2.5 dB	0.8	moderate
6LINE-PAM5	3.17 dB	0.833	moderate
7LINE-PAM6	$\simeq 3$ dB	1	High

four lines with 6-PAM modulation in each line.

In this scheme, patterns $AAAA$ and $BBBB$ are used for constructing a constellation with $MSED = 4$. There are 162 points in this constellation and therefore seven bits can be transmitted with this scheme. From the original 162-point constellation, 34 points that have higher energy have been removed to form a 128-point constellation. It can be shown that this scheme provides roughly 3 dB gain over the uncoded 4-PAM while its data rate is only 13% less than the data rate of 4-PAM. The last row in Table 4.2 represents a scheme for transmitting 14 bits over 7 lines. Therefore, it has the same throughput as the 4-PAM scheme ($(bits/symbol)_n = 1$), and it also provides about 3 dB gain over 4-PAM. However, its complexity is much more than the complexity of 4LINE-PAM6 scheme.

Using an uncoded 3-level PAM on four lines results in a 3^4 -point constellation. Therefore, another option is to transmit six bits over four lines using 3-PAM in each line. However, the gain of this method over the ordinary 4-PAM is roughly 2 dB and $(bits/symbol)_n = 0.75$, which is not as good as the corresponding values in 4LINE-PAM6.

This section has introduced several signaling schemes that can be used in chip-to-chip

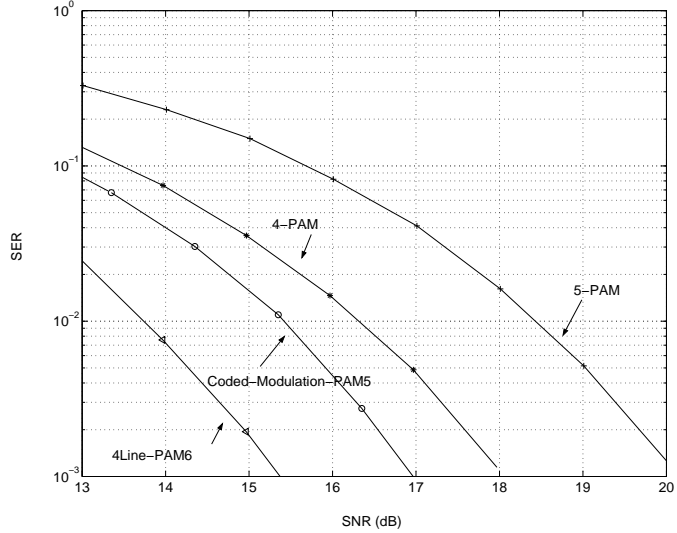


Figure 4.6: Simulation Result for 4-PAM, 5-PAM, 4LINE-PAM6, and Coded-Modulation-PAM5 schemes

communication. Table 4.3 summarizes the performance of some of the schemes that have been studied. The second column in this table shows the performance improvement of each method over the 4-PAM scheme and the third column shows the normalized data rate for each method. To obtain the performance improvement of each scheme, we make the MSED of each scheme equal to the MSED of 4-PAM, thereby obtaining approximately the same BER, and calculate the extra power that the traditional 4-PAM scheme needs.

Among the schemes in Table 4.3, the 4LINE-PAM6 method is the best signaling scheme for high-speed inter-chip applications since it is a low-complexity method that has the second largest $(bits/symbol)_n$. A low-complexity method for an analog implementation of 4LINE-PAM6 is proposed in Section 4.4.2.

To verify the results of Table 4.3 for 4LINE-PAM6 scheme and compare the performance of this method with the performance of Coded-Modulation-PAM5 and regular 4-PAM, a model in Simulink is developed. This model uses the additive white Gaussian noise (AWGN) model for the channel. Fig. 4.6 shows the simulation results, the symbol error rate (SER) versus the signal to noise ratio (SNR), for several signaling schemes. As shown in this figure, the performance of the 4LINE-PAM6 method is roughly 2.7 dB better than the performance of the 4-PAM scheme at a SER of about 10^{-3} . However, the expected gain for this method is

3 dB. The reason for this small difference (0.3 dB) is the fact that each point in the 4LINE-PAM6 constellation has more neighboring points, points in the constellation with minimum distance away from the original point, compared to the 4-PAM constellation. Fortunately at high SNRs, where the SER-versus-SNR curve has a larger slope, this difference would be even smaller and almost the full 3-dB gain over 4-PAM can be achieved.

It should be mentioned that some interconnect applications are peak-power limited, hence the comparison between the proposed method should be performed when the peak power is the same for both methods. Fig.4.7 shows the simulation results in this case. The horizontal axis shows the noise attenuation in dB and vertical axis shows the symbol error rate. The expected gain in this case is about 1.6 dB less than the expected gain in the previous case. This 1.6 dB comes from the fact that if the peak signal power in both 4-PAM and 4LINE-PAM6 schemes is the same, the average power of a 4LINE-PAM6 scheme is 1.6 dB less than that of a conventional 4-PAM scheme.

As shown in Fig. 4.7, simulation results show that the gain in this case is roughly 1.4 dB as we expected. A similar approach to the one in section 4.1.1 is used to take crosstalk into account. Fig. 4.8 shows a performance improvement of about 2.5 dB and 4 dB for $g = 0.05$ and $g = 0.07$, respectively. The above 4 dB gain translates to roughly 5.6 dB power saving. The rest of this chapter specifies coding gain based on power saving at certain BER and to obtain the gain for peak-power-limited applications, 1.6 dB should be deducted from the mentioned coding gain. It should be also noted that to avoid excessively long simulations, they are not extended to high SNRs.

4.2 A Realistic Channel Model for Chip-to-Chip Applications

Fig. 4.9 shows the general block diagram of a chip-to-chip communication system. The transmitter is modelled by a voltage source, an output impedance (Z_s) and a package. Similarly the receiver is modelled by a receiver package and an impedance (Z_L), which is the input impedance of the receiver. Different kinds of transmission lines, such as microstrip and stripline, can be used for PCB traces between two chips. In general, PCB traces can be

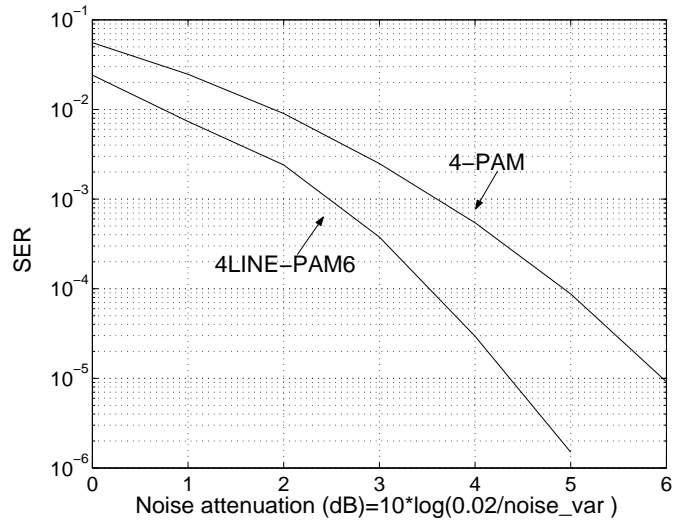
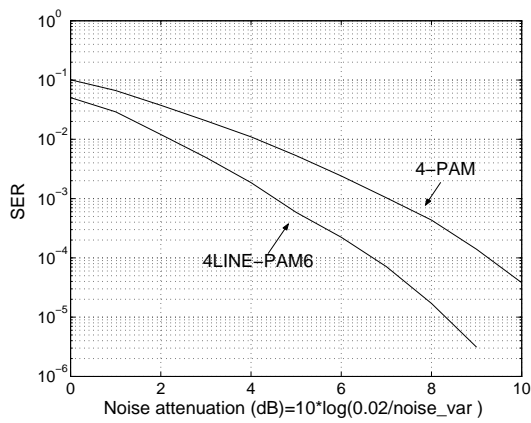
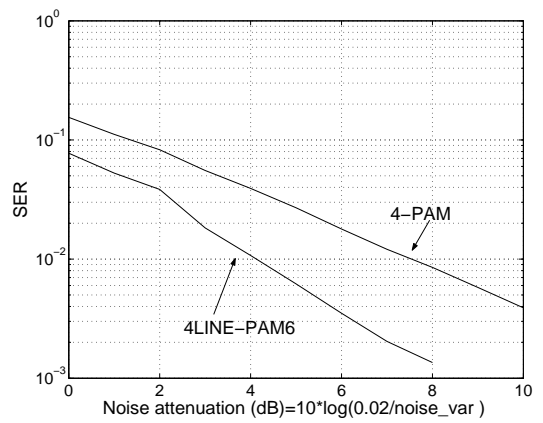


Figure 4.7: Simulation Result for 4-PAM, 4LINE-PAM6 in peak-power-limited case



(a) $g=0.05$



(b) $g=.07$

Figure 4.8: Performance of the 4LINE-PAM6 and 4-PAM methods in the presence of crosstalk in peak-power-limited case.

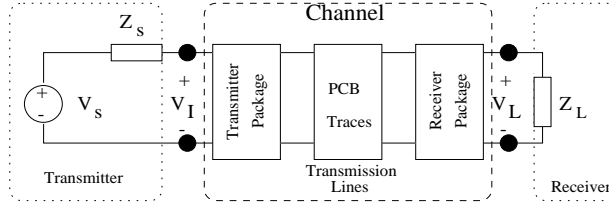


Figure 4.9: A general block diagram for a chip-to-chip communication system.

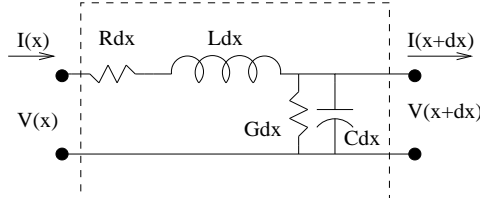


Figure 4.10: Per-unit-length model of a transmission line.

modelled as transmission lines.

For perfect termination, Z_L and Z_s should be equal to the characteristic impedance of the transmission line [55]. In practice, it is very difficult to have perfect termination and therefore there would be some residual reflections. Attenuation of the transmission line is also another important parameter in this system, which causes inter-symbol interference (ISI). Throughout this thesis, ISI refers to dispersion-induced ISI only to differentiate it from the intersymbol-interference due to the reflections.

So far, a channel model that takes into account the effect of crosstalk and/or AWGN has been used in the simulations. This would be a good model for a channel with perfect termination and no ISI. Nevertheless, the main sources of noise for this application in practical systems are usually ISI and residual reflections due to the imperfect termination. Consequently, a more realistic model for the channel should take into account the effect of ISI and reflection. This section presents a general channel model for inter-chip communication applications.

The two-port characterization of transmission line is derived from the per-unit-length two-port model shown in Fig. 4.10. The R , L , G , and C represent the resistance, inductance, conductance, and capacitance per unit length of the transmission line, respectively. Voltage and current at any point of the transmission line relate through these differential equations

[55]:

$$\begin{aligned}\frac{d^2V}{dx^2} &= \gamma^2 \cdot I, \\ \frac{d^2I}{dx^2} &= \gamma^2 \cdot V,\end{aligned}\tag{4.1}$$

where

$$\gamma = \alpha + j\beta = \sqrt{(R + j\omega L) \cdot (G + j\omega C)}.\tag{4.2}$$

The solution to the set of differential equations in (4.1) is two opposite-direction voltage/current waves:

$$\begin{aligned}V(x) &= V_0^+ \cdot e^{-\gamma x} + V_0^- \cdot e^{\gamma x}, \\ I(x) &= I_0^+ \cdot e^{-\gamma x} + I_0^- \cdot e^{\gamma x}.\end{aligned}\tag{4.3}$$

It can also be shown that the ratio of V_0^+/I_0^+ and V_0^-/I_0^- are constant and equal to the characteristic impedance of transmission line, Z_0 :

$$Z_0 = \frac{V_0^+}{I_0^+} = \frac{V_0^-}{I_0^-} = \sqrt{\frac{R + j\omega L}{G + j\omega C}}.\tag{4.4}$$

In a transmission line with length d , the load voltage and current, V_L and I_L , can be obtained by replacing x with d in (4.3). (4.3) and (4.4) can be used to obtain a two-port ABCD representation for the transmission line [55]:

$$\begin{pmatrix} V(0) \\ I(0) \end{pmatrix} = \begin{pmatrix} \cosh(\gamma d) & Z_0 \cdot \sinh(\gamma d) \\ \frac{1}{Z_0} \cdot \sinh(\gamma d) & \cosh(\gamma d) \end{pmatrix} \cdot \begin{pmatrix} V(d) \\ I(d) \end{pmatrix}.\tag{4.5}$$

A 2-D field solver (W-element in HSPICE) is used to obtain the *RLCG* for two typical transmission lines for inter-chip applications: microstrip and stripline with $Z_0 = 100\Omega$. These *RLCG* parameters are frequency dependent and therefore ABCD representation in (4.5) can also take into account the skin effect and dielectric loss [56].

It is straightforward to obtain the two-port ABCD representation of package models and source and load impedances. Multiplying those two-port representations results in the two-port representation of the entire channel, and thereby the transfer function V_L/V_S .

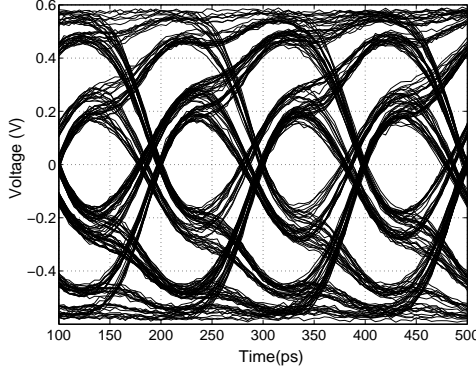


Figure 4.11: Eyediagram for a 10 Gb/s link at the receiver (channel: package, 300-mm microstrip, package) when $Z_s = 80$ and $Z_l = 120$

4.3 Simulation Results

This sections shows the simulation results for 3LINE-PAM2 and 4LINE-PAM6 schemes in typical chip-to-chip communication channels. The channel model in these simulations consists of two parts. The first part uses the magnitude and the phase of the channel transfer function obtained by the method presented in Section 4.2 to find the impulse response of the channel. This impulse response is used to find the output of the channel. The second part adds jitter to the clock and white Gaussian noise to the output of the first section. Therefore, this model is a general model that takes into account the effect of jitter, ISI, reflection and additive white Gaussian noise.

4.3.1 Simulation Results for the 3LINE-PAM2 Scheme

A typical channel model for this application can be obtained with the proposed method in Section 4.2. Fig. 4.11 shows the eye-diagram of a 10 Gb/s link at the receiver for a channel comprised of a transmitter package, 0.3-m microstrip, and a receiver package for a binary signaling scheme. Here, the source and load impedances are selected to be 80Ω and 120Ω , respectively. Fig. 4.12 shows the magnitude of the transfer function for this channel. In addition, the channel model is modified to take into account the effect of clock jitter. Fig. 4.13 shows the performance of 3LINE-PAM2 for this model at 10Gb/s. As shown in Fig 4.13a, the proposed method provides roughly 5 dB performance improvement at $\text{BER}=10^{-3}$.

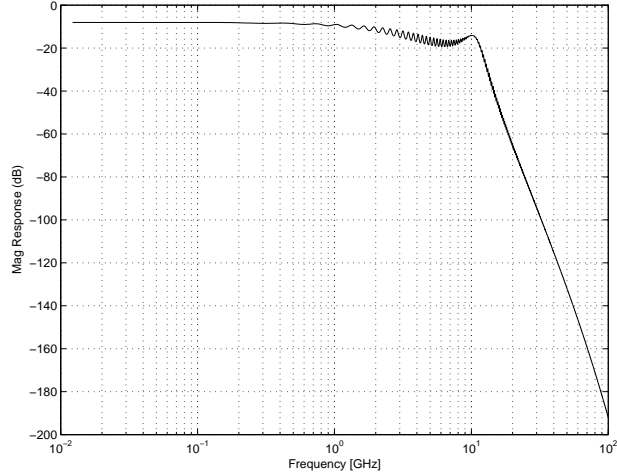


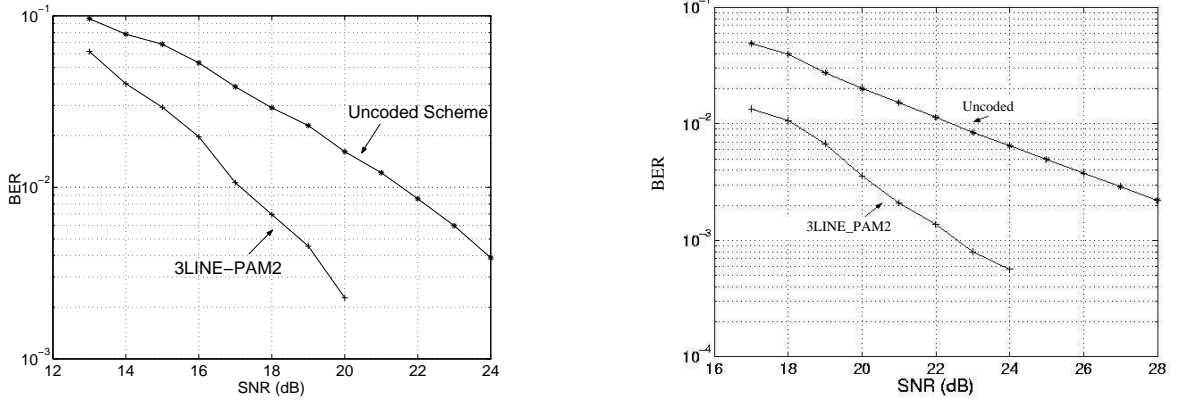
Figure 4.12: Channel frequency response for the case of microstrip and $Z_L = 120\Omega$, $Z_S = 80\Omega$, $d = 0.3\text{m}$

Significant performance improvement of about 8 dB is achieved with a 20 ps p-p jitter (see Fig. 4.13b).

Simulation results in this section show that the 3LINE-PAM2 scheme is significantly less sensitive to jitter, ISI, and residual reflection than the ordinary 2-PAM signaling scheme. As shown in Fig 4.11, the eye-height of the received signal is roughly 0.3 V, which translates to a noise margin of only 0.15 V, for a 2 V signal swing at the transmitter. Consequently, in an advanced technology in which the signal swing cannot be more than 1 V, the use of a coding scheme to achieve the required performance becomes more appealing. More specifically, the proposed method might even eliminate the need for an equalizer in a high-speed inter-chip application. Nevertheless, an equalizer can be used along with a coding scheme to further improve the performance of the system. Indeed, using both equalizer and coding is common in most of the communication applications.

4.3.2 Simulation Results for the 4LINE-PAM6 Scheme

Two typical channel models for high-speed inter-chip communication applications are used in this section to determine the performance of 4LINE-PAM6 scheme in the presence of jitter, ISI and residual reflections. One set of simulations for each channel has been performed in MATLAB to compare the performance of this method with the ordinary 4-PAM scheme.



(a) Without jitter

(b) With 20 ps peak-to-peak jitter

Figure 4.13: 3LINE-PAM2 simulation results.

The corresponding simulation results for each channel are presented in this section. These results show that the 4LINE-PAM6 scheme is significantly less sensitive to jitter, ISI and residual reflections.

Case I: ($Z_l = 110\Omega$, $Z_s = 90\Omega$, $Z_0 = 100\Omega$ for a 0.20-m stripline)

The first channel is composed of a 0.2-m stripline, transmitter package, receiver package and source and load termination resistors. A 10% terminations mismatch is considered for the source and load resistors ($Z_l = 110\Omega$, $Z_s = 90\Omega$) to create some residual reflections. Fig. 4.14 shows the magnitude of the channel transfer function in this case. Since the data rate used for this simulation is 10Gb/s (5GS/s), the frequency range of interest is 0-2.5 GHz. The attenuation of the channel in this case is roughly 4.5 dB at 2.5 GHz and therefore this channel introduces moderate ISI.

Fig. 4.15a illustrates the SER versus SNR for the 4LINE-PAM6 and 4-PAM schemes, which shows roughly 6 dB gain for the 4LINE-PAM6 scheme over 4-PAM at BER around 10^{-2} . Since the reasonable SER for chip-to-chip communication is of the order of 10^{-15} and the two curves in Fig. 4.15a are slowly diverging, the gain at high SNRs would be even better. Fig. 4.15b shows the performance of the 4LINE-PAM6 in the presence of a 20-ps p-p jitter. As shown in this figure, 4LINE-PAM6 shows roughly 7 dB gain over 4-PAM

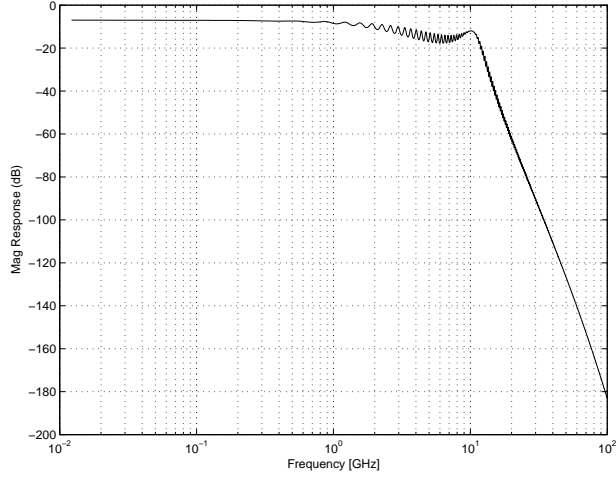


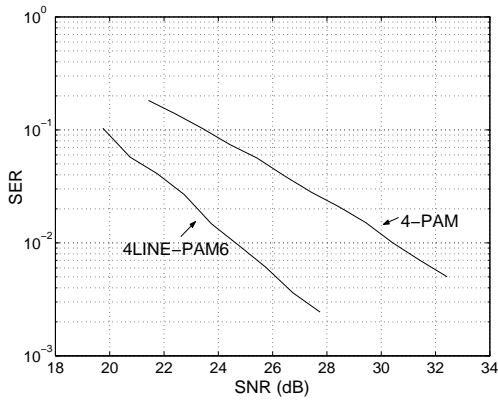
Figure 4.14: Channel frequency response for the case of stripline $Z_L = 110\Omega$, $Z_S = 90\Omega$, $d = 0.2\text{m}$

at SER around 10^{-2} , which is better than the simulation result without jitter. Therefore, 4LINE-PAM6 is less sensitive to jitter than 4-PAM.

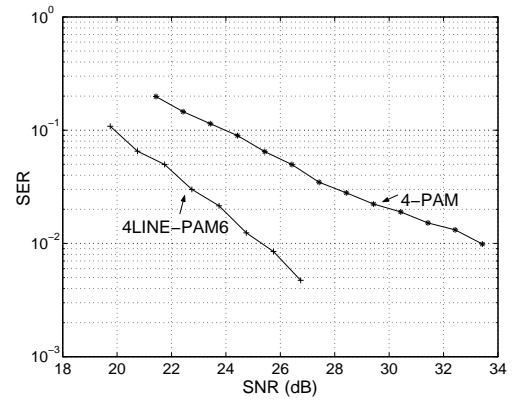
Case II: ($Z_l = 105\Omega$, $Z_s = 95\Omega$, $Z_0 = 100\Omega$ for a 50-mm microstrip)

To assess the performance of 4LINE-PAM6 at 20Gb/s, another model for the channel is used, which models a high-speed transmitter package, a high-speed receiver package and a 50-mm microstrip. A 5% termination mismatch is considered for the source and load resistors ($Z_l = 105\Omega$, $Z_s = 95\Omega$). Fig. 4.16 shows the magnitude of the channel transfer function in this case. The attenuation of the channel is roughly 6 dB at 5 GHz.

Fig. 4.17a, which illustrates the SER versus SNR for the 4LINE-PAM6 and 4-PAM schemes, shows roughly 4 dB gain for the 4LINE-PAM6 scheme over 4-PAM at $\text{BER}=10^{-3}$. Fig. 4.17b shows the performance of the 4LINE-PAM6 in the presence of a 10-ps p-p jitter. As illustrated in this figure, 4LINE-PAM6 scheme shows roughly 4.7 dB gain over 4-PAM at $\text{SER}=10^{-3}$, which is again better than the simulation result without jitter.



(a) Without jitter



(b) With 20ps peak-to-peak jitter

Figure 4.15: Case I: $Z_L = 110\Omega$, $Z_S = 90\Omega$, 0.2-m stripline.

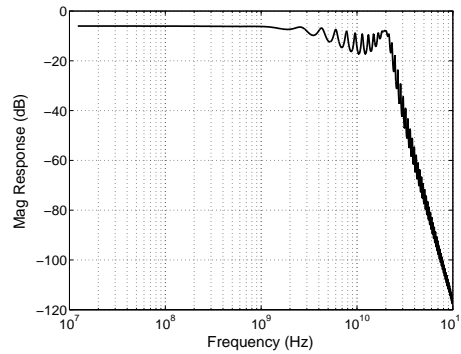
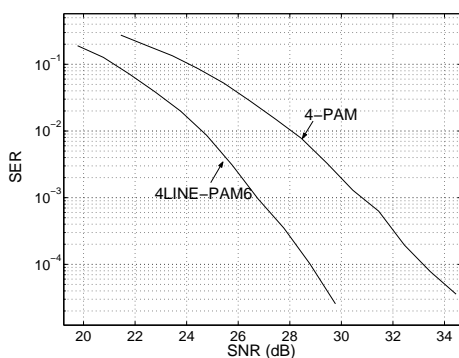
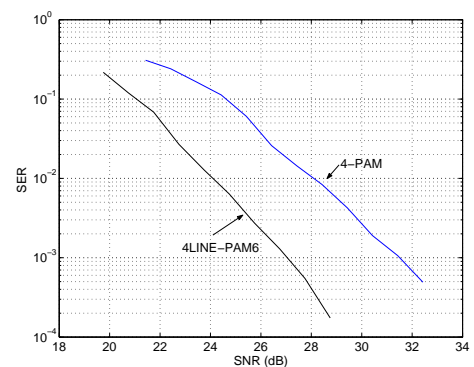


Figure 4.16: Channel frequency response for the case of microstrip with high-speed package and $Z_L = 105\Omega$, $Z_S = 95\Omega$, $d = 50\text{mm}$



(a) Without jitter



(b) With 10ps peak-to-peak jitter

Figure 4.17: Case II: $Z_S = 95\Omega$, $Z_L = 105$, 50-mm microstrip at 20Gb/s.

4.4 Analog Implementation

As mentioned earlier, the main challenge in high-speed interconnect applications is to come up with low-complexity signaling scheme that not only provides some coding gain but also can be implemented at high-speed. This section presents low-complexity architectures for 3LINE-PAM2 and 4LINE-PAM6 schemes.

4.4.1 Analog Implementation of the 3LINE-PAM2 Scheme

In an optimal decoder for the 3LINE-PAM2 scheme, assuming all constellation points are equally likely, the distance of the received signal and all four points in the constellation should be calculated and the point that has the minimum distance is decoded as the output. Assume the received signal is (x, y, z) . The Euclidean distances of this signal to the transmitted codewords $(-1, -1, -1)$, $(-1, 1, 1)$, $(1, -1, 1)$, and $(1, 1, -1)$ are $(x + 1)^2 + (y + 1)^2 + (z + 1)^2$, $(x + 1)^2 + (y - 1)^2 + (z - 1)^2$, $(x - 1)^2 + (y + 1)^2 + (z - 1)^2$, and $(x - 1)^2 + (y - 1)^2 + (z + 1)^2$, respectively. Cancelling all common terms and dividing them by 2 leads to the following terms for distances: $x + y + z$, $x - y - z$, $-x + y - z$, $-x - y + z$.

Hence, the decoding algorithm is composed of two steps: the first step is to calculate these distances and the second step is to find the smallest one by means of 6 comparators. The former step is unnecessary since we can further simplify the comparisons. For example $x + y + z > x - y - z$ is equivalent to $y > -z$. Therefore, the transmitted information can be decoded by using only six comparators and several logic gates as shown in more detail in Fig. 4.18. The receiver architecture comprises of six comparators, three *AND* and two *OR* gates. Moreover, as shown in Fig. 4.18, the encoder in the transmitter is simply an *XOR* gate. Hence, the low-complexity of transceiver architecture makes its high-speed implementation feasible.

4.4.2 Analog Implementation of the 4LINE-PAM6 Scheme

In an optimal decoder for 4LINE-PAM6 scheme, assuming all points are equally likely, the distance of the received signal and all 128 points in the constellation should be calculated and the point that has the minimum distance is decoded as the output. Obviously, this method

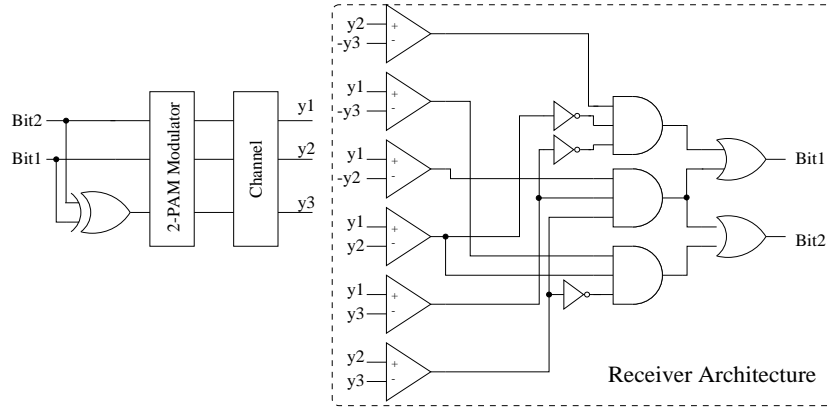


Figure 4.18: Transceiver architecture for the 3LINE-PAM2 method

is prohibitively complex and a suboptimal method with lower complexity is more desirable. A low-complexity method for analog implementation of 4LINE-PAM6 is proposed. This method shows a negligible performance degradation (less than 0.02 dB) compared with the optimal scheme. Only *AAAA* and *BBBB* patterns are used in the 4LINE-PAM6 method. Since each pattern has 81 points, the total number of points is 162. The proposed decoder is actually an optimal decoder for the 162-point constellation. Therefore its output would be a point in this constellation.

This method uses a simple architecture to specify the transmitted pattern (*AAAA* or *BBBB*) for the received signal. Once this is known, the decoding would simply be the decoding of an ordinary 3-level PAM, which needs only two comparators.¹ The first step, as shown in Fig. 4.19, is to find the distances of the received signal in each line with the closest point in subsets $A = \{-2.5, -0.5, 1.5\}$ and $B = \{-1.5, 0.5, 2.5\}$, d_A and d_B . The transmitted pattern can then be found by this inequality:

$$d_{A1}^2 + d_{A2}^2 + d_{A3}^2 + d_{A4}^2 < d_{B1}^2 + d_{B2}^2 + d_{B3}^2 + d_{B4}^2 . \quad (4.6)$$

If the output of the comparison in (4.6) is true, the transmitted pattern would be *AAAA*. The implementation of the decision in (4.6) needs circuitry that provides signals proportional to the square value of the d_{Ai} or d_{Bi} , which is not straightforward. Interestingly, d_{Bi} can be

¹We would like to acknowledge Prof. Frank Kschischang for his helpful comments, which have led to the development of the proposed low-complexity architecture for this receiver.

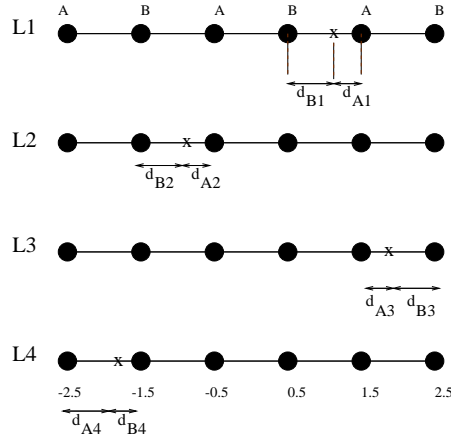


Figure 4.19: Main idea of an analog implementation of the 4LINE-PAM6 scheme.

expressed in terms of d_{Ai} as follows,

$$d_{Bi} = \begin{cases} d_{Ai} - 1 & \text{if } y_i > 2.5, \\ 1 - d_{Ai} & \text{if } -2.5 < y_i < 2.5, \\ 1 + d_{Ai} & \text{if } y_i < -2.5. \end{cases} \quad (4.7)$$

where y_i is the received signal in the i^{th} line. This leads to the following expression for the square value of d_{Bi} :

$$d_{Bi}^2 = (1 - m_i \times d_{Ai})^2, \quad (4.8)$$

where

$$m_i = \begin{cases} -1 & \text{if } y_i < -2.5, \\ 1 & \text{if } y_i > -2.5. \end{cases}$$

Substituting (4.8) in (4.6) results in:

$$\sum_{i=1}^4 (m_i d_{Ai} - 0.5) < 0. \quad (4.9)$$

So it would be sufficient to find $m_i d_{Ai}$ for each line and then add them all up. This addition is straightforward in current mode circuitry. It is straightforward to show that

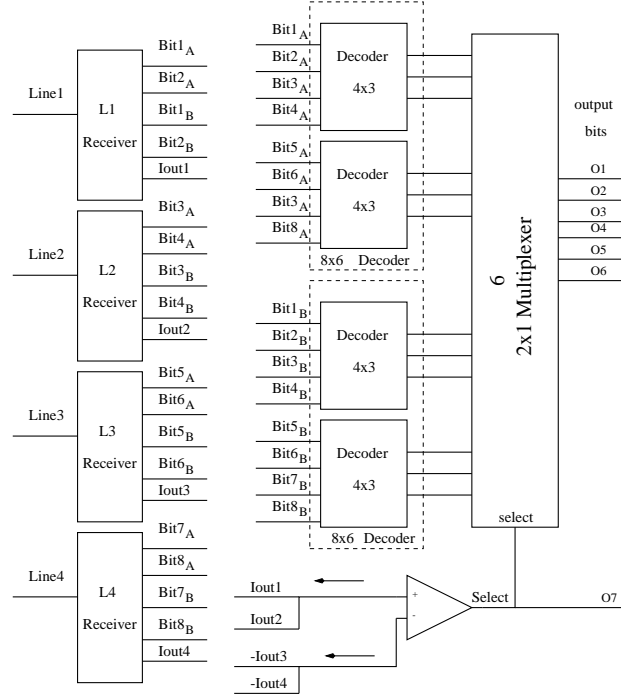


Figure 4.20: Block diagram of an analog implementation of 4LINE-PAM6 scheme.

$m_i d_{Ai}$ can be obtained by

$$(m_i d_{Ai} - 0.5) = \begin{cases} y_i + 2 & \text{if } y_i < -1.5, \\ -y_i - 1 & \text{if } -1.5 < y_i < -0.5, \\ y_i & \text{if } -0.5 < y_i < 0.5, \\ -y_i + 1 & \text{if } 0.5 < y_i < 1.5, \\ y_i - 2 & \text{if } y_i > 1.5. \end{cases} \quad (4.10)$$

Receiver Architecture

Equations (4.9) and (4.10) can be used to detect the transmitted pattern of the received signal. Knowing the transmitted pattern, only two comparators for each line are required to decode the received signal. Fig. 4.20 shows the general block diagram of the receiver for implementing this algorithm. The detail of the front-end block for each line is shown in Fig. 4.21. As shown in this figure, two comparators decode the signal for $AAAA$ pattern and similarly the other two comparators decode the signal for $BBBB$ pattern. I_{out} output of this block would be proportional to the $m_i d_{Ai} - 0.5$ for each line. Fortunately, the required

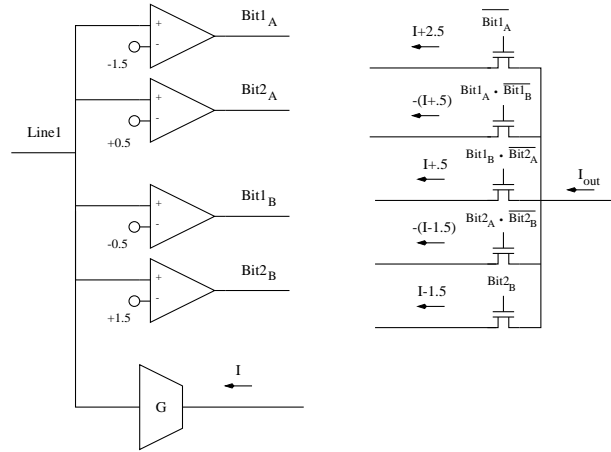


Figure 4.21: The detail of the receiver block (L1-L4) for each line.

thresholds for obtaining the I_{out} are identical to those used in these comparators (see Fig. 4.21 and Equation (4.10)). Therefore, no additional comparators are needed to obtain I_{out} for each line. As shown in Fig. 4.20, a comparator is used to perform the comparison in (4.9). The output of this comparator, "Select" signal, specifies the transmitted pattern for the received signal.

There are 64 points in each subconstellation, $AAAA$ or $BBBB$. This means that decoding a point in each subconstellation specifies six bits and the seventh bit would be the "Select" signal. As shown in Fig. 4.20, the output of the first stage of the receiver has eight bits corresponding to each subconstellation. These eight bits are mapped to six bits using an 8×6 digital decoder. In other words, two sets of six bits corresponding to the two subconstellations are decoded individually. Six 2×1 multiplexers are used to select one of these sets based on the "Select" signal.

Structure of 8×6 Decoder

As mentioned earlier, two 8×6 decoders map the output of the first stage of the receiver to 12 bits (two sets of 6 bits). To further simplify the structure, as shown in Fig. 4.20, each of the two 8×6 digital decoders is decomposed into two 4×3 digital decoders. Since the design methodology is similar for all of these 4×3 digital decoders, the design of the top decoder in Fig. 4.20 will be explained only.

The four inputs of this decoder ($Bit1_A$, $Bit2_A$, $Bit3_A$, $Bit4_A$) are the outputs of "L1

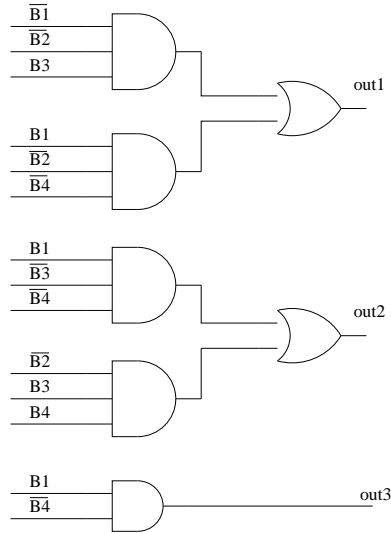


Figure 4.22: The detail of the 4×3 decoder.

Table 4.4: Mapping design for 4×3 decoder.

Signal1	Signal2	$Bit1_A$	$Bit2_A$	$Bit3_A$	$Bit4_A$	out1	out2	out3
-2.5	-0.5	0	0	1	0	1	0	0
-2.5	1.5	0	0	1	1	1	1	0
-0.5	-2.5	1	0	0	0	1	1	1
-0.5	-0.5	1	0	1	0	1	0	1
-0.5	1.5	1	0	1	1	0	1	0
1.5	-2.5	1	1	0	0	0	1	1
1.5	-0.5	1	1	1	0	0	0	1
1.5	1.5	1	1	1	1	0	0	0

Receiver” and ”L2 Receiver” blocks in Fig. 4.20. The first six columns of Table 4.4 show the relation between the signal on line1 and line2 and these four bits. Since the signal in line1 cannot be simultaneously larger than 0.5 and smaller than -1.5, $Bit1_A$ and $Bit2_A$ can have only three different possibilities (00, 10, 11). The same argument is true for $Bit3_A$ and $Bit4_A$. Therefore, the four input bits of this smaller decoder have nine different possibilities. The all-zero case is selected to be invalid since it corresponds to the constellation point (-2.5, -2.5), which is far from origin and thus needs more power.

The remaining eight different possibilities can be mapped to three output bits of the decoder. This mapping needs to be carefully designed to reduce BER. The mapping based

on Gray code is a well known scheme for a constellation with uniformly-distributed points. However, for some systems with irregular constellations, such as the mapping for this decoder, the common method of mapping is invalid. For a constellation with M points, there are a total of $M!$ different mappings and the search for the optimal mapping for large M is impractical. A suboptimal algorithm for this purpose is proposed in [57].

Fortunately, in this case, M is equal to 8 and looking for an optimal mapping is possible. The goal is to assign small Hamming distances to small Euclidean distances. A good mapping that needs a low-complexity circuitry for its implementation is shown in Table 4.4. Fig. 4.22 shows the required combinational circuitry for this mapping. As shown in this figure, this circuit is simple and it needs only 5 *AND* gates and 2 *OR* gates.

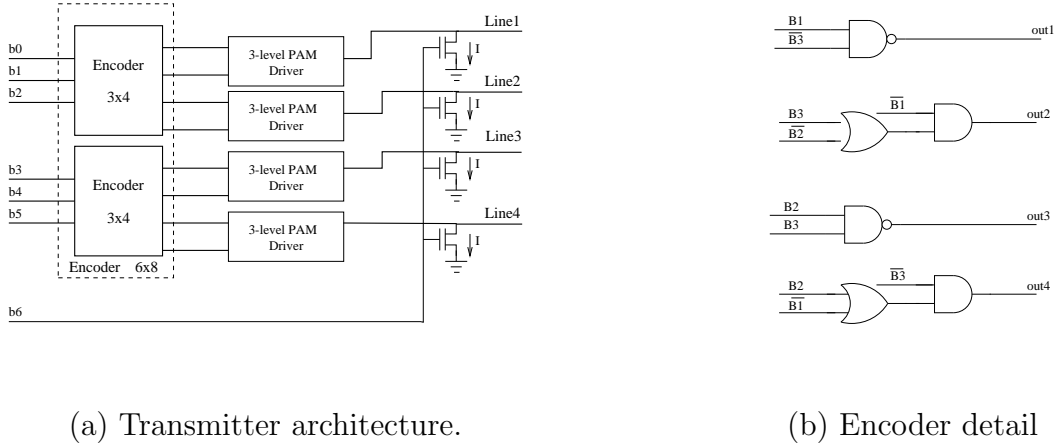
Transmitter Architecture

It seems that the 4LINE-PAM6 scheme needs a 6-PAM transmitter. Nevertheless, having only *AAAA* and *BBBB* patterns for the transmitted signals can also simplify the transmitter structure. Since the points in the subsets $A = \{-2.5, -0.5, 1.5\}$ and $B = \{-1.5, 0.5, 2.5\}$ are a shifted version of each other, the structure of the transmitter is basically a 3-PAM transmitter. As shown in Fig. 4.23a, a current-mode three-level transmitter can be used in each line to generate signals for all different possibilities of the first 6 bits. The last input bit specifies the transmitted pattern. If this bit is "1" a fixed current is added to all lines.

Similar to the receiver, a decoder is needed to map the six input bits to eight bits, two bits for each line, to make the data ready for transmission. This can be done with two 3×4 digital encoders. The required circuitry for these digital encoders is shown in Fig. 4.23b. This simple circuitry needs only two *AND*, two *NAND*, and two *OR* gates.

Simulation Results

Fig. 4.24 shows the simulation results for this analog implementation method and an optimal decoder, which are obtained by MATLAB [53] for an AWGN channel. The performance of this method is almost identical to the performance of the optimal method. Particularly, the performance of this method is roughly 0.02 dB worse than the performance of the optimal method at SER around 10^{-3} . Simulation results for a digital implementation of 4LINE-



(a) Transmitter architecture.

(b) Encoder detail

Figure 4.23: Transmitter structure

PAM6 with 4-bit quantization is also shown in Fig. 4.24 to show the advantage of this analog implementation over a digital implementation with 4-bit quantization.

As shown in Fig. 4.24, the performance of a digital implementation with 4-bit quantization is around 1 dB worse than the performance of the optimal implementation of 4LINE-PAM6. At the same time, this digital implementation needs much more circuitry than the analog one. This analog implementation needs only 17 comparators whereas the digital implementation needs four 4-bit analog to digital converters (60 comparators).

It is also useful to compare the complexity of this method with the complexity of the ordinary 4-PAM schemes. The ordinary 4-PAM scheme needs three comparators for each line, whereas the analog implementation of 4LINE-PAM6 requires four comparators and one transconductance amplifier for each line. Thus a small increase in the complexity results in a large performance improvement.

4.4.3 Circuit-Level Simulations

The receiver architecture for 4LINE-PAM6 method was designed and simulated in a $0.18\mu\text{m}$ CMOS technology with Spectre [58]. As shown in Fig. 4.21, the receiver architecture needs operational transconductance amplifiers (OTA) to convert the received signal from voltage to current. Simulation results show that 5-6 bit linearity is sufficient for this block. Since high-speed implementation is the primary concern here, a simple architecture that can work

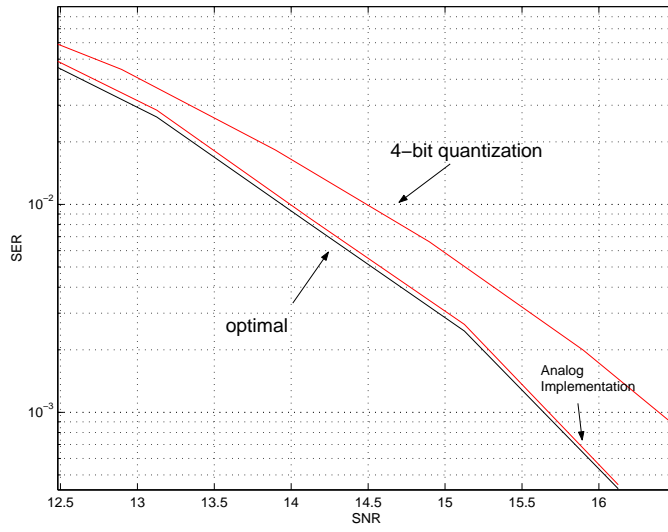


Figure 4.24: Simulation results for different implementations of the 4LINE-PAM6 scheme.

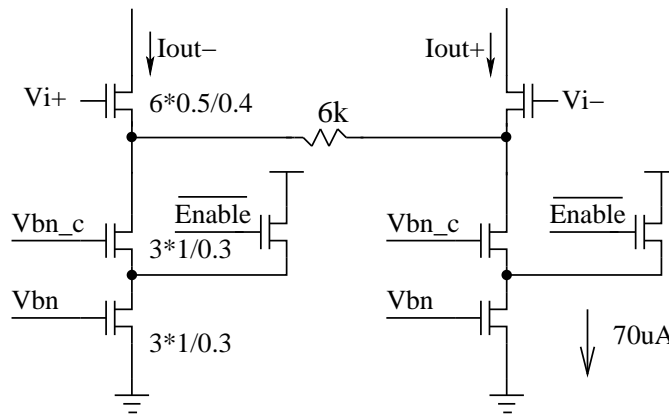


Figure 4.25: Schematic of the transconductance amplifier.

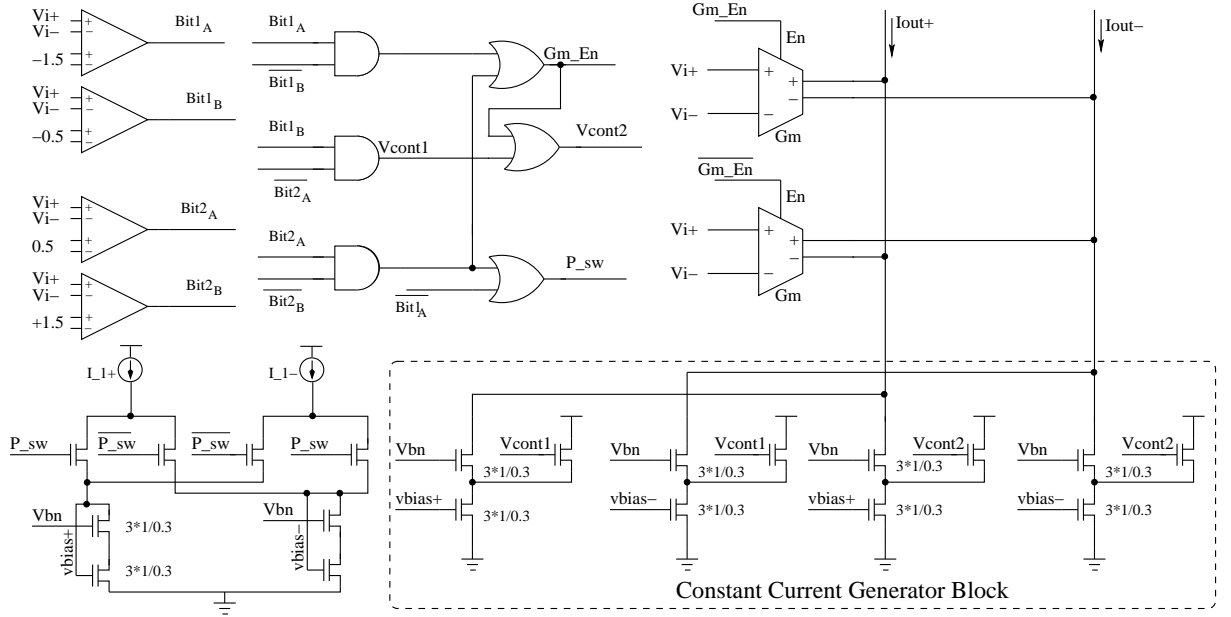


Figure 4.26: Detail of circuit implementation for receiver blocks (L1-L4 in Fig. 4.20).

at high-speed and satisfies the required linearity condition is chosen.

Fig. 4.25 shows the schematic of such an amplifier. As shown in this figure, the conventional architecture for a high-speed OTA is modified to have an extra feature. This new feature enables the OTA to be turned off by pulling up 'Enable' signal, which in turn steers the tail currents into two dummy branches and makes the output current equal to zero. This switching architecture not only increases the speed but also reduces the required voltage headroom of the conventional method for turning off the OTA, using a switch in series with the current mirrors. Simulation results for this block show that its linearity is roughly 6 bits.

This modified OTA is especially useful for high-speed implementation of 4LINE-PAM6 receiver. As illustrated in (4.10) the output of the front-end blocks in the receiver has two terms. A first term is proportional to the received signal (y'_i or $-y'_i$) and the other term is a constant value that depends on the received signal. Fig. 4.26 shows the circuit realization of the front-end block of the 4LINE-PAM6 receiver. As shown in this figure, two OTAs are used to generate currents proportional to y'_i and $-y'_i$. Based on the input signal, one of them is turned on and the other one is turned off. 'Constant Current Generator' block generates the constant term in (4.10). Instead of turning off unnecessary current sources,

they are steered into dummy branches to increase the speed of circuit. Simulation results show that the proposed circuit is functional up to 2 GS/s. Using an interleaving technique, it should be possible to speed up the circuit to 4 GS/s by means of two parallel circuits at 2 GS/s.

As stated before, 4LINE-PAM6 method can reduce the transmitted power by roughly 6 dB. Since 20 mA current is required for a differential signal swing of 1 V in a typical 4-PAM driver with 50 Ω source and load termination, the 4LINE-PAM6 method can save roughly 10 mA in the transmitter. Simulation results show that for the case of 2 GS/s the required supply current for the overhead circuitry in the 4LINE-PAM6 receiver is roughly 1mA. Therefore not only the transmitted power, but also the total power of the transceiver can be lowered by using the 4LINE-PAM6 scheme.

4.5 Summary

This chapter proposed several coding schemes for chip-to-chip applications. These coding schemes can be used as an attempt to approach the theoretical Shannon limit. The main contribution, here, is to propose coding scheme with low-complexity decoders. These coding schemes achieve roughly 3 dB coding gain in the case of an additive white Gaussian noise (AWGN) model for the channel. Moreover, a realistic model for the channel is developed that takes into account the effect of crosstalk, jitter, reflection, ISI, and AWGN. The proposed signaling schemes are significantly less sensitive to those noise sources. In particular, two coding schemes (3LINE-PAM2 and 4LINE-PAM6) that show better performance were highlighted and simulation results show that they provide a coding gain of 5-8 dB in the presence of jitter, ISI, and residual reflections. These methods are significantly less sensitive to crosstalk, which is the dominant noise in most of the microstrip interconnects. Finally the presented low-complexity architectures for analog implementations of 3LINE-PAM2 and 4LINE-PAM6 makes their high-speed implementations feasible. This was also confirmed by circuit-level simulation for the 4LINE-PAM6 receiver at 2 GS/s.

Chapter 5

A Power-Efficient 4-PAM Signaling Scheme for Inter-Chip Links Using Coding in Space

The proposed 4LINE-PAM6 scheme in Chapter 4 is not as effective for peak-power-limited applications. This chapter proposes a coding scheme that employs a convolutional encoder and a sequence detector. This scheme, which uses a 4-PAM modulation, provides 3-5 dB coding gain over the regular 4-PAM scheme without increasing the peak-power of the conventional 4-PAM scheme.

The proposed method is explained in Section 5.1. Section 5.2.1 compares simulation results of this method with those of a conventional 4-PAM scheme. Section 5.3 introduces different architectures for implementation of the proposed signaling scheme. A high-speed, low-complexity analog implementation of this method is explained in Section 5.4. Finally, a 4-PAM receiver for this signaling scheme along with a conventional 4-PAM receiver is designed and implemented in $0.18\mu m$ CMOS technology and Section 5.5 presents the experimental results for this chip.

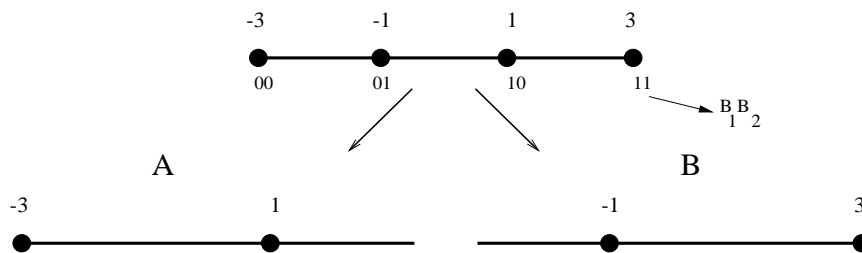


Figure 5.1: Set partitioning for a 4-point constellation

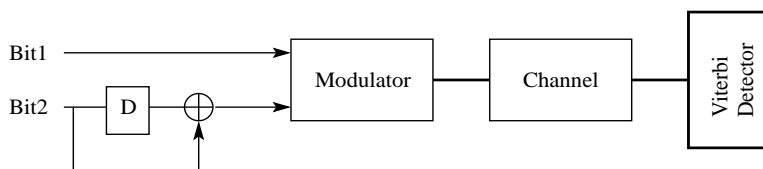


Figure 5.2: A possible coding scheme for inter-chip applications

5.1 Proposed Coding Scheme for Inter-Chip Communication

As mentioned in Chapter 4, the idea of coded-modulation can be exploited to introduce coding schemes with less complexity suitable for high-speed chip-to-chip communication. The proposed signaling scheme in this chapter is based on a coding scheme that is proposed in [24, pp. 680]. Although the idea is general and can be applied to different constellations, here, only a method for a 4-point constellation is explained.

Similar to coded-modulation, as shown in Fig 5.1, a four-point constellation can be partitioned into two subsets. The minimum distance of the points in each subset is twice of the one in the original 4-point constellation. Consider transmitting two bits per symbol, one bit to select the subset A or B, and the other bit to select the points within each subset. Obviously, decoding the bit that selects subset A and B is more important because the minimum distance of the points inside each subset is twice of the minimum distance of the original 4-PAM. Therefore, protecting the important bit with a coding scheme is desirable. A simple method is to use a simple convolutional coding scheme such as duobinary ($1 \oplus D$). Fig. 5.2 shows a block diagram of the transceiver based on this idea. As shown in this figure, a Viterbi detector can be used to decode the signal in the receiver.

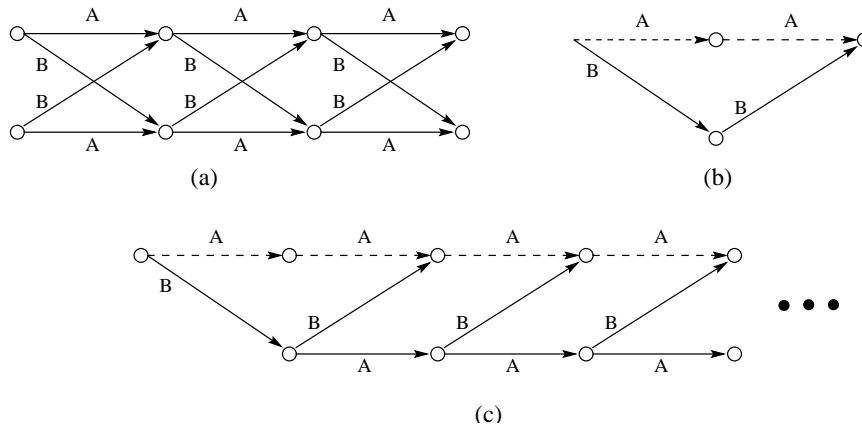


Figure 5.3: a. A trellis for the scheme of Fig. 5.2 b. A minimum-distance error event c. A set of minimum distance events. There are infinite number of minimum-distance events

The trellis of this scheme is shown in Fig. 5.3a and a minimum-distance error event is illustrated in Fig. 5.3b. In this event, the correct path is shown by the dashed line. With $d(a_n, a'_n)$ denoting the Euclidean distance (ED) between channel signals a_n, a'_n , it is desirable to maximize free ED,

$$d_{free} = \min_{a_n \neq a'_n} \left[\sum_n d^2(a_n, a'_n) \right]^{1/2}, \quad (5.1)$$

between all pairs of the possible channel-signal sequences $\{a_n\}$ and $\{a'_n\}$. It should be noted that if maximum-likelihood (ML) detector is used, the error-event probability will approach to the lower bound [23]:

$$Pr(e) > N(d_{free}) \cdot Q(d_{free}/2\sigma). \quad (5.2)$$

In this equation, $N(d_{free})$ is the number of error event with distance d_{free} , and $Q(\cdot)$ is the Gaussian error function. The d_{free} of the minimum distance error event is $\sqrt{2}$ times of the d_{free} of the uncoded system. If this is the only probable error event, the total gain would be about 3 dB. However, no coding gain should be expected since two source bits are represented with an alphabet of size four and there is no redundancy. Although the d_{free} of the system is increased, there are actually infinite number of minimum-distance error events (see Fig. 5.3(c)). This explains why this method does not provide any gain.

An interesting way to achieve an appreciable coding gain is to force the two-state trellis to return to state zero at every fourth symbols [24, pp.680], as shown in Fig. 5.4. Note that in the fourth symbol interval the coder does not have a choice of set A or B. Thus, only one

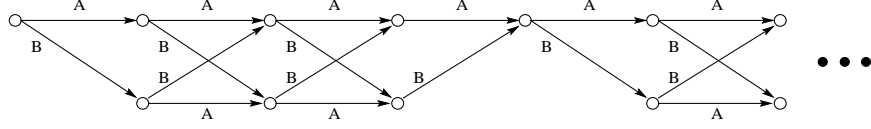


Figure 5.4: Modified trellis to force the state to return to the zero state every fourth symbol bit instead of two can be transmitted in the fourth symbol interval. Now, there are at most three error events with the minimum distance starting at any given time, so the probability of error event can be approximated by $3Q(d_{free}/2\sigma)$. Here, d_{free} is $\sqrt{2}$ times of the d_{free} of uncoded system and nearly the full 3 dB improvement can be realized at high SNRs.

As shown in Fig. 5.2, the state of the trellis at each time step depends only on the second input bit, Bit2, and the first bit, Bit1, can be either zero or one in each branch. This means that there are actually two parallel branches between every two states of the trellis. Fig. 5.5 shows the actual trellis for one time-step. Fig. 5.5b shows the branch metric corresponding to each branch. It can be shown that these branch metrics can be simplified to those in Fig. 5.5c. In each branch metric, y represents the received signal. The two branches between every two states can be merged by taking the minimum of the two branch metrics. Therefore, the branch metrics can be calculated by

$$A = B11 = B00 = \begin{cases} 2y + 2 & \text{if } y < -1, \\ 0 & \text{if } y > -1, \end{cases} \quad (5.3)$$

and

$$B = B01 = B10 = \begin{cases} y & \text{if } y < 1, \\ -y + 2 & \text{if } y > 1. \end{cases} \quad (5.4)$$

The proposed method needs one Viterbi decoder for each line in the bus. This increases the complexity of the receiver. To alleviate this problem and to reduce the latency of the decoder, the sequence can be transmitted in space over a bus rather than sequentially in time. The Viterbi algorithm can also be applied in space. This idea proposes a new scheme for chip-to-chip communication. In this scheme, hereafter referred to as 4LINE-PAM4 scheme, seven input bits are converted to eight bits by means of a convolutional encoder in space as shown in the Fig. 5.6a. Therefore, the rate of this coding scheme is 7/8. These eight bits

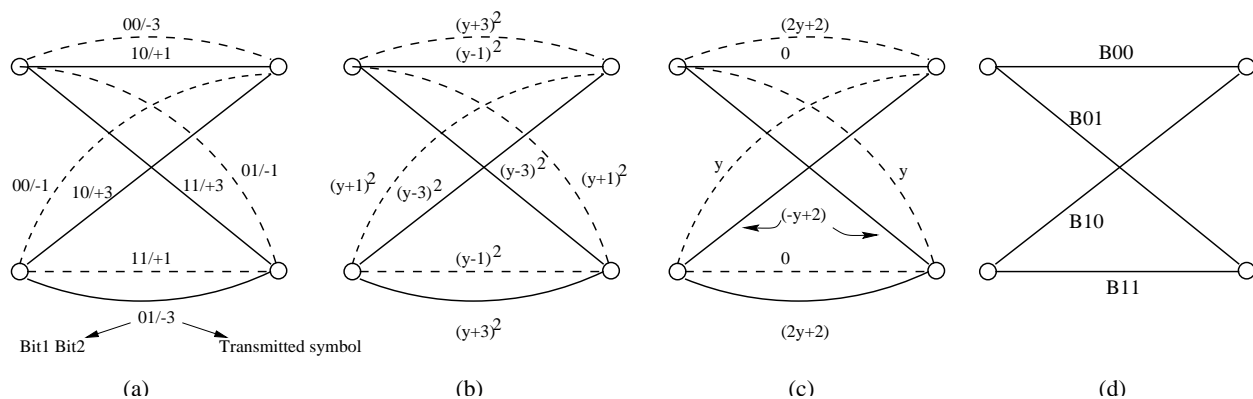


Figure 5.5: a. Trellis for one time-step, b. The branch metrics corresponding to each branch, c. Simplified branch metrics, d. General form of the trellis for one time-step

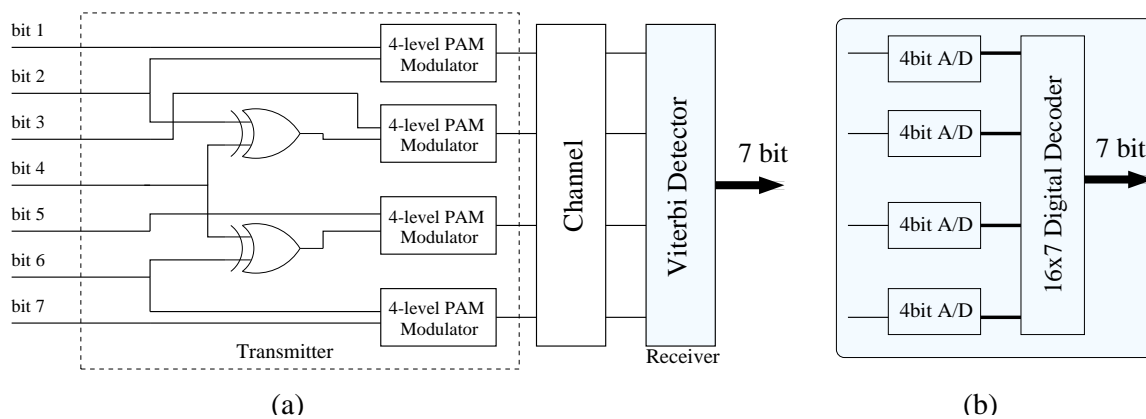


Figure 5.6: a. Block diagram of transmitter and the receiver using Viterbi decoder in space, b. Digital implementation of the Viterbi detector

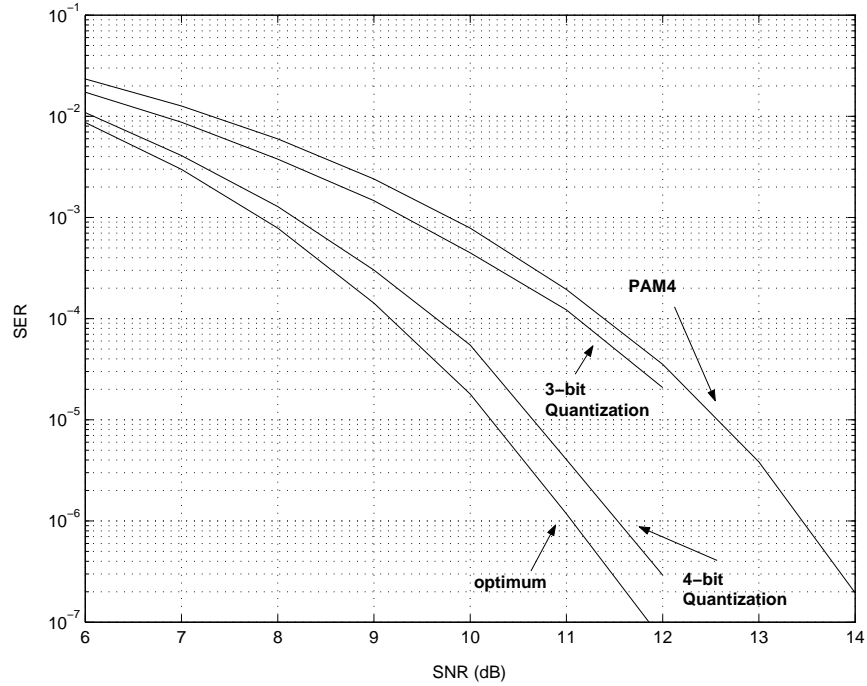


Figure 5.7: Performance comparison of different schemes.

can be transmitted over four lines in a bus using a 4-PAM modulation. A Viterbi decoder can be used for decoding the original seven bits from the received signal.

5.2 Simulation Results

The performance improvement of the proposed 4LINE-PAM4 scheme compared to the conventional 4-PAM method has been verified by simulations with two different channel models.

5.2.1 Simulation Results with an AWGN Channel Model

With an additive white Gaussian noise (AWGN) model for the channel, the 4LINE-PAM4 scheme and the ordinary 4-PAM scheme were simulated in Matlab. Simulation results show that the performance of 4LINE-PAM4 scheme is roughly 2.6 dB better than the performance of the 4-PAM at symbol error rate (SER) of 10^{-6} (see Fig. 5.7).

5.2.2 Simulation Results with a More Realistic Channel Model

AWGN model is a good model for a channel with perfect termination and no ISI. However, the main sources of the noise for inter-chip application in practical systems are usually ISI and residual reflection due to the imperfect termination. Therefore, a more realistic model for the channel, which is similar to the one in Section 4.2, is used to verify the performance of the proposed method in two practical situations.

Case I: ($Z_l = Z_s = 55\Omega$, $Z_0 = 50\Omega$, $d = 0.3m$)

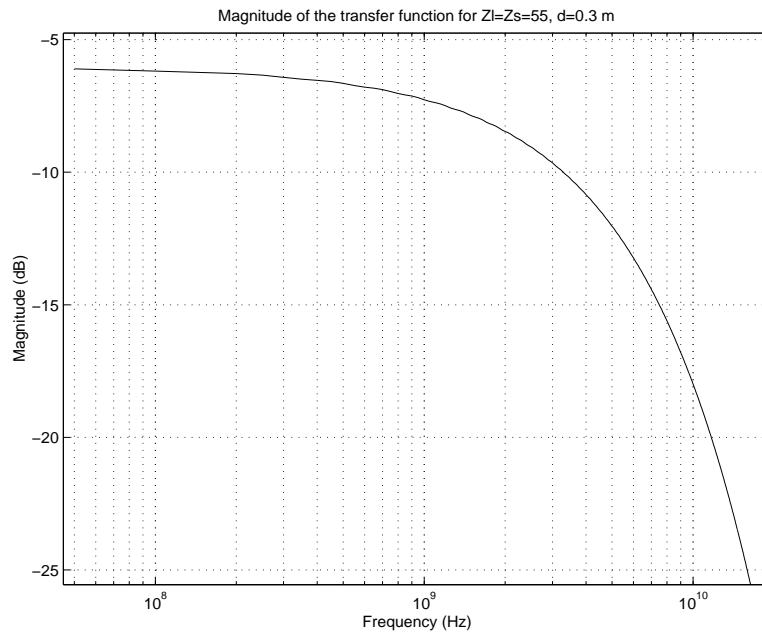
In this case, source and load terminations are %10 more than the characteristic impedance ($Z_0 = 50\Omega$). A 0.3-m microstrip is considered as a medium for this channel. Fig. 5.8a shows the magnitude of the channel transfer function for this channel. As shown in this figure, the attenuation of the channel is roughly 4 dB at 2.5 GHz and therefore this channel introduces moderate amount of ISI. However, there would be a small residual reflection since the source and load impedances are close to 50Ω (perfect termination). Fig. 5.8b, which illustrates the SER variation with SNR for 4LINE-PAM4 and 4-PAM schemes, shows roughly 5 dB gain for the 4LINE-PAM4 scheme.

Case II: ($Z_l = \infty$, $Z_s = 55\Omega$, $Z_0 = 50\Omega$, $d = 0.2m$)

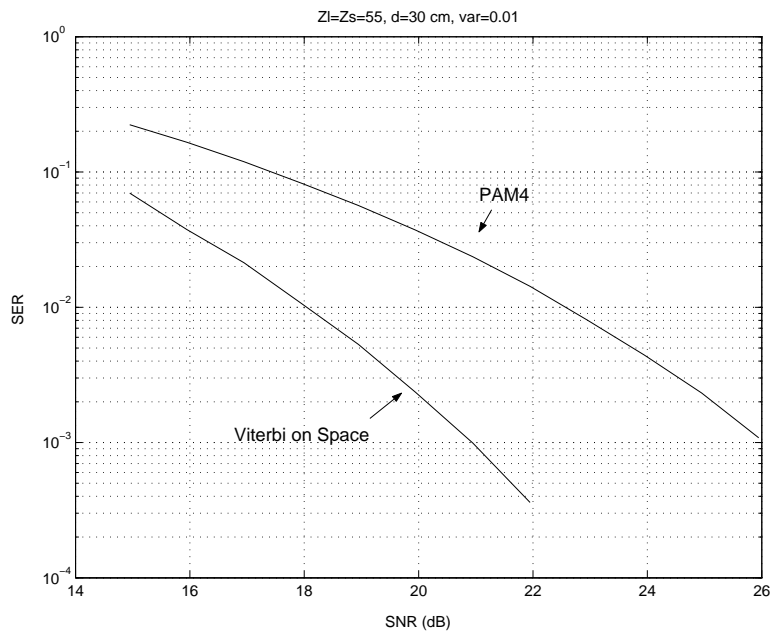
Here, a 0.20-m microstrip with a 55Ω source termination ($Z_s = 55\Omega$) and infinite load impedance ($Z_l = \infty$) is considered for the channel. Fig. 5.9a illustrates less than 3 dB attenuation for this channel at 2.5 GHz. Therefore, the amount of ISI in this case is less than that of the previous case. However, it introduces more reflection because of the poor return-loss of the channel at the receiver. Fig. 5.9b shows the performance of both 4LINE-PAM4 and 4-PAM schemes in this case. Again the 4LINE-PAM4 scheme shows roughly 4.5 dB performance improvement over 4-PAM at SER around 10^{-3} .

5.3 Different Implementations for the Proposed Method

The complexity and latency of the Viterbi decoder make the use of that restricted to moderate-speed link in the order of several hundreds of *MHz* [44]. For high-speed ap-

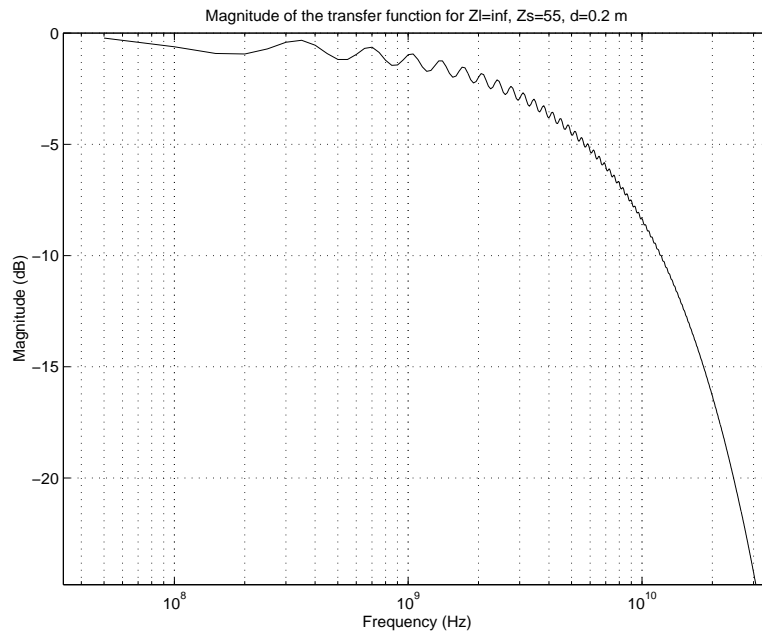


(a) Channel frequency response

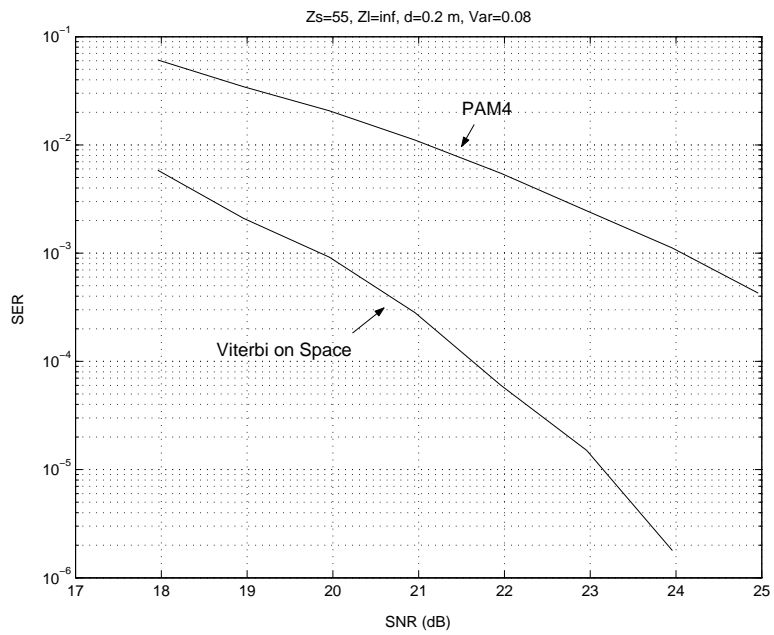


(b) SER versus SNR

Figure 5.8: Case I: $Z_L = Z_S = 55\Omega$, $d = 0.3$ m.



(a) Channel frequency response



(b) SER versus SNR

Figure 5.9: Case II: $Z_S = 55\Omega$ $Z_L = \infty$, $d = 0.2$ m.

plications, pipeline techniques can be used, which further increases the receiver complexity.

An alternative method for the implementation of the Viterbi algorithm in the receiver is to use ADCs at the front-end of the receiver and implement the algorithm digitally. Fig. 5.7 shows the performance of the Viterbi decoder with 3-bit and 4-bit quantization along with that of the optimal implementation. As shown in this figure, the performance of the Viterbi decoder with 4-bit quantization is roughly 0.6 dB worse than the optimal implementation. Therefore, a possible scheme is to use a 4-bit ADC for each line (see Fig. 5.6b). This results in 16 bits since there are four lines in the architecture. A digital 16×7 decoder can be used to retrieve the original data. However, a digital implementation needs an ADC for each line and therefore its complexity makes it unsuitable for most applications.

As an alternative, a tree representation can be used instead of the trellis. The trellis of this scheme has to go back to state zero at every fourth symbol (Fig. 5.3). Therefore, there are only 8 different paths in the trellis. This is shown in Fig. 5.10. Here, the problem is to find the minimum state metrics among the leaves in the tree. Obtaining branch metrics would be straightforward in current-mode circuitry using (5.3) and (5.4).

Fig. 5.11 shows a simple method for the receiver architecture of this signaling scheme. Each compare select unit (CSU) compares its two inputs and the output of this block would be the minimum of the two inputs. Although this method is simple, it cannot be used at high speed due to the time that each CSU needs to generate the minimum of its two inputs at its output. Moreover, this method also suffers from error accumulation.

A winner-take-all (WTA) circuitry can be used instead of the minimum-finder block in Fig. 5.11 as shown in Fig. 5.12 [59]. Ideally, all output currents except the one that corresponds to the maximum input current would be zero. Therefore, the maximum/minimum state metric can be found easily with this method. However, HSPICE simulation results show that the resolution of this circuit is not sufficient for this method at high speed (larger than 1Gb/s) in an advanced technology with a small power supply. An improved method that can be implemented at higher data rates is introduced in the next section.

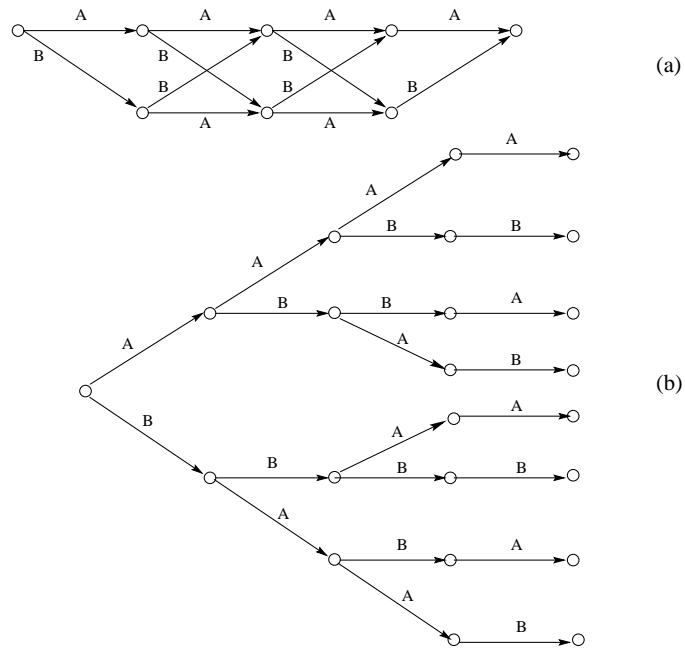


Figure 5.10: Using a tree instead of trellis for decoding the proposed scheme

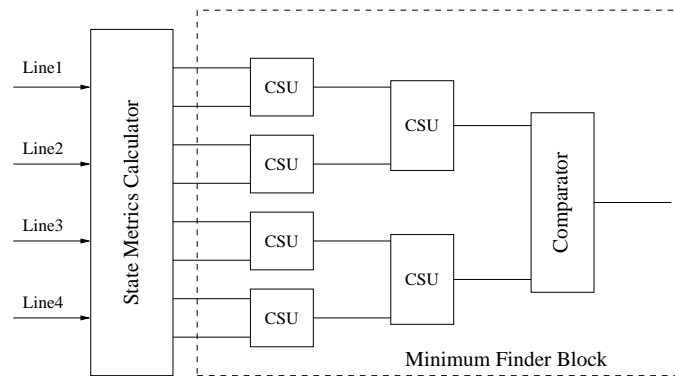


Figure 5.11: Block diagram of the receiver

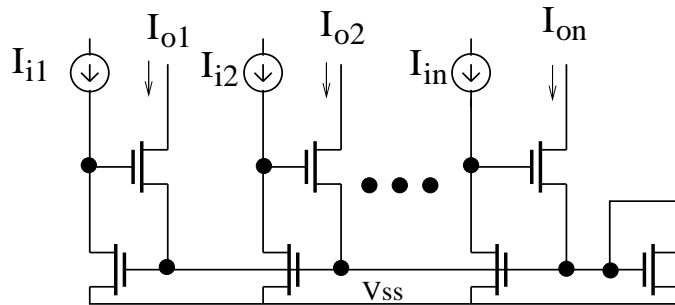


Figure 5.12: The current maximum selector circuitry

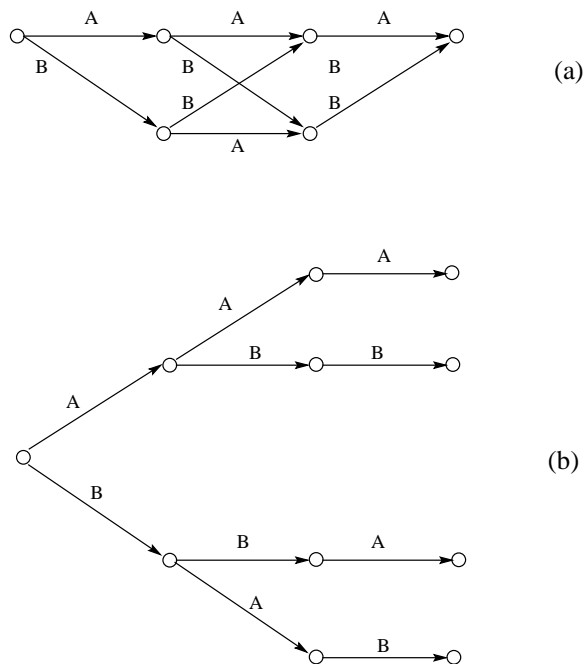


Figure 5.13: Using a tree instead of trellis for decoding the proposed scheme with rate=5/6

5.4 Analog Implementation

If a flash-type method, comparing every couple of state metrics, is used as the minimum-finder block in Fig. 5.11, 28 comparators are needed. This means a large circuit overhead for this method compared to the uncoded 4-PAM scheme. As mentioned earlier, the rate of the 4LINE-PAM4 scheme is $7/8 = 0.87$. Interestingly, by forcing the trellis to go back to state zero at the third symbol instead of the fourth, the number of leaves in the tree reduces to four, see Fig. 5.13. Therefore, the flash-type method needs only six comparators. This means that by reducing the rate from $7/8$ to $5/6$ (less than 5% reduction), the complexity of the receiver can be reduced significantly. This new method, 3LINE-PAM4 can also provide a better coding gain especially at low SNRs. The reason is that in the new trellis (see Fig. 5.13a), there are only two minimum distance error events and therefore the probability of error is roughly $2Q(d_{free}/2\sigma)$, which is a factor of $2/3$ smaller than the probability of error for 4LINE-PAM4 scheme with rate $7/8$. This has been verified by simulating both methods for the case of an AWGN channel (see Fig. 5.14).

Fig. 5.15 shows the transceiver architecture of the 3LINE-PAM4 scheme, which is implemented in this work. As shown in this figure, 3LINE-PAM4 encoding needs only one *XOR*

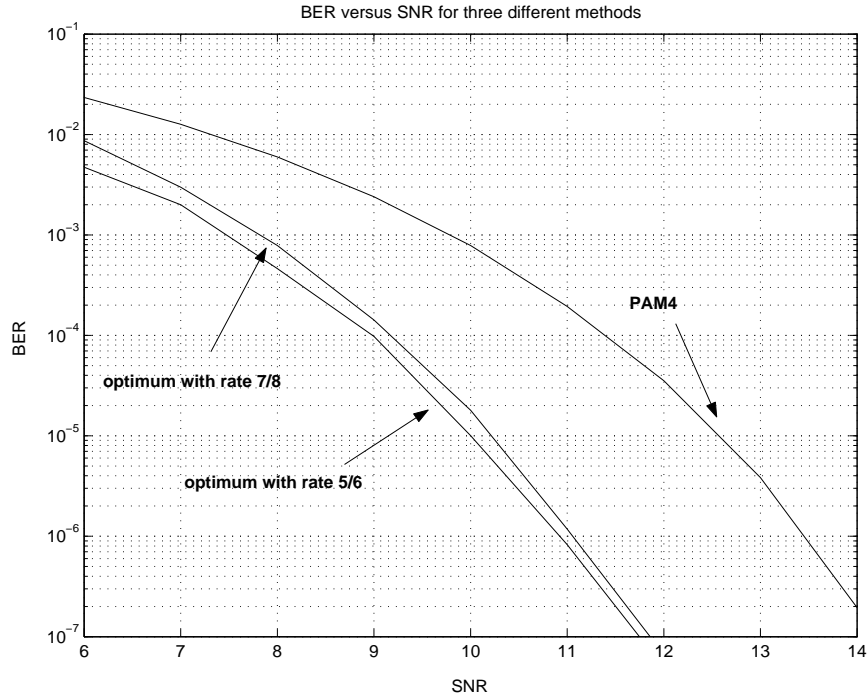


Figure 5.14: Performance comparison for 4-PAM, 3LINE-PAM4 and the 4LINE-PAM4 scheme

gate and the transmitter is essentially the same as the transmitter of a conventional 4-PAM scheme.

The receiver architecture for this scheme is shown in Fig. 5.16. The state-metric calculator unit computes the costs for all four paths in the trellis or equivalently for all four leaves in the tree of Fig. 5.13. As shown in equations (5.3) and (5.4), calculating the two branch metrics corresponding to each input line needs two comparators. Therefore, there would be a total of six comparators in the state-metric-calculator unit. The detail of this unit is shown in Fig. 5.16b.

The original five bits can be retrieved from the output of the comparators in the state-metric calculator unit along with the output of the comparators in the second stage of the receiver. These 12 outputs are fed into a 12 by 5 digital decoder that decodes the original bits in the receiver. Fig. 5.17 shows the required circuitry for this decoder.

Fig. 5.18 illustrates the circuit implementation for the branch-metric-calculator unit, which is a direct implementation of (5.3) and (5.4). As shown in this figure, differential pairs are used instead of switches to increase the speed of this implementation. Also, four

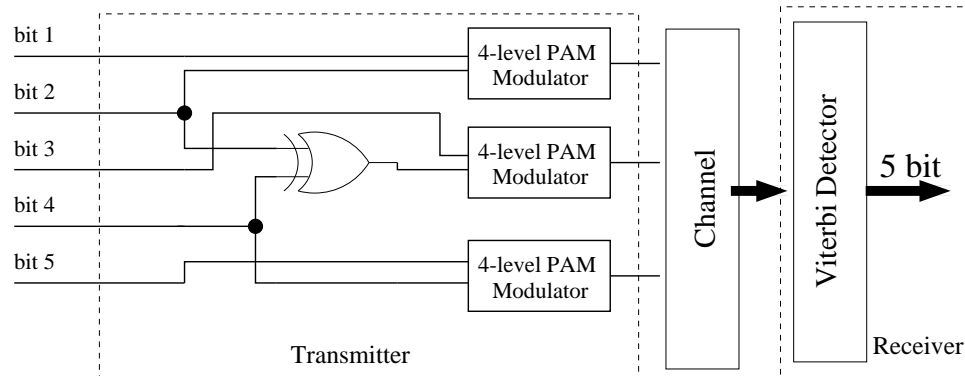


Figure 5.15: Block diagram of the transceiver.

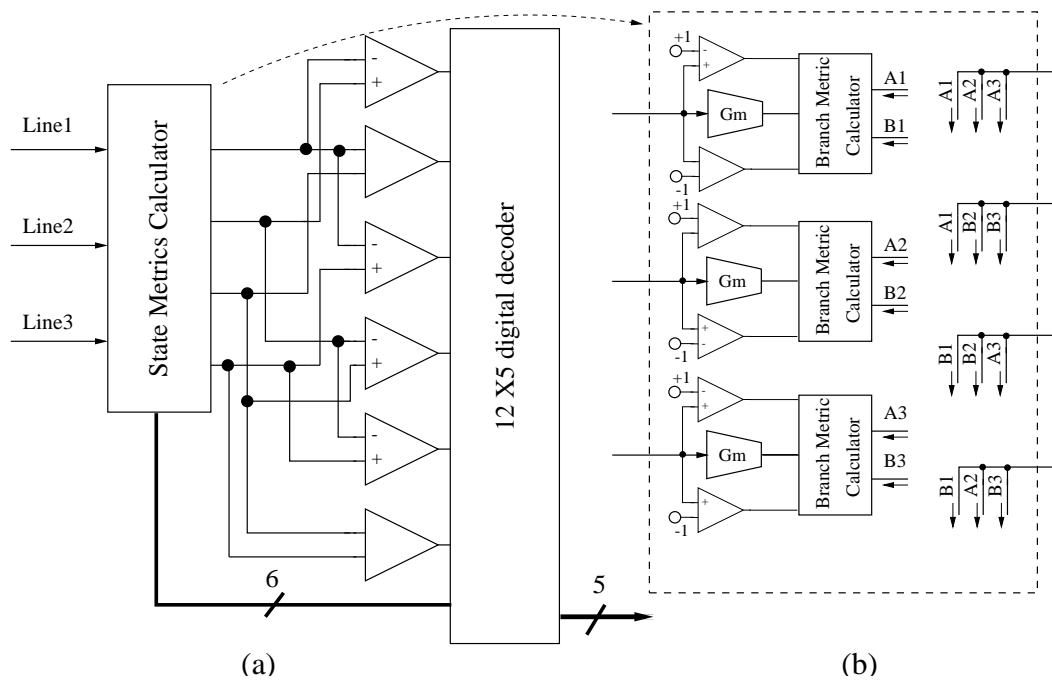


Figure 5.16: a. Block diagram of the receiver, b. Detail of the state-metric calculator unit.

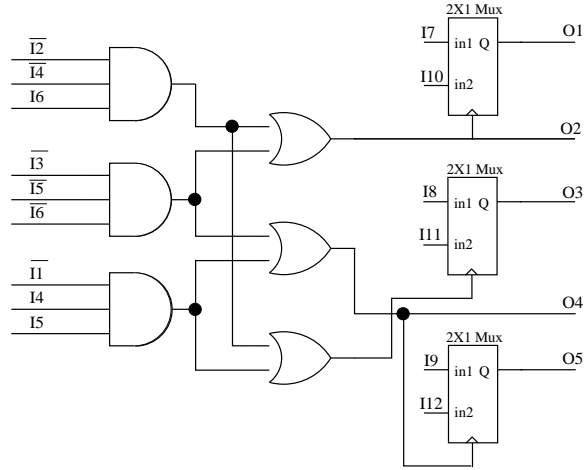


Figure 5.17: Detail of the 12X5 digital decoder.

transconductance amplifiers, instead of one, are used for calculating each branch metric to eliminate the need for current mirrors and thus the additional delay in the signal path.

Current 'I₂₊' and 'I₂₋' in this simplified schematic are used for generating the constant terms in (5.3) and (5.4). The state metric for each leaf in the tree (Fig. 5.13) can be calculated by adding the three branch metrics for each path. This is straightforward in current-mode circuitry and can be performed by just connecting the outputs of branch-metric calculator units.

As shown in Fig. 5.19, a simple architecture is selected for a high-speed implementation of the transconductance block [60]. Simulation results for this block show that its linearity is roughly 6 bits.

Another important unit in this design is the high-speed comparator block. The simplified schematic of the comparator is shown in Fig. 5.20. The comparator is a slight variation of the comparators in [61], [60]. This comparator comprises of a preamplifier and a latch. During the track time, preamplifier amplifies the input signal. The amplified voltage is then latched at the rising edge of the clock ('Latch' signal in the schematic). A preamplifier is used to reduce the input referred offset of the comparator. Monte-Carlo simulations shows that the comparator offset is roughly 35 mV. In order to remove the precharge phase from the output and to present a constant (non-data dependent) load to the comparator, the comparator is followed by an SR-latch [38].

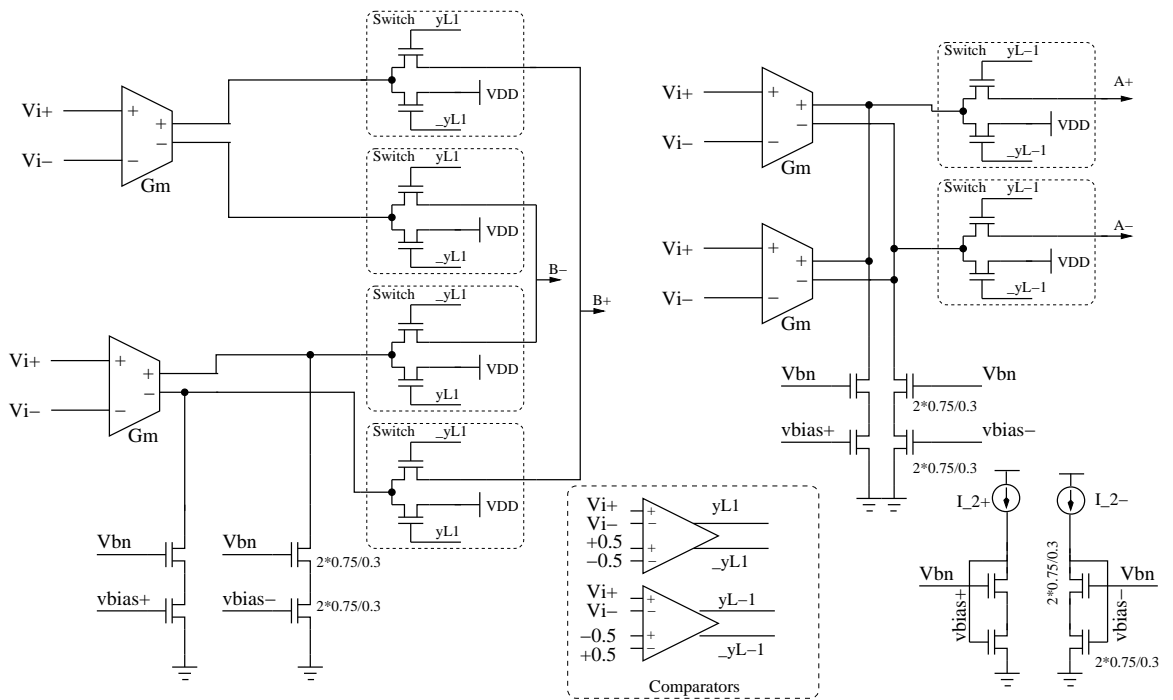


Figure 5.18: Branch metric calculator schematic.

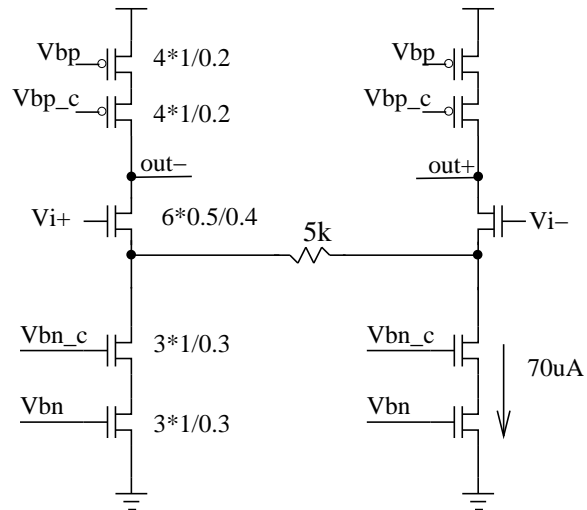


Figure 5.19: Detail of the transconductance amplifier.

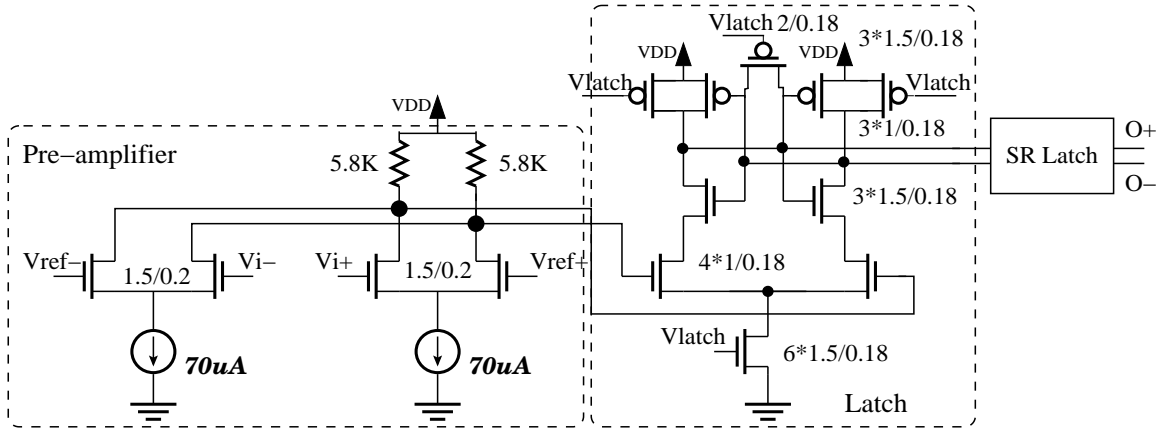


Figure 5.20: Simplified schematic of a comparator.

The 3LINE-PAM4 scheme and a conventional 4-PAM receiver are designed in a $0.18\mu m$ standard CMOS technology. The functionality of the 3LINE-PAM4 scheme has been verified by HSPICE simulation. Simulation results show that the implemented 3LINE-PAM4 receiver can achieve 5 Gb/s data rate. At 5Gb/s the 3LINE-PAM4 receiver draws 14.5mA for decoding 5 bits (2.9mA/bit). Whereas the regular 4-PAM receiver draws 3.6mA for decoding 2 bits (1.8mA/bit). Therefore, the 3LINE-PAM4 method requires 1.1mA more power-supply current per bit compared to the conventional 4-PAM method.

As stated before, 3LINE-PAM4 method can reduce the transmitted power by 3-5 dB. Since 20 mA current is required for a differential signal swing of 1V p-p in each channel of a typical 4-PAM driver with $50\ \Omega$ source and load termination, this method can save roughly 6 mA per channel in the transmitter. Therefore not only the transmitted power, but also the total power of the transceiver can be lowered by using the proposed 3LINE-PAM4 scheme.

5.5 Experimental Results

The receiver for the 3LINE-PAM4 scheme along with a regular 4-PAM receiver was implemented and fabricated in a $0.18\mu m$ standard CMOS technology. Fig. 5.21 shows the $1.88mm \times 1.18mm$ chip micrograph. Careful layout techniques has been used to reduce the skew between different channels in the 3LINE-PAM4 receiver. The design was pad-limited and the active area of the regular 4-PAM receiver is $0.026\ mm^2$ whereas the active area of

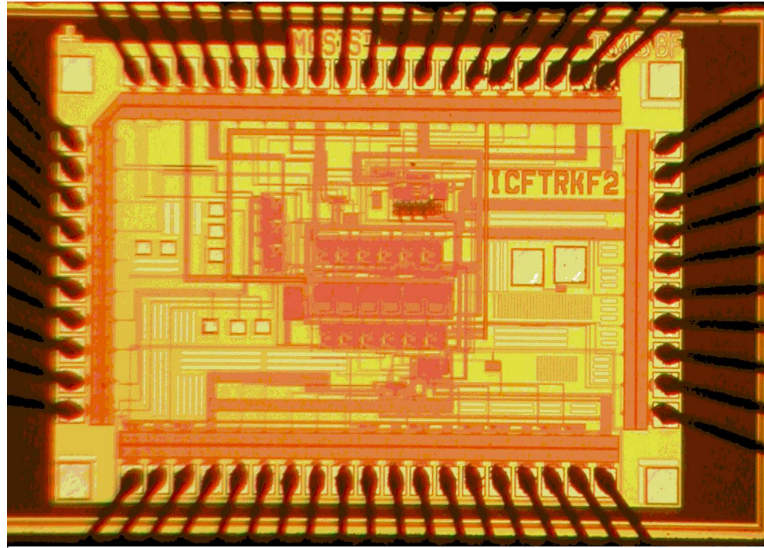


Figure 5.21: Chip micrograph.

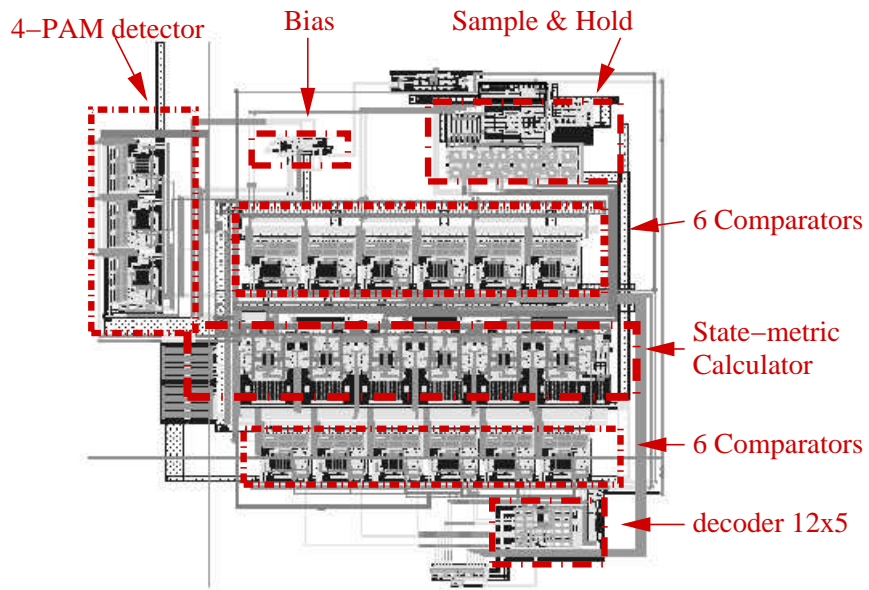


Figure 5.22: Layout of the active area of the chip.

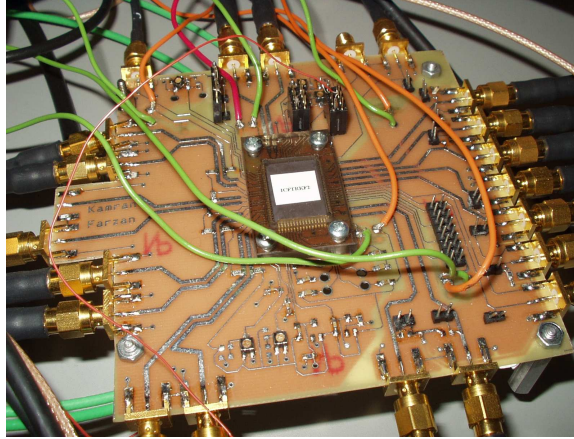


Figure 5.23: PCB test fixture used for obtaining the experimental results.

the 3LINE-PAM4 receiver is 0.2 mm^2 . Fig. 5.22, which depicts the chip layout, can be used to compare the required area for each building block of the receiver. Since the 3LINE-PAM4 receiver decodes 5 bits simultaneously the per-bit area overhead of this method is 0.027 mm^2 .

An 80-pin ceramic flat package (CFP80) is used for packaging of this chip. A printed circuit board test fixture is designed and fabricated for testing this chip. Fig. 5.23 shows this test fixture along with the fabricated chip. This board has two layer and high-speed signal traces are designed to have 50Ω characteristic impedance. PCB design methods are used to minimize the skew between different input channels of the 3LINE-PAM4 receiver.

A LabVIEW virtual instrument (VI) was developed for low-frequency measurement [62] (see Fig. 5.24). This VI generates the random data and 4-PAM signals at a rate of 1 KS/s for each channel. In addition, it adds AWGN to the generated signal of each channel. This VI uses National Instrument (NI) data acquisition cards with 12-bit digital to analog converters (DACs) to generate the required signals for the measurement. Digital input cards are used to capture the outputs of the chip. This VI also compares the transmitted and received data to measure the BER for both methods. Fig. 5.25 shows the experimental results for BER-versus-SNR measurements using this VI. As shown in Fig. 5.25, the performance of the 3LINE-PAM4 scheme is roughly 2.3 dB better than that of the regular 4-PAM receiver, which is similar to the simulation results for an AWGN channel. This can be expected since at low-speed an AWGN model is a good approximation for the channel.

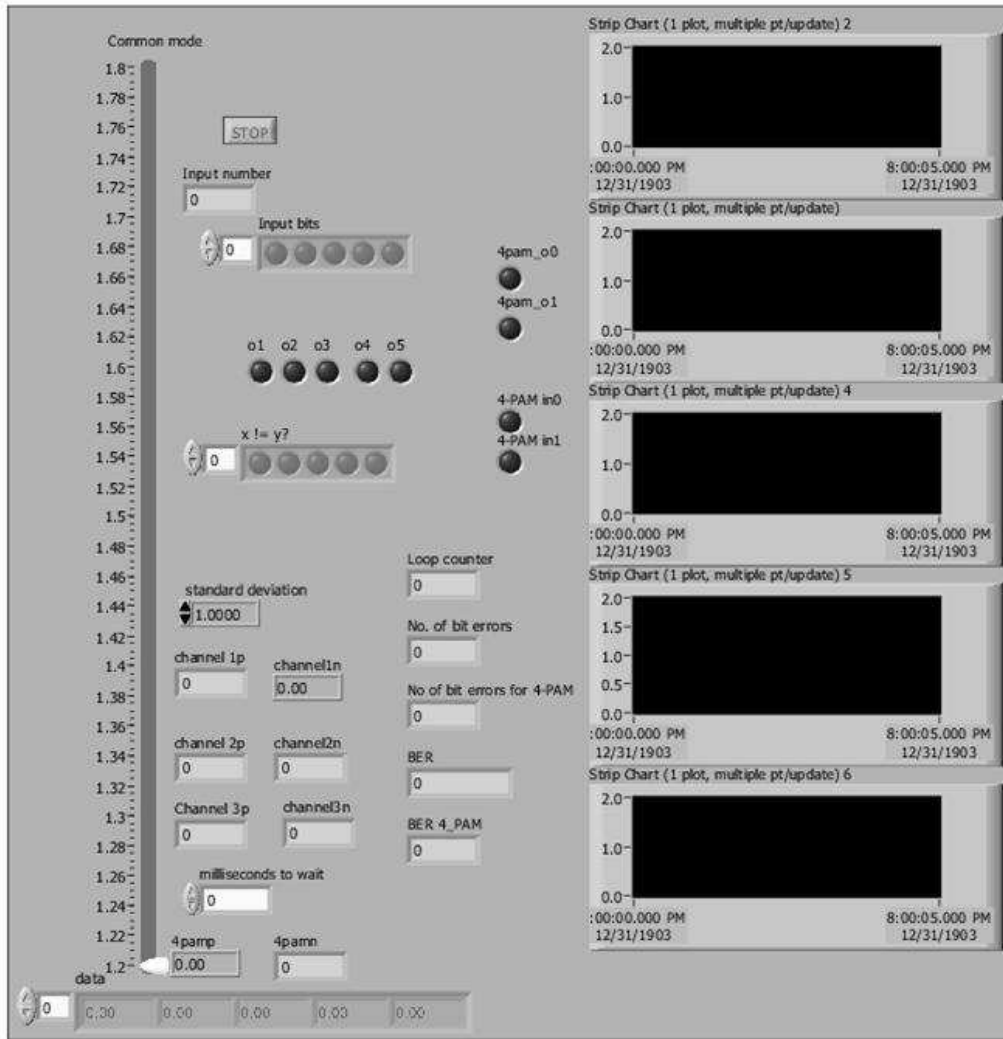


Figure 5.24: Virtual instrument developed in LabVIEW.

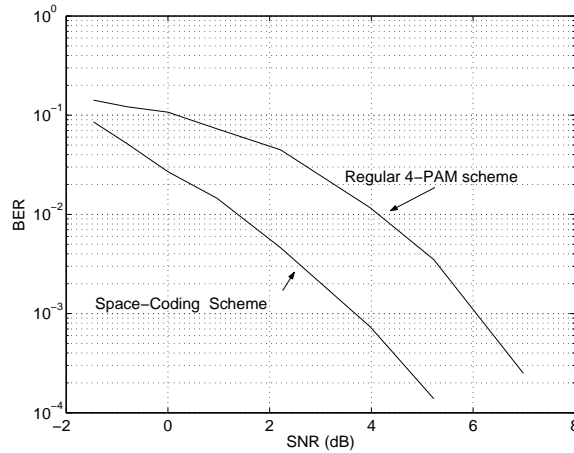


Figure 5.25: Measured BER versus SNR for two methods.

The functionality of 3LINE-PAM4 scheme at high-speed has been verified by a Parallel Bit Error Ratio Tester (ParBERT). Fig. 5.26 shows the test setup for this experiment. Each output of the ParBERT has only two levels and a combiner is used for generating each 4-PAM signal. Fig. 5.27 shows the output of the combiner for 340 MS/s. Experimental results show that the receiver is functional up to 2.5Gb/s. This is achieved without any on-chip skew-cancellation circuitry and higher speed can be achieved with on-chip skew-cancellation. Fig. 5.28, which depicts the output of the non-ideal combiner at 1GS/s, shows significant degradation in the eye at this speed. This is the main speed-limiting factor in this measurement.

Ideally, we need a high-speed arbitrary waveform generator with four channels in order to generate high-speed 4-PAM signals superimposed by additive white Gaussian noise for BER measurement of this chip. Unfortunately due to the lack of the required equipment, the BER measurement was not feasible at high speed. However, experimental results at 900 Mb/s show that the 3LINE-PAM4 scheme needs a smaller SNR for a given performance. Reducing the signal swing by 10% and 20% results in BERs of $6 * 10^{-7}$ and $4.2 * 10^{-2}$, respectively, for 4-PAM and BERs of $< 10^{-12}$ and $8 * 10^{-10}$ for 3LINE-PAM4. Finally, experimental results show that the 3LINE-PAM4 receiver draws roughly 12mA from 1.8V power supply at 2.5Gb/s.

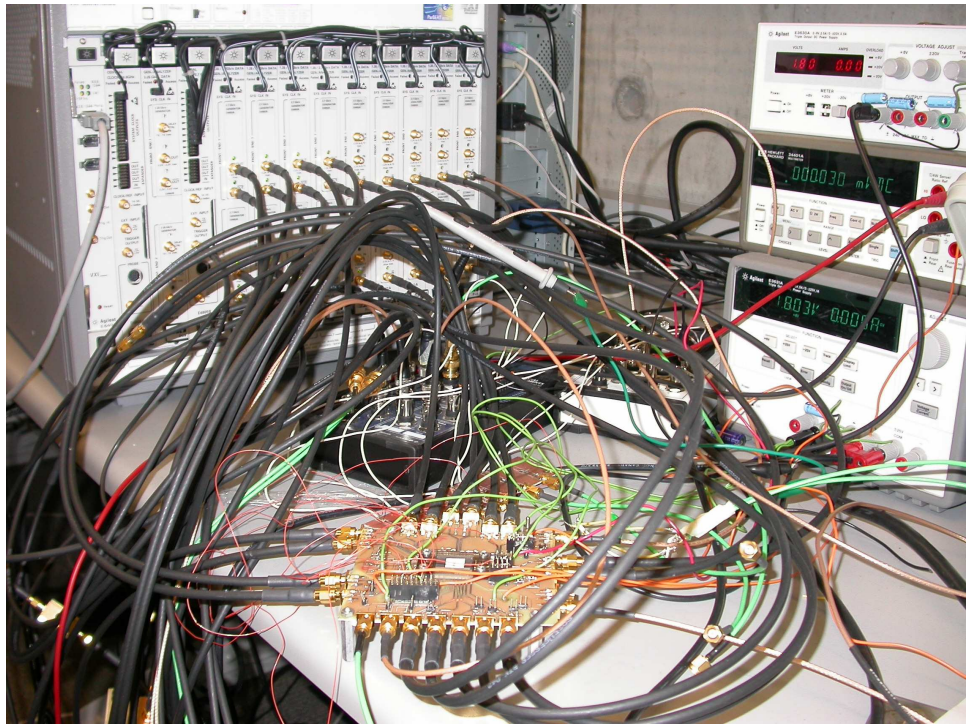


Figure 5.26: Test setup for high-speed measurement.

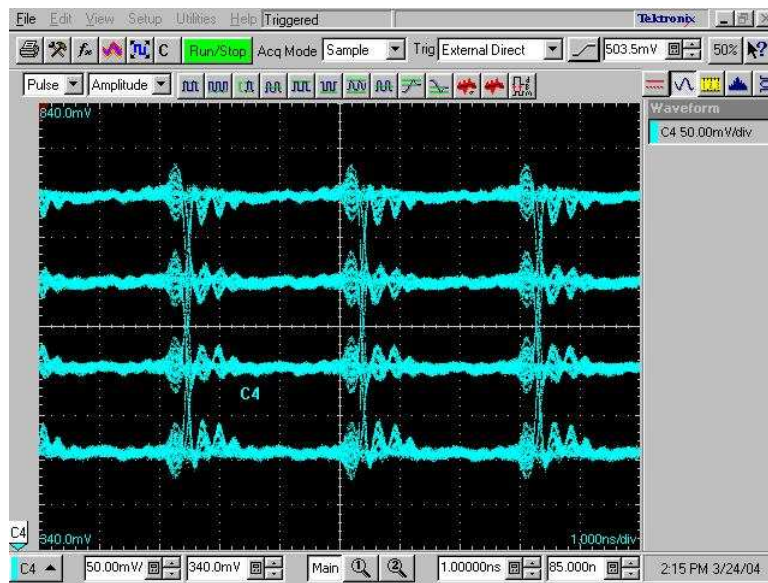


Figure 5.27: Output of the ParBERT at 340 MS/s.

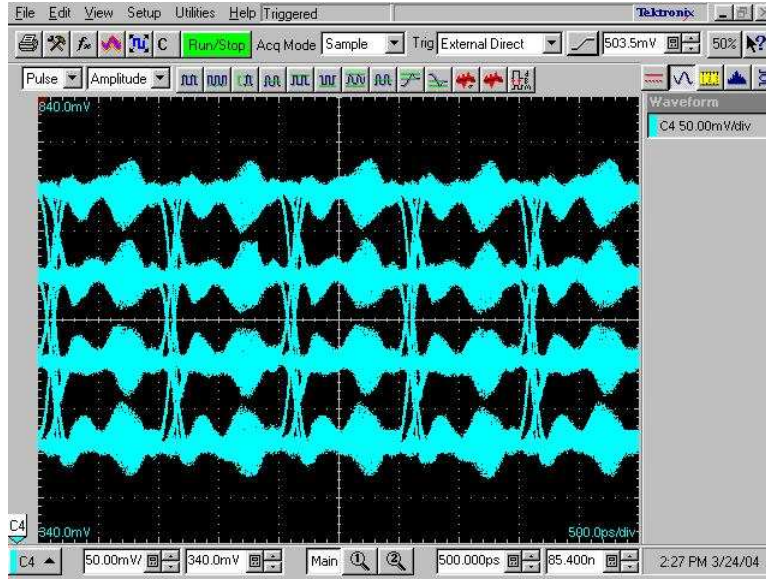


Figure 5.28: Output of the ParBERT at 1 GS/s.

5.6 Summary

Multi-level signaling can be used to reduce the number of signal paths or to increase the data rate. Two coding schemes, 4LINE-PAM4 and 3LINE-PAM4, that can reduce the transmitted power of multi-level signaling have been introduced. These coding schemes have roughly 3-5 dB coding gain over the uncoded 4-level PAM. Moreover, analog implementations for these methods have been presented. The proposed low-complexity analog implementation of 3LINE-PAM4 makes its high-speed implementation feasible. This 3LINE-PAM4 scheme is designed and implemented in a $0.18\mu\text{m}$ standard CMOS technology. This coding scheme transmits 5 bits over 3 differential lines of 4-PAM resulting in a $5/6$ code rate. Experimental results not only verified the coding gain at low-speed, but also showed that the chip is functional up to 2.5Gb/s. The measured supply current for the proposed method was 12.5mA at 2.5Gb/s and the active area of the implemented receiver based on this method was only 0.2 mm^2 .

Chapter 6

A CMOS 10-Gb/s Power-Efficient 4-PAM Transmitter

The potential benefits of 4-PAM signaling for increasing data rates in physical short-bus systems have been shown in [29–32]. However, transmitted power is often increased to compensate for the impact of multi-level signaling on bit error rate (BER). Since there are several drivers in a parallel bus signaling system, the power dissipation of each driver is extremely important and therefore power-efficient drivers are desirable.

Different architectures for drivers in general are explained in Section 6.1. Section 6.2 explains the proposed power-efficient architecture. Section 6.3 presents simulation results. The transmitter architecture is described in Section 6.4, and finally, the experimental results are presented in Section 6.5.

6.1 Driver Architectures

As shown in Fig. 6.1, there are generally two different architectures for drivers: unipolar and bipolar [34]. In unipolar architectures, current I is steered either in the right or left transistor in the current-steering differential pair. In this case, the single-ended output voltages would be either V_{DD} or $V_{DD} - RI$. Therefore, the differential swing is RI . However, in bipolar drivers (Fig. 6.1b), the output voltages would be either $V_{DD}/2 + RI$ or $V_{DD}/2 - RI$, which makes the differential swing equal to $2RI$. Since the power dissipation in both architectures

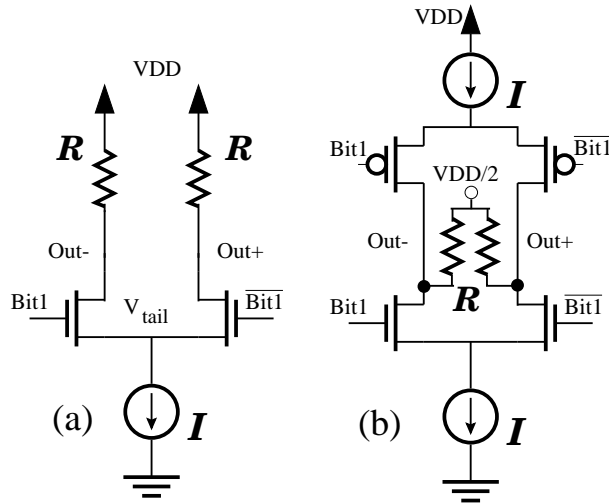


Figure 6.1: Driver architectures: (a) Unipolar, (b) Bipolar

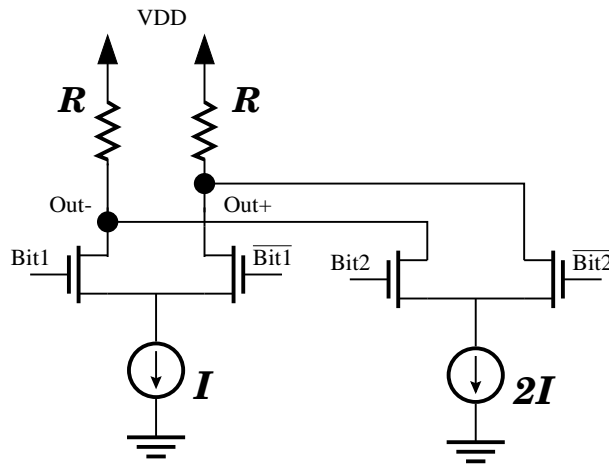


Figure 6.2: A common 4-PAM driver architecture

is practically the same, $V_{DD} \cdot I$, the bipolar architecture needs half the power of unipolar architectures for a given swing.

Note that since the top and bottom tail currents are equal, the power dissipation of the power supply $V_{DD}/2$ is ideally zero. In practice however, there might be some mismatches between the two current sources. In this design, replica bias is used to ensure the top and the bottom tail current sources remain equal over process, voltage and temperature (PVT) variation. This ensures negligible power dissipation in the power supply $V_{DD}/2$. Since the supplied current from this source is small, it is generated on-chip.

There is another source of power-inefficiency in typical multi-level PAM drivers. As

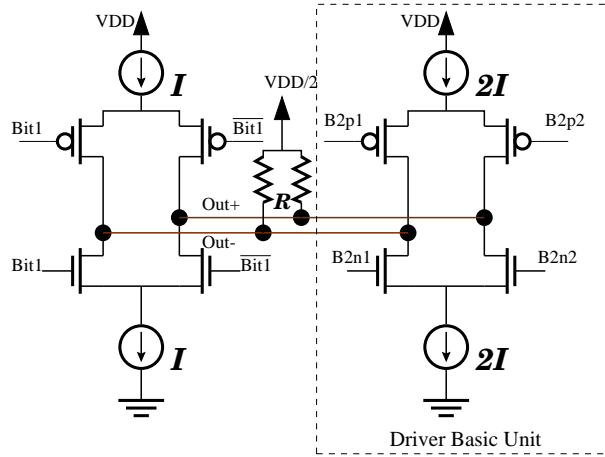


Figure 6.3: Basic architecture of the power-efficient 4-PAM Driver

shown in Fig. 6.2, the tail current sources of a typical 4-PAM driver are always on and output differential levels ($\pm RI, \pm 3RI$) are obtained by steering current in different branches. For example, level $+RI$ is generated when *Bit1* is '0' and *Bit2* is '1'. This means that the power dissipation would be the same for all four levels. However, the right current source in Fig. 6.2 ($2I$) can be turned off whenever levels $\pm RI$ are to be transmitted. Since the driver power dissipation is directly proportional to the total current ($P = V_{DD} \cdot I_{total}$), the driver power consumption can be reduced by a factor of 1.5 for 4-PAM. This power saving would be even more significant in PAM schemes with more than 4 levels (1.75 for 8-PAM).

6.2 Power-Efficient Driver Topology

The reported high-speed multi-level drivers have used power-inefficient unipolar architectures [29–33]. The line driver, which is proposed here, has a novel power-efficient architecture. Fig. 6.3 shows the basic architecture of the proposed 4-PAM driver. As shown in this figure, this driver uses a bipolar topology to reduce the power. In this architecture, the driver is composed of two basic units. To further reduce the power, the right unit can be turned off whenever $\pm RI$ are to be transmitted. The top current source can be turned off by pulling up both $B2p1$ and $B2p2$ inputs, while for turning off the bottom current source, $B2n1$ and $B2n2$ signals should be pulled down. Another advantage of this architecture is its modularity. This 4-PAM topology can be easily changed to 6-PAM by just adding another

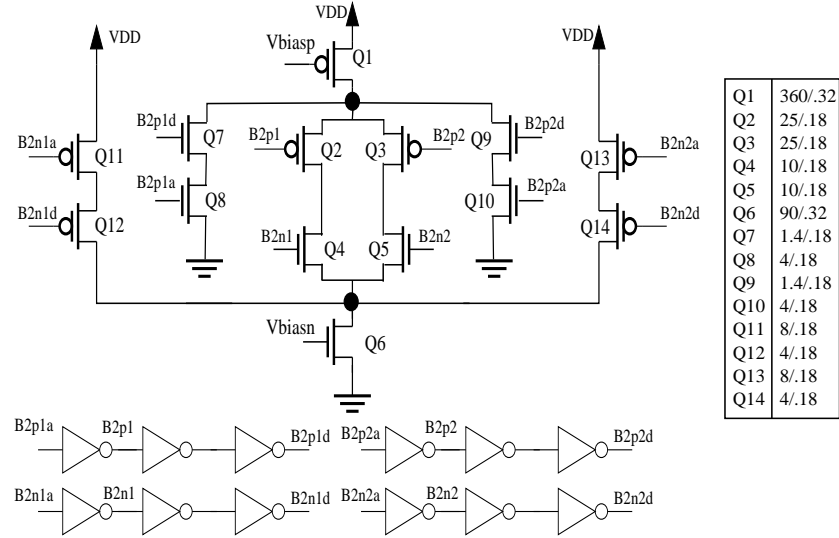


Figure 6.4: Detail of each driver basic unit

basic unit.

While the architecture in Fig. 6.3 significantly reduces the power, switching current sources reduces the maximum operating speed of the driver due to the required settling time of current sources at the switching time. A data-look-ahead technique is used to overcome this problem. As shown in Fig. 6.4, four branches in each unit are used to pre-switch the current sources. This increases the achievable data rate of the driver. The mechanism can be described with an example.

Assume both current sources in Fig. 6.4 are off and signal $B2n1$ is about to go from low to high. Since signal $B2n1$ is the output of an inverter whose input signal is $B2n1a$, signal $B2n1a$ goes low before the signal $B2n1$ goes high (one inverter delay, roughly 40ps). Since the signal $B2n1d$ is the delayed version of signal $B2n1$ (two inverter delay, roughly 80ps), at the transition of $B2n1a$, $B2n1d$ is still low. This turns on transistors $Q11$ and $Q12$, which then turns on transistor $Q6$. By the time that signal $B2n1$ goes from low to high, after one inverter delay, the current in transistor $Q6$ has settled down. In other words, the current source is turned on slightly before the transition of signal $B2n1$. Two inverter delays after the transition of signal $B2n1$, signal $B2n1d$ goes high and turns off the $Q11 - Q12$ branch.

Another advantage of this architecture is the fact that it eliminates the need for a pre-driver. The pre-driver's function is to switch the gate voltages on the two current-steering

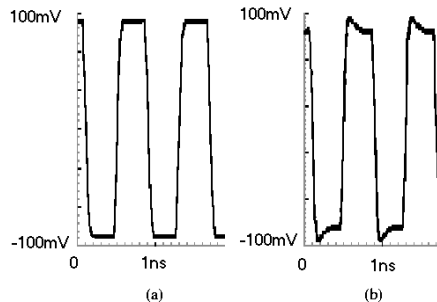


Figure 6.5: The output of a 2-level bipolar driver: (a) With pre-switching, (b) Without pre-switching

transistors, see for example Fig. 6.1a, in each driver segment in such a way that current steers smoothly from one output to the other. To achieve this, tail transistors should stay in the saturation region. Thus, driver inputs should switch between V_{DD} and a voltage slightly less than $V_{tail} + V_{th}$ (where V_{th} is the threshold voltage of the transistor) [34]. However, without a pre-driver, the crossover voltage of the inputs of the driver, $Bit1$ and $\overline{Bit1}$, cannot hold both steering devices on, and thus the tail transistor will fall out of saturation. This not only reduces the speed of the driver, but also creates overshoot and undershoot in the output.

Fortunately, the proposed pre-switching technique also alleviates this problem and there is no need for the pre-driver. Fig. 6.5 shows the output of a 2-level bipolar driver, which is similar to the one in Fig. 6.1b, in two different cases: with and without pre-switching technique. Eliminating the pre-driver from the transmitter block diagram can significantly reduce the transmitter power since a high-speed pre-driver is a power-hungry block.

6.3 Simulation Results

A 4-PAM transmitter is designed and implemented based on the proposed power-efficient driver architecture. The entire transmitter was simulated with HSPICE in a $0.18\mu m$ CMOS technology. The simulated driver power consumption is 10.5 mW and simulation results show that pre-switching circuitry consumes only 1 mW at 10 Gb/s. Simulation results also show that the dynamic power of the driver is roughly 20% of its static power at 10Gb/s.

An important factor in transmitter design is the variation of differential and single-ended

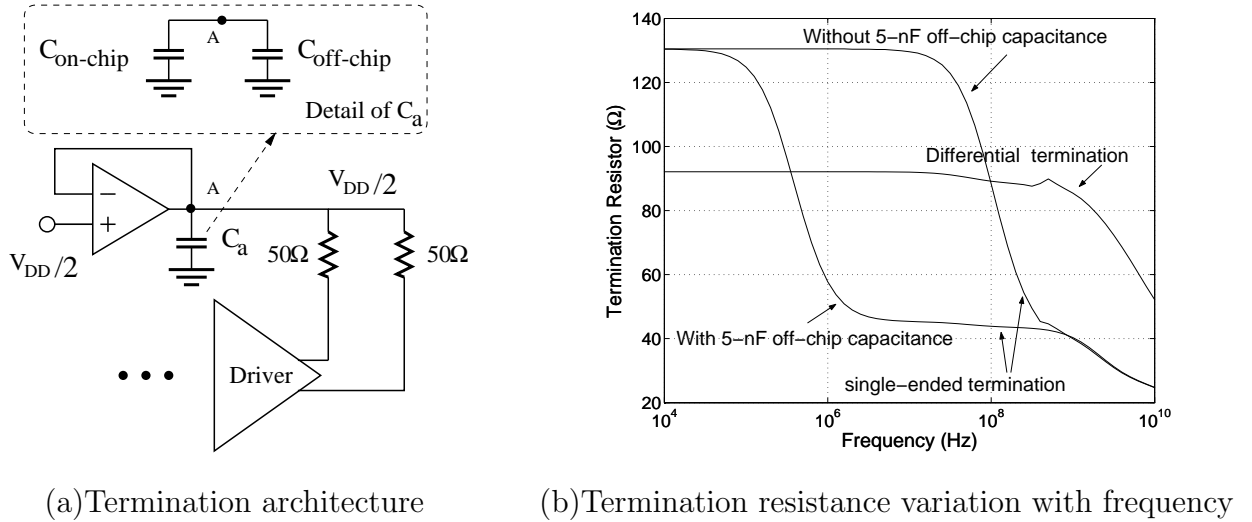


Figure 6.6: Termination structure and its simulation results

termination with frequency. Ideally, with perfect matching, only differential termination is important. However, due to the mismatch between the two differential outputs, single-ended termination is also important. Fig. 6.6a shows the general termination architecture. As shown in this figure, a simple on-chip buffer is used for generating the $V_{DD}/2$ power supply. The current of this power supply is practically zero and the differential termination is always 100Ω in parallel with the driver output impedance. Therefore, this structure is expected to show small differential-termination variation with frequency. This is verified by HSPICE simulation (see Fig. 6.6b).

On the other hand, in this architecture, single-ended termination could vary significantly with frequency. At high frequencies, the capacitance C_a in Fig. 6.6a is a short circuit and therefore the single-ended termination would be roughly 50Ω . However at low frequencies, there would be some impedances between node A and ground and this would change the single-ended termination. This can be alleviated by increasing the capacitance at node A . In this design capacitance C_a is composed of one on-chip and one off-chip capacitance. Fig. 6.6b show the single-ended termination variation with frequency for a 10pF on-chip capacitance with and without a 5nF off-chip capacitor. As shown in Fig. 6.6b, the single-ended termination is 50Ω for a wide range of frequency when a 5-nF off-chip capacitor is used. Fortunately, increasing the capacitance at node A also decouples the noise of $V_{DD}/2$

reference voltage.

6.4 Transmitter Architecture

Fig. 6.7 illustrates the block diagram of the transmitter. A pseudo-random-bit-sequence (PRBS) unit consisting of two $2^7 - 1$ PRBS generators produces the random data. Each of the four encoders in Fig. 6.7 has six outputs, which correspond to the driver inputs. Fig. 6.8 shows the required circuitry for each encoder. As shown in this figure the circuitry for each encoder is very simple and consists of only 3 inverters, two NAND gates and six flip-flop gates. Four encoders work in parallel to generate the data at a quarter of the driver speed. This lower speed makes the design of the PRBS and encoder units much simpler.

4:1 multiplexers are used for serializing data from four parallel branches, outputs of the encoders. Therefore, the outputs of the multiplexers (inputs of the driver) are at full speed. The multiplexer architecture is shown in Fig. 6.9a. It consists of two stages and each stage doubles the data rate. Two clocks with 90° phase shift, CLK1 and CLK2, are used in the multiplexer. A pseudo-NMOS XOR, Fig. 6.10, is used as a frequency doubler to generate a high frequency clock. The basic architecture of each 2:1 multiplexer, Fig. 6.9b, is similar to a dual-edge-trigger flip-flop [63].

In this design, switches use only NMOS transistors to reduce the switch parasitic capacitance and therefore to increase the speed. Threshold voltages of those inverters that are right after the switches, are designed to be less than $V_{DD}/2$ by careful sizing. This compensates the poor behavior of these switches for passing a 'high' signal. Moreover, on-chip termination is used to reduce the ringing due to package parasitics [34].

Fig. 6.11 shows the circuitry for the on-chip termination. An external control voltage can be used to adjust the resistance of the NMOS transistor, which works in triode. This structure is used to improve the termination resistor linearity since the nonlinearity of the NMOS transistor does not directly affect the termination resistor linearity.

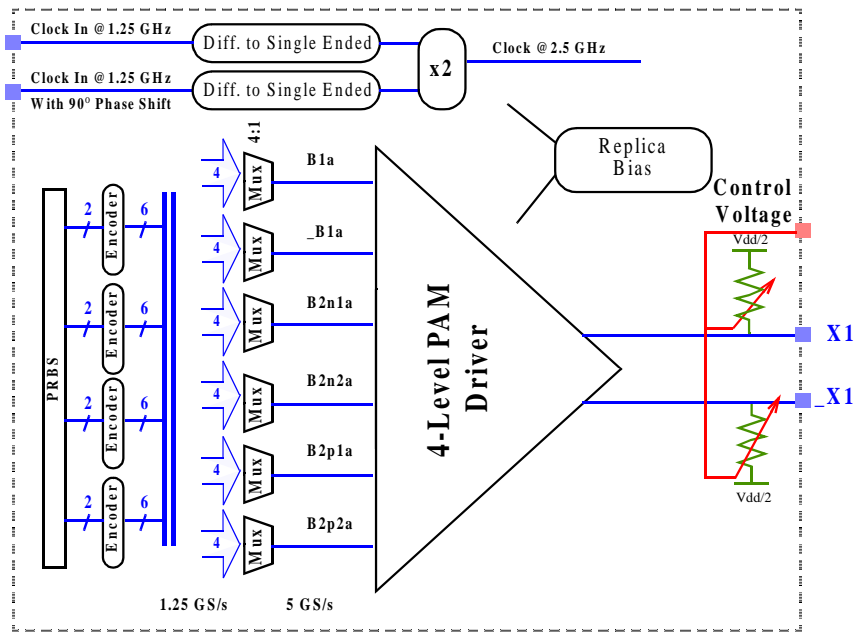


Figure 6.7: Transmitter block diagram

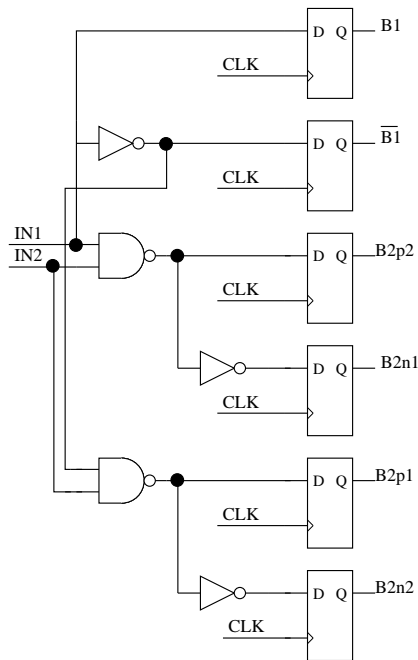


Figure 6.8: Encoder circuitry

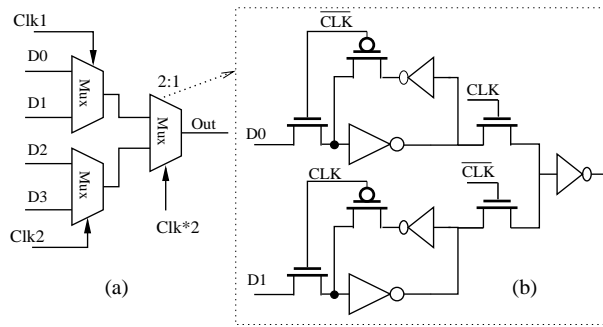


Figure 6.9: Multiplexer architecture

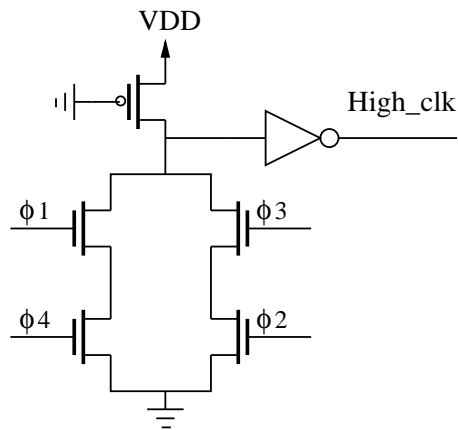


Figure 6.10: The circuitry for generating high-frequency clock

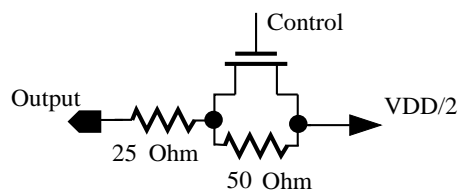


Figure 6.11: On-chip termination circuit

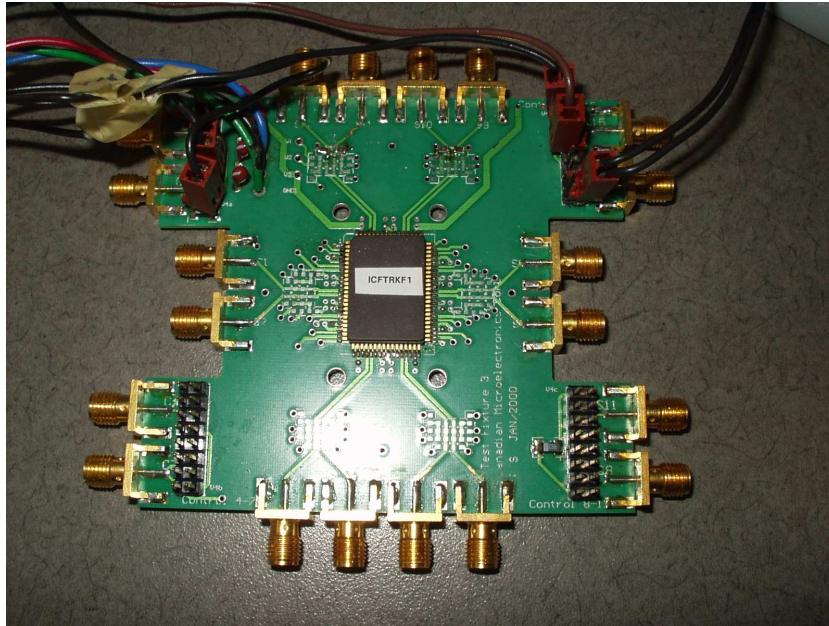


Figure 6.12: Printed circuit board test fixture used for experimental measurement

6.5 Experimental Results

The 4-level PAM transmitter was implemented and fabricated in a $0.18\mu\text{m}$ standard digital CMOS technology and an 80-pin ceramic flat package (CFP80) was used for packaging. Fig. 6.13 shows the $2.6\text{mm} \times 1.5\text{mm}$ chip micrograph. As shown in this figure, there are two transmitters and some test circuitry in the chip and the design is pad-limited. Each transmitter occupies 0.16mm^2 . As shown in Fig. 6.12, a printed circuit board test fixture is used for measurement. This board, which is provided by Canadian Microelectronic Corporation (CMC), supports testing integrated circuits operating at frequencies up to 4.8 GHz (-3dB).

The entire transmitter draws 39 mA from a 1.7-V power supply. The driver and $V_{DD}/2$ reference generator consume roughly 12.5 mW at 7 Gb/s, which is the lowest reported power at this speed. Fig. 6.14 shows the eye at 7 Gb/s at the output of the transmitter. The transmitter has a random output jitter of 22 ps (peak to peak) at 7 Gb/s. A better jitter benchmark for multi-level signaling is the eye-opening. This transmitter has a maximum eye height of 140 mV and an eye width of 200 ps over a 0.8-m cable and a 30-mm printed circuit board (PCB) trace at 7 Gb/s. An eye-opening of 200 ps at the 3.5 GS/s rate corresponds

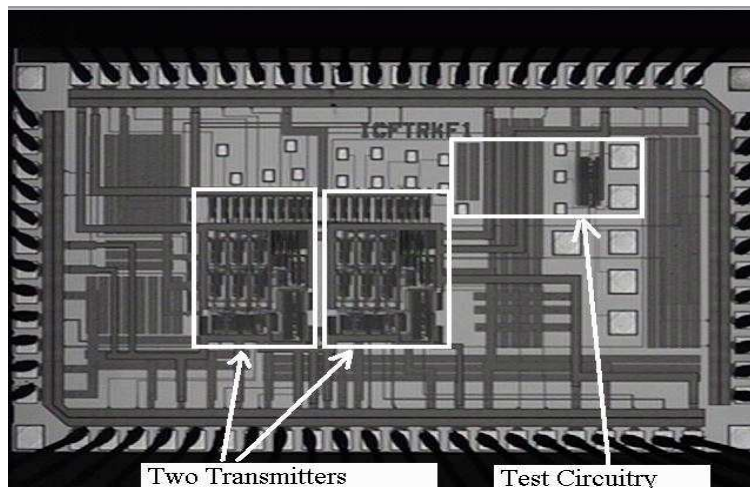


Figure 6.13: Transmitter chip micrograph

to a 70% eye-opening.

By increasing the speed from 7 Gb/s towards 10 Gb/s with a 1.7-V power supply the eye-opening gets smaller. This problem has been resolved by increasing the power supply voltage from 1.7 V to 2 V and increasing the input reference current by 25%. Fig. 6.15a shows the eye-diagram at 8 Gb/s. As shown in this figure, the eye has a maximum eye height of 140 mV and an eye width of 160 ps at 8 Gb/s.

Although the eye-diagram at 10 Gb/s, Fig. 6.15b, is open, the duty-cycle of the clock is not 50% and this produces eyes with different widths. This shows the importance of the clock duty-cycle at high-speed. To solve this problem, duty-cycle-correction circuits can be used [64]. An alternative scheme is to use delay locked loop (DLL) to generate four different quadrature clocks. The small performance degradation of this transmitter due to clock duty-cycle imbalance is not related to the proposed power-efficient architecture.

Table 6.1 summarizes the test results along with the results of other state-of-the-art designs. As shown in this table, the driver power of this design is much smaller than the other designs. However, power supplies and maximum swing of these designs should be normalized to have a fair comparison. The third row of this table shows the normalized values for the driver power of these designs referenced to the design in [32]. For 10Gb/s measurement, since power supply and the input reference current have been increased by 20% and 25%, respectively, the power dissipation is expected to increase roughly by a factor

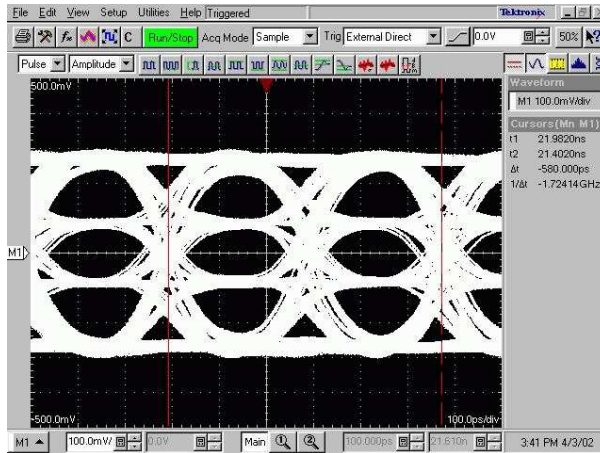


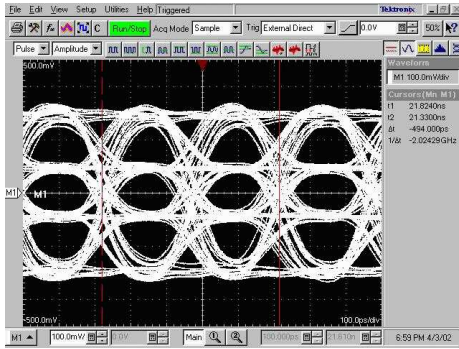
Figure 6.14: Eye diagram at 7Gb/s over 0.8m cable and 30mm printed circuit board channel

Table 6.1: Test result summary

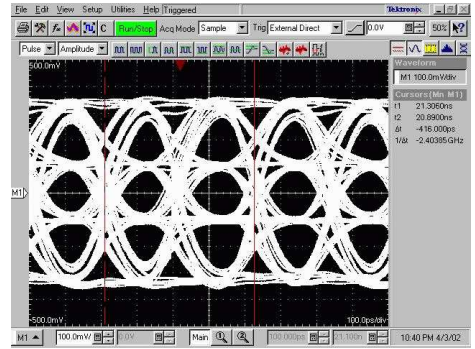
Specification	This design (7Gb/s)	This design (10Gb/s)	design in [32]	design in [31]	design in [33]
Driver Power	12.5mW*	20mW*	220mW	-	-
Normalized Driver Power	0.33	0.45	1	-	-
Total power	66mW	120mW	1W**	1.5W**	400mW**
Data rate	7Gb/s	10Gb/s	8Gb/s	10Gb/s	1.3Gb/s
Max. Swing (p-p)	600mV	600mV	2V	2V	1.1V
Power supply	1.7V	2V	3V	3.3V	3.3V
Technology	0.18 μ m	0.18 μ m	0.3 μ m	0.4 μ m	0.5 μ m

*This includes the power consumption of $V_{DD}/2$ reference generator.

**This value shows the total power of the transmitter and receiver.

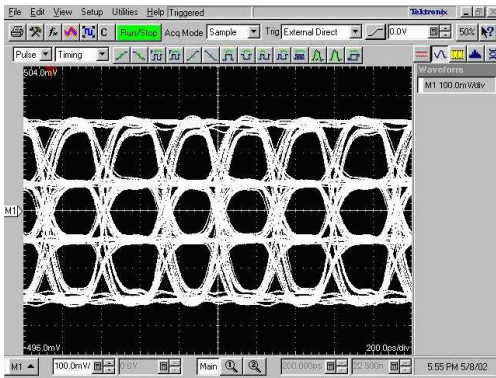


(a) Eye at 8Gb/s

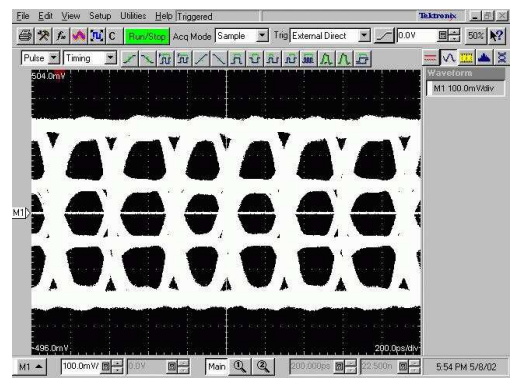


(b) Eye at 10Gb/s

Figure 6.15: Eye diagram at 8 Gb/s and 10Gb/s over 0.8-m cable and 30-mm printed circuit board channel



(a) Eye-diagram without noise



(b) Eye-diagram with noise

Figure 6.16: Eye-diagram with and without power-supply noise

of 1.5. This is reasonably confirmed by the experimental results in Table 6.1, bearing in mind that the dynamic power dissipation also increases with speed.

To determine the performance of the driver in the presence of the driver power-supply noise, a 40mV(rms) white noise is added to the 1.8 V driver power supply. It should be noted that the driver on-chip power-supply noise is attenuated by the on-chip decoupling capacitors. Fig. 6.16 shows the eye-diagram for two different cases: with and without this power-supply noise. As shown in this figure, the proposed power-efficient architecture is not significantly sensitive to the power-supply noise.

6.6 Summary

A novel power-efficient bipolar architecture for multi-level PAM transmitter is presented. This architecture reduces the driver power by employing a bipolar architecture, reducing current when transmitting small voltage levels, and eliminating the need for pre-driver. A data-look-ahead technique is used for high-speed implementation of this architecture. Moreover, a 4-PAM transmitter based on this architecture was fabricated in $0.18\mu m$ digital CMOS technology. The transmitter achieves 3.5 GS/s (7 Gb/s) with a 1.7-V power supply and 5 GS/s (10 Gb/s) with a 2-V power supply. The driver draws 6.7 mA from a 1.7-V supply at 7 Gb/s and 10 mA from a 2-V power supply at 10 Gb/s. Interestingly, this power-efficient architecture can also be used to significantly reduce the power consumption of current-steering digital to analog converters (see Appendix B).

Chapter 7

Conclusion and Future Work

The exponential growth in speed and integrations levels of digital integrated circuits has increased the demand for high-speed chip-to-chip communication bandwidth to maximize the overall system performance. However, the cost and power of these links have also increased dramatically. This dissertation addresses circuit techniques and signaling schemes that can be used to reduce the cost and power of high-speed interconnects.

7.1 Contributions

A technique called incremental signaling was proposed which allows for N differential signals to be communicated via as few as $N + 1$ signal paths. This system rejects the common-mode noise and even-order distortion terms. However, the BER performance would be 3 dB worse than a fully differential system with respect to independent noise sources. The proposed MLSD methods can be used to regain this 3 dB.

There has been a great interest in the use of multi-level signaling to increase the signaling bandwidth without increasing the clock frequency in a system. Multi-level signaling can also be used to reduce the number of signal paths in a link. However, circuit and system improvements are required to compensate the impact of multilevel signaling on SNR. For a given performance, multi-level signaling often needs a higher power than that of the binary signaling.

Channel coding can be used to reduce the required power/SNR of multi-level signaling

schemes. Several suitable coding schemes for high-speed links are proposed. The main contribution, here, is to propose coding schemes with low-complexity decoders. The proposed coding schemes achieve roughly 3 dB coding gain in the case of an additive white Gaussian noise (AWGN) channel. Moreover, a realistic model for the channel is developed that takes into account the effect of crosstalk, jitter, reflection, ISI, and AWGN.

The proposed signaling schemes are significantly less sensitive to those interferences compared to the conventional uncoded 4-PAM. In particular, two coding schemes, 4LINE-PAM6 and 3LINE-PAM4, that show better performance are highlighted and simulation results show that they provide a coding gain of 5-8 dB in the presence of jitter, ISI, and residual reflections. These methods are also less sensitive to crosstalk. Low-complexity architectures and circuit techniques for high-speed implementations of these schemes are proposed. The functionality and performance of the 4LINE-PAM6 scheme at 4 Gb/s has been verified by circuit-level simulation with Spectre.

In addition, the 3LINE-PAM4 scheme is implemented and fabricated in a $0.18\mu\text{m}$ digital CMOS technology. Experimental results not only verified the coding gain at low-speed, but also showed that the chip is functional up to 2.5Gb/s. The measured supply current for the proposed method was 12.5mA and the active area of a receiver based on this method was only 0.2 mm^2 .

Finally, new circuit techniques and architectures for high-speed multi-level transmitters were also investigated. A novel power-efficient bipolar architecture for multi-level PAM transmitter is presented. This architecture reduces the driver power by employing a bipolar architecture, reducing current when transmitting small voltage levels, and eliminating the need for a pre-driver. A data-look-ahead technique is used for high-speed implementation of this architecture. Moreover, a 4-PAM transmitter based on this architecture was fabricated in a $0.18\mu\text{m}$ digital CMOS technology. The transmitter achieves 3.5 GS/s (7 Gb/s) with a 1.7-V power supply and 5 GS/s (10 Gb/s) with a 2-V power supply. The driver draws 6.7 mA from a 1.7-V supply at 7 Gb/s and 10 mA from a 2-V power supply at 10 Gb/s. In addition, we have shown that this power-efficient architecture can also be used to reduce the power consumption of high-speed current-steering digital to analog converters.

7.2 Future Work

Although the proposed incremental signaling in Chapter 3 provides many advantages of fully differential signaling schemes with reduced number of signal pins, high-speed hardware implementation of the proposed modified Viterbi algorithms is an open area of research. Alternatively, one possible implementation is to use an ADC for each line in the receiver and implement the algorithm digitally. However, a practical application should first justify the use of this extra complexity.

The search for low complexity coding schemes for high-speed interconnects can be an ongoing research area. Channel characterization is also crucial. An accurate channel model is essential for coming up with optimum coding schemes for high-speed interconnects. It is also interesting to implement and measure the performance of 4LINE-PAM6 signaling scheme, proposed in Chapter 4, in an actual inter-chip link. Exploiting parallelism along with on-chip per-pin skew cancellation to achieve higher data rates is also another interesting follow-up for this project. Since the scaling of CMOS provides such high bandwidths, one possible application is optical interconnects where the bandwidth of the medium is extremely high. However, electronic components to drive and receive the optical signals have more stringent criteria than electrical links. The receivers require resolution on the order of several millivolts and the drivers often require driving voltages greater than 3V [6], [5]. The design of these circuits is a difficult challenge. It is also interesting to investigate the performance of the proposed coding schemes for optical channels.

In this work, we assume a short-distance channel that does not need equalization. However, in practice, there are many high-speed applications, such as backplane, that requires channel equalization. It is an interesting project to investigate the feasibility of combining the proposed coding schemes with an analog equalizer in the receiver to address this issue. In addition, ADCs can be used to convert the received signal to digital and perform the proposed coding schemes along with equalization in digital domain. Moreover, it would be interesting to investigate the performance of the proposed schemes in a more advanced technology to improve the performance and speed of these methods.

One of the most important limiting factor for the speed of a conventional parallel bus is

the skew between different signal traces in the bus. This has been the main motivation of current trend toward serial interfaces. One possible research topic is to search for coding and modulation techniques that can reduce the sensitivity of parallel busses to skew. This can ease up the specifications on skew compensation blocks and push the bandwidth of electrical parallel links even further. In general, if CMOS technology continues to scale to perform such complex digital signal processes fast enough for multi-Gb/s communication, the use of more complex coding and modulation techniques will improve the performance of electrical links, obviating the need for high-cost optical interconnects.

Coding theory is a broad area and there might be numerous coding techniques that can be used for this application. Exploiting different coding techniques such as dual codes and lattice codes might results in signaling scheme with a better performance or less complexity. One possible coding scheme is to use the same idea of the proposed method in Chapter 5, but force the trellis to return to state zero on the second time interval. This results in a moderate reduction of the receiver complexity with a slight reduction of the code rate.

Finally, the proposed architecture for high-speed multi-level driver can be employed to significantly reduce the power consumption of high-speed current-steering digital-to-analog converters.

References

- [1] R. Mooney, C. Dike, and S. Borkar, “A 900 Mb/s bidirectional signaling scheme,” *IEEE J. Solid-State Circuits*, vol. 30, pp. 1538–1543, Dec. 1995.
- [2] N. Kushiyama *et al.*, “A 500-megabyte/s data-rate 4.5 M DRAM,” *IEEE J. Solid-State Circuits*, vol. 28, pp. 490 – 498, Apr. 1993.
- [3] Y. Ota and R. Swartz, “Multichannel parallel data link for optical communication,” *IEEE LTS*, vol. 2, pp. 24 – 32, May 1991.
- [4] M. Horowitz, C.-K. K. Yang, and S. Sidiropoulos, “High-speed electrical signaling: Overview and limitations,” *IEEE Micro*, pp. 12–24, Jan. 1998.
- [5] R. Farjad-Rad, “A CMOS 4-PAM Multi-Gbps serial link transceiver,” Ph.D. dissertation, Stanford University, Stanford, 2000.
- [6] C.-K. K. Yang, “Design of high-speed serial links in CMOS,” Ph.D. dissertation, Stanford University, Stanford, 1998.
- [7] S. Abdalla, “A 7.2Gb/s/pin 8-bit parallel bus transmitter using incremental signaling in 0.18 μ m CMOS,” Master’s thesis, Univ. of Toronto, Toronto, 2002.
- [8] S. Sidiropoulos, “High performance inter-chip signalling,” Ph.D. dissertation, Stanford University, Stanford, 1998.
- [9] “RapidIO:an embedded system component network architecture,” White Paper, Motorola, Feb. 2000. [Online]. Available: www.tundra.com/tdc_files/library/70A7000_WP001_01.pdf

- [10] D. Mayhew and V. Krishnan, "PCI express and advanced switching: evolutionary path to building next generation interconnects," in *11th Symposium on High Performance Interconnects*, Aug. 2003.
- [11] A. V. Bhatt, "Creating a third generation I/O interconnect," White Paper, Intel, 2002. [Online]. Available: www.intel.com/technology/pciexpress/downloads/3rdGenWhitePaper.pdf
- [12] *IEEE Standard for Low-Voltage Differential Signals (LVDS) for Scalable Coherent Interface (SCI)*, IEEE Computer Society Std. 1596.3, 1996.
- [13] D. R. Cecchi, M. Dina, and C. W. Preuss, "A 1Gb/S SCI line in 0.8 μm BiCMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. of Tech. Papers*, Feb. 1995, pp. 326–327.
- [14] W. G. Rivard and M. J. S. Smith, "A 1.2 μm CMOS differential I/O system capable of 400 Mb/s transmission rates," in *Proc. Fifth Annual ASIC Conf. and Exhibit*, 1992, pp. 427–431.
- [15] J. H. Quigley, J. S. Caravella, and W. J. Neil, "Current mode transceiver logic,(CMTL) for reduced swing CMOS, chip to chip communication," in *Proc. Sixth Annual ASIC Conf. and Exhibit*, 1993, pp. 452–455.
- [16] *IEEE Standard for Scalable Coherent Interface (SCI)*, IEEE Computer Society Std. 1596, 1992.
- [17] *IEEE Standard for High-Bandwidth Memory Interface Based on Scalable Coherent Interface (SCI) Signaling Technology (RamLink)*, IEEE Computer Society Std. 1596.4, 1996.
- [18] J. Sonntag *et al.*, "An adaptive PAM-4 5 Gb/s backplane transceiver in 0.25 μm CMOS," in *Proc. IEEE Custom Integrated Circuits Conference CICC*, May 2002.
- [19] N. Tan and S. Eriksson, "Low-power chip-to-chip communication circuits," *Electronics Letters*, vol. 30, pp. 1732–1733, Oct. 1994.

- [20] P. P. Sotiriadis and A. P. Chandrakasan, "Low-power bus coding techniques considering inter-wire capacitances," in *Proc. IEEE Custom Integrated Circuits Conference CICC*, May 2000.
- [21] ———, "Bus energy reduction by transition pattern coding using a detailed deep submicrometer bus model," *IEEE Trans. Circuits Syst. I*, vol. 50, Oct. 2003.
- [22] M. Hatamian *et al.*, "Design consideration for gigabit ethernet 1000Base-T twisted pair transceivers," in *Proc. IEEE Custom Integrated Circuits Conference CICC*, May 1998.
- [23] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, Jan. 1982.
- [24] E. A. Lee and D. G. Messerschmit, *Digital Communication*. Kluwer Academic Publishers, 1994.
- [25] M. V. Ierssel, T. Esmailian, A. Sheikholeslami, and S. Pasupathy, "Signaling capacity of FR4 PCB traces for chip-to-chip communication," in *Proc. IEEE International Symposium on Circuit and System ISCAS*, May 2003.
- [26] K. Farzan and D. A. Johns, "Power-efficient chip-to-chip signaling schemes," in *Proc. IEEE International Symposium on Circuit and System ISCAS*, May 2002.
- [27] S. Benedetto, G. Montorsi, and D. Divsalar, "Concatenated convolutional codes with interleavers," *IEEE Commun. Mag.*, pp. 102–108, Aug. 2003.
- [28] O. Kwon and R. Pease, "Closely packed microstrip lines as very high-speed chip-to-chip interconnects," *IEEE Trans. Comp., Hybrids, Manufact. Technol.*, vol. CHMT-10, pp. 314–320, Sept. 1987.
- [29] J. L. Zerbe *et al.*, "1.6 Gb/s/pin 4-PAM signaling and circuits for a multidrop bus," in *Proc. IEEE VLSI Symp. Circuits*, June 2000, pp. 128–131.
- [30] J. L. Zerbe, P. S. Chau, C. W. Werner, W. F. Stonecypher, H. J. Liaw, G. J. Yeh, T. P. Thrush, S. C. Best, and K. S. Donnelly, "A 2 Gb/s/pin 4-PAM parallel bus interface

- with transmit crosstalk cancellation, equalization, and integrating receiver,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. of Tech. Papers*, Feb. 2001, pp. 66–67.
- [31] R. Farjad-Rad, C.-K. K. Yang, M. A. Horowitz, and T. H. Lee, “A 0.4 μm CMOS 10-Gb/s 4-PAM pre-emphasis serial link,” *IEEE J. Solid-State Circuits*, vol. 34, pp. 580–585, May 1999.
- [32] R. Farjad-Rad, C.-K. K. Yang, and M. A. Horowitz, “A 0.3 μm CMOS 8-Gb/s 4-PAM serial link transceiver,” *IEEE J. Solid-State Circuits*, vol. 35, pp. 757–764, May 2000.
- [33] D. J. Foley and M. P. Flynn, “A low-power 8-PAM serial transceiver,” *IEEE J. Solid-State Circuits*, vol. 37, pp. 310–316, Mar. 2002.
- [34] W. J. Dally and J. W. Poulton, *Digital Systems Engineering*. Cambridge University Press, 2000.
- [35] R. Tummala, “Electronic packaging research and education in the 21st century at PRC,” in *IEMT/IMC Symposium*, Apr. 1998.
- [36] H. Johnson and M. Graham, *High-Speed Digital Design*. Prentice-Hall PTR, 1995.
- [37] S. H. Hall, G. W. Hall, and J. A. McCall, *High-Speed Digital System Design*. Wiley-IEEE Press, 2000.
- [38] C.-K. Yang, V. Stojanovic, S. Modjtahedi, M. Horowitz, and W. Ellersick, “A serial-link transceiver based on 8-GSample/s A/D and D/A converters in 0.25 μm digital CMOS,” *IEEE J. Solid-State Circuits*, vol. 36, pp. 1684 – 1692, Nov. 2001.
- [39] E. Yeung and M. Horowitz, “A 2.4Gb/s/pin simultaneous bidirectional parallel link with per-pin skew compensation,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. of Tech. Papers*, Feb. 2000, pp. 256–257.
- [40] M. E. Lee, W. J. Dally, and P. Chiang, “A 90mW 4Gb/s equalized I/O circuit with input offset cancellation,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. of Tech. Papers*, Feb. 2000, pp. 252–253.

- [41] B. Razavi, *Design of Analog CMOS Integrated Circuits*. McGraw-Hill, 2001.
- [42] R. Farjad-Rad, W. Dally, H.-T. Ng, R. Senthinathan, M.-J. Lee, R. Rathi, and J. Poulton, “A low-power multiplying DLL for low-jitter multigigahertz clock generation in highly integrated digital chips,” *IEEE J. Solid-State Circuits*, vol. 37, pp. 1804 – 1812, Nov. 2002.
- [43] M. Tomlinson, “New automatic equalizer employing modulo arithmetic,” *Electronics Letters*, vol. 7, pp. 138–139, Mar. 1971.
- [44] M. H. Shakiba, D. A. Johns, and K. W. Martin, “Bimos circuits for analog viterbi decoders,” *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 1527–1537, Dec. 1998.
- [45] G. D. Forney, “The Viterbi algorithm,” in *Proc. IEEE*, vol. 61, Mar. 1973, pp. 268–278.
- [46] S. A. Altekari and J. K. Wolf, “Improvements in detectors based upon colored noise,” *IEEE Trans. Magn.*, vol. 34, pp. 94–97, Jan. 1998.
- [47] J. D. Coker, E. Eleftheriou, F. L. Galbraith, and W. Hirt, “Noise-predictive maximum likelihood (NPML) detection,” *IEEE Trans. Magn.*, vol. 34, pp. 110–117, Jan. 1998.
- [48] A. Kavcic and J. M. F. Moura, “Correlation-sensitive adaptive sequence detection,” *IEEE Trans. Magn.*, vol. 34, pp. 763–771, May 1998.
- [49] L. G. Tallini and B. Bose, “Balanced codes with parallel encoding and decoding,” *IEEE Trans. Comput.*, vol. 48, pp. 794–814, Aug. 1999.
- [50] D. E. Knuth, “Efficient balanced codes,” *IEEE Trans. Inform. Theory*, vol. 32, pp. 51–53, Jan. 1986.
- [51] M. H. Shakiba, D. A. Johns, and K. W. Martin, “Analog implementation of a class iv partial response viterbi decoder,” in *Proc. IEEE International Symposium on Circuit and System ISCAS*, May 1994.
- [52] M. H. Shakiba, “Analog viterbi detection for partial response signaling,” Ph.D. dissertation, Univ. of Toronto, Toronto, 1997.

- [53] Matlab website. [Online]. Available: <http://www.mathworks.com>
- [54] L. F. Wei, "Trellis coded modulation with multi-dimensional constellations," *IEEE Trans. Inform. Theory*, July 1987.
- [55] T. Starr, J. M. Cioffi, and P. J. Silverman, *Understanding Digital Subscriber Line Technology*. Prentice-Hall PTR, 1999.
- [56] T. C. Carusone. High-speed link model. [Online]. Available: <http://www.eecg.toronto.edu/~tcc>
- [57] N. Gup and L. B. Milstein, "Mapping design for general multidimensional communication systems," in *Proc. Military Communications Conference*, 1999, pp. 35–39.
- [58] K. S. Kundert, *The Designer's Guide to SPICE & SPECTRE*. Kluwer Academic Publishers, 1995.
- [59] S. Vlassis and S. Siskos, "High-speed and high-resolution WTA circuit," in *Proc. IEEE International Symposium on Circuit and System ISCAS*, May 1999.
- [60] D. A. Johns and K. W. Martin, *Analog Integrated Circuit Design*. John Wiley, 1997.
- [61] M. E. Lee, W. J. Dally, and P. Chiang, "Low-power area-efficient high-speed I/O circuit techniques," *IEEE J. Solid-State Circuits*, vol. 35, pp. 1591–1599, Nov. 2000.
- [62] Graphical development software. [Online]. Available: <http://www.ni.com/labview>
- [63] G. M. Blair, "Low-power double-edge triggered flipflop," *Electronics Letters*, vol. 33, pp. 1004–1006, May 1997.
- [64] S.-J. Jang, Y.-H. Jun, J.-G. Lee, and B.-S. Kon, "ASMD with duty cycle correction scheme for high-speed DRAM," *Electronics Letters*, vol. 37, pp. 845–847, Aug. 2001.
- [65] H. Kobayashi, "Correlative level coding and maximum-likelihood decoding," *IEEE Trans. Inform. Theory*, vol. 17, pp. 586–594, Jan. 1971.
- [66] J. F. Hayes, "The Viterbi algorithm applied to digital data transmission," *IEEE Commun. Mag.*, vol. 13, pp. 15–20, Mar. 1975.

- [67] A. van den Bosch, M. Borremans, M. Steyaert, and W. Sansen, "A 10-bit 1-GSample/s nyquist current-steering CMOS D/A converter," *IEEE J. Solid-State Circuits*, vol. 36, pp. 315–324, Mar. 2001.
- [68] P. Vorenkamp, J. Verdaasdonk, R. van de Plassche, and D. Scheffer, "A 1 GS/s 10b digital-to-analog converter," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. of Tech. Papers*, Feb. 1994, pp. 52–53.
- [69] D. Seo, A. Weil, and M. Feng, "A 14 bits, 1 GS/s digital-to-analog converter with improved dynamic performance," in *Proc. IEEE International Symposium on Circuit and System ISCAS*, May 2000.
- [70] K. Farzan and D. A. Johns, "A CMOS 7-Gb/s power-efficient 4-PAM transmitter," in *European Solid State Circuit Conference ESSCIRC*, Sept. 2002.
- [71] B. Razavi, *Principles of Data Conversion System Design*. IEEE Press, 1993.
- [72] M. P. Tiilikainen, "A 14-bit 1.8-V 20-mW 1 – mm² CMOS DAC," *IEEE J. Solid-State Circuits*, vol. 36, 2001.

Appendix A

MLSD Probability of Error

Derivation

Fig. A.1 depicts the trellis diagram for a $(1 - D)$ encoded binary sequence. The state at each step, k , of the trellis is an element of the binary alphabet $\{0, 1\}$. Since the input sequence $[u_1 \ u_2 \ u_3 \ \dots]$ is finite in length, so is the trellis diagram. Furthermore, it is known that the initial and final states of the correct path through the trellis must both be 0. Following the approach used in [65] we seek $P_e(l)$, the probability that an adversary path from $k = t$ to $k = t + l$ that follows the state trajectory $[\bar{s}_t \ \bar{s}_{t+1} \ \dots \ \bar{s}_{t+l}]$ will be chosen over the correct path with state trajectory $[s_t \ s_{t+1} \ \dots \ s_{t+l}]$ where $\bar{s}_t = s_t$ and $\bar{s}_{t+l} = s_{t+l}$. If we define b_{ij} as the branch metric from state j to state i at step k , the metric of the adversary path minus the metric of the correct path is

$$w_l = \sum_{k=t+1}^{t+l} (b_{\bar{s}_{k-1}\bar{s}_k}(k) - b_{s_{k-1}s_k}(k)). \quad (\text{A.1})$$

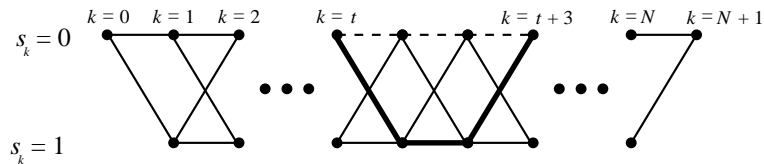


Figure A.1: Trellis diagram for a binary decode system showing the correct path (bold line) and an adversary (dashed line) of length $l = 3$.

Assuming the Euclidean squared error of the detected sequence is to be minimized, a suitable branch metric is given by [66], which is

$$b_{ji}(k) = (j - i)u_k - (j - i)^2 A . \quad (\text{A.2})$$

The path “closest” to the correct one will deviate from it at $k = t + 1$, then run parallel to it until $k = l - 1$. If only the closest adversary path is considered (a reasonable assumption for low bit error rates) all terms in the summation (A.1) equal zero except for the first and last. Substituting (A.2) into the remaining terms yields:

$$w_l = (\bar{s}_{t+1} - s_{t+1})u_{t+1} + (s_{t+l-1} - \bar{s}_{t+l-1})u_{t+l} - A((\bar{s}_{t+1} - s_t)^2 + (s_{t+l} - \bar{s}_{t+l-1})^2 - (s_{t+1} - s_t)^2 - (s_{t+l} - s_{t+l-1})^2) . \quad (\text{A.3})$$

Therefore, w_l is a Gaussian random variable with an expected value of $-2A$ and $P_e(l)$ is the probability that w_l is greater than zero. For $l > 2$, the variance of w_l is

$$\begin{aligned} \text{var}[w_l] &= \text{var}[u_{t+1}] + \text{var}[u_{t+l}] , \\ &= \text{var}[y_{t+2} - y_{t+1}] + \text{var}[y_{t+l+1} - y_{t+l}] , \\ &= \text{var}[y_{t+2}] + \text{var}[y_{t+1}] + \text{var}[y_{t+l+1}] + \text{var}[y_{t+l}] , \\ &= 4\sigma^2 . \end{aligned} \quad (\text{A.4})$$

and $P_e(l)$ is

$$P_e(l) = Q\left(\frac{2A}{\sqrt{4\sigma^2}}\right) = Q(\eta), \quad l > 2 . \quad (\text{A.5})$$

For the special case $l = 2$, (A.3) simplifies to

$$w_2 = (\bar{s}_{t+1} - s_{t+1})(u_{t+1} - u_{t+2}) - A((\bar{s}_{t+1} - s_t)^2 - (s_{t+1} - s_t)^2 - (s_{t+2} - s_{t+1})^2) , \quad (\text{A.6})$$

and the variance of w_2 is

$$\begin{aligned} \text{var}[w_2] &= \text{var}[u_{t+1} - u_{t+2}] , \\ &= \text{var}[(y_{t+2} - y_{t+1}) - (y_{t+1} - y_t)] , \\ &= \text{var}[y_{t+2}] + \text{var}[2y_{t+1}] + \text{var}[y_t] , \\ &= 6\sigma^2 . \end{aligned} \quad (\text{A.7})$$

Therefore, $P_e(2)$ is

$$P_e(2) = Q\left(\frac{2A}{\sqrt{6}\sigma^2}\right) = Q\left(\frac{\sqrt{6}}{3}\eta\right). \quad (\text{A.8})$$

A union bound for the probability of a bit error can now be obtained by summing $P_e(l)$ over all possible values of l :

$$\begin{aligned} P_e &< \sum_{l=2}^{N+1} (l-1) \left(\frac{1}{2}\right)^{l-2} P_e(l), \\ &= P_e(2) + \sum_{l=3}^{N+1} (l-1) \left(\frac{1}{2}\right)^{l-2} P_e(l), \\ &= Q\left(\frac{\sqrt{6}}{3}\eta\right) + Q(\eta) \sum_{l=3}^{N+1} (l-1) \left(\frac{1}{2}\right)^{l-2}. \end{aligned} \quad (\text{A.9})$$

The term $(l-1)$ is included because $(l-1)$ bit errors are caused by incorrectly choosing an adversary path of length l . The term $(1/2)^{l-2}$ represents the fraction of all paths of length l which have an adversary. Although the summation in (A.9) is finite, extending it to an infinite series has little effect on the result for any reasonable N . Hence,

$$\begin{aligned} P_e &< Q\left(\frac{\sqrt{6}}{3}\eta\right) + Q(\eta) \sum_{l=3}^{\infty} (l-1) \left(\frac{1}{2}\right)^{l-2}, \\ &= Q\left(\frac{\sqrt{6}}{3}\eta\right) + 3Q(\eta), \\ &= Q\left(\frac{\sqrt{6} \cdot SNR}{3}\right) + 3Q(\sqrt{SNR}). \end{aligned} \quad (\text{A.10})$$

Appendix B

A Power-Efficient Architecture for High-Speed D/A Converters

High-speed analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) are key components for advanced digital receivers and high-speed instruments. Increasing amount of effort in the integration of digital and analog systems on one chip intensifies the importance of the interface (DACs and ADCs) between these systems. CMOS current steering D/A converters are widely used for many high-speed applications since they are fast and cost effective. Since high-speed high-precision DACs would require large area and power consumption, the effort for reducing the power consumption is extremely important. Almost all of the reported high-speed DACs have used power-inefficient unipolar architectures [38, 67–69]. The proposed architecture for power-efficient bipolar drivers in Chapter 6 can be used to significantly reduce the power consumption of high-speed current-steering DACs.

B.1 A Power-Efficient Topology for DACs

This section explain the proposed power-efficient architecture for a 2-bit DAC for better illustration. Fig. B.1 shows the basic architecture of the proposed 2-bit DAC. As shown in this figure, this driver uses bipolar topology to reduce the power. In this architecture, the DAC composed of two basic units. To further reduce the power, the right unit can be turned off whenever $\pm RI$ are to be generated. This can be done by proper selection of its inputs ($B2n1$, $B2n2$, $B2p1$, and $B2p2$). The top current source can be turned off by pulling up both $B2p1$ and $B2p2$ inputs, while for turning

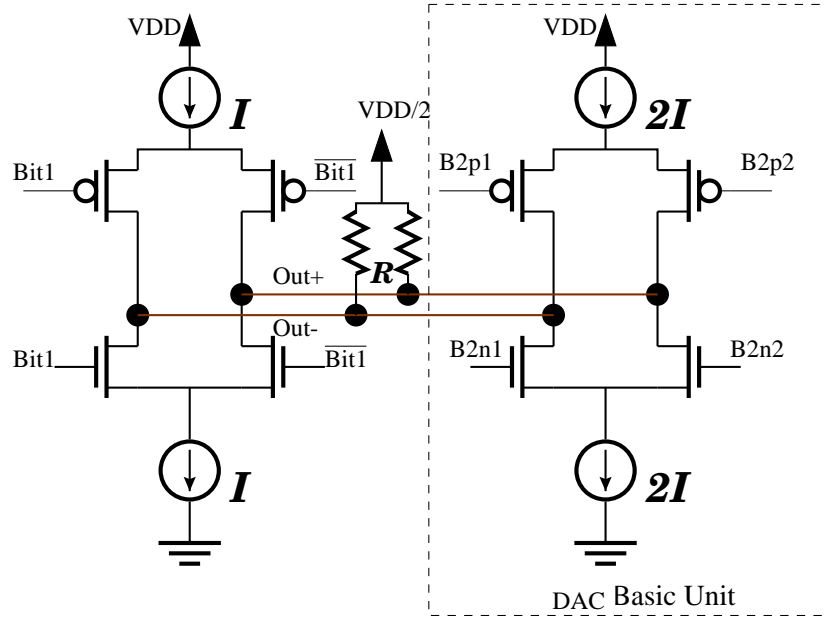


Figure B.1: Basic architecture of the power-efficient 2-bit DAC

off the bottom current source, $B2n1$ and $B2n2$ signals should be pulled down. Another advantage of this architecture is its modularity. This topology can be easily extended to a 6-bit DAC by just adding four basic units. Therefore, this architecture can be used as a general power-efficient architecture for high-speed D/A Converters.

The proposed architecture in Fig. B.1 can significantly reduce the power. However, switching current sources reduces the maximum operating speed of the driver since current sources need some time to settle at the switching time. A data look ahead technique is used to overcome this problem. As shown in Fig. B.2, four branches in each unit are used to pre-switch the current sources. This increases the achievable data rate of the driver. The mechanism can be described with an example. Assume that both current sources in Fig. B.2 are off and signal $B2n1$ is about to go from low to high. Since signal $B2n1$ is the output of an inverter whose input signal is $B2n1a$, signal $B2n1a$ goes low before the signal $B2n1$ goes high (one inverter delay, roughly 40ps). Since the signal $B2n1d$ is the delayed version of signal $B2n1$ (two inverter delay, roughly 80 ps), at the transition of $B2n1a$ from high to low $B2n1d$ is still low. This turns on transistors $Q11$ and $Q12$, which then turns on transistor $Q6$. By the time that signal $B2n1$ goes from low to high, after one inverter delay, the current in transistor $Q6$ has settled down. In other words, the current source is turned on slightly before the transition of signal $B2n1$. Two inverter delays after the transition of signal $B2n1$, signal $B2n1d$ goes high and turns off the $Q11 - Q12$ branch.

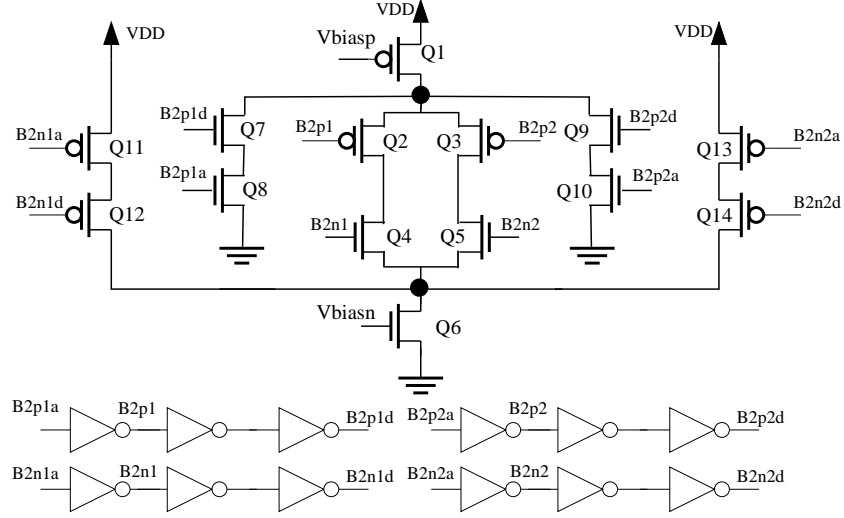


Figure B.2: Detail of each DAC basic-unit

Another advantage of this architecture is the fact that it eliminates the need for a pre-driver, which is usually used before each unity-current cell in DACs. The pre-driver's function is to switch the gate voltages on the two current-steering transistors, see for example Fig. 6.1a, in each current-cell in such a way that current steers smoothly from one output to the other. To achieve this, tail transistors should stay in saturation region. Thus, driver inputs should switch between V_{DD} and a voltage slightly less than $V_{tail} + V_{th}$, where V_{th} is the threshold voltage of the transistor and V_{tail} is the Drain voltage of transistor Q6 in Fig. B.2 [34]. However, without a pre-driver, the crossover voltage of the inputs of the driver, $Bit1$ and $\overline{Bit1}$, cannot hold both steering devices on, and thus the tail transistor will fall out of saturation. This not only reduces the speed of the driver, but also creates overshoot and undershoot in the output. Fortunately, the proposed pre-switching technique can also alleviate this problem and there is no need for a pre-driver block [70]. Eliminating the pre-driver significantly reduces the DAC power since a high-speed pre-driver is a power hungry block. The next section explains the block diagram of a 6-bit DAC, which is designed based on this architecture.

B.2 6-bit Power-Efficient DAC Architecture

Binary weighted DACs are area efficient and simple. However, they suffer from large differential nonlinearity (DNL) and integral nonlinearity (INL). On the other hand, unary decoded architecture that uses thermometer code to switch every unit current-cell separately has the disadvantage of

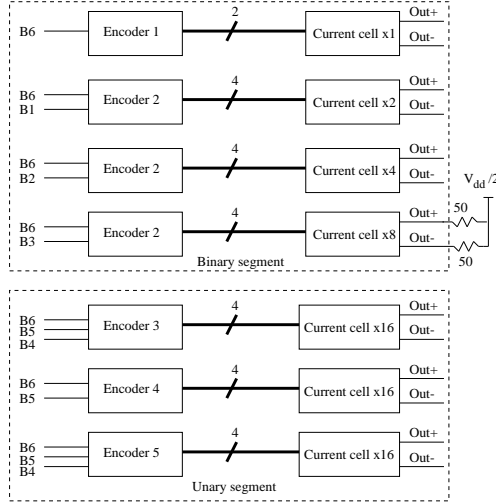
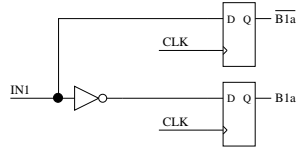


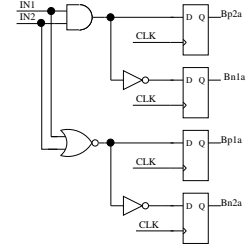
Figure B.3: The general block diagram of the 6-bit power-efficient DAC

large complexity and power consumption. Segmented architecture can be used to alleviate these problems [71]. Fig. B.3 illustrates the block diagram of the power-efficient 6-bit DAC. B6-B1 are the input bits in binary format. As shown in this figure, the proposed DAC uses binary architecture for the 4 least significant bits (LSB) and unary architecture for the 2 most significant bits (MSB). Therefore, four binary current-cells, similar to DAC basic-unit in Fig. B.2, are used for the 4 LSBs (B4-B1). The MSB (B6) is also the sign bit and inverting B6 changes the direction of current in all current-cells. The smallest ($\times 1$) and the largest ($\times 8$) current-cell in the binary segment steer $100\mu A$ and $800\mu A$ current, respectively. Since the smallest current cell in the binary segment is always on, only two inputs are enough for its corresponding DAC-basic-unit. Other 3 current-cells need 4 input signals. As shown in Fig. B.3, four encoders are used for generating these inputs. Fig. B.4 shows the detail of these encoders.

As shown in Fig. B.3, 3 unary current-cells, corresponding to the 2 MSBs (B6, B5), are used for unary segment of the D/A converter. Interestingly, converting binary to thermometer code and generating the necessary input signals for the unary current-cell can be merged in one single encoder. This reduces the complexity and latency of these encoders. Fig. B.5 shows the detail of unary encoders used in the DAC block diagram (Fig. B.3). As shown in Fig. B.4 and Fig. B.5, D-type flip-flops are used to synchronize the inputs of all current-cells.

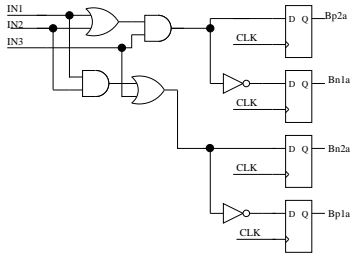


(a) Encoder 1

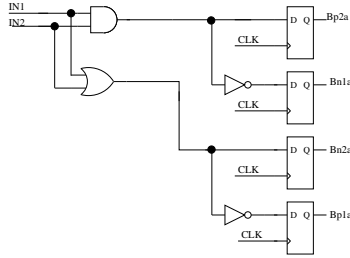


(b) Encoder 2

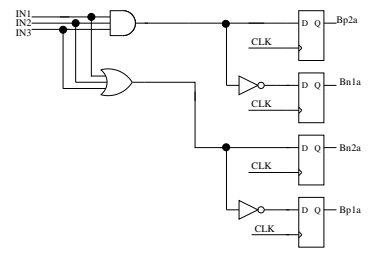
Figure B.4: Binary segment encoders



(a)Encoder3



(b)Encoder4

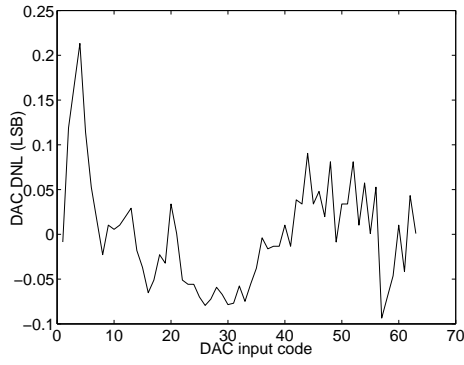


(c)Encoder5

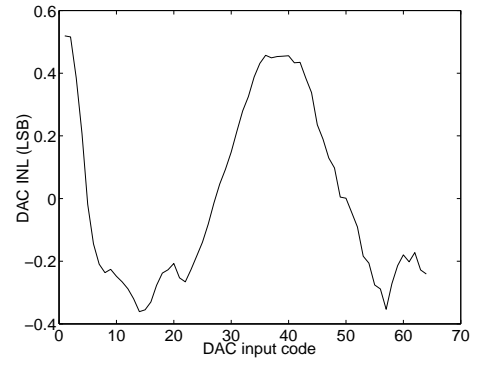
Figure B.5: Unary segment encoders

B.3 Simulation Results

The proposed DAC is designed and simulated with HSPICE in $0.18\mu\text{m}$ CMOS technology. Fig. B.6 shows the simulated DNL and INL profiles versus DAC input code. As shown in this figure, the DNL and INL are less than 0.22 LSB and 0.5 LSB, respectively. Fig. B.7a shows the output of the DAC for a 5MHz sine wave input with a 1GHz sampling frequency. The DAC spectrum is shown in Fig. B.7b and the DAC signal to noise plus distortion ratio (SNDR) for this case is 35.43dB. Table B.1 summarizes the result of this design and 3 other state-of-the-art designs in the literature. As shown in this table, although the output swing of this design is 1.5 times of the ones in [67] and [38], the DAC power consumption (24mW) at 1GS/s is much smaller than the power consumption of other designs. Although this architecture reduces the power significantly, its power-supply current is signal dependent. Therefore, careful layout and design should be used to reduce its destructive effect.

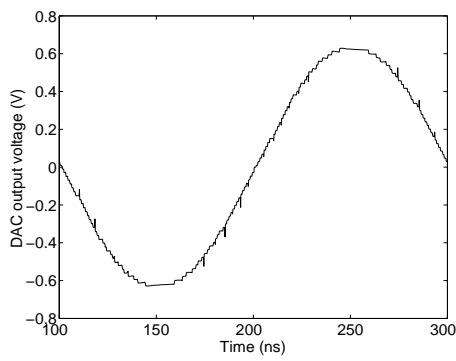


(a)DAC DNL

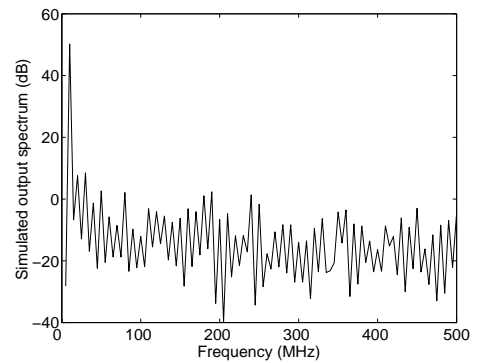


(b)DAC INL

Figure B.6: The simulated DNL and INL profiles versus DAC input code



(a)DAC output



(b)Simulated output spectrum

Figure B.7: DAC output and its spectrum for a 5 MHz sine wave input

B.4 Summary

A power-efficient bipolar architecture for D/A converters is presented. A data-look-ahead technique is used for high-speed implementation of this architecture. Interestingly, this technique also eliminates the need for the pre-driver block. Moreover, a 1GS/s 6-bit DAC based on this architecture is designed in $0.18\mu\text{m}$ CMOS technology. Simulation results show that this DAC consumes only 24mW at 1GS/s, which is the lowest power reported at this speed.

Table B.1: Result summary

Specification	This design ⁺	Design in [67] ⁺⁺	Design in [38] ⁺⁺	Design in [72] ⁺⁺
Total power	24mW	110mW	1.2W	20mW
Update rate	1GS/s	1GS/s	8GS/s	100MS/s
Max. Swing	1300mV fully diff.	800mV single-ended	800mV fully diff.	800mV fully diff.
Power supply	1.8V	3V/1.9V *	2.5V	1.8V
no. of bits	6 bits	10 bits	8 bits	14 bits
DNL	0.22LSB	0.15LSB	Not rep.	$< 0.5LSB$
INL	0.48LSB	0.2LSB	Not rep.	$< 0.5LSB$
Technology	CMOS 0.18 μm	CMOS 0.35 μm	CMOS 0.25 μm	CMOS 0.18 μm

* 3V for analog and 1.9V for digital power supply.

+ Simulated result, ++ Experimental result.