CHANNEL-LIMITED HIGH-SPEED LINKS: MODELING, ANALYSIS AND DESIGN

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING

AND THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Vladimir Stojanović

September 2004

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.

Mark Horowitz

Mark A. Horowitz
(Principal Advisor)

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.

John M. Cioffi

John M. Cioffi

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.

Joseph M. Kahn

Joseph M. Kahn

Approved for the University Committee on Graduate Studies

Abstract

Today's high-speed interfaces are limited by the bandwidth of the communication channel, tight power constraints and noise sources that differ from those in standard communication systems. The wire bandwidth limitations make straight circuit solutions inefficient, and the power and area constraints make standard digital communication approaches infeasible. This thesis presents a system-level link design approach, tightly integrating the noise and channel properties with communication algorithms and circuit-level power and speed constraints.

After describing the issues that high-speed I/Os need to overcome, we create a model that correctly represents the statistics of the various noise sources that affect the system's performance. Our new link model maps the timing noise into effective voltage noise revealing the critical impact of high-frequency transmit jitter. This model estimates the performance limits of the system, and indicates the components which most limit the link performance. The capacity of typical high-speed link backplane channels is between 50 and 100 Gb/s, which is much higher than 3 Gb/s data rates of currently deployed baseband links. We then allocate our limited area and power resources to those issues that are most critical to overall performance.

In order to estimate the data rates of practical baseband architectures, we solve the power constrained optimal linear precoding problem and formulate a bit-error rate (BER) driven optimization, including all link-specific noise sources and hardware constraints. Using this optimization framework, we show that practical data rates are mainly limited by inter-symbol interference due to complexity constraints on the number of precoder and equalizer taps, and then by slicer resolution and sampling jitter that limit the higher bandwidth utilization provided by multi-level modulations. Better circuits are needed to improve the bandwidth utilization to more than 2 bits/dimension in baseband. With current circuit technology and precision, it seems that links which use both PAM2 and PAM4 modulation, with a combination of transmit pre-emphasis and decision feedback equalization (DFE) can achieve 5-12 Gb/s data rates.

With only minor modifications, the hardware needed to implement a PAM4 system can be used to implement a loop-unrolled single-tap DFE receiver. To get the maximum performance from either technique in practice, the link has to be tuned to match the specific channel it is driving. To achieve this with low cost we designed an adaptive equalization technique using data based update filtering that allows continuous updates while minimizing the required sampler front-end hardware and significantly reduces the cost of implementation in multi-level signaling schemes. A transceiver chip was designed and fabricated in a 0.13 μm CMOS process to investigate dual-mode PAM2/PAM4 operation and the modifications of the standard adaptive algorithms necessary to operate in high-speed link environments. The experimental data match the statistical link model predictions extremely well, within a couple of mV, even at BERs lower than the required 10^{-15}.

Acknowledgments

I would first like to thank my two advisors. Prof. Mark Horowitz is the best advisor I know – the Shannon limit – an ideal that I just wish I will be able to approach as the years go by. I would like to thank him for his tolerance, patience in letting me explore different fields, for always making me understand and interpret the true nature of things and great technical guidance.

I would also like to thank my undergraduate thesis advisor, Prof. Vojin Oklobdžija for always supporting me and giving me the best advice, regardless of consequences for him. I especially value his courage and trust in bringing me to work with him and helping me to enroll at Stanford. We have enjoyed many years of joint work, but I thank him most for being patient with me at the very beginning and using his vision to open the doors of the research world to me.

I would like to thank Rambus for the financial support and for providing a great environment for me to do research and interact with outstanding people. I am very grateful to Jared Zerbe, Andrew Ho, and Fred Chen of Rambus, for being great colleagues and friends. I wish to thank all the people in Rambus XG team for all the help and support in building the chips and for constantly educating me about the signal integrity and high-speed circuit design.

I also wish to thank MARCO IFC for initial financial support and later affiliation.

The people in my research group provided a great creative environment and I thank them for that, especially Elad Alon and Amir Amirkhany for all the creativity and work on joint projects. I wish to thank Elad for always being interested and ready to help. His curiosity and great research energy led to many hours of “destabilizing” and many great ideas.

I would like to thank Dr. George Ginis for joint work on parts of the project and Prof. John Cioffi for always providing useful feedback and being my thesis reader. I also would like to thank Prof. Stephen Boyd for being on my orals committee and for outstanding lectures on convex optimization, which lured me into the field of optimization and enabled the formulation of many problems in this dissertation. I am especially grateful to Prof. Joseph Kahn for being on my orals committee, reading my thesis and more than anything else engaging me in a very interesting research in optical links.

I wish to thank Prof. Michael Flynn for initial support at Stanford (when I needed it the most), Prof. Ken Yang for great time working together and Prof. Borivoje Nikolić for work on VLSI optimization projects.

I would like to give my special thanks to Marianne Marx for having faith in me when nobody at Stanford did. I would like to thank Teresa, Penny, Taru, Deborah, Pamela for being great admins for the group and helping me in numerous issues.

My warmest thanks go to Svjetlana, Danijela and Dejan for their great friendship and support.

I also wish to thank my sister Tamara, Maurizio and my whole family.

My special thanks and enormous gratitude go to my mother Nada for always believing in me (even at the times I did not), and supporting me with her infinite motherly love. I just hope that some day I will repay the debt by trying to approach the parenting ideal that she set by her example.

I wish to thank my wife Ivana, kids Marija and Marko for giving me the reason to be happy, and strength to work and live. I owe a special thanks to them for always supporting me with their patience and love, and bearing with times of my mental and physical absence.

Going through this period, I was extremely fortunate to be shielded by two best kinds of love, motherly love of my mother Nada, and that of my wife Ivana. Each in their own way gave me the strength to stay on the course, and for that I dedicate this thesis to them.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In the past decade we have witnessed the integration of computer systems into a more global context of Information Technology Systems, which communicate and process information. Scaling of integrated circuit technology has continually increased the data processing capabilities of integrated circuits in these systems. On the other hand, data communication has continually caused bottlenecks in these systems at various levels of system hierarchy. The challenges and demand associated with overcoming these bottlenecks have caused a boom in development of the systems and techniques that improve the efficiency of data communication.

Breakthroughs in communication and signal processing techniques have resulted in faster Internet access due to development of modems [1-3] over twisted pair telephone channels. Increase in data rates of optical links in Internet backbones [4] has caused the need for faster data routing nodes (i.e. core routers) [5]. At the lowest level of the system hierarchy, significant effort has been devoted to increasing the data rates in chip-to-chip communication, from communication between the data-shuffling chips inside the internet router [6,7], to communication between processors and memory chips in computers [8-10].

Simple input/output (I/O) drivers integrated in these chips have been replaced by more and more sophisticated high-speed link circuits. While chips in modems have incorporated the latest signal processing techniques to overcome the severe bandwidth limitations of telephone channels, chip-to-chip high-speed links have mostly focused on improvements in circuits needed to sustain the desired data rates (from 100’s Mb/s in the early 1990’s to 10’s of Gb/s today) overcoming the limitations of a given integrated circuit technology.

This improvement in chip I/O performance (which scaled faster than processor frequency) has led to expectations of continued improvements in the I/O rates. However, the nature of the I/O design problem is changing. Today internal circuits can run at 10’s of Gb/s, but the performance of the link is limited by the bandwidth of the channel – the electrical path from one die to the other. The obvious question now is how to continue to scale I/O performance, and what, if anything, will ultimately limit pin bandwidth.

This thesis is an attempt to bridge the gap between high-speed link design and high-speed communication system design in order to overcome the bandwidth limitations of today’s high-speed links. We apply the analysis and techniques used in communication system design to the unique problems posed by the high-speed, channel-limited link design. By analyzing the specific properties of the high-speed link system, and by classifying and analyzing the noise sources of these systems, we are able to decide where communication techniques can be applied most cost effectively and how to apply them.

1.1 Organization

In order to efficiently trade-off the data rates with power and complexity in high-speed links, we need to understand how the link’s performance depends on different components in the system. We first need to understand the way the signal is degraded in the wires connecting the chips, and then characterize the behavior and magnitude of the noise sources that affect the signal.

We look at these components in Chapter 2 where we describe the high-speed link environment. In that chapter, we first look at the channel properties using an example of a high-speed backplane link connecting high-speed serializer-deserializer chips (placed on two linecards of an Internet router) over a backplane. Next we describe the link specific noise sources and models to characterize their propagation through the system.

With these two components, the channel and the noise, in Chapter 3 we perform system-level analysis and optimization. By analyzing the system with the channel and noise models developed in Chapter 2 and link-specific hardware constraints, we first determine what the ultimate limits are, then find the most efficient communication techniques that work with practical design constraints.

Since practical high-speed links have many design constraints, it is very important to understand the most critical effects and focus the system and circuit design on these most sensitive issues. In Chapter 4 we formulate the statistical system-level link model based on the noise and the channel models from Chapter 2 and analyze the performance of the communication techniques and topologies described in Chapter 3. We use this performance analysis to predict the most efficient link architectures with link specific noise sources and hardware constraints.

Chapter 5 describes the experimental system built using the analysis in Chapter 4. This system incorporates novel adaptive equalization and modulation, and corresponding clock and data recovery techniques. It is also used to verify the link modeling and analysis.

In Chapter 6, we give predictions about the scaling of data rates in high-speed links derived from our link models and experimental results. These results indicate that new link architectures are needed to further scale the data rates over existing channels. In the conclusion, we also outline some of the possible directions for further extensions of this work.

Before we start developing the material in this dissertation, it is useful to get some background information about high-speed links. So, in the next section, we first look at the historical development of high-speed link designs, and then address the challenges in today's high-speed link design that this thesis is trying to solve.

1.2 Background: High-Speed Link System

Due to the limited number of I/O pins in a chip's package and density constraints on the number of wires between the chips, high-speed links usually serialize blocks of parallel data for off-chip transmission. The system diagram of a typical high-speed link is shown in Figure 1.1.

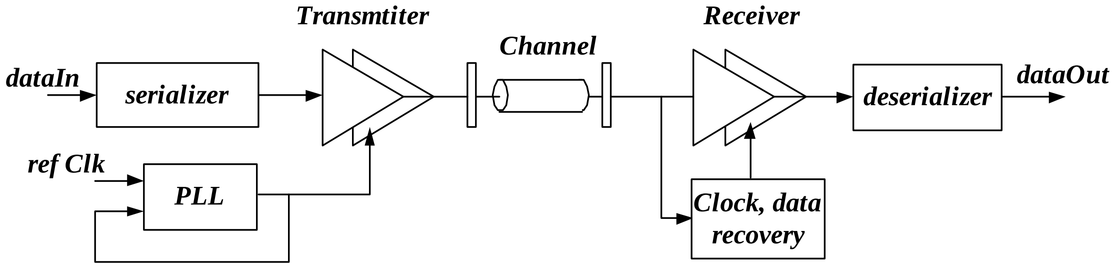


Figure 1.1: High-speed link block diagram

A phase locked loop (PLL) acts as a timing generator in a high-speed link; it generates a high-frequency transmit clock by multiplying the low frequency reference clock. A clock and data recovery (CDR) circuit on the receiver side usually incorporates a PLL and some additional circuits needed to synchronize the receiver with the incoming data stream. These timing blocks are critical for high-speed operation of the link since they provide accurate spacing of transmitted data symbols and sampling of the signal waveforms at the receiver.

One of the earliest efforts in integrated high-speed links started with design of high-speed current integrating receivers [11], which at sub-Gb/s data rates provided a very good approximation to the matched filter [12] for a pulse, and significantly improved the robustness of the link by averaging the noise, especially timing jitter.

With data rates entering the Gb/s region, high-speed I/O design became limited by the speed of the underlying technology[1]. This resulted in the work on multiplexed transmitters and receivers [13] which managed to overcome the intrinsic gate-speed limitations. The key to these techniques lies in precise phase generation, using ring-based voltage controlled oscillators to derive multiple phases and phase-interpolators to obtain higher resolution of phase tuning. This precise phase generation also enabled the use of oversampling receivers, where each bit was 3x oversampled to provide for both data and clock recovery [14].

Further advances in high-speed I/O led to improvements in the design of timing

[1] The width of the shortest pulse that can be propagated through an inverter chain in a given circuit technology is 3-4 FO4, where FO4 is the delay of an inverter loaded by the four identical inverters. This limits the period of the on-chip clock to 6-8 FO4 delays.

loops, such as semi-digital dual delay-locked loops (DLLs) [15], and adaptive bandwidth phase locked-loops (PLLs) and DLLs with regulated supply CMOS buffers [16]. Numerous other papers on PLLs and DLLs [17-20], and CDR loops [21-23] have been published in the last decade, enabling the continuous increase in high-speed link data rates.

During all this time, the wires connecting the chips appeared essentially lossless and given that the intrinsic voltage noise is very low, the focus in link design was on high-speed I/O and on reduction in timing noise, which dominated the link performance. However, as the data rates kept scaling, somewhere around 3 Gb/s links started being limited by the wire bandwidth. In order to address these issues, several designs started to use one or two taps of transmit pre-emphasis [24-26] to compensate for channel bandwidth limitation, or used four-level signaling [27,28], instead of binary NRZ signaling, to increase the data rates without increasing the signaling rate significantly beyond the channel bandwidth. In both cases, the distance between signal levels decreases due to the limited headroom of the I/O driver. Because of this effect, coupled with stringent bit-error-rate (BER) requirements in high-speed links, we need to look carefully into the interference and noise sources in order to determine how many signal levels to transmit and at what rate, in order to achieve the maximum data rate for which the link still works with the guaranteed BER target. Since these equalization and modulation techniques increase the cost and complexity of the link, the biggest design challenge in today’s links is to find the architecture that most efficiently achieves the desired data rate.

Chapter 2

High-Speed Link Environment

In order to trade-off link power and complexity with performance, designers of today's links find it necessary to have the capability to predict the link performance at design time. To fully understand the factors that limit the link performance, we first need to find out how the communication channel between the two high-speed link chips degrades the signal. In order to do this, in this chapter we first describe the link environment using a backplane link as an example. Then we focus on the signal interference caused by the channel in that environment. We show that interference strongly degrades the quality of the signal and we look at the physical properties of the link channels, to gain more insight into the origin of that interference.

Having described the impact of the interference on the quality of the signal, we then look at the noise sources in the system. We refer all the noise components to the input of the receiver and then rank them in order to find the ones that are most important. To do this, we map the timing noise from both transmitter and receiver, to the effective voltage noise as seen by the receiver using superposition-based models. We use these models later in Chapter 4 to show that timing noise generated by PLLs and CDR loops is larger than generic voltage noise. These models are also used to develop the system optimization framework in Chapter 3 which then enables the exploration of the link design space in Chapter 4.

2.1 Channel

The characteristics of the link channel depend strongly on the application. In this work we use a backplane link as our design example, although the analysis method we develop can be applied to any link design.

2.1.1 Backplane Environment – A Physical Example

A typical backplane environment is illustrated in Figure 2.1, where the high-speed serializer-deserializer chips connect the two linecards over a backplane. Such backplanes can usually be found in large Internet routers [29, 30] or more recently in racks of Blade servers [31].

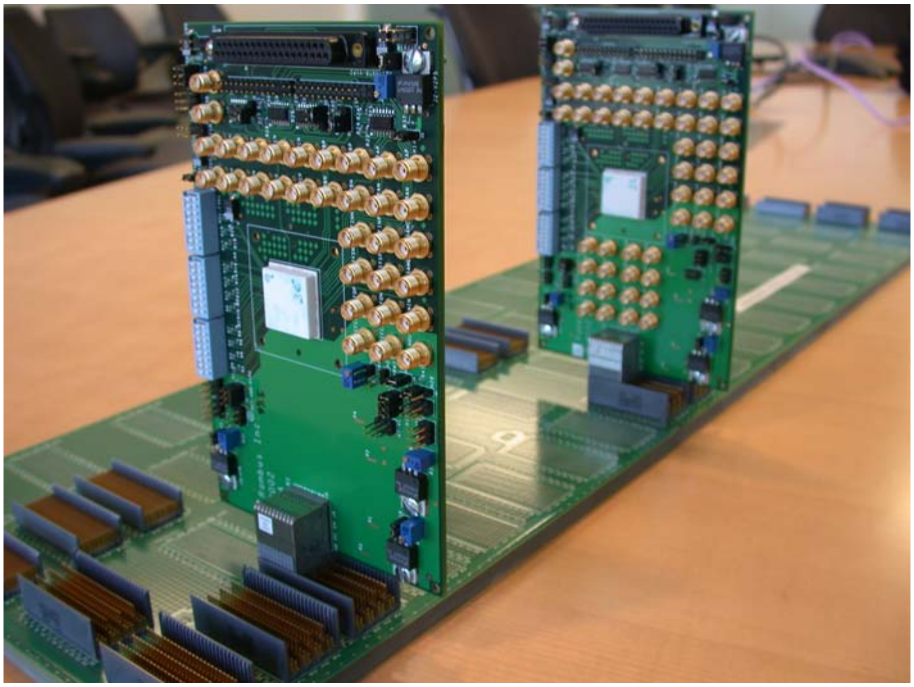


Figure 2.1: Backplane environment, high-speed serializer-deserializer chips connecting the two linecards over a backplane. Photo is courtesy of Rambus, Inc.

Linecards accept the optical connections from the external network (e.g. SONET), or from other distributed racks of equipment [32]. High-speed serializer-deserializer chips then communicate the data between the linecard and the switchcard, which then redirects the data stream to the linecard that contains the desired output port of the router.

In this system, the chips are mounted in packages that are soldered to the linecard. The linecards plug into the backplane using dense through-hole connectors [33]. The cross-section of the system shown in Figure 2.2 makes it easier to see the full signaling path.

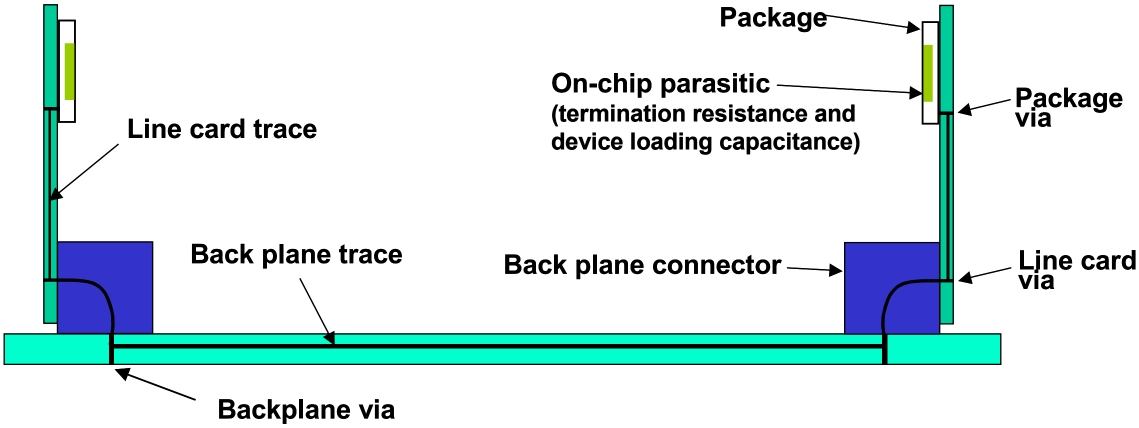


Figure 2.2: Backplane system cross-section indicating different sections of the signaling path [34]

The channel in Figure 2.2 is the full path from one die to the other die. On-chip 50 Ω termination resistors and device capacitance create parasitic low-pass filters. The signal has to traverse a number of different traces in order to arrive from source to destination. Along the long backplane traces we have increasing line attenuation with frequency, due to skin-effect and dielectric loss [35]. While this line attenuation causes additional low-pass filtering of the signal, sometimes more detrimental effects come from the short traces (e.g. vias, or connector traces) that connect the components of the system together. We use these traces to get from the package into the linecard and to connect the linecard to the backplane. These short traces can create large impedance mismatches and cause reflections that can significantly degrade the quality of the signal. We can model all these components in the system relatively accurately, and create the frequency response of the channel. Before getting into the details and explaining the physical phenomena that cause the above mentioned effects, let us first look from a higher level at how these effects, such as loss and reflections, impact the quality of the signal transmission. This will provide the motivation to look more deeply and understand the physical properties of the system.

Electrical link channels are approximately stationary and band-limited, with very slow changes due to temperature and humidity [36]. However, the variations among different channels in a backplane are large due to different components and physical dimensions of these channels.

The channels shown in Figure 2.3 all belong to the same backplane. The loss slope changes significantly from channel to channel due to different channel lengths, and there is also large variability in the frequency response, due to notches caused by some vias and routing layers used in the backplane. This variability creates a problem for high-speed link design, since we need to find a link architecture that works for all the channels in the backplane.

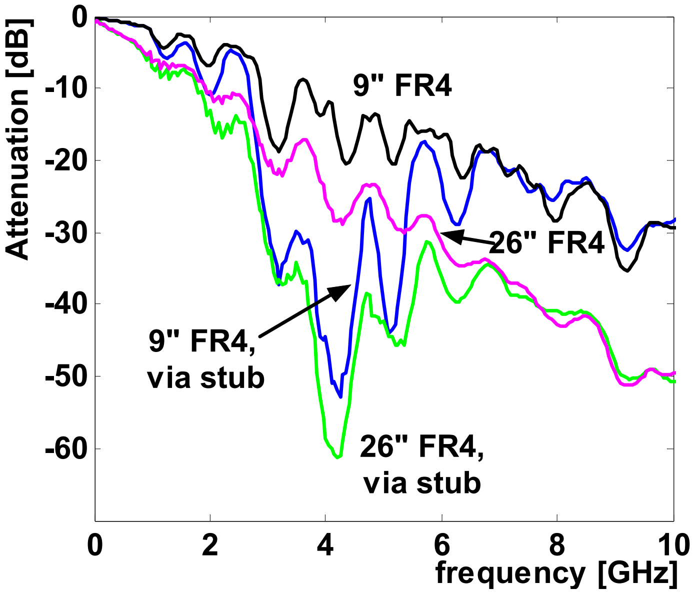


Figure 2.3: Variability in the frequency response of several channels within the same backplane [37]

Despite variability, these channels are mostly low-pass. This means our nice narrow pulse at the input of the channel will be significantly attenuated and much wider at the output of the channel, as shown in Figure 2.4. The dots at the received pulse indicate the symbol-spaced samples. We see that at some even relatively larger latencies there are ripples in the received pulse waveform due to reflections from impedance discontinuities. In addition to that, the first pre-cursor and post-cursor samples are very large due to pulse dispersion from low-pass filtering. Both effects would make it very difficult to correctly detect bits that are transmitted in a sequence.

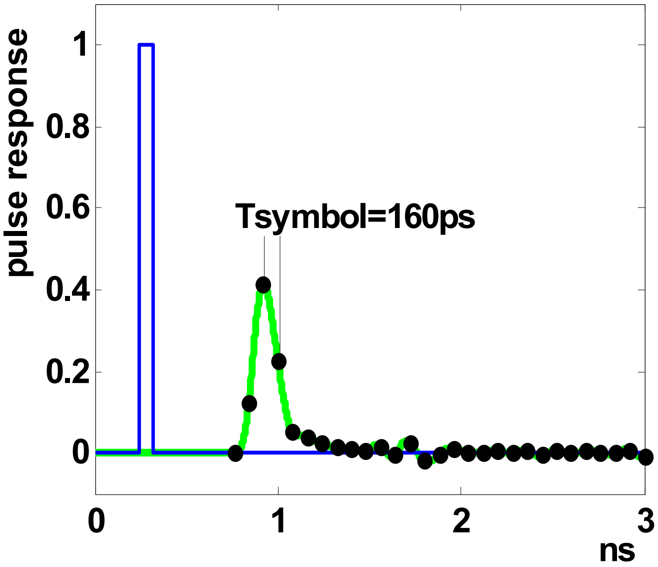


Figure 2.4: The response of the channel to 160 ps wide pulse, [37]

For example, in Figure 2.5, a received sample that corresponds to bit zero, in a one-zero-one pattern sent from the transmitter, drops to only 0.3 due to interference from the previous bit by 0.2 and next bit by 0.1. As a result, this bit is received in error.

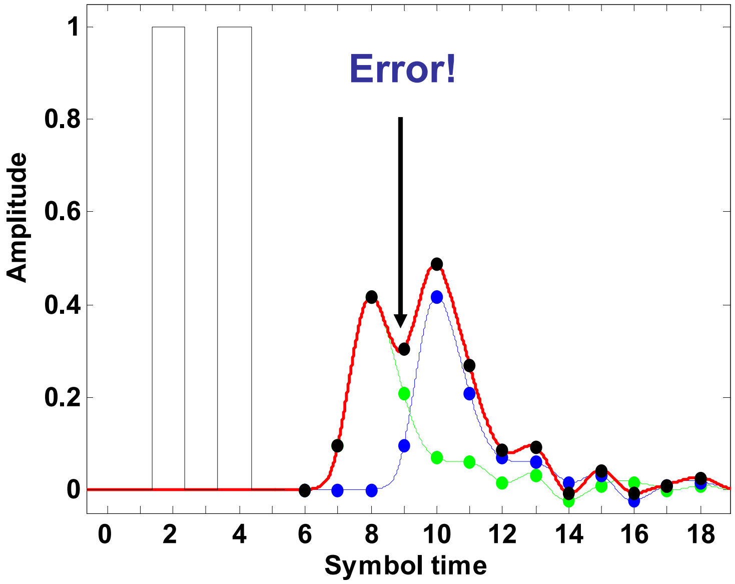


Figure 2.5: The effect of intersymbol interference

This intersymbol interference (ISI) effect is deterministic, since we can always repeat it by transmitting the same data pattern. It is obvious that this effect becomes worse as the width of the transmitted bit decreases. As such, ISI is clearly one of the most significant effects that limit the achievable data rates in high-speed backplane links. In order to better understand the character of ISI we need to look at the physical properties of the backplane system. This analysis will also reveal some additional sources of interference.

2.1.2 Interference

At this point we seek to get a better understanding of the physical effects that cause the degradation of the signal illustrated in the previous section. In backplane systems, interference occurs not only between symbols that travel on the same wire, due to the limited bandwith of the wire, but also between different wires due to electro-magnetic coupling of signals traveling in densely spaced wires (e.g. in the board, connector or package).

2.1.2.1 Inter-Symbol Interference

As we discussed previously, dispersion and reflections are two main causes of ISI. They are based on two fundamentally different mechanisms, so it is worth exploring them both in more detail.

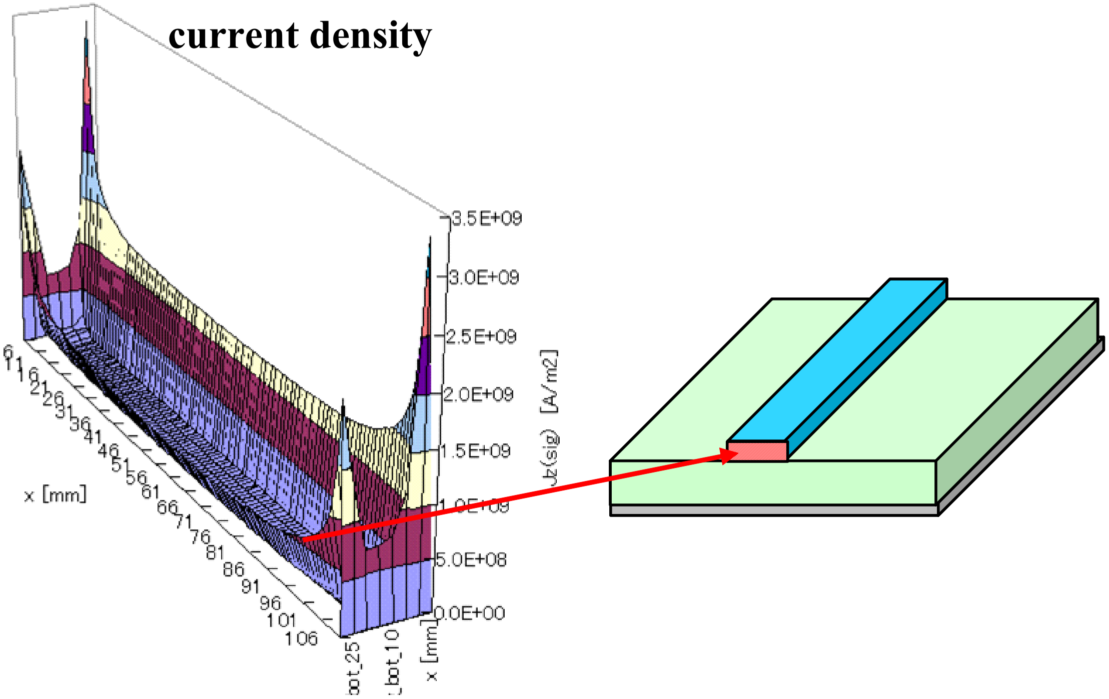
Dispersion

At frequencies well into the gigahertz range, the wire traces in the backplane start behaving like lossy transmission lines. As mentioned earlier, skin-effect and dielectric loss are two contributing effects causing the loss to increase with frequency.

Skin-effect is manifested as crowding of the higher-frequency current toward the surface of the conductor. In Figure 2.6a, we show the current density of a 1 GHz signal at the microstrip cross-section shown in Figure 2.6b. In Figure 2.6b we illustrate the physical definition of a microstrip (a wire trace over a dielectric and ground plane). Higher frequency currents experience more loss, since they incur higher resistance, due to the flow over a smaller cross-section. The resistance, and hence the loss due to skin-effect, are proportional to the square-root of frequency

$$R_{AC}(f) = \frac{2.16\cdot 10^{-7}}{\pi D}\sqrt{p_r \cdot f} \tag{2.1}$$

where D is the wire diameter (Ω/in) and p_r is the relative resistivity of the wire compared to copper [35].



(a) (b)

Figure 2.6: a) Illustration of skin effect - current crowding of 1 GHz signal in a cross-section of a microstrip, b) Microstrip definition

Dielectric loss is attributed to the energy loss in the dielectric surrounding the transmission line. This loss increases proportionally to signal frequency:

$$\alpha_D = \frac{\pi\sqrt{\varepsilon_r}}{c} f \tan\delta \tag{2.2}$$
(2.2).

where *tan δ* is the loss tangent, c is the speed of light and ε_r is the relative permitivity [35].

Dielectric loss is usually specified with only the loss tangent and strongly depends on the type of insulator material, e.g. FR4 (0.035), Polyamide (0.025), GETEK (0.01), Rogers4350 (0.004), Teflon (0.001). Most legacy backplanes use FR4 material, which has the highest loss tangent, while newer backplanes use either Rogers or one of the NELCO materials with lower loss.

Due to the linear dependence in frequency, the dielectric loss dominates over the skin-effect at very high frequencies. The crossover frequency depends on the material properties and dimensions of the trace. The two effects are illustrated in Figure 2.7 for FR4 material, and we see that the crossover occurs at around 500 MHz.

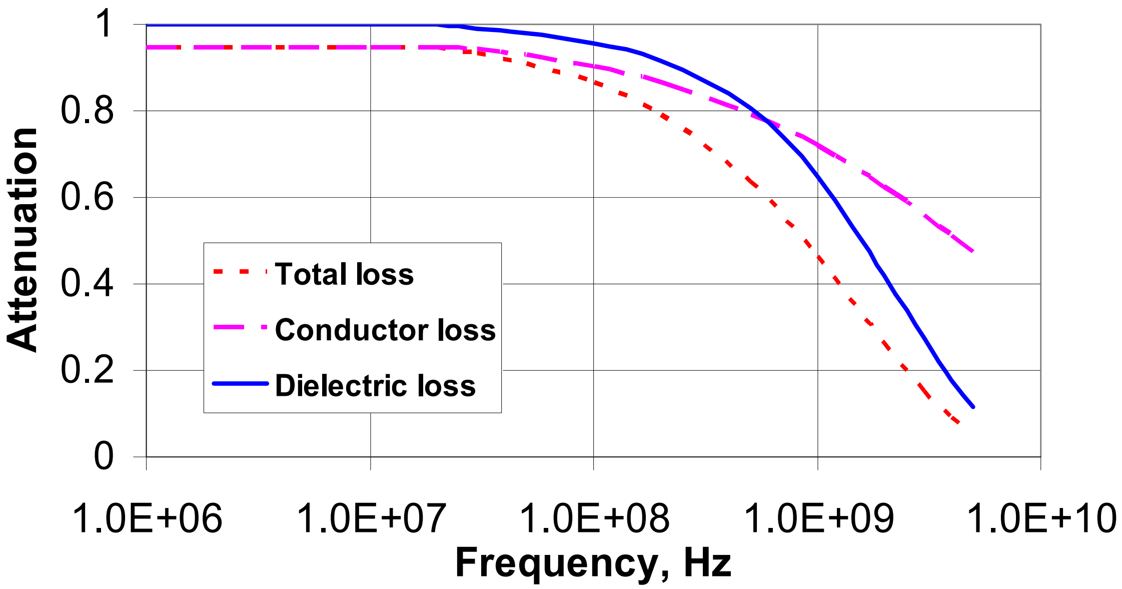


Figure 2.7: Crossover between skin-effect and dielectric loss, FR4 8 mil wide and 1 m long 50 Ω strip line [38]

Reflections

The other ISI component occurs from reflections, i.e. multiple bounces of the signal from impedance discontinuities. A signal transitioning from one transmission line to another line with different impedance, as in Figure 2.8a, suffers a reflection of magnitude

$$R = \frac{Z_2 - Z_1}{Z_2 + Z_1}. \tag{2.3}$$

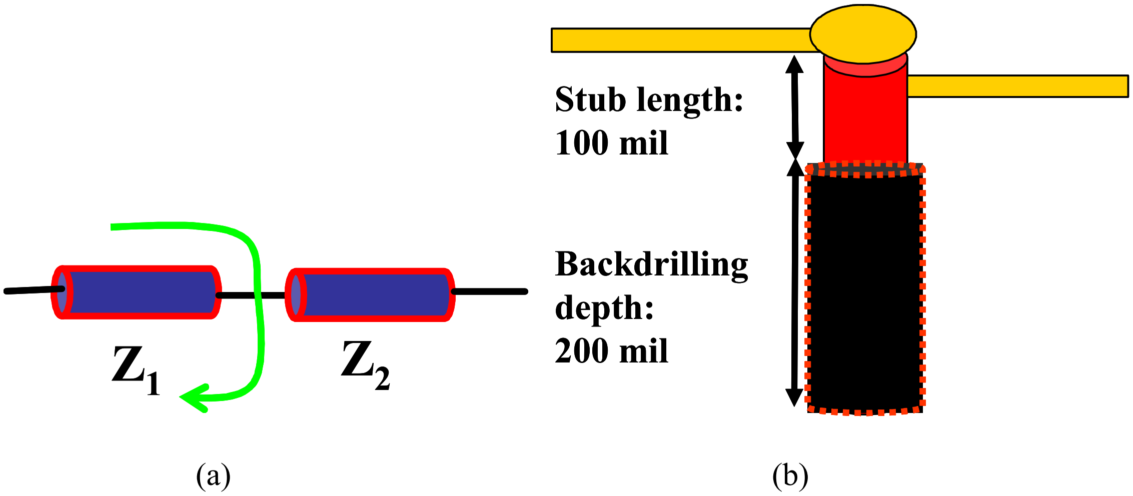


Figure 2.8: Reflections from impedance discontinuities: a) General principle b) Frequency selective impedance discontinuity from the via stub and via length reduction by backdrilling [34]

Some impedance discontinuities can have the same magnitude at all frequencies, for example a discontinuity between a trace with 45 Ω impedance (due to 10% manufacturing error) and a 50 Ω termination resistor at the receiver. From Equation (2.3) we compute that this impedance mismatch causes approximately 5% reflection independent of the signal frequency. Other impedance discontinuities can be frequency dependent, like via stubs in Figure 2.8b. Here the stub acts as a capacitor, which reflects high frequency energy. Recently, manufacturers have started to backdrill the vias in post-production to shorten the length of via stubs which provides better impedance matching and reduces the amount of reflections.

In the backplane channel, reflections from impedance discontinuities occur at several distinct points, causing multiple bounces of the signal, which can persist for a relatively long time, e.g. up to 80 symbols at 5 Gsymbol/s rate[2]. We have already identified backplane via stubs as sources of frequency selective reflections. The second most dominant source of reflections is the frequency dependent impedance discontinuity due to parasitic device capacitance at both the transmitter and receiver. By considering only these two most dominant sources of reflections we can simplify the analysis, as shown in Figure 2.9.

[2] The rule of thumb for the speed of signal propagation is about 150 ps/inch of backplane trace for 8 mil x 1 mil traces in FR4 dielectric (this slightly varies with dimensions of the trace and type of dielectric).

Type A reflections occur either between the connector and transmitter (A_T) or between the connector and receiver (A_R). Other groups of reflections are identified as B, C, D, depending on the bouncing paths. The longest primary reflection path is for type D, which travels across the backplane exactly three times before it hits the receiver. In addition to the primary reflections, there are also their repetitions from the second round of reflections, for example A2 from the repeated round-trip between the connector and transmitter or receiver.

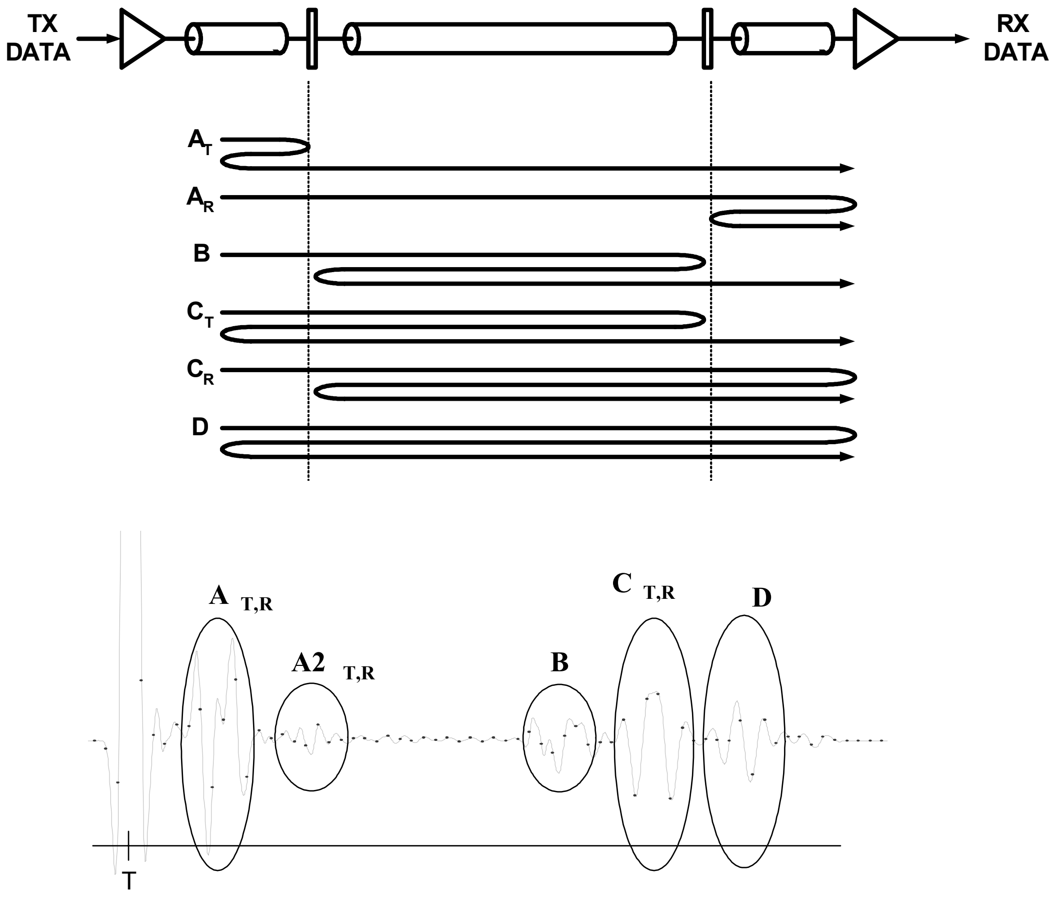


Figure 2.9: Reflections groups within a backplane (top) bounce diagram (bottom) reflections marked in the pulse response (time T marks the location of the main sample) [39]

As we will see later, these very long latencies, e.g. 40 bits for type B reflections in 20” FR4 backplane at 6.25 Gb/s binary transmission, complicate the link architecture since significant hardware resources are required to keep the data in the link for that long and compensate for that many reflections.

2.1.2.2 Inter-Channel Interference (Crosstalk)

The same short traces in connectors and vias, which suffer from impedance mismatches and cause reflections, also suffer from density constraints which cause significant inter-channel interference (crosstalk) between signal lines. The strongest crosstalk occurs between the signal lines in chip packages and connectors [35,40]. Standard crosstalk between transmission lines on the linecard and in the backplane [35] has a much smaller effect.

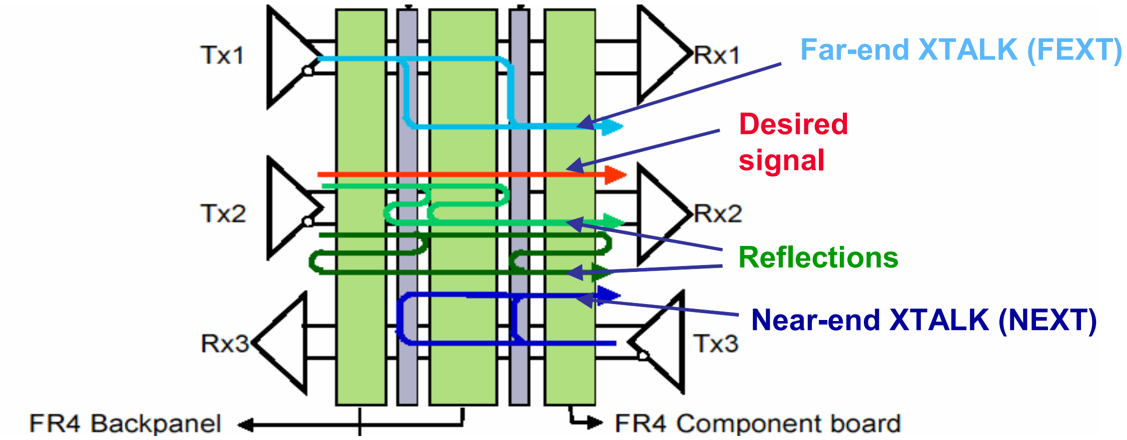


Figure 2.10: Crosstalk definitions [40]

Crosstalk can be divided into far-end (FEXT) and near-end (NEXT) crosstalk. As shown in Figure 2.10, FEXT occurs when the aggressor signal travels in the same direction as the victim. The NEXT occurs when the aggressor signal travels in the opposite direction, and can be much more critical since the strong aggressor signal can couple into an attenuated victim signal in the connector or package located on the receive side. Since crosstalk is caused either by capacitive or inductive coupling of different signal lines (more predominantly inductive in modern dense connectors), it has high attenuation at low frequencies. Due to the low-pass filtering of the channel, FEXT is also attenuated at high-frequencies. Therefore FEXT crosstalk channels are mostly band-pass, while NEXT channels are high-pass.

These frequency characteristics of crosstalk channels are illustrated in Figure 2.11. We see that for frequencies above 4GHz, NEXT becomes stronger than the received signal. This increase in crosstalk energy at higher frequencies presents a big problem for future scaling of link data rates.

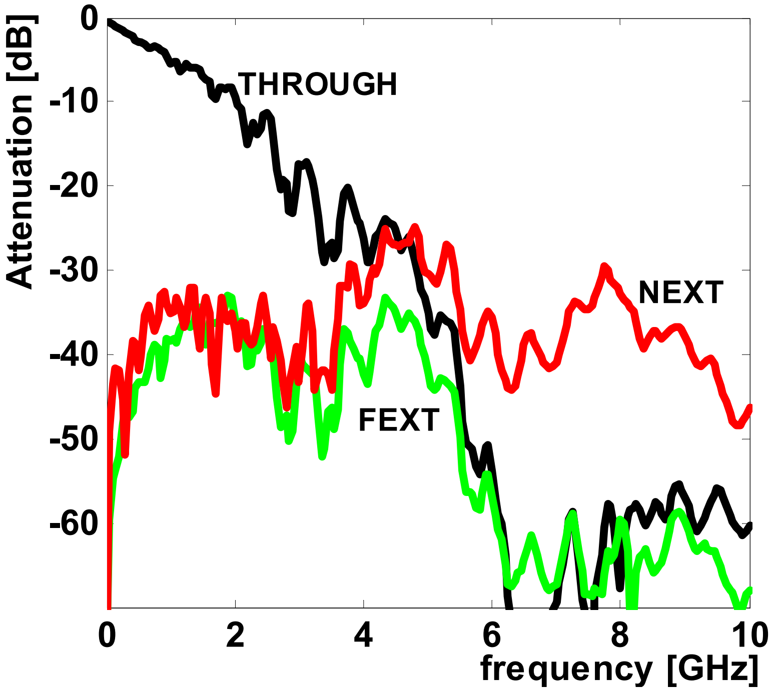


Figure 2.11: Crosstalk frequency response, compared to through channel.

As we will see later in the thesis, although crosstalk will be a serious issue for future links, it is still not the most dominant impairment for current or even next generation links. Currently residual ISI is the dominant error. In this work, we focus on the circuit and system techniques to compensate the ISI, and accurately analyze the effects of both ISI and crosstalk. Techniques for crosstalk suppression, although theoretically well developed in digital communications [41-44] are currently prohibitively complex to implement. We address some approaches to crosstalk in Appendix C in the context of MIMO systems with transmit pre-emphasis [45].

In this section, we have seen the properties of different interference sources in high-speed links. These are deterministic effects that can always be replayed by repeating the transmitted data patterns and are currently limiting the performance of the high-speed link systems. From the perspective of communication theory we can always compensate for these effects so in the end, noise, rather than deterministic effects like interference, imposes a fundamental limit on the link data rate. To investigate these fundamental limits and characterize the performance of high-speed links, we need to understand the link-specific noise sources.

2.2 Noise Sources

While in many communication systems components can be designed so well that thermal noise is the real limiting factor, in high-speed link systems, high-throughput requirements yield circuits that result in non-negligible system noise. This system noise is located in both time and voltage domains. For example, the phase noise (i.e. jitter) of the transmitted signal or the received sampling clock is timing noise. Examples of voltage domain noise include limited sampling resolution, thermal device noise and supply noise. In a strict theoretical sense only thermal voltage noise and phase noise resulting from thermal voltage noise can be labeled *noise*. All other terms, such as supply noise or phase noise due to supply noise, are actually interference from a large number of signal paths on a chip, communicating data between the logic stages. However, given that in most cases, the number of these events is very, very large and often intractable, we can consider them as random events and call them *noise*.

2.2.1 Voltage Noise

This section describes the link-specific noise sources that originate in the voltage domain. It first looks at the sources of thermal noise in the system, and then describes the impact of supply and substrate noise. Finally, it examines the noise that is a result of the limited resolution of either the receiver or the equalizer coefficient settings.

2.2.1.1 Thermal Noise

Thermal noise has traditionally been neglected in high-speed links, since it was assumed that the thermal noise of a resistor and transistor device is very small when compared to the signal magnitude. However, with increased link signaling rates these noise sources are slowly emerging as potentially important because as the bandwidth of the signal increases, it collects more noise power.

The root causes of thermal noise in links are 50 Ohm terminations at the receiver. The device noise of receiver circuits also adds several dB of noise figure to the termination noise level. Since this additional device noise strongly depends on the type of the receiver, we just look at the device noise of the input differential pair, common to many types of link receivers. An example analysis of the main noise components, including the noise in the receiver termination and receiver pre-amplifier, is shown in Table 2.1 illustrating the ballpark values of thermal voltage noise for a typical link.

Table 2.1 Random voltage noise, termination and receiver pre-amplifier [46]

Noise source	σ [mV]	PSD [dBV][3]	BW=5 GHz, I_{dc}=2.5 mA, γ=2.5, R_{load}=100 Ω
R_{term}=50 Ω	0.035	-90	$\sigma^2 = \frac{4\mathrm{k}\cdot\mathrm{T}\cdot\mathrm{BW}}{\mathrm{R}_{\mathrm{term}}}\cdot\left(\frac{R_{term}}{2}\right)^2$
Thermal drain	0.078	-82	$\sigma^2 = 4\mathrm{k}\cdot\mathrm{T}\cdot\gamma\cdot g_{d0}\cdot\mathrm{BW}\cdot\mathrm{R}_{\mathrm{load}}$
Shot noise	0.2	-74	$\sigma^2 = 2q\cdot\mathrm{I}_{dc}\cdot\mathrm{BW}\cdot\mathrm{R}_{\mathrm{load}}^2$
Gate noise	0.2	-74	$\sigma^2 = \frac{4\mathrm{k}\cdot\mathrm{T}\cdot 2\gamma\cdot\mathrm{BW}}{5g_{d0}}$

For example, the total input referred random noise for a receiver with 5 GHz noise bandwidth has an rms value of roughly 0.3 mV, which is roughly 40 dB down from the equalized signal level at the receiver.

Although the input bandwidth of the link is limited by the on-chip parasitics (due to transmit and receive circuits and electro-static discharge protection circuits), we assume that this bandwidth will always scale approximately with the signaling rate. In that scenario thermal noise spectral density of the resistive termination is around $(1\ \mathrm{nV})^2$/Hz, which is ~70 dB down from the peak output energy of a typical link transmitter at 10 GHz Nyquist frequency, with transmitter output voltage swing constrained to ±500 mV. This very high transmit signal-to-noise ratio (SNR) indicates that in order to truly estimate the performance of the link we need to consider other noise sources as well.

2.2.1.2 Supply and Substrate Noise

While the supply and substrate noise do not directly impact the performance of the link, they do so indirectly by inducing jitter in transmit and receive timing generation loops [47], and by modulating the input-referred receiver offset [48]. Many theoretical and experimental studies of substrate and supply noise were conducted [49, 50], but they are

[3] The PSDs are multiplied by the Nyquist frequency, to obtain the noise power at each frequency, hence the unit [dBV] instead of customary [dB V^2/Hz].

both application specific and lack the full statistical and spectral description of the noise. Alon in [51] first presented the circuits and techniques that enable measurements of full statistical and spectral properties of the supply noise, on an example of a high-speed link. This supply voltage distribution as a function of time is shown in Figure 2.12. We can recognize the periodic deterministic effect of the on-chip clocks, but there is also a random component of the supply noise.

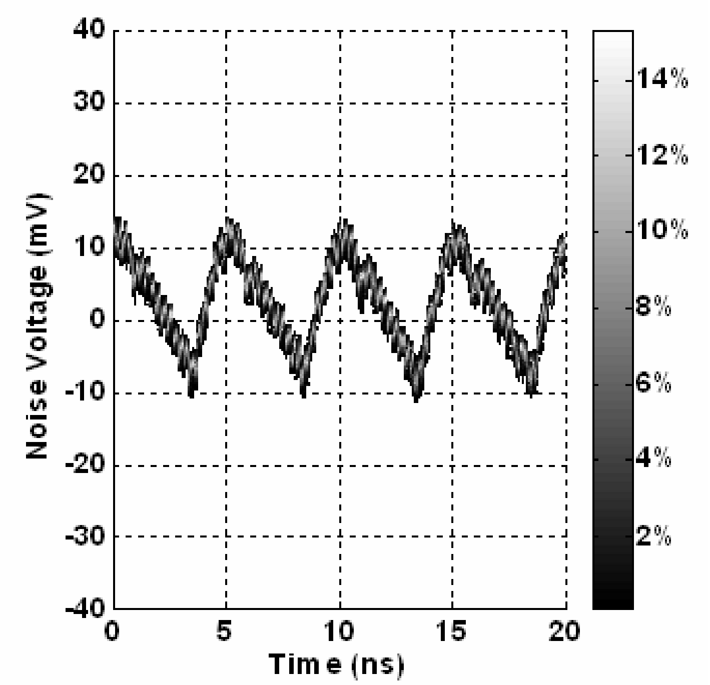


Figure 2.12: Supply noise distribution measured at the PLL supply [51]

As we will see later in this section, in order to estimate the impact of supply noise on timing jitter, we really need to look at the noise spectrum. Although results in [51] indicate that supply noise is cyclo-stationary and therefore cannot have a uniquely defined power spectral density [52,53], we can assume to the first order that the cyclo-stationarity is lost when the supply noise is transferred to timing jitter[4].

If we disregard the timing reference in obtaining the power spectral density of noise, we *stationarize* it [54]. In Figure 2.13 we can see the power spectral density of the stationarized supply noise. The spikes at 200 MHz, 400 MHz and other frequencies

[4] If a noisy cyclo-stationary process is filtered with a band-limited filter, the noise at the output is stationarized to first order [54], i.e. its spectrum becomes time-invariant.

represent the random switching events due to the the flow of random data through the latches and flip-flops, modulated by different clocks in on-chip clock domains.

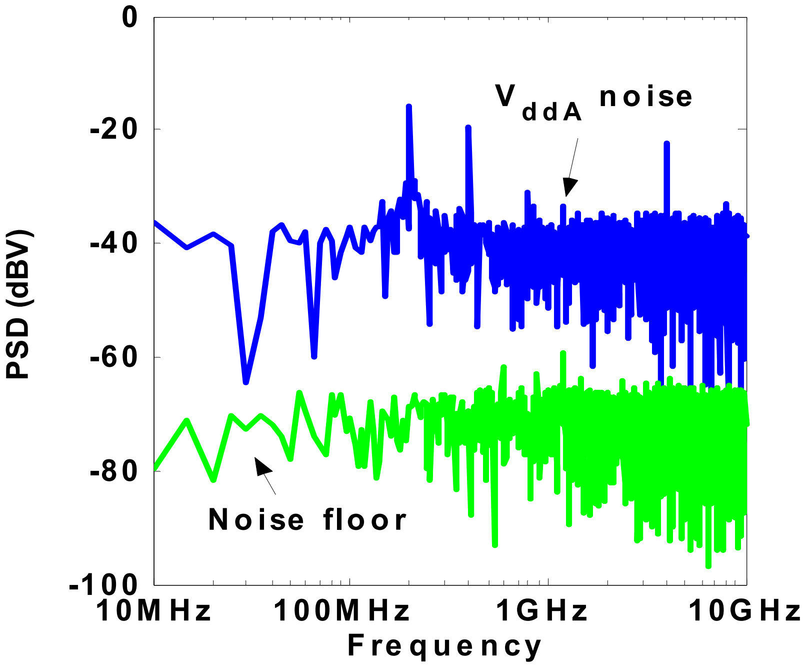


Figure 2.13: Power spectral density of PLL supply noise, including the average noise from on-chip clocks [51]

Except for these spikes, the spectrum of the random supply noise, from random switching of data in the logic stages within the clock cycle, between the flip-flops, is fairly white and stretches to very high frequencies, 10 GHz in Figure 2.13. This is because the switching events between the gates are separated by gate delays of around 50 ps in 0.13 µm technology, which indicates that noise events up to roughly 20 GHz are possible.

We will see later in this chapter how the supply noise impacts the timing noise in links. In addition to that, it also impacts our ability to efficiently cancel the built-in receiver offsets, as we will see in the next section.

2.2.1.3 Receiver Resolution

Sampling resolution is the minimum voltage level that can be distinguished by the receiver comparator in the absence of other noise sources. It is determined by several factors including receiver static offset, input-referred supply noise and the input voltage (overdrive) required for the comparator to obtain the decision within a certain period of time[5]. Static offset occurs due to transistor mismatch from statistical process variations [56]. While it has a statistical nature, the values are fixed once the chips are fabricated. These offsets can be corrected to the first order, but this same mismatch limits the ability of the receiver to reject on-chip supply noise [48]. The value of these noise sources depends on the design of the receiver. In most high-speed links the input is fed into a regenerative stage with little pre-amplifier gain. For these systems, the uncorrected offsets can be ±60 mV and the residual error is non-negligible. We will use ±10 mV as the required sampling resolution based on the experimental data in [57,58]. This value represents the residual error plus the required overdrive.

2.2.1.4 Estimation Errors/Quantization

Many links use analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) to help reduce the ISI of the channel. For these designs quantization errors in the transmitter and receiver (DAC and ADC step size) add noise to the system. These errors are uniformly distributed with σ determined by the size of the quantization steps. In our analysis we will use 10 mV steps and 10% estimation errors, which have an rms value of of a few mV[6].

As we will see in Chapter 4 when we talk about link performance analysis, including all the voltage noise sources that we described is relatively straightforward as long as we take into account their statistics accurately.

2.2.2 Timing Noise

Timing noise has traditionally been considered separately from voltage noise. In this section we first show how our new link model maps the timing noise into effective voltage noise at the receiver input, and then we discuss the origins of this timing noise.

[5] Since comparators are regenerative elements [55], their gain depends on the time we wait for the result. So for very small input signals we need to wait for a longer time to get an output signal that is large enough to be considered “digital”. When the input signal is so small that the comparator takes nearly infinite time to resolve we say that the comparator is metastable [59].

[6] Detailed derivation of the propagation of the estimation error and quantization noise through the system and to the input of the receive slicer is presented in Appendix A.

2.2.2.1 Mapping to Voltage Noise

The motivation for mapping the timing noise to effective voltage noise can best be understood in an example shown in Figure 2.14. We would ideally like to position the receiver sampler at the time instant with the biggest voltage margin, i.e. a point with the biggest eye opening, as indicated by the green line.

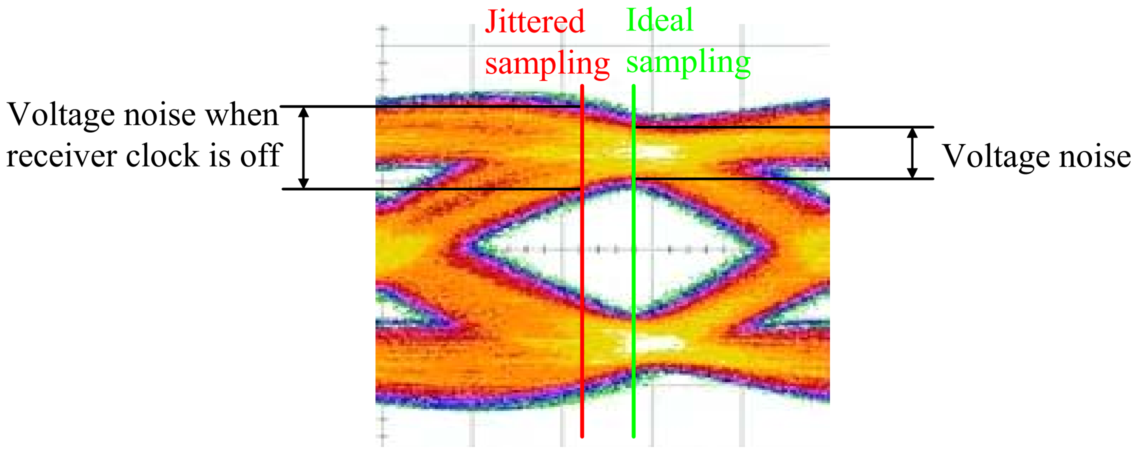


Figure 2.14: Received eye diagram with effective voltage noise due to receiver sampling jitter

In reality, however, since there is timing noise in the system, the sampling may be slightly off as indicated by the red line, and we see that at that point we have a larger voltage uncertainty than at the ideal sampling point. How much noise is added depends on the properties of the channel, the magnitude of the timing jitter and also the data pattern. If we can somehow characterize this effect, we will be able to map the timing noise to effective voltage noise and aggregate all noise sources into one domain. This combination would enable a compact link performance model.

The goal is to map the effect of both transmitter and receiver jitter to voltage noise at the receive slicer (i.e. comparator) input. As we have seen in Figure 2.14, receiver jitter modulates the sampling position of the slicer, while the transmitter jitter modulates the position of both the beginning and the end of the transmitted symbol. We start the mapping procedure by first looking at the effect of transmitter jitter.

Figure 2.15 shows a decomposition of a noisy symbol into a noiseless symbol *(a)* and two noise pulses[7] caused by the jitter *(b)*. Independence of the jitter process, ε, from the data stream, b, implies the independence of signals *(a)* and *(b)* in Figure 2.15. Since the two noise pulses are much narrower than the impulse response of the channel filter and the reference symbol pulse, we can approximate them with delta functions as we did in [37]. When such noisy symbols pass through the channel filter, our approximation by delta functions is effectively equivalent to a zero-order approximation of the convolution integral.

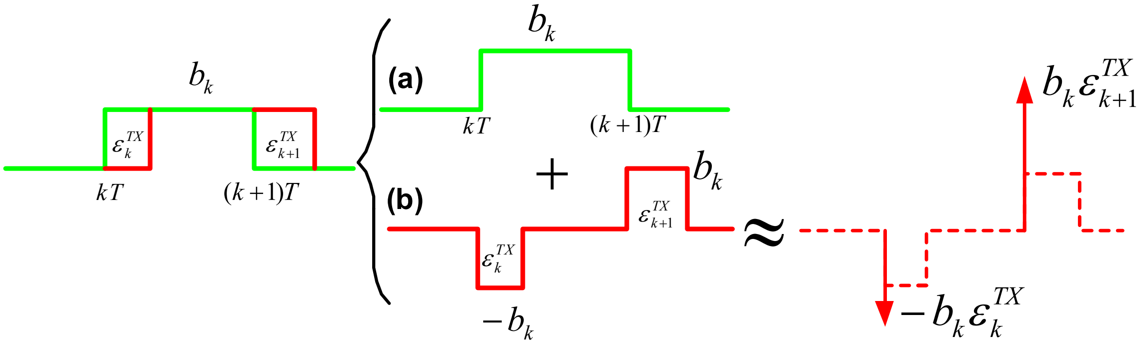


Figure 2.15: Jittered pulse decomposition. A symbol transmitted with jitter is converted to a symbol with no jitter *(a)*, plus a noise term where the widths of the noise symbols *(b)* are equal to $\varepsilon_k{}^{TX}$ and $\varepsilon_{k+1}{}^{TX}$.

We can see from this model that high frequency transmitter jitter modulates the energy of the transmitted symbol, since jitter pulses in Figure 2.15 are then uncorrelated. This is very undesirable because when uncorrelated, their powers add as they propagate through the channel, increasing the effective jitter-induced voltage noise.

Low frequency transmitter jitter is less detrimental because it shifts larger portions of data pattern without changing much the energy of the individual symbols. This is because the jitter pulses in Figure 2.15 are then correlated and we get partial cancellation when they are mapped to voltage noise. Similar to low frequency transmitter jitter, receiver jitter is equivalent to the shift of the whole transmit sequence, since time references are relative, as shown in Figure 2.16.

Using the model in Figure 2.15 we will formally derive the autocorrelations of the effective voltage noise from both transmitter and receiver jitter, but let us first look at the

[7] In real high-speed link system implementations, basis functions are usually in the form of a square pulse as considered here, but the arguments are valid for any arbitrary pulse shape.

results of the model.

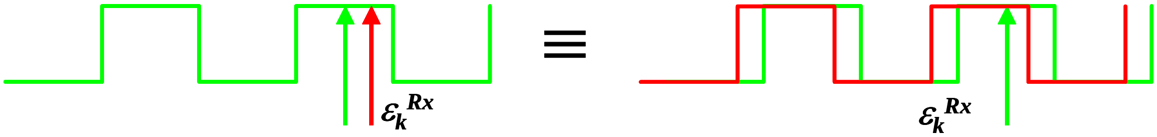


Figure 2.16: Equivalence of receiver jitter and shift of the whole transmit sequence

In Figure 2.17 we can see that the power spectral density of the voltage noise induced by transmitter jitter is larger in case where the transmitter jitter is white than in case where it is correlated. In our example, we use the correlated jitter produced by a PLL with a supply noise bandwidth of 100 MHz.

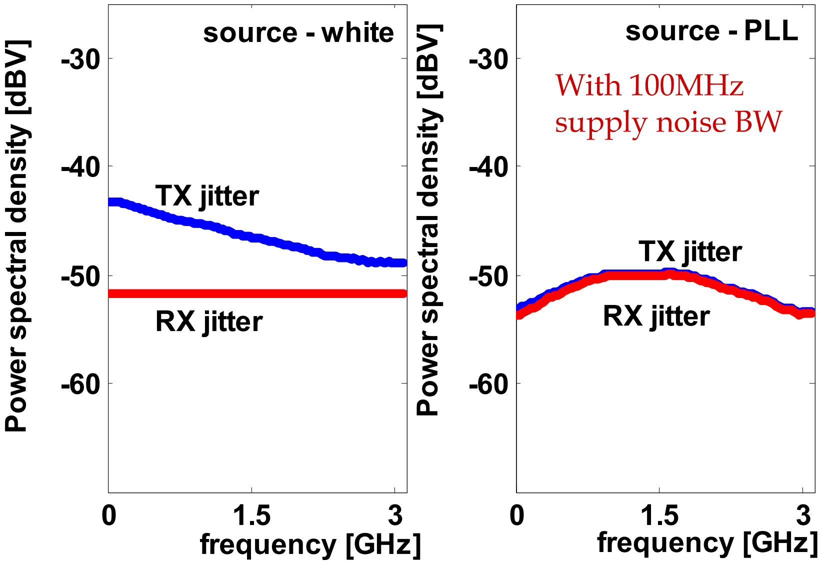


Figure 2.17: Power spectral density (PSD) of voltage noise at sample time, due to transmitter and receiver jitter

Assuming both transmit and receive jitter are white and Gaussian with 1.4 ps rms at 6.25 Gsymbol/s, they induce effective noise voltages at the slicer input of 3 mV and 1.6 mV rms, respectively[8]. In the case of correlated receiver jitter, the power spectral density of the effective noise from receiver jitter changes slightly from white to correlated, but the total power does not change. An important observation for the correlated case is that the power spectral densities of voltage noise induced by the

[8] Transmitter output swing is ±500 mV, with the channel as in Figure 2.11.

transmitter and receiver jitter are almost identical. This supports the intuitive explanation that the receiver jitter and relatively low frequency transmitter jitter induce similar effective voltage noise.

More formally, we can define the transfer function from jitter to input referred noise. With the assumptions stated in Figure 2.15, we create the system model, shown in Figure 2.18, where noiseless symbols pass through the standard channel pulse response block *p(nT)*, while the noise pulses pass through the impulse response block offset by half the symbol time *h(nT+T/2)*. As we saw earlier, this model can be used to estimate the effect of receiver jitter by shifting the entire transmit sequence by the amount of receive jitter. This corresponds to making ε the same for all k for each sequence.

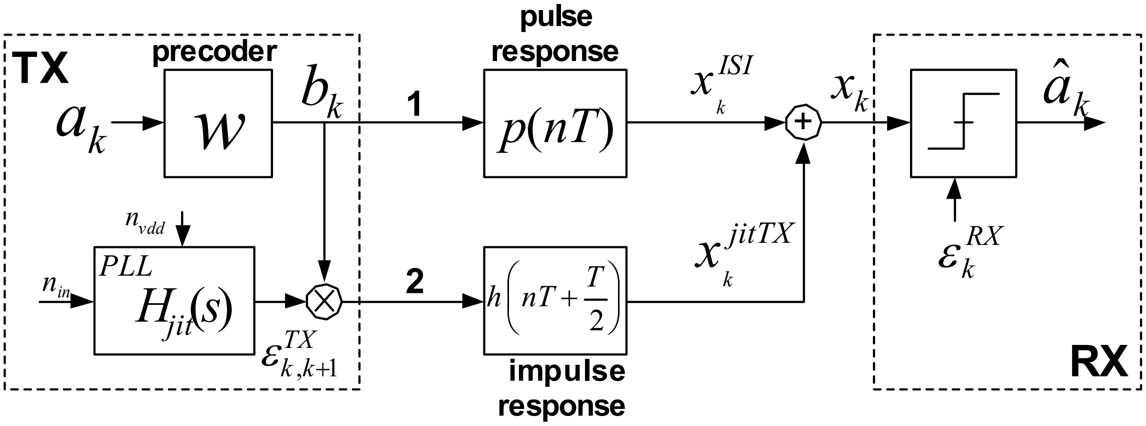


Figure 2.18: System model with transmitter and receiver jitter. Since the noise pulses caused by transmitter jitter are narrow, they are represented by impulses located at the edges of the symbol (half a symbol from the symbol sample point), and after passing through the channel act as independent additive noise to the input data.

The resulting expressions for samples at *kT*, Equation (2.4), corrupted by ISI, Equation (2.5), and voltage noise due to transmit and receive jitter, Equations (2.6) and (2.7) are:

$$x_k = x_k^{ISI} + x_k^{jitTX} + x_k^{jitRX} \tag{2.4}$$

$$x_k^{ISI} = \sum_{n=-sbS}^{sbE} b_{k-n} p_n \tag{2.5}$$

$$x_k^{jitTx} = \sum_{n=-sbS}^{sbE} b_{k-n}\left(h_{n-1}\varepsilon_{k-n+1}^{TX} - h_n\varepsilon_{k-n}^{TX}\right) \tag{2.6}$$

$$x_k^{jitRx} = \varepsilon_k^{RX} \sum_{n=-sbS}^{sbE} b_{k-n}\left(h_n - h_{n-1}\right) \tag{2.7}$$

where ε_k^{TX} and ε_k^{RX} are samples of the transmit and receive jitter, b_k the value of the transmitted symbol, with *sbS* and *sbE* as start and end indices of the impulse response sequence, and $p_n=p(nT)$ and $h_n=h(nT+T/2)$ samples of pulse and impulse responses of the channel, at nT and $nT+T/2$, respectively.

Now we can derive the autocorrelation functions of the transmit, Equation (2.6), and the receive, Equation (2.7), jitter-induced voltage noise, following the procedure that we used earlier in [60].

To help compensate the ISI, the transmitted signal b_k is often a filtered version of the true input a_k. In case of linear transmit precoding, $b_k=\underline{w}^T\underline{a}_k$, where $\underline{w}$ is the precoding vector and $\underline{a}_k$ is the transmit alphabet vector, the autocorrelation functions of the voltage noise from transmit and receive jitter can be shown to be of the form[9]:

$$R_{x^{jitTX,RX}}(m) = \underline{w}^T E\left(\mathbf{A}(k)\underline{J}^{TX,RX}(k)\underline{J}^{TX,RX}(k+m)^T\mathbf{A}(k+m)^T\right)\underline{w} \tag{2.8}$$

with

$$\mathbf{A}(k) = [\underline{a}_{k+sbS} \ldots \underline{a}_{k-sbE}];\;\; \underline{a}_k = [a_{k+preW} \ldots a_{k-postW}]^T$$

$$\underline{J}^{TX,RX}(k) = \left[J_{-sbS}^{TX,RX}(k) \ldots J_{sbE}^{TX,RX}(k)\right]$$

$$J_n^{TX}(k) = \varepsilon_{k-n+1}^{TX}h_{n-1} - \varepsilon_{k-n}^{TX}h_n\,;\; J_n^{RX}(k) = \varepsilon_k^{RX}(h_n - h_{n-1})$$

where **A**(k) is the transmit alphabet matrix, and *preW* and *postW* are the number of taps before and after the main equalizer tap.

From Equation (2.8), the autocorrelation can be compactly written as:

[9] Note that in this form, the variance of voltage noise due to transmit and receiver jitter ($R_x^{jitTX,RX}(0)$ sample of the autocorrelation in (2)) is actually a square of the *l2* norm in $\underline{w}$ since the inner matrix is positive semi-definite. This convex form will be used later in Chapter 3.

$$R_{x_{jitTX}}(m) = \underline{w}^T \mathbf{S}_{\mathbf{m}}^{\mathbf{TX}} \underline{w}\,,$$
$$\mathbf{S}_{\mathbf{m}}^{\mathbf{TX}} = E_a \sum_{j=-sbS}^{sbE} \sum_{k=-sbS}^{sbE} \mathbf{I}_{\mathbf{m+j-k}} [h_{j-1}\ h_j] \begin{bmatrix} R_{\varepsilon^{TX}}{}^{(m+j-k)} & -R_{\varepsilon^{TX}}{}^{(m+j-k-1)} \\ -R_{\varepsilon^{TX}}{}^{(m+j-k+1)} & R_{\varepsilon^{TX}}{}^{(m+j-k)} \end{bmatrix} \begin{bmatrix} h_{k-1} \\ h_k \end{bmatrix} \tag{2.9}$$

$$R_{x_{jitRX}}(m) = \underline{w}^T \mathbf{S}_{\mathbf{m}}^{\mathbf{RX}} \underline{w}\,,$$
$$\mathbf{S}_{\mathbf{m}}^{\mathbf{RX}} = E_a R_{\varepsilon^{RX}}(m) \sum_{j=-sbS}^{sbE} \sum_{k=-sbS}^{sbE} \mathbf{I}_{\mathbf{m+j-k}} [h_{j-1}\ h_j] \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} h_{k-1} \\ h_k \end{bmatrix} \tag{2.10}$$

where E_a is the average energy of the transmit alphabet, a, $R_{\varepsilon^{TX}}(m) = E(\varepsilon_k^{TX} \varepsilon_{k+m}^{TX})$ and $R_{\varepsilon^{RX}}(m) = E(\varepsilon_k^{RX} \varepsilon_{k+m}^{RX})$ are m^{th} samples of the autocorrelation functions of transmit and receive jitter, respectively, and $\mathbf{I}_{\mathbf{n}}$ is the identity matrix shifted right by n places. In this derivation, we assume that both jitter and discrete data processes are stationary and that descrete data sequence is uncoded (i.e. uncorrelated).

2.2.2.2 Origin of Timing Noise – PLL

As we saw earlier in the analysis of jitter propagation through the channel, the magnitude of jitter-induced voltage noise strongly depends on the jitter spectrum. In order to understand what the jitter spectrum looks like, we need to look at the source of timing generation on a chip, which is usually a PLL. A typical second-order PLL is shown in Figure 2.19, together with dominant noise sources. The heart of the PLL is a voltage controlled oscillator (VCO), which generates the clock that is distributed to the output of the PLL via the clock buffer and then fed-back through an optional divide circuit to be compared with the reference clock. The phase detector generates the error signal which is then filtered by the loop filter to create the control voltage for the VCO and steer its phase to align it with the reference clock.

The effect of noise on the PLLs has been extensively studied [47, 61, 62]. While Hajimiri in [61] and Demir in [62] mostly focused on the performance of the voltage controlled oscillators and cyclo-stationary aspects of phase noise, caused by device noise, Mansuri in [47] has recently illustrated the impact of the most significant noise sources in high-speed link PLLs, namely supply noise and reference clock phase noise. Since this noise is critical in most loops we repeat some of this analysis next.

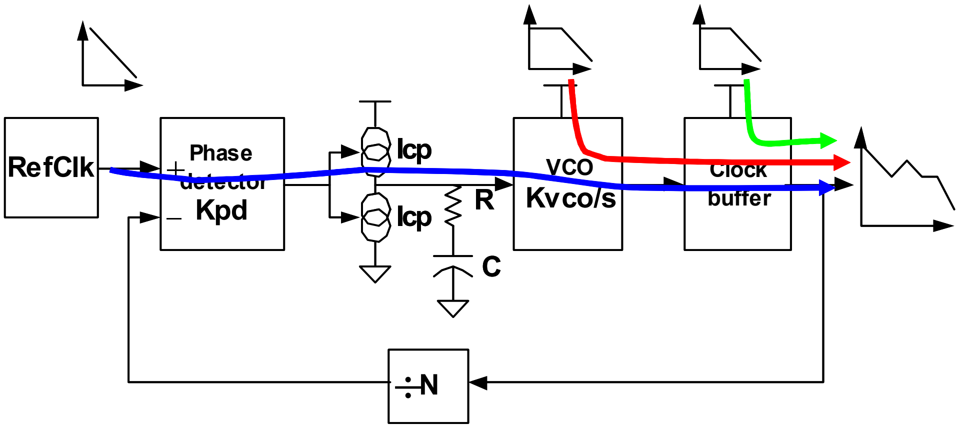


Figure 2.19: The most significant noise sources for a high-speed link PLL: Input clock jitter, VCO supply noise and clock buffer supply noise. Each is transformed through the loop to have a different spectral response.

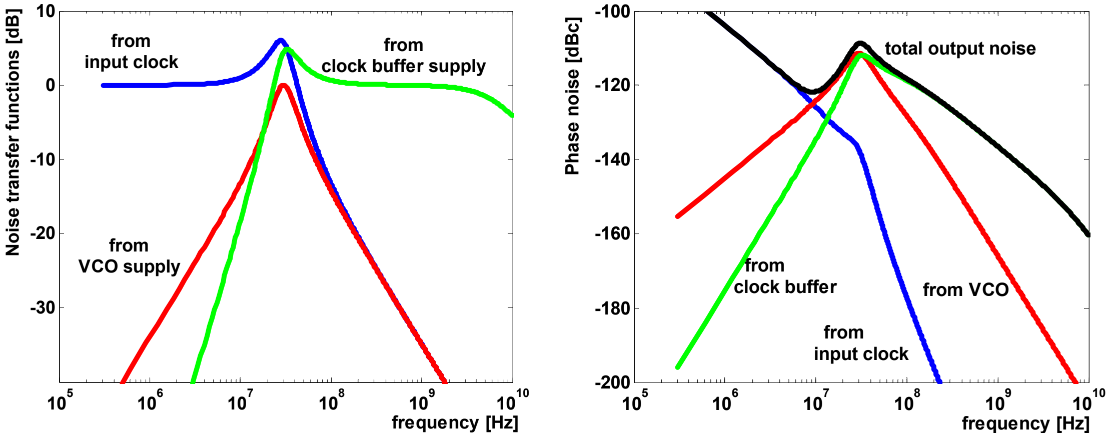
Approximating a PLL as a linear second order system, we can obtain the noise transfer function from the power supply of the VCO and the clock buffer to the output of the PLL as well as the transfer function from the reference clock input to the output of the PLL. Figure 2.20a shows the transfer functions of the noise.

We see that VCO supply noise is band-pass filtered to the output, while clock buffer supply noise is high-pass filtered[10], and reference clock noise is low-pass filtered. To find the magnitude of the jitter, we need both the power spectrum of the supply noise, and the sensitivity of both the VCO and the clock buffers to their supply noise. Even if we assume the supply noise is white, the PLL jitter cannot be assumed white, and hence autocorrelation of the jitter samples has to be taken into account in the noise analysis. In addition, since the supply noise is bounded, the approximation of the jitter distribution should be bounded as well.

Now that we have the noise transfer functions we can use the measured power supply spectral density shown in Figure 2.13 to obtain the phase noise spectrum at the output of the PLL, similar to the procedure that we used in Figure 2.20b to plot the phase noise assuming the power supply filtered to 100 MHz of bandwidth. From the jitter

[10] Actually, the clock buffer supply noise transfer function is also band-pass but with much higher bandwidth determined by the total delay of the clock buffer. The gain is determined by the number of buffer stages, i.e. samples of the supply noise that add to the jitter at the output of the buffer chain.

spectrum, we can then compute the jitter autocorrelation functions used in Equations (2.9) and (2.10).



(a) (b)

Figure 2.20: a) Noise transfer functions from different supplies to the output of the PLL, b) Power spectral densities of PLL phase noise components assuming the supply noise is uniformly distributed with 20 mV p-p and filtered by the on-chip supply network with bandwidth of 100 MHz. For our example design [39] the VCO sensitivity is 0.15 ps/mV, and the buffer sensitivity is 0.75 ps/mV.

Unfortunately, the PLL is not the only source of timing noise in the system. In many applications, to avoid excessive wire, package and connector pin density, we do not explicitly send the timing reference signal to the receiver on a separate wire for every link[11]. In the case of high-speed serial links we usually extract the timing information from the incoming data stream, since the separate timing reference wire per link represents 100% overhead in terms of the aggregate cross-sectional bandwidth. In addition to this we still need to have some circuits that compensate for phase difference between the data and time-reference wires due to wire length mismatch. For these reasons, high-speed links usually have a CDR loop which incorporates a PLL and some extra circuits to synchronize the receiver with the incoming data stream, by extracting the time reference from the data stream. As we will see in the next section, these extra circuits add some timing uncertainty to the jitter which exists at the output of the PLL, increasing the total amount of timing noise in the system.

[11] This is called source-synchronous clocking and is used mostly in parallel links like memory or system busses since then the relative wiring overhead is small compared to the width of the buss, and it simplifies receiver design, as in the Rambus memory interface.

2.2.2.3 CDR Analysis

There are many ways to synchronize the receiver with the incoming data stream [63]. One of the most common techniques to do this is to oversample the incoming symbol by a factor of two [64], as shown in Figure 2.21. One comparator samples at the center of the symbol, denoted by d_n and d_{n-1}, and one at the symbol transitions (e_n). Since transitions have the highest voltage gradient vs. time, the timing error information derived from transition samples will have the highest quality. By using the data samples to detect the transitions and using the edge samples to detect the timing error the receiver can always detect if it is early or late and choose the correct phase to align itself with the incoming data.

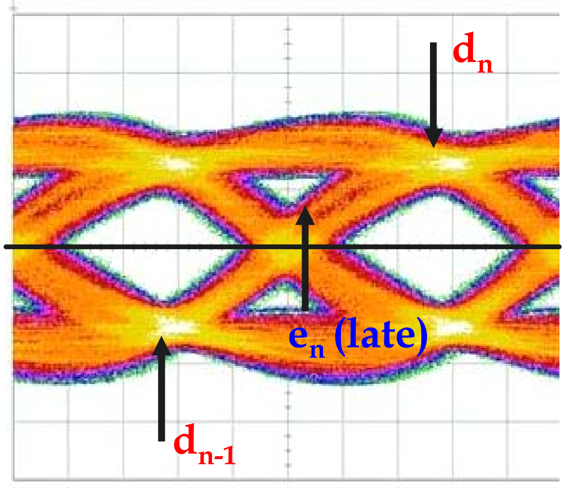


Figure 2.21: Timing detection by 2x oversampling

One of the most popular CDR techniques involves the use of identical samplers for data slicing and phase detection, in order to cancel the sampler's setup time. In this thesis, we will use that approach for phase detection together with a simple first-order loop shown in Figure 2.22.

In this circuit, the phase mixer creates a number of finely spaced phases from the PLL and the control loop selects the phases for edge and data clocks, keeping them in quadrature and aligned with the incoming data stream. Using a comparator identical to the data slicer as a phase detector simplifies the implementation of the loop, since the setup time of the data comparator is compensated for automatically and the rest of the loop control is digital. This binary type of phase detector results in a bang-bang control loop, which is non-linear and in general very hard to analyze.

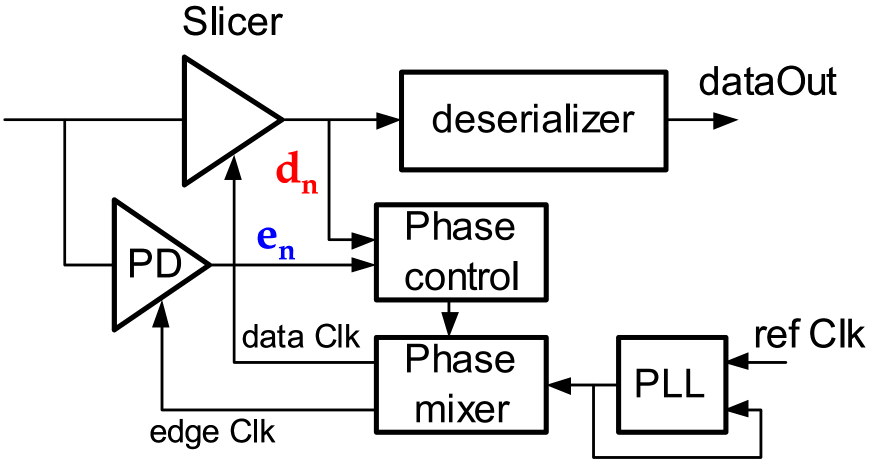


Figure 2.22: Receiver with first-order CDR loop

In the literature, there have been two separate methodologies to characterize this type of CDR system. The communications camp uses Markov chain analysis of the loop [65-67], while IC designers usually linearize the loop and treat it as linear control system [23,68]. Both approaches have their limitations.

As illustrated in Figure 2.23, possible phase positions of the recovered clock are shown as states in a Markov chain. The transitions between the states are governed by the *hold*, *up* and *down* decisions to hold, advance or retard the current phase ϕ_i. In an environment with noise and ISI these transitions have some probabilities $p_{hold,i}$, $p_{up,i}$ and $p_{dn,i}$ associated with them for every phase state.

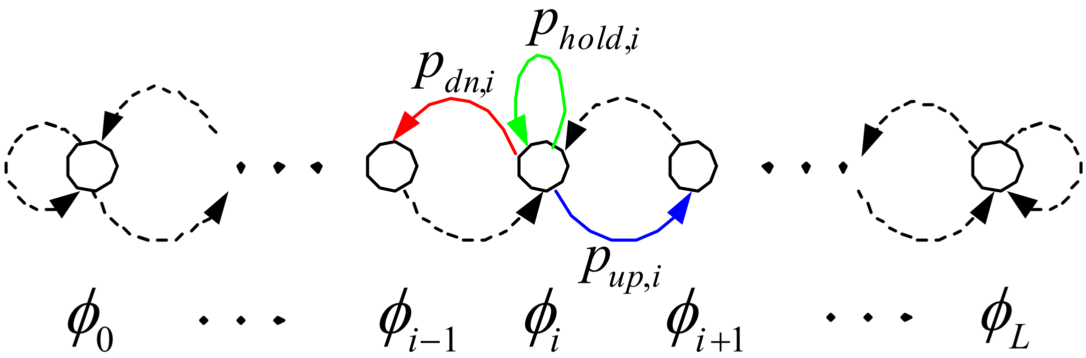


Figure 2.23: First-order Markov chain phase-state model. Each state represents a different phase position, and the arcs are the probability of transition, given that position.

The probabilities are generated by filtering (using a moving-window, random-walk filter [69], or an accumulate-reset filter [66]) the phase update information, i.e. *early* or *late* decisions. This filtering increases the probability of making the right decision, as shown in Figure 2.24, where for binary transmission, *early* and *late* decisions are filtered with the accumulate and reset majority voting filter that votes one *up/dn/hold* on every block of four *early/late* decisions. The probabilities of *early/late* decisions are found from the statistics of the input data and noise.

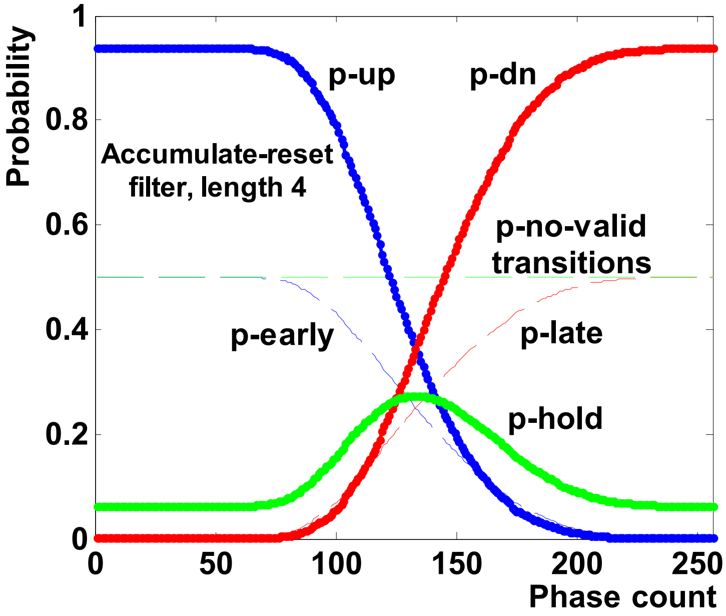


Figure 2.24: The raw input probabilities (p-early, p-late, p-no-valid transition) are converted by a filter to state transition probabilities (p-up, p-dn, p-hold) for each possible phase location. Note that in a random non-return-to-zero binary data stream, 50% of all data does not contain a transition to lock to.

Once we know the transition probabilities for each of the phase states, as in Figure 2.24, we can form the Markov chain transition matrix T and compute the steady-state phase probabilities by looking either at the eigenvalues of the transition matrix or solving for transitions iteratively

$$\underline{p}_{n+1}^{\phi} = T \cdot \underline{p}_{n}^{\phi} \tag{2.11}$$

where $\underline{p}^{\Phi}$ is the phase probability distribution vector.

The resulting steady-state phase probability distribution is shown in Figure 2.25. In this example, the CDR loop can choose between 256 available phases per each received symbol. The maximum of the phase probability distribution indicates the nominal "lock" point of the CDR loop. Due to noise and interference, the CDR loop deviates from the lock position with a probability that is represented as the steady-state phase probability in Figure 2.25. This is really just the histogram of the jitter added by the CDR loop.

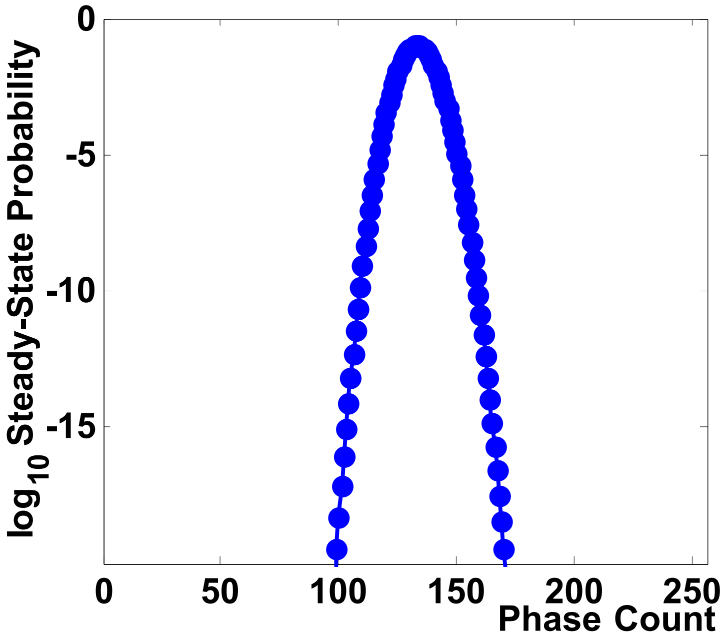


Figure 2.25: A plot of the steady-state phase probabilities for the transition probabilities in Figure 2.24.

Unfortunately, real systems can depend on more than just the previous state of the system, which violates the first-order Markov chain model. In channels heavily affected by ISI and colored noise, the first-order Markov chain can only be applied on an accumulate-reset filter in cases where the length of the strong correlation of the ISI or the noise is similar to the length of the filter. For random-walk filters, or longer correlation lengths, higher-order Markov chains have to be used [67].[12]

To avoid this issue with Markov models, one can linearize the loop, for example using a first-order quantizer approximation [23, 68], in which the phase detector of a

[12] Another issue with Markov model is it ignores the latency of the CDR feedback loop, which leads to dither jitter. For systems where the phase steps are small, the input jitter is larger than the dither jitter and the latter can be ignored.

second-order CDR loop can be replaced by a gain element and an additive white noise source with variance equal to that of the dither jitter. This enables a frequency domain analysis of the loop, and creates the jitter tolerance mask for the CDR loop. One problem with this approach is that it does not preserve the accurate statistics of the input noise (since it approximates the quantization noise as uniform or sometimes Gaussian [23]). These approximate statistics cause inaccuracies in estimation of the steady-state phase distributions. On the other hand, by using loop linearization, we can correctly take into account the loop delay [70], which is essential for low-frequency jitter tracking, because of the peaking in the CDR transfer function that occurs from excessive delay.

As we will show in Chapter 4, much of the jitter is due to factors related to the bit stream. To analyze this jitter, we will use a Markov model and assume that strong correlation of noise and ISI exists only within a window covered by the CDR accumulate-reset loop filter. We also assume that the weak correlation of the residual ISI due to long latency reflections and associated colored noise can be ignored (assumed to be uncorrelated). This approach works well with different edge selection algorithms present in multi-level and other more advanced signaling techniques, since a Markov model is essentially a state-driven model and lends itself nicely to imposing additional conditions or filtering of transitions between states.

It is also worth noting that the Markov chain framework can easily be reduced to the case of a baud-rate CDR [71-73], which is becoming increasingly popular on band-limited channels due to its compatibility with symbol-spaced equalization.

2.3 Summary

Ultimately, the limitation to the number of bits that we can transmit across a channel is determined by the signal to noise and interference ratio at the receiver. The larger this ratio, the more distinguishable are the levels that one can transmit in each symbol, increasing the effective bit rate. Unfortunately at high symbol rates, the interference levels are often quite high. The ISI caused by dispersion (pulse widening due to band-limited channel) and reflections (multi-path effect due to impedance discontinuities in the channel) can be as large as the desired signal. Crosstalk occurs at points with dense wiring, like connectors and packages. While still problematic, it is much smaller than ISI, and can be effectively dealt with by wiring rules that prevent near-end crosstalk, which causes the larger voltage errors.

While interference dominates the voltage errors in a raw system, these voltage errors can, in theory, be completely corrected for. Thus, this chapter also looked at fundamental sources of voltage noise at the receiver. This noise comes from two sources, direct voltage noise, and voltage noise caused by the timing noise or jitter of the system. While the direct thermal noise is becoming a more important voltage source, it still is not the dominant noise term in this system. Instead, dominant noise sources are the residual voltage offsets in the input receivers and voltage noise caused by timing errors.

The magnitude of the voltage noise caused by timing jitter makes it critical that this term is modeled accurately, and this chapter showed a relatively simple approach for accomplishing this. The model showed that high-frequency jitter causes the largest noise, and hence the spectrum of the jitter is critical to understand. This spectrum can be obtained by using a model of the PLL generating the clock, and the measured spectrum of the supply noise, since in these systems the jitter is generally set by supply noise.

Before using the models developed in this chapter to estimate link performance, we first need to remove as much of the interfering signals as we can, given the link power and area constraints. The next chapter looks at this problem in more detail.

Chapter 3

System-Level Design

Many standard communication systems today operate close to the capacity of their channels [3,74-76]. All these systems currently support data rates of up to tens of Mb/s. These amazing advances have been enabled by the exponential growth in IC technology. For these data rates, we can design chips in current IC technology that are sufficiently fast and complex enough to host the sophisticated coding and signal processing algorithms needed to approach the Shannon capacity limit.

On the other hand, high-speed link systems operate at Gb/s data rates. This puts more than two order of magnitude tougher throughput requirements on the chip design. In current technology, we simply cannot port the same sophisticated algorithms to link chips that operate at Gb/s data rates without incurring excessive power and complexity. For example, we can neither afford the high-resolution GS/s ADCs that are needed to apply sophisticated digital communication algorithms [3], nor implement efficiently the recursive soft-decoding and error-correction techniques needed to approach the capacity [77-80]. Instead, we need to resort to simpler digital communication techniques, i.e. modulation and equalization, to avoid or compensate the ISI up to very low BERs. We even need to modify these simple techniques in order to satisfy the link-specific throughput and power constraints. This requires careful system analysis and optimization.

In this chapter, we present the topologies and algorithms which enable efficient ISI compensation in high-speed links, within the link-specific constraints. In order to first orient ourselves, we look at the capacity limits for link channels with link-specific noise sources. We then show some practical equalization and modulation topologies and derive an optimization framework [60] that includes all dominant link-specific noise sources and constraints. We will use this framework in the next chapter to compare the link capacity to the data rates achievable by the optimized practical multi-level signaling and equalization topologies. Finally, we also derive approximate adaptive algorithms [45,81] that work with this architecture.

3.1 Limits

It is desirable to find the Shannon capacity limit of a practical link channel with realistic, link-specific noise sources for two reasons. First, the limit determines the effective usable bandwidth of the channel (hence specifying the circuit speed requirements), and second, it gives the maximum possible data rate. These two parameters are critical in determining the longevity of copper wires as an electrical signaling medium.

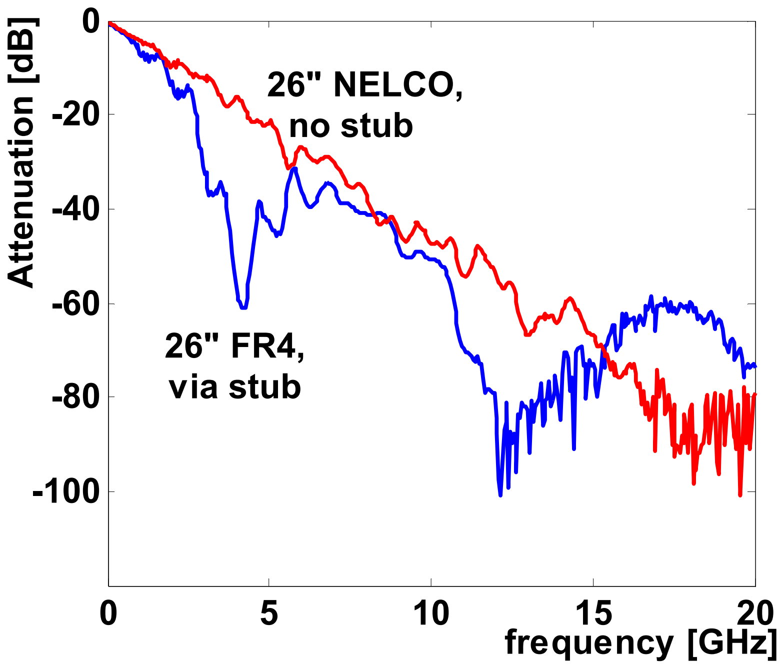


Figure 3.1: Baseline channels; a legacy channel with via stub discontinuities and FR4 dielectric, and a newer microwave-engineered channel with no stubs and NELCO6000 dielectric.

We base our analysis on two channels, shown in Figure 3.1, which represent the two opposite ends of the whole range of channels over which today's links are required to operate.

The channel shown in blue in Figure 3.1 represents a class of older, legacy channels with impedance discontinuities from via stubs and connectors, causing notches in the frequency domain, and an older dielectric, FR4, causing a higher loss slope. The newer channel, shown in red, is microwave-engineered to minimize impedance discontinuities from via stubs and connectors, and thus has a much smoother roll-off. It also has a NELCO6000 dielectric with a smaller loss slope.

3.1.1 Capacity Analysis

In the previous chapter, we described the properties of the link-specific noise sources, and in this section we try to estimate their impact on the capacity of the link. To make the analysis more accurate, we also impose a peak transmit power constraint, present in high-speed links due to limitations in the transmitter driver circuits.

In estimating the capacity of the link, we start from the well-known waterfilling solution defined for thermal Gaussian noise [82]. To make the analysis more applicable to links we add to the thermal noise term the effective voltage noise from phase noise[13]. This phase noise term is proportional to the signal energy, and is derived in detail in Appendix B[14].

Assuming the source signal distribution as Gaussian, for a fixed peak-to-average ratio (PAR), capacity achieved by this modified waterfilling with $\Gamma=1$ (gap[15], defined in [83]) is a concave optimization problem[16]:

[13] While this is exact for thermal noise, which is Gaussian, it is not exact for phase (carrier) noise since the capacity is achieved in that case when the sum of the signal and voltage noise due to phase noise is Gaussian. However, given that phase noise variance is usually much smaller than one, the Gaussian distribution of the signal overwhelms the distribution of voltage noise due to phase noise and the resulting sum is mostly Gaussian.

[14] Although we derived the expressions for sampling jitter in a baseband link in Chapter 2, in capacity analysis we need to use the effective voltage noise from phase noise of the carrier tone, since capacity cannot be achieved with baseband, but rather with coded multi-tone techniques.

[15] This "gap approximation" represents the gap to capacity of a certain coding and modulation scheme with a given probability of error – for example, the gap for uncoded PAM modulation at BER=10^{-7} is 9.5 dB and BER=10^{-15} is around 13.3 dB.

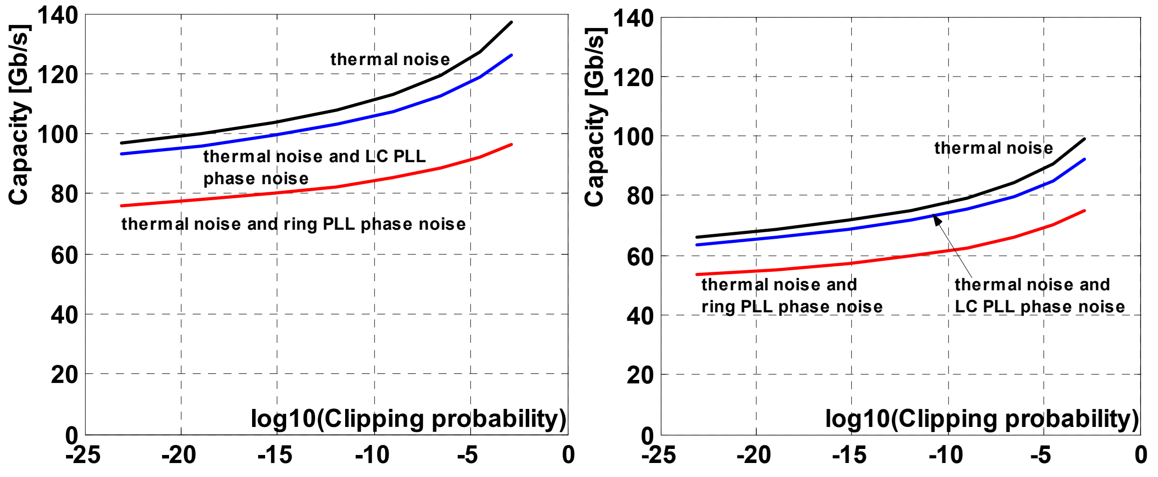
[16] This can be easily shown by examining the convexity in t on any energy line $E_n=E_{on}+tE_{sn}$ [84].

$$\lim_{N\to\infty}\left(\underset{E_n}{\text{maximize}}\;\; b=\frac{1}{2}\sum_{n=1}^{N}\log_2\left(1+\frac{E_n\|H_n\|^2}{\Gamma\left(\sigma_{thermal}^2+E_n\|H_n\|^2\sigma_{\theta n}^2\right)}\right)\right) \tag{3.1}$$

$$\begin{aligned} s.t. \quad & \sum_{n=1}^{N} E_n = NE_{avg} = NE_{peak}PAR^{-1} \\ & E_n \geq 0,\;\; n=1,\dots,N \end{aligned}$$

where $\sigma^2_{thermal}$ is the thermal noise spectral density, $\sigma^2_{\theta n}$ is the sum of transmitter and receiver variances of phase noise of tone *n,* similar to Equation (B.2), and *N* is the total number of tones.

The capacity curves with thermal noise are shown in Figure 3.2 for the two baseline channels. Due to the peak power constraint and very low BER requirements, we are interested in plotting capacity curves vs. clipping probability of the transmitted signal, determined by the PAR. The clipping probability is another representation of average signal energy, as the average signal energy is mapped to clipping probability through the peak-power constraint.



(a) (b)

Figure 3.2: Capacity curves vs. clipping probability, for best (a) and worst channels (b), with thermal noise, phase noise from LC and ring oscillator based PLL.

As we increase the average energy of the signal, the clipping probability increases, assuming that signal has Gaussian distribution and fixed peak power constraint at the transmitter. Given that we can always use the codes to reduce the effect of PAR [85] we plot the capacity for a wide range of average signal energies resulting in different clipping probabilities without explicitly using techniques that reduce the PAR.

The capacity of these channels with thermal noise only is very high, between 70 and 120 Gb/s. The capacity degrades only slightly, by up to 5%, when we add the phase noise of state-of-the-art frequency synthesizers based on the LC oscillators [86] with a standard deviation of phase of around 0.5°. However, if we want a greater frequency tuning range, we need to use frequency synthesizers based on the ring oscillators [39]. The phase noise of such a synthesizer originating from thermal and supply noise in carrier distribution buffers, results in up to 5° of phase noise standard deviation, which degrades the capacity by about 20%. In addition to this, with higher phase noise, the capacity becomes less dependent on the signal energy (and therefore the clipping probability), since phase noise introduces a signal proportional noise source, as shown in Equation (B.2).

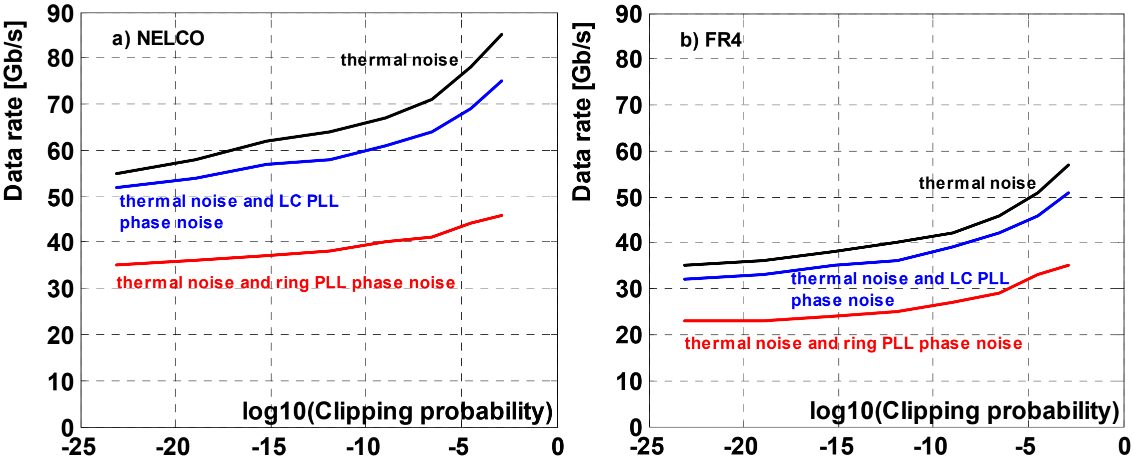
3.1.2 Data Rates of an Uncoded Multi-Tone System with Integer Modulation[17]

Since using soft-decoders is prohibitively expensive in high-speed links, iterative techniques like turbo codes or LDPC codes, which closely approach the capacity [77, 87], cannot be used. It is therefore of practical interest to see what data rates are achievable by just using uncoded integer constellations (like quadrature amplitude modulation - QAM, see 3.2.2) with waterfilling. Figure 3.3 shows integer loading curves with thermal and phase noise for a gap of 13.3 dB, corresponding to a BER of 10^{-15}. By modifying the gap approximation and the Levin-Campello loading algorithm [88] to include the effects of the carrier phase noise, we note that the degradation in the data rate from phase noise is slightly more pronounced than that in the capacity. The reason for this difference can be seen by looking at the signal energy.

With increase in the signal constellation, the required energy per channel increases faster than the minimum distance of the integer constellation points, causing the phase

[17] Using the *Gap* approximation and greedy loading.

noise (which is proportional to energy), to become a more limiting factor than in the case of the capacity, where both the signal and the noise are proportional to the energy and scale evenly.



(a) (b)

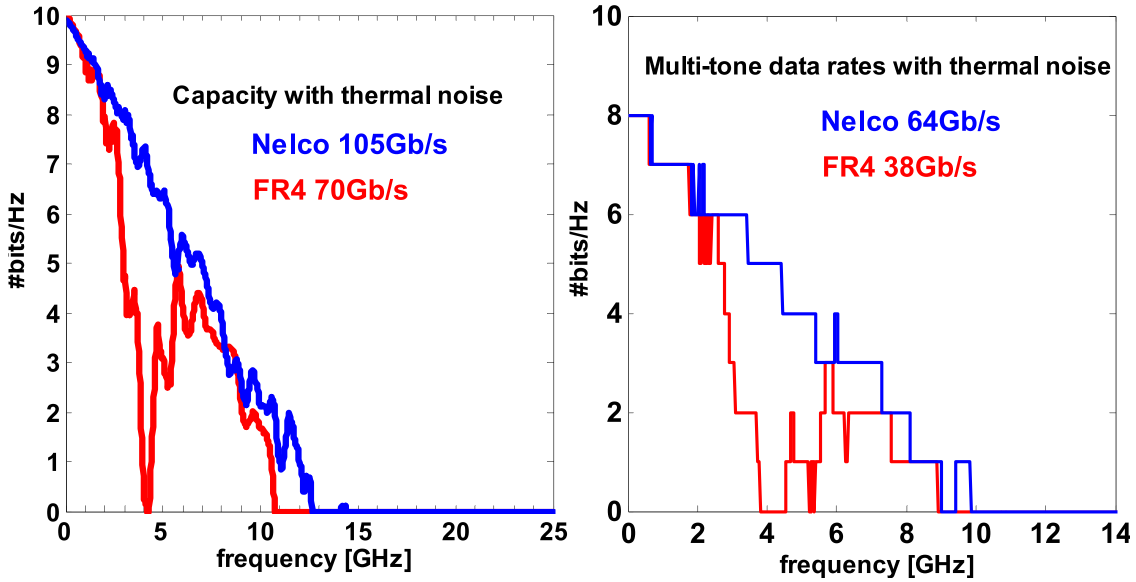
Figure 3.3: Data rate curves for integer loading with gap for BER=10^{-15}, for best (a) and worst channels (b), with thermal noise, phase noise from LC and ring oscillator based PLL, obtained using the modified Levin-Campello loading algorithm, to account for the effects of phase noise.

The capacity estimates and the data rate results in Figure 3.2 and Figure 3.3 show that even for very low BER requirements, and realistic noise sources, the achievable data rates are very high - much higher than the 3.125 Gb/s used in the backplane links that are currently deployed in routers.

Besides the maximum achievable data rates, this analysis also reveals the usable bandwidth of the channel. As we said earlier, this is very important for practical implementation purposes since the maximum bandwidth sets the throughput requirements for the circuits and is very important in these wideband applications. We can find the usable channel bandwidth by plotting the bit loading vs. frequency, for both the capacity achieving case and uncoded discrete multi-tone.

Example bit loading vs. frequency for thermal noise only in both channels is shown in Figure 3.4a for the capacity, and Figure 3.4b for the uncoded multi-tone with PAM/QAM constellations. The loading algorithms effectively use the channel up to the

Nyquist frequencies of 10-12 GHz for both baseline channels in Figure 3.4, indicating that the fundamental noise sources are too small to limit the data rates significantly. This also represents the upper bound on the usable channel bandwidth in any type of link, as in a real system we can only get more noise than just thermal noise from a 50 Ω termination.



(a) (b)

Figure 3.4: Example bit loading: a) Capacity achieving bit loading, b) Integer bit loading assuming uncoded PAM/QAM constellations (for average energy corresponding to clipping probability of 10^{-15} in Figure 3.2 and also with BER of 10^{-15} in Figure 3.3).

For practical baseband systems, it turns out that the main obstacle to achieving the data rates projected in Figure 3.3 is the efficient elimination of ISI. Currently, state-of-the-art next-generation links are still baseband and being designed to achieve up to 5-10 Gb/s rates over similar channels [39,89]. To overcome the effects of the ISI, these links are starting to use some of the techniques that are well-known in standard digital communications, such as modulation and equalization [12]. However, due to the tight power constraints and high-throughput, the resources in a link have to be allocated to compensate for the ISI in the most energy-efficient manner. In addition to these constraints, the IC circuit constraints like peak output power (driver headroom) also propagate to the algorithm level in practical links, creating the necessity to change the well-known algorithms in order to build practical links.

3.2 Baseband Communication Techniques for High-Speed Links

In this section we first give an overview of the existing equalization and modulation techniques in high-speed links and then extend these techniques to better handle bandlimited channels. With these new topologies, we then formulate the system optimization and analysis framework. This enables us to explore the design space by estimating the performance of different topologies in Chapter 4 with link-specific noise sources and hardware constraints.

3.2.1 Equalization

One of the simplest ways to compensate for ISI is to use a linear finite impulse response (FIR) filter as the equalizing filter at the receiver to compensate for the filtering effects of the channel and flatten the resulting response [12]. More advanced techniques involve a combination of linear feed-forward and feedback filters, called decision-feedback equalization (DFE) [90,91]. In standard communications, it is generally preferred to do equalization at the receiver since in that case, the receive equalizer can adapt to the channel and we do not need to send any information back to the transmitter. Recent dial-up modems [1,2] are exceptions since they use non-linear transmit precoding techniques like those developed by Tomlinson [92] and Harashima [93], or Laroia [94] to avoid the error accumulation in the feedback path of the DFE[18].

Power and speed constraints in high-speed links require different choices for implementation of these standard communication algorithms. We first discuss the specific topologies that are used and propose solutions to alleviate some of their problems. Then, we derive the algorithms to optimize these structures.

3.2.1.1 Equalization Topologies

In high-speed links it is often easier to implement filters at the transmitter, since a fast DAC is easier and cheaper to build than a fast ADC. The first attempt to perform

[18] Precoders are also used to enable combination with trellis coding and soft Viterbi decoders [1,2].

equalization in high-speed links was made by Dally and Poulton [24] by implementing the transmit pre-emphasis with a two-tap analog FIR filter. The filter was made by connecting drivers of different sizes driven by the delayed versions of data and with inverted sign to equalize the dispersion component of the ISI, as shown in Figure 3.5.

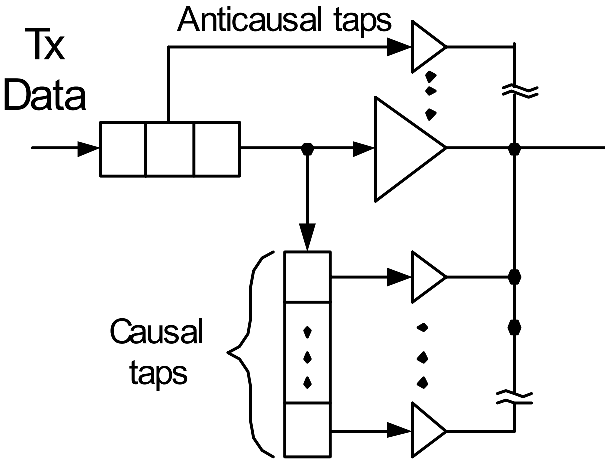


Figure 3.5: Analog FIR implementation of transmit pre-emphasis

This equalizer compensates for the effect of the channel by attenuating the low frequencies of the signal to match the attenuation at higher frequencies and flatten the channel up to the Nyquist frequency. This attenuation is a result of the peak-power constraint of the transmitter due to the limited headroom of the circuits used in the output drivers.

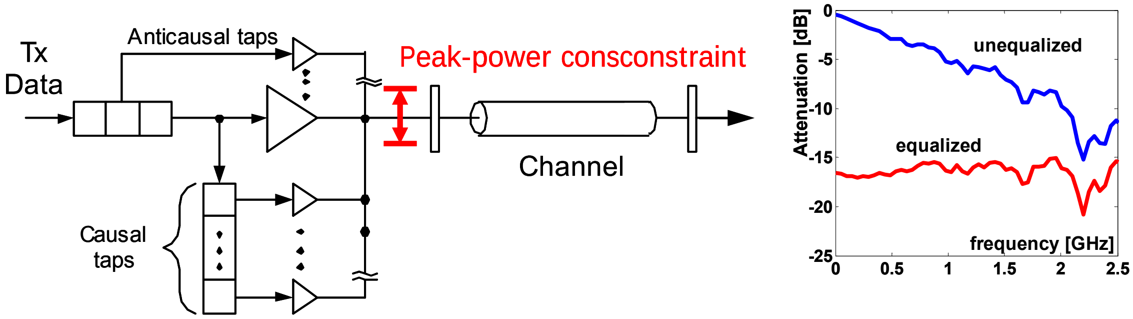


Figure 3.6: Transmit pre-emphasis with peak-power constraint

Rather than building a linear filter to equalize the pulse response, DFE uses the history of the received symbols to cancel the trailing ISI that is present in the channel. If the channel pulse response is known, we can subtract the residual signals from all the symbols we have already seen, as shown in Figure 3.7, leaving only the signal from the symbol of interest.

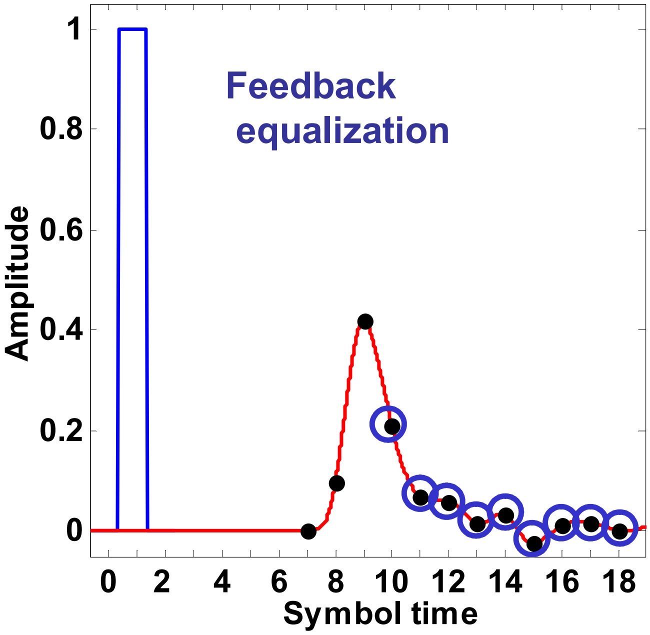


Figure 3.7: Effect of feedback equalization

In communications, DFE has been used heavily instead of linear filtering, to circumvent the problem of noise amplification, [12]. These systems implement DFE in two basic ways: either in the digital domain, or as a mixed-signal circuit. For the digital domain the input is initially digitized at a high enough resolution and rate, and the DFE is implemented in a digital signal processor. High-bit-rate Digital Subscriber Lines (HDSL) [3] and Gigabit Ethernet [95] are implemented this way. For mixed-signal implementation, the input quantizer only needs to resolve the input symbol. The output of this quantizer feeds a finite impulse response (FIR) filter that then drives a DAC whose output is subtracted from the input signal. This approach has been proposed for use in analog disk-read channels [96].

Both approaches however, are impractical in multi-Gb/s high-speed links. The digital approach requires a very fast and accurate ADC. While high-speed ADCs have been created [97], they would be too expensive (area/power) to use in a practical link. The second approach suffers from latency problems. For a 6.25 Gb/s binary link, we have 160 ps to resolve the input, drive the DAC, and have the DAC outputs settle to the required precision[19].

As shown by Kasturia [98] and Parhi [99], and more recently by Sohn [100], one tap of feedback equalization can be achieved by using loop unrolling to avoid the bottleneck in the latency of the feedback loop. Since we cannot run the feedback loop fast enough, we unroll it once and make two decisions each cycle. One comparator decides the input as if the previous output was a 1, and the other comparator decides the input as if the previous bit was a 0. Once we know the previous bit, we select the correct comparator output, as shown in Figure 3.8.

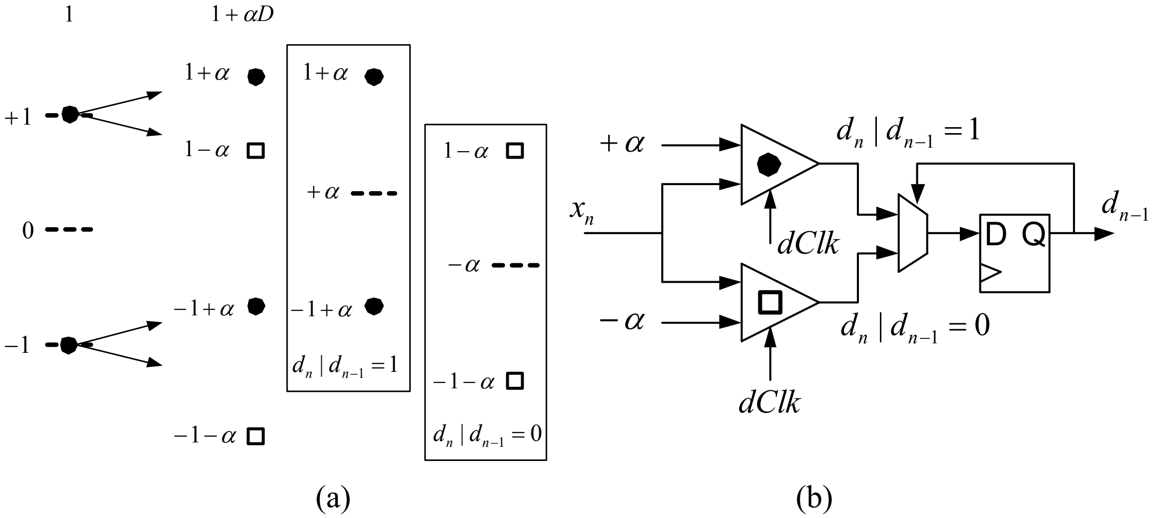


Figure 3.8: One tap DFE using loop-unrolling: a) Transmitted binary signal levels corrupted by ISI split to ±1±α levels at the receiver and can be recovered with two slicers offset by the amount of ISI ±α b) Practical implementation of the one tap DFE using loop-unrolling.

Instead of just one data sampler, for binary signaling the receiver has two samplers that are offset by ±α, anticipating the impact of trailing ISI α from a previously sent symbol of value of ±1. This method can be applied to two or more taps of feedback; however, the number of required receivers is M^L, where M is the number of signal levels and L is the number of feedback taps, and each receiver nominally has M-1 comparators. Usually, only a small amount of unrolling is needed to bridge the latency gap.

[19] In 0.13 µm technology, so far only one design has shown promise to close the loop on the first DFE tap [121], however authors did not disclose the power consumption which is critical in highly integrated, power-limited link applications such as crossbar ASICs in routers.

Corrections for symbols that are far enough away from the current sample do not have a latency problem. This makes it easier for the DFE to correct for the long latency interference caused by the reflections from the connectors, vias, transmit/receive parasitic capacitances, and other termination mismatches. To prevent the complexity of the resulting DFE and parasitic output capacitance from becoming a problem, we proposed the use of a tap-selective DFE, where only the most significant taps are used [39]. This next generation link architecture is shown in Figure 3.9. The feedback equalizer can be implemented either as a current-mode driver feeding back into the channel or by modulating the threshold voltage of the receiver comparator. The current-mode driver implementation has the additional advantage of canceling all the secondary reflections, as illustrated previously in Figure 2.9, by sending the cancellation waveform along with the primary reflection off the mismatched receiver termination, but generally requires longer latency.

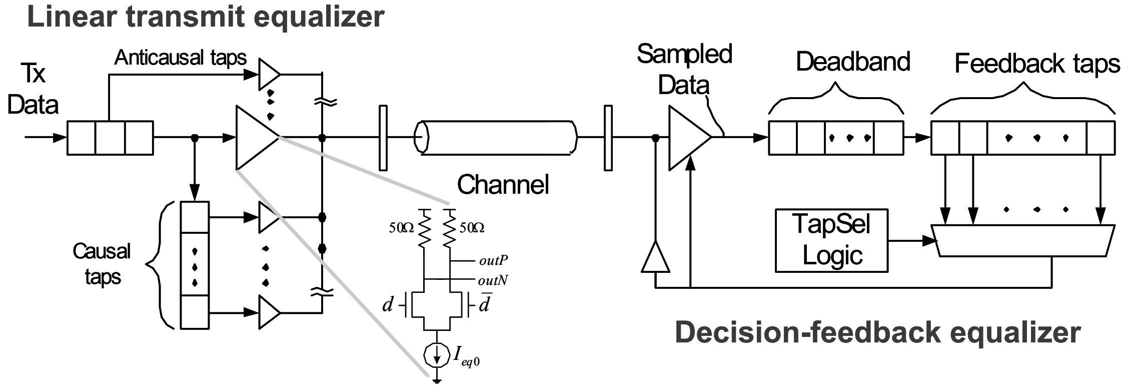


Figure 3.9: Baseband high-speed link architecture with transmit pre-emphasis filter and tap-selective feedback filter [39].

The tap selective DFE addresses the most important tradeoff involved in building a current-mode feedback filter[20]: adding more taps cancels more reflections but also increases the parasitic device capacitance at the 50 Ω terminated summation node, i.e. the input to the receiver comparator. As noted earlier, this parasitic capacitance forms the low-pass filter on the signal path, and it also creates frequency dependent impedance mismatch in the 50 Ω termination impedance, causing more reflections in the channel.

[20] The current-mode feedback filter is very similar to the analog transmit pre-emphasis FIR filter. It is just fed by the data that has already been received.

Decision-feedback equalization is not entirely free of noise amplification or equivalently, signal attenuation, since leading ISI (see Figure 3.7), can only be eliminated using a linear filter preceding the feedback stage. In the link architecture in Figure 3.9, the linear transmit pre-emphasis filter can be used for that purpose. Such a linear transmit precoder replaces the standard receiver based feed-forward filter.

3.2.1.2 Equalization Algorithms

With an increase in the desired data rates, the precoder and the feedback equalizer lengths increase significantly, decreasing the power efficiency of the link. Additionally, the precoding loss[21] increases, limiting the achievable data rates in the presence of noise. In order to estimate and optimize the performance of such architectures, we derive a convex optimization framework that incorporates the link-specific noise sources in convex form to obtain globally optimal precoder and feedback filters. Following the optimal system formulation, we also derive a practical adaptive algorithm that addresses the system topology and circuit constraints.

Optimal Transmit Precoding Combined with Feedback Equalization - (Quasi-Concave Formulation)

It is well known that in the linear receiver equalizer problem, a minimization of the mean square error (MSE)[22] after unbiasing results in maximum signal-to-interference-and-noise ratio (SINR) and minimum BER[23], [91]. In a system with a linear precoder, we saw that the peak-power constraint at the transmitter causes the precoder to attenuate the low frequencies of the signal to match the biggest attenuation of the signal within the Nyquist band, as shown in Figure 3.6. While we do get the flat frequency response of the whole system at the receiver, the attenuation of the equalized received signal depends strongly on the loss in the channel, and it varies from channel to channel. In this situation, it is difficult to formulate a unified error function at the receiver that we could use to tune both the transmit precoder coefficients and the receiver decision levels.

[21] This is similar to the linear receive equalizer noise amplification problem. While the *l2* norm of the linear equalizer amplifies noise in the receiver, the *l1* norm of the transmit precoder attenuates the transmitted signal in the case of the peak transmit power constraint.

[22] MSE is a quadratic norm in equalizer taps, and therefore is convex in equalizer taps.

[23] BER here is defined assuming a mean distortion approximation, i.e. approximating the residual ISI as Gaussian noise.

Interestingly, we can easily formulate the correct error function if we add a variable gain element to the receiver [45], as shown in Figure 3.10. In this way the variable gain element amplifies the unknown received signal and compares it to a known target value. Considering now the whole system with a linear precoder, in Figure 3.10, we can formulate the MSE criterion

$$MSE(\underline{w}, g) = E_a\left(1 - 2g\underline{w}^T\mathbf{P}\underline{1}_\Delta + g^2\underline{w}^T\mathbf{P}\mathbf{P}^T\underline{w}\right) + g^2\sigma^2 \tag{3.2}$$

where $\underline{w}$ is the precoding vector, **P** is the Toeplitz matrix of the channel pulse response, g is the scalar receiver gain, $\underline{1}_\Delta$ is the system delay vector defined as $[0\ 0\ \ldots\ 0\ 1\ 0\ \ldots\ 0]^T$ where the one is in position Δ+1 and represents system delay. The average energy of the transmitted alphabet a is E_a, and σ is the standard deviation of the AWGN source at the receiver. This form is similar to that of the finite-length linear receiver equalizer problem [101]. For detailed description on the structure and notation of this matrix model, see the first part of Appendix C.

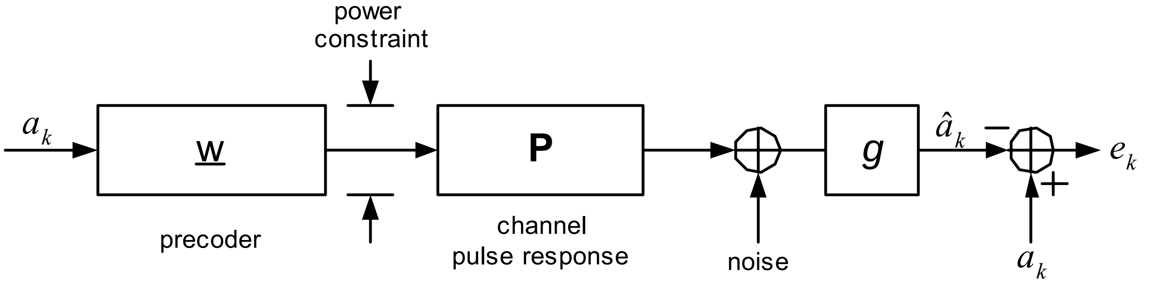


Figure 3.10: Precoding system with transmit power constraint and scalar gain in the receiver.

Due to the power constraint, the precoder is not able to compensate the loss of the signal in the channel, but rather just compensate the ISI, while the gain element in the receiver compensates for the amplitude loss of the received signal. Thus the gain element effectively causes noise amplification.

In previous work on the optimization of a linear precoder, approximately optimal methods are derived, without using the gain element in the MSE criterion [102]. It is also shown that the MSE criterion is sometimes inferior to the zero-forcing solution (ZFE), scaled to satisfy the power constraint. We extend that work by showing that the minimization of the MSE formulated using the receiver gain element, Equation (3.2), is equivalent to SINR maximization, and therefore minimizes the BER.

From Equation (3.2) we can derive the optimal gain g^* as a function of the precoder $\underline{w}$:

$$g^*(\underline{w}) = \frac{\underline{w}^T \mathbf{P} \underline{1}_\Delta}{\underline{w}^T \mathbf{P}\mathbf{P}^T \underline{w} + \sigma^2 / E_a} \tag{3.3}$$

which, when substituted back in Equation (3.2) yields

$$\frac{1}{SINR_{biased}} \overset{\Delta}{=} \frac{MSE(\underline{w}, g^*)}{E_a} = 1 - \frac{E_a(\underline{w}^T \mathbf{P} \underline{1}_\Delta)^2}{E_a \underline{w}^T \mathbf{P}\mathbf{P}^T \underline{w} + \sigma^2} = \frac{1}{SINR_{unbiased} + 1} \tag{3.4}$$

where $SINR_{unbiased}$ represents the "true" (unbiased) signal-to-interference-and-noise ratio, and is defined as

$$SINR_{unbiased} \overset{\Delta}{=} \frac{E_a(\underline{w}^T \mathbf{P} \underline{1}_\Delta)^2}{E_a \underline{w}^T \mathbf{P}(\mathbf{I} - \underline{1}_\Delta \underline{1}_\Delta^T)(\mathbf{I} - \underline{1}_\Delta \underline{1}_\Delta^T)^T \mathbf{P}^T \underline{w} + \sigma^2} \tag{3.5}$$

where $\underline{w}^T\mathbf{P}\underline{1}_\Delta$ represents the main tap of the received pulse response, and $\underline{w}^T\mathbf{P}(\mathbf{I}\text{-}\underline{1}_\Delta\underline{1}_\Delta^T)(\mathbf{I}\text{-}\underline{1}_\Delta\underline{1}_\Delta^T)^T\mathbf{P}^T\underline{w}$ the square of the *l2* norm of the residual ISI in the precoded pulse response.

The identity in Equation (3.4) shows that minimization of the MSE defined as in Equation (3.2) indeed results in maximization of the unbiased SINR. However, the convenient quadratic cost function is lost, and the resulting problem is to maximize the SINR, which is a fractional quadratic programming problem known to be non-convex [45, 103].

Since our final goal is to minimize BER, starting from Equation (3.5) directly, we note that the argument of the BER function is the square root of Equation (3.5), resulting in the ratio of $\underline{w}^T\mathbf{P}\underline{1}_\Delta$, an affine function in $\underline{w}$, and the *l2* norm of $\underline{w}^T\mathbf{P}(\mathbf{I}\text{-}\underline{1}_\Delta\underline{1}_\Delta^T)$ and σ, which is convex in $\underline{w}$. It can be shown that maximization of this ratio is a quasiconcave programming problem with a global optimum [103], and can be efficiently solved by, for example, bisection [84].

Given that our final target is to minimize the actual BER, the BER function used in the optimization must be a very close approximation of the actual BER. Due to the very low BER requirements in high-speed links, it has been shown that the Gaussian approximation of ISI, which leads to a BER function defined as $Q\left(\sqrt{SINR_{unbiased}}\right)$ is usually not very accurate for BERs <10^{-5} [104]. While we explore this discrepancy in more detail in Chapter 4, here we want to look at the physical causes in order to derive a better approximation.

The main reason for this discrepancy is that in fixed length precoders/equalizers ISI energy is dominated by a few very big residual components of dispersion ISI, but the total number of ISI taps is large due to reflections (which are much smaller than residual dispersion components). Since such ISI is not identically distributed, it cannot be well approximated with a Gaussian distribution. To avoid this effect, we propose a mix of peak distortion and mean distortion criteria to achieve higher accuracy in BER approximation. It is only necessary to assume that a few big residual ISI taps are frequent enough to be considered as a constant shift from the mean value of the received signal, and the rest of the taps can be then well approximated with a Gaussian distribution. The resulting optimization is still quasiconcave:

$$\begin{aligned} \underset{\underline{w}}{\text{maximize}}\ \gamma &= \frac{0.5 d_{\min} \underline{w}^T \mathbf{P} \underline{1}_\Delta - V_{peak} \left\| \underline{w} \mathbf{P} \mathbf{I}_{PD} \right\|_1 - offset}{\left(E_a \underline{w}^T \mathbf{P} (\mathbf{I} - \underline{1}_\Delta \underline{1}_\Delta^T - \mathbf{I}_{PD})(\mathbf{I} - \underline{1}_\Delta \underline{1}_\Delta^T - \mathbf{I}_{PD})^T \mathbf{P}^T \underline{w} + \sigma^2 \right)^{1/2}} \\ s.t. \quad & \left\| \underline{w} \right\|_1 \le 1 \end{aligned} \tag{3.6}$$

where the *l1* norm of $\underline{w}$ is limited to 1 to satisfy the peak output power constraint, and $\mathbf{I}_{PD}$ is a diagonal matrix that selects the residual ISI components to be considered for peak distortion. The average energy of the transmit alphabet, E_a, and minimum distance in transmit alphabet constellation, $d_{min,}$ assuming multi-level pulse amplitude modulation (PAM), are related to peak transmitter voltage V_{peak} by

$$d_{\min} = \frac{2V_{peak}}{M-1},\ E_a = T_{symbol}\frac{V_{peak}^2(M+1)}{3(M-1)},\ M = 2^b. \tag{3.7}$$

Since the variances of the voltage noise due to transmitter and receiver sampling jitter are convex (quadratic) functions of the precoder taps, we can also add the impact of the sampling jitter, from Equations (2.9) and (2.10), to the noise term σ^2 in (3.6), so the resulting noise variance is $\sigma^2 = \underline{w}^T\boldsymbol{S_0}^{\boldsymbol{TX}}\underline{w} + \underline{w}^T\boldsymbol{S_0}^{\boldsymbol{RX}}\underline{w} + \sigma^2_{thermal}$. The effect of the limited slicer resolution is added to (3.6) as the term *offset*. In this way, we managed to include all of the described link-specific noise sources into the optimization framework in (3.6). The quasiconcave formulation of the optimization problem in (3.6) guarantees a globally optimal solution for the linear precoder, i.e. the one that achieves the minimum BER.

This framework can easily incorporate DFE in addition to transmit precoding. The optimal setting for the feedback taps nullifies (zero-forces) the corresponding causal ISI in the precoded received signal. Thus, prior to determining the precoder coefficients, we only need to pre-process the channel Toeplitz matrix in such a way as to put "don't care" values on those residual ISI samples of the signal with precoding, whose indices correspond to the feedback taps that are to be used. This can be achieved by simply eliminating the columns of the channel Toeplitz matrix whose indices correspond to the time index of the feedback taps. Such a "punctured" Toeplitz matrix is then used in (3.6) to obtain the optimal transmit precoder coefficients. The feedback taps then just zero-force the remaining response at particular tap indices.

We use this optimization framework to evaluate the performance limits of the practical implementations in Chapter 4. However, implementing this optimization framework in hardware is expensive due to the complexity of the computing engine required for the optimization algorithm. This section describes an adaptive solution, which obeys the circuit and system constraints, but requires minimal hardware support.

Adaptive Formulation

We use the sign-sign LMS algorithm (a derivative of the well-known least-mean square (LMS) algorithm [105]) to adapt the equalizer taps since it is one of the simplest adaptive algorithms. It creates updates for the tap coefficients ($\underline{w}$) based only on the sign of the data and the measured error

$$w_{n+1}^{k} = w_{n}^{k} + \Delta_{w} sign(d_{n-k}) sign(e_{n}) \tag{3.8}$$

where *n* is the time instant, *k* is the tap index, d_n is the received data and e_n is the error of the received signal with respect to the desired data level, *dLev*.

Although variants of the LMS algorithm are used for adaptation of the equalization filters at the receiver, it is possible to reformulate the algorithm using filtered-X LMS [106] and arrive to adaptive loop formulations for the transmit pre-emphasis filter, which when simplified result in Equation (3.8) for the sign-sign variant of the algorithm. For more details about the loop reformulation see Appendix C.

As mentioned earlier in the optimization framework, one issue in using transmit pre-emphasis based equalization is that the ideal reference level *dLev* from which the error signal is created is unknown *a priori*. This problem arises because the peak output swing constraint in the transmitter forces the equalizer to attenuate the low-frequency components of the signal to match the loss of the signal at high frequencies, as in Figure 3.6. Thus, the amount of voltage swing available at the receiver depends on the frequency characteristics of the channel.

In our earlier work on adaptive algorithms for time-domain multiplexed high-speed link [45], described in detail in Appendix C, we used the variable gain element in the receiver to amplify the unknown received signal to the known target value. We use this same concept to derive the error functions for the system optimization framework in the previous section. This approach would result in the variable gain loop with updates

$$g_{n+1} = g_{n} + \Delta_{g} sign(d_{n}) sign(e_{n}) \tag{3.9}$$

However, a more practical and power efficient approach for high-speed links is to adaptively adjust the reference level of the data slicer, rather than amplifying the signal. Thus we create a second loop which adjusts *dLev* to track the signal level using the following updates

$$dLev_{n+1} = dLev_{n} - \Delta_{dLev} sign(e_{n}) \tag{3.10}$$

At each iteration, the reference level loop in Equation (3.10) adjusts the reference level and provides the error signal e_n for both the reference level loop and equalizer tap loops Equation (3.8). The peak-to-peak error and *dLev* setting are shown in Figure 3.11, for the initial through the final iteration of the algorithm [81].

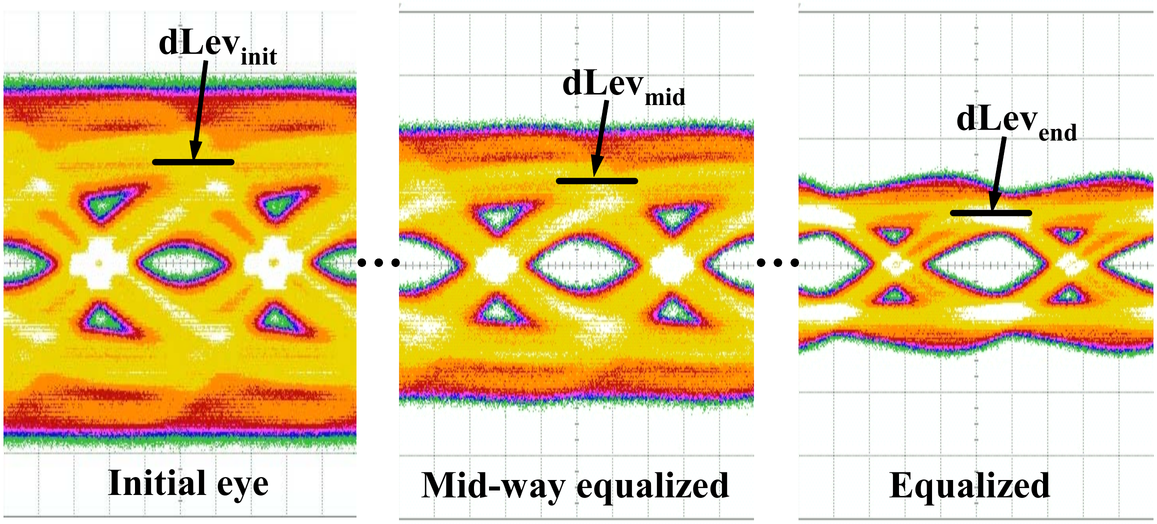


Figure 3.11: Scaling of the *dLev* reference loop Equation ((3.10)) in a dual-loop interaction with the equalizer loop, Equation (3.8). As the signal gets more equalized, scaling in the transmitter decreases the value of the received signal, and the reference loop adjusts *dLev* accordingly.

In order to obtain the highest signal levels at the receiver, maintain transmit output peak constraint, and avoid the trivial stability point of both loops (at zero tap magnitudes and signal level), the proposed values of the equalization taps after every iteration in Equation (3.8) need to be rescaled such that the sum of their magnitudes always equals the maximum allowed by the peak swing constraint[24].

One simple, implementation-driven approximation of this rescaling modifies the update algorithm such that the update on the main tap is computed from the updates of the other taps and the peak constraint requirements, rather than using its own update information. In Appendix D we derive these normalization algorithms in detail and discuss the implementation cost. Recently, similar adaptive algorithms have been proposed by Stonick *et al* for transmit pre-emphasis [107] and Jaussi *et al* [108] for

[24] Note that this rescaling does not alter the optimality of the final solution in terms of the zero-forcing, since due to Karuch-Kuhn-Tucker optimality conditions [84] the optimal equalizer solution in this case has to satisfy the peak-power constraint with equality.

receive equalization. We will discuss the implementation differences between these approaches and our approach in more detail in Chapter 5.

3.2.2 Modulation (Multi-Level Signaling)

Before we start evaluating the performance of different link architectures we need to introduce the concept of modulation techniques that are used in high-speed links.

Multi-level modulations, pulse-amplitude modulation (PAM) or quadrature-amplitude modulation (QAM), have been used very effectively for communication over band-limited channels. For example, dial-up modems [1,2] and HDSL [3] use QAM for communication over band-limited telephone channels, improving the bit rate by transmitting more bits/s/Hz rather than increasing the signaling rate and hence the required bandwidth. Similarly, Gigabit Ethernet [95] uses PAM with five signal levels (PAM5) for communication over copper twisted pairs in a local area network (LAN). Example signal constellations for PAM and QAM are illustrated in Figure 3.12.

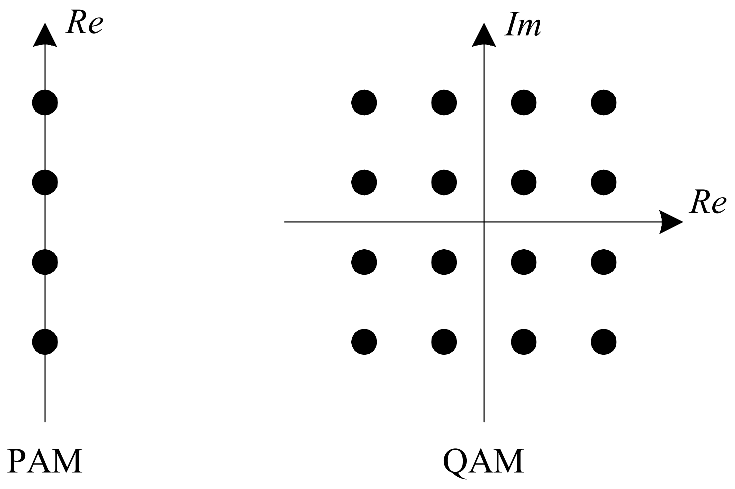


Figure 3.12: Example constellations for PAM4 and QAM16

Recently, high-speed link designs [27,28,39,107,109] have started using PAM4 instead of binary (PAM2) signaling in an attempt to make better use of the high signal-to-noise ratio in the available bandwidth of the dominantly low-pass high-speed link channel. An example comparing the two modulations is shown in Figure 3.13. If we use only PAM2, the only way to increase the data rate is to increase the signaling rate. On a predominantly low-pass channel, such as the one in high-speed links, this means that we are forced to signal beyond the bandwidth of the channel, and as we saw in Chapter 2, this increases the ISI. If, on the other hand, we halve the signaling rate and use four levels instead of two, to carry two bits per symbol, we end up with less ISI because the signal fits more nicely into the available bandwidth of the channel. For example, rather than signaling at 5 GHz to achieve 10 Gb/s with PAM2 signaling, we can use PAM4 and decrease the Nyquist frequency to 2.5 GHz while achieving the same bit rate.

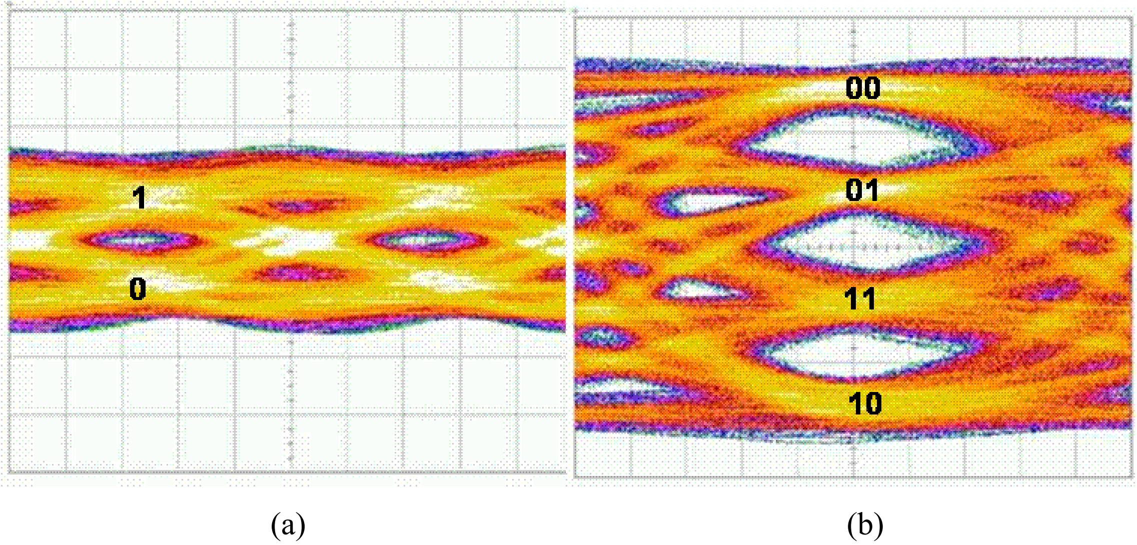


Figure 3.13: Received signal eye diagrams for 6.25 Gb/s data rate with (a) PAM2 and (b) PAM4 modulations (both diagrams have identical time and voltage axis grids).

When the modulation is used jointly with equalization, the total receiver signal magnitude depends on the highest channel attenuation in the Nyquist band. Since the links channels are predominantly low-pass, this means that the modulation with higher Nyquist frequency (for example PAM2 over PAM4, for same bit rate) will incur more loss. On the other hand, due to the peak-power constraint, the multi-level PAM has to fit all the signal levels within the same headroom thereby decreasing the distance between the signal levels and therefore the received eye opening.

This tradeoff is clearly illustrated in Figure 3.14. The channel with smaller loss slope, shown in blue, has around 15 dB attenuation at 3.125 GHz Nyquist frequency for PAM2, and around 5 dB of attenuation at 1.5625 GHz Nyquist frequency for PAM4. Due to the headroom limitation in the transmitter, PAM4 has inherent loss of 3x or 10 dB with respect to PAM2, since we need to fit three eyes into the headroom constraint in PAM4 case compared to only one eye in case of PAM2. Given that the channel loss between the two Nyquist frequencies is around 10 dB, this cancels out the initial disadvantage of PAM4 and we see on the upper right of Figure 3.14, that the equalized eye diagrams for PAM2 and PAM4 have roughly the same eye openings. The true benefit of PAM4 is seen on the other channel with significantly larger rolloff around the PAM2 Nyquist frequency resulting in attenuation of around 35 dB, compared to unchanged 5 dB at PAM4 Nyquist frequency. With the 10 dB penalty of PAM4, the equalized PAM4 is still 25 dB better than equalized PAM2 for this channel, as can be seen in the comparison of the PAM2 and PAM4 eye diagrams on the lower right of Figure 3.14.

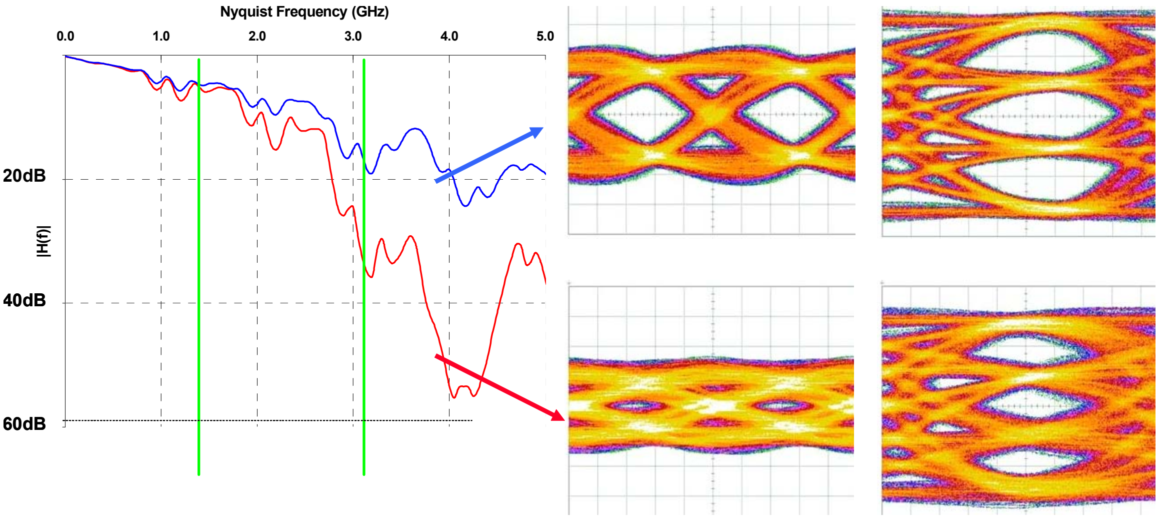


Figure 3.14: Comparison of PAM2 and PAM4 on different channels [39]

The 10 dB penalty is actually just a rule of thumb, and the trade off is a bit more complicated because of different distribution of ISI and jitter in PAM2 and PAM4 cases, as well as the impact of crosstalk. Given the decrease in the received signal relative to the maximum power transmitted into the channel, multi-level modulation schemes are generally more sensitive to residual errors from ISI and crosstalk than normal binary signaling. We will revisit these issues in detail in the next chapter where we estimate the performance of the links, based on the extensions of the noise and interference models from Chapter 2 and system optimization framework developed in this chapter.

3.3 Summary

Link capacity estimates for the two baseline channels with thermal termination noise, phase noise and peak-power constraint, indicate that capacity of these links is between 60 and 120 Gb/s depending on the channel. By looking at the bit loading vs. frequency, even with just thermal noise, we see that the useful bandwidth does not exceed 12 GHz.

This bandwidth limit is very important. Many people today are trying to build 40 Gb/s PAM2 transceivers, pushing the link circuits to operate at very high-speeds, and hoping that some day, they will be able to also integrate signal processing algorithms that would enable them to use these fast signaling circuits on real backplane channels. Our results, however, show that this is not the way we should be thinking about link design. We should design signaling circuits with sample rates not higher than 2-3 times the usable bandwidth of the channel, and instead focus on building more precise link circuits and decreasing the amount of system noise. As we will see later in the next chapter, this is also the case even with relatively simple baseband links since, when we exclude the residual ISI, the link-specific noise sources limit the spectral efficiency that we can get.

With uncoded multi-tone modulation using integer PAM and QAM constellations we can achieve data rates between 35 and 60 Gb/s depending on the channel, which is still much higher than the 3.125 Gb/s data rates of the currently deployed baseband links. The reason for this is that baseband links are limited mostly by ISI.

Most energy-efficient link architectures for ISI compensation use analog FIR filtering in the transmitter combined with tap-selective analog feedback filtering in the receiver. Digitally controlled analog FIRs are used since they provide addition and multiplication operations with no extra energy than required by normal link drivers. In current technology, closing the feedback loop with latency smaller than link symbol time either requires a lot of energy, hence techniques such as loop-unrolling can enable efficient implementations if used in binary signaling and applied to a small number of taps.

To enable performance evaluation of these different topologies, we developed an optimization framework that incorporates all link impairments and the peak-power constraint, resulting in globally optimal equalization setting that minimizes the BER.

While useful for exploration of the design space and performance comparison, this optimization framework is too complex to implement in hardware. Therefore we also developed sub-optimal zero-forcing adaptive algorithms based on the sign-sign LMS algorithm.

The main modifications of the standard sign-sign LMS algorithm include adaptation of transmit pre-emphasis filter with peak-power constraint by using a dual-loop algorithm and iterative tap rescaling to satisfy the peak-power constraint. The additional loop tracks the level of the received signal, and extracts the error information for both itself and the equalizer loop. At each iteration, the updated equalizer taps need to be rescaled to obey the peak-power constraint and avoid non-linear clipping noise as well as the trivial solution of the dual-loop algorithm when all the taps are equal to zero. This algorithm enables an efficient implementation as discussed in detail in Chapter 5.

In addition to the equalization techniques, we also need to use modulation in order to increase the spectral efficiency of the link. As we will see in the next chapter, PAM2 and PAM4 are the two modulation types that are robust enough to be used in current high-speed links. A practical rule of thumb is that whenever a channel has a roll-off of more than 10 dB per octave, there is a potential for PAM4 modulation to be better than PAM2, since this is the loss that PAM4 incurs at the start by fitting three eyes in the transmitter headroom constraint. The accurate comparison of different equalization and modulation techniques with link-specific noise sources and hardware constraints is the topic of our next chapter.

Chapter 4

Performance Analysis

Our goal at this point is to combine the topology of the link system, the link-specific noise sources and different equalization algorithms to evaluate the performance of the system using a BER criterion. This criterion is important since high-speed links are required to operate at BERs that are typically lower than 10^{-12}. While this is not an unusual requirement in many digital communication systems, what is unusual is that this is the uncoded BER specification. High-speed links have very tight power constraints and as we noted earlier it would be very hard to incorporate any coding scheme[25] that could reduce the BER requirement in links by several orders of magnitude. For these power reasons, the equalizers and all other system components like receiver slicers, PLL and CDR have to work especially hard to bring the BER down to below 10^{-12}.

At the point where links started being limited by the channel, somewhere between 1 and 2 Gb/s, it was hard to understand what really limits the link performance and thus how to improve it. Unfortunately, there were no good system models that could enable power/performance tradeoffs. When attempted, this analysis borrowed models from either computer systems or from digital communications, and as we will see, both had issues when applied to high-speed links.

[25] Codes that are used in links are simple transition density/elimination codes, like IBM's well-known 8b10b code [127] or newer multi-level transition elimination and density codes like 4s5s [107,128] that target improved CDR operation and reduction in reflections and crosstalk.

The worst case analysis used in computer systems [110,111] assumes the worst case correlation of all noise and interference sources and effectively targets zero BER operation. As such, it can be very pessimistic for high-speed links - especially if the noise sources are uncorrelated and the probability of the worst case interference is very low.

On the other hand, people also tried to use digital communications analysis in the context of high-speed links. In this approach [12], all noise and interference sources are mostly modeled with Gaussian distributions. This approximation works well near the mean of noise and interference distributions, i.e. for BERs that are not lower than roughly 10^{-4}, and with very long equalizers, which is the reason why it is used in digital communications. There, the inner and outer codes take care of lowering the BER from 10^{-4} uncoded, to target system BERs of 10^{-9} and lower. Unfortunately, neither the high uncoded BER assumption nor the long equalizer tap assumption can be applied to links. These differences lead to a big discrepancy between the Gaussian model estimates and real BER for the low BERs that are targeted in links.

To address these issues, we developed a new link model. We have already seen in Section 2.2.2 , how this new model integrates both voltage and timing noise in one domain and propagates different noise sources through the system from their source to the destination at the receiver slicer. In this chapter, we will complete the model by showing how to compute accurate ISI and crosstalk distributions.

In parallel with our work on these accurate link models, several other authors have also looked at these issues. Ahmad in [112] attempted to compute the BER of the whole link accurately, by computing the joint conditional probability distributions of transmit and receive jitter, and ISI and crosstalk. Although mathematically correct, his approach was very computationally intense since the computation of joint probability mass functions (PMFs) is a very hard combinatorial problem. Later both Casper [113] and Ahmad [104] revised Ahmad's previous approach by first computing accurate ISI and crosstalk distributions and then computing BER by using this interference distribution as conditional for noise. Both models, however, lacked an accurate and fast method of including the impact of transmit and receive jitter, and the CDR loop. In Section 2.2.2 we illustrated an efficient solution to this problem [37, 60]. Similar approach for jitter analysis was also published shortly after our initial publication, by Balamurugan *et al*

[114] and also by Hanumolu *et al* [115].

In the rest of the chapter we first integrate the interference and noise analysis into one system model and compute the BER of the link. Then we look at the sensitivity of the system performance to each of the system parameters and noise sources. This analysis is the culmination of the link system modeling work in this dissertation. It identifies the architectures for next generation baseband links, indicating that residual ISI and circuit precision limit the number of modulation levels to PAM2 and PAM4 in current technology. The implications of this analysis are used to drive the design and specification of the link sub-systems and overall link architecture. It appears that these new link architectures are capable of achieving 5-12 Gb/s over a variety of backplane channels.

4.1 System BER Modeling

In this section we first revisit the method proposed by Casper [113] and Ahmad [104] to compute the accurate interference distributions. Then we integrate this into our new system model with timing and voltage noise sources and CDR loop behavior to illustrate the performance of the whole system.

4.1.1 Convolution Method for PMFs of ISI and Crosstalk

Assuming that we transmit uncoded, random and independent data through the link, the received signal is just a sum of random variables (transmitted symbols) weighted by the pulse response of the channel, as a result of the convolution through the channel. The expression is a bit more complicated with transmit pre-emphasis since then, transmitted symbol b_k in Equation (2.5), is actually itself a weighted sum of random data a_k.

Nevertheless, the resulting received signal is still the weighted sum of independent random variables and its probability distribution is just a convolution of the PMFs of the weighted random variables [116]. Both Casper and Ahmad used this approach to compute accurate interference distributions, and we will also use it in our analysis.

This concept is illustrated in a simple example shown in Figure 4.1. If we use binary signaling for each transmitted symbol (+/-1), the probability mass functions of each symbol weighted by residual ISI of 0.3 and 0.1 will be still two delta functions at 50% probability but positioned at +/-0.3 and +/-0.1 values, respectively. This just indicates with what probability the residual ISI will impact the received signal if we transmit independent random binary symbols with values of +/-1. In this scenario there are four possible ways that ISI can impact the symbol +0.3+0.1, +0.3-0.1, -0.3+0.1 and -0.3-0.1, each with 25% probability. This is exactly what we get if we just convolve the two PMFs in Figure 4.1.

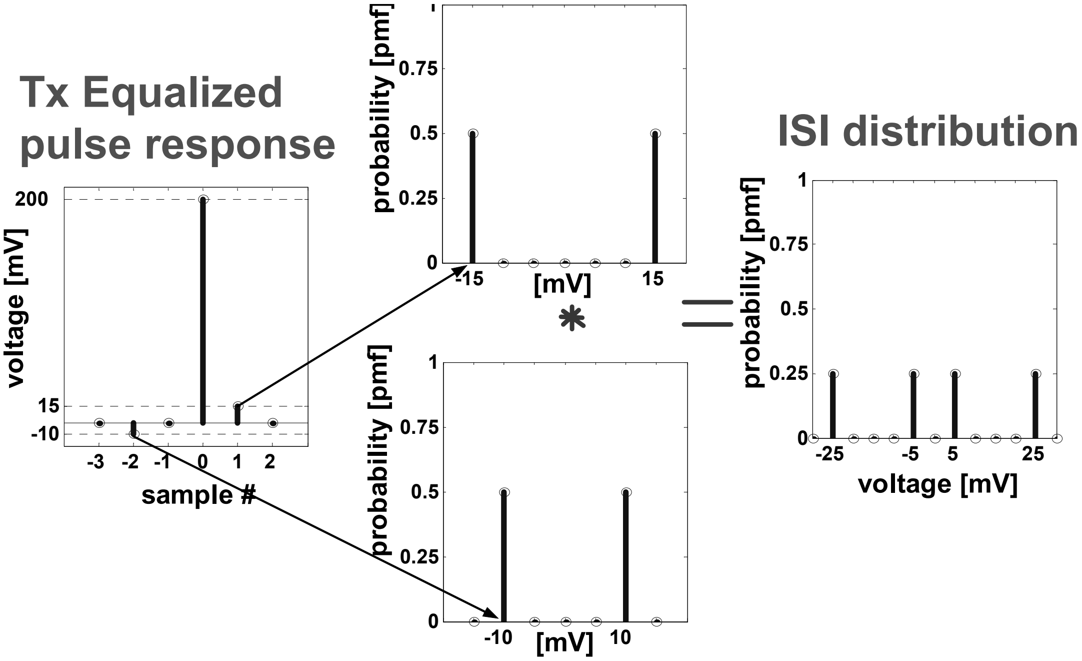


Figure 4.1: Pulse response ISI samples scale the PMFs of transmitted data symbols (+/-1), which when convolved result in the PMF of the residual ISI.

Using this method, we can compute the accurate PMFs of both ISI and crosstalk. The important point is that the complexity of computation increases linearly in the number of taps since we just have nested convolutions, while in time-based Monte-Carlo simulation, the space increases exponentially with number of channel taps.

For example, the PMFs of the residual ISI, after applying 5 taps of transmit equalization at 6.25 Gb/s to a representative channel, are illustrated in Figure 4.2. The plot on the left shows the PMF of the residual ISI at the data samples, and the plot on the right at symbol transitions as seen by the CDR phase detector[26].

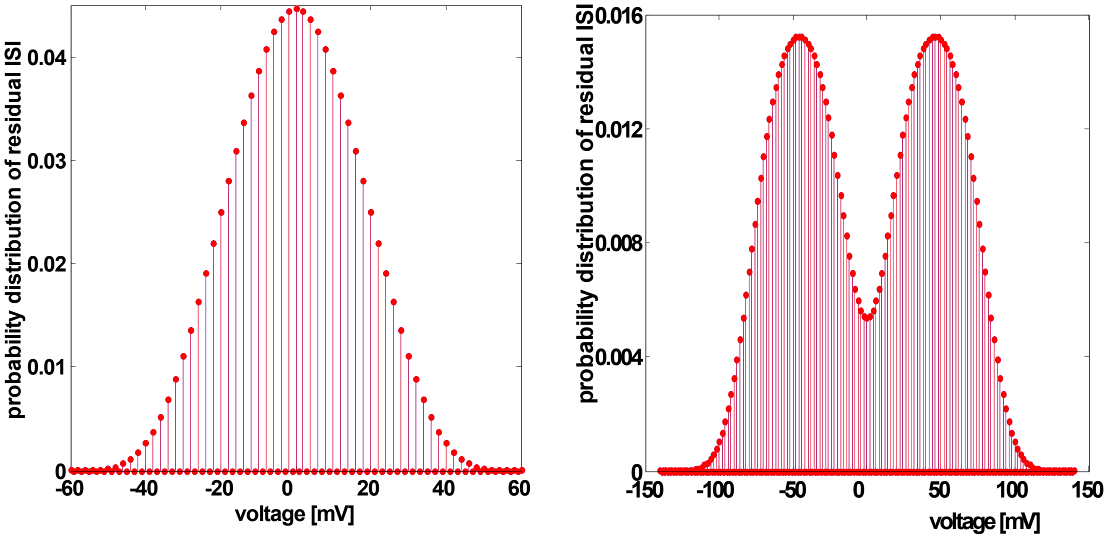


Figure 4.2: Residual ISI PMFs at data samples (left) and transitions (right), with 5 taps of transmit pre-emphasis.

The PMFs in Figure 4.2 are bounded. This causes large discrepancies between the accurate ISI distributions and their Gaussian approximations, since Gaussian distributions are unbounded.

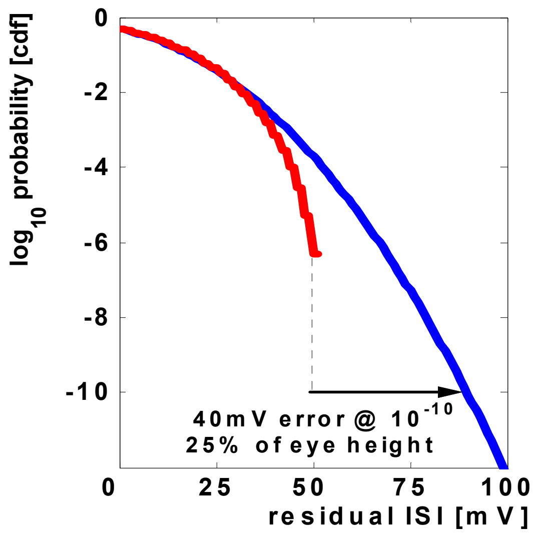


Figure 4.3: Error of the Gaussian approximation of the residual ISI distribution, impact on data samples, 5 taps of transmit pre-emphasis.

[26] Since transmit pre-emphasis is symbol spaced, it nicely controls the data samples but introduces bi-modal ISI distribution at symbol transitions which creates problems for the CDR loop.

In Figure 4.3, we plot the cumulative distribution function (CDF) of both the accurate ISI distribution and its Gaussian approximation. As we commented earlier, we see that the agreement is very good near the mean (for probabilities higher than 10^{-4}), which is why this approximation is used in digital communications. However, for probabilities lower than 10^{-10}, the Gaussian approximation is way off, which is a problem for high-speed links that work in this region. For example, at a probability of 10^{-10}, we get a 40mV error in the distribution width estimate, which is 25% of the eye opening in this particular case. This error is huge and thus we need to use the accurate ISI distributions.

The next figure illustrates the differences between the steady-state CDR phase probabilities when signal distributions are modeled using the Gaussian approximation, blue, and the accurate ISI PMF, red. We see that in this example, at probability of 10^{-10} the error in the peak phase estimates is about 50%, which is a lot.

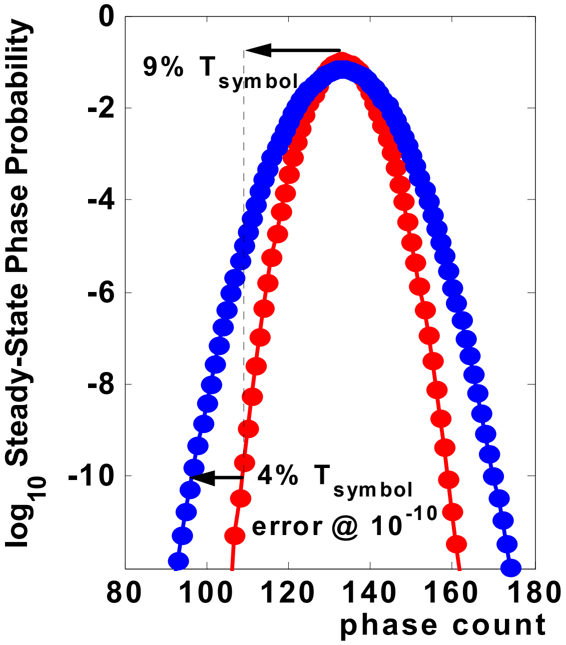


Figure 4.4: Comparison of steady-state probability distributions of CDR phases computed with Gaussian model of ISI, blue, and with accurate ISI model, red.

In the next section we show how to combine the ISI distribution and CDR phase distribution to compute the distributions of the received signal samples as seen by the receiver decision circuits.

4.1.2 Including the CDR Loop

We saw in the previous section that by using the PMF convolution procedure outlined in Figure 4.1 we can obtain the PMF of the received signal at a given sampling phase, as shown in Figure 4.2, for two discrete phases, data sample time and symbol transition. In order to understand the signal behavior during the whole symbol time, we can extend this procedure by changing the phase of the receive sampler in increments that are normally controlled by the CDR loop. In this way we can plot the "statistical eye diagram" of the received signal. For example, in Figure 4.5[27], we see the statistical distributions of the received signals, one with PAM2 at 3.125 Gb/s and the other with PAM4 at 6.25 Gb/s.

The ISI distribution in both examples is centered at the nominal received signal levels, ±200 mV in the PAM2 case, and ±200 and ±100 mV in the PAM4 case.

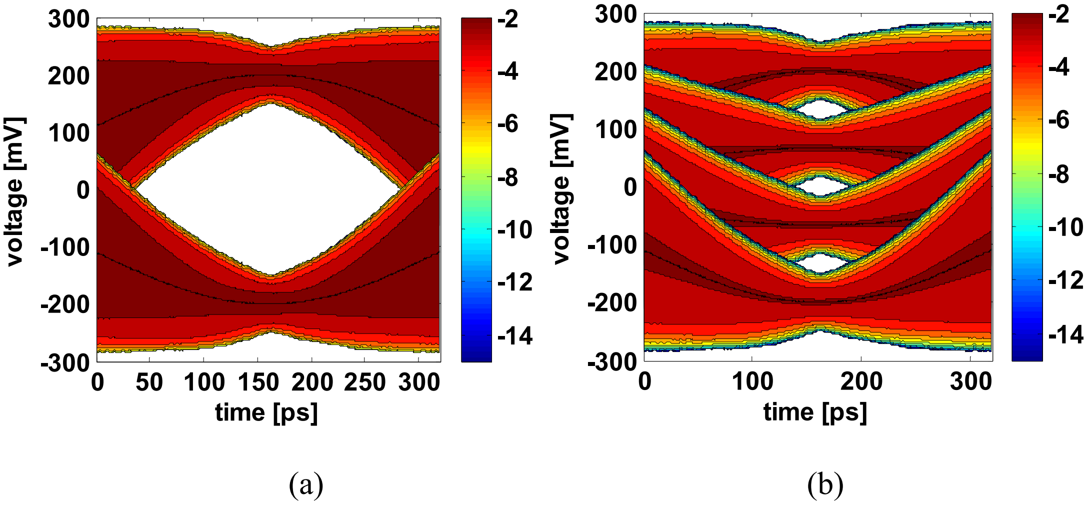


Figure 4.5: Statistical distribution of the received signal vs receiver phase, FR4 baseline case (a) PAM2 signal at 3.125 Gb/s (b) PAM4 signal at 6.25 Gb/s. Probabilities that are lower than 10^{-15} are shown as white space.

Although both examples use the same frequency range of the channel since the Nyquist frequency is the same, we get different ISI distributions around the nominal

[27] In this plot we show only the distribution of the signal in the victim channel as a function of time, assuming there is no crosstalk. In case we have crosstalk, that crosstalk is also treated as random interference and depending on the FEXT and NEXT pulse responses, crosstalk signal distributions change throughout the symbol time, assuming that the whole system is fully mesochronous, i.e. driven from one clock frequency generator. There are cases where aggressor signals are plesiochronous with respect to the victim signal which means that the two channels operate at slightly different frequencies. The crosstalk signal then "walks" in time across the victim signal and effectively creates the same crosstalk distribution at each time instant within a received symbol period.

signal levels since the transmitted symbol probability distributions are different for PAM2 and PAM4.

The plots in Figure 4.5 were obtained by sweeping the receiver phase through the symbol time. If we let the CDR loop lock to these signals, the loop would use the available phases with some probability, as we have seen earlier in Figure 2.25 and Figure 4.4, depending on the amount of interference and noise in the channel that would cause the CDR loop to choose the phase slightly off the nominal lock point, i.e. the mean of the phase distribution. In Figure 4.6, we show the CDR phase probabilities for the PAM2 and PAM4 examples discussed in Figure 4.5. In this example, each of the received symbols is 320 ps long and covered with 256 CDR phase steps, from an eight-bit phase interpolator.

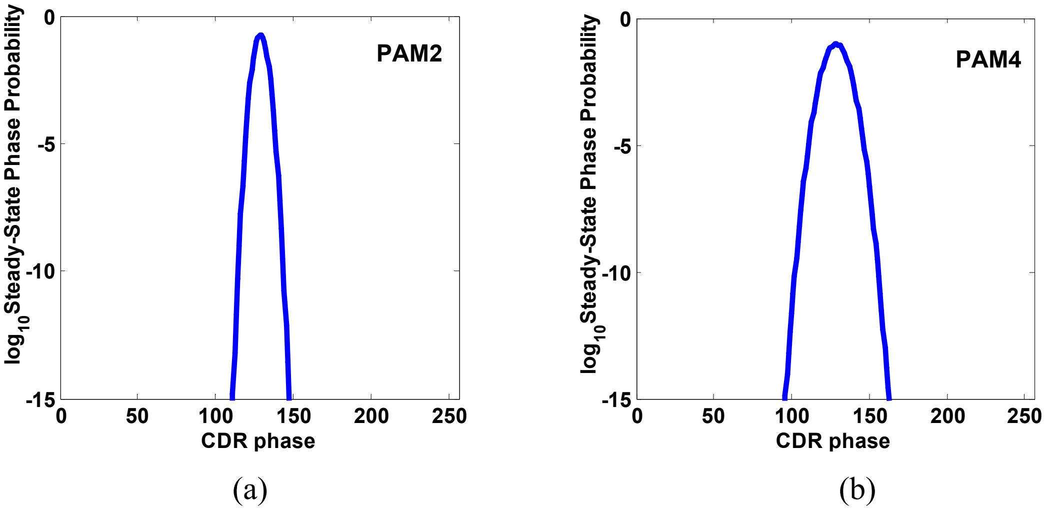
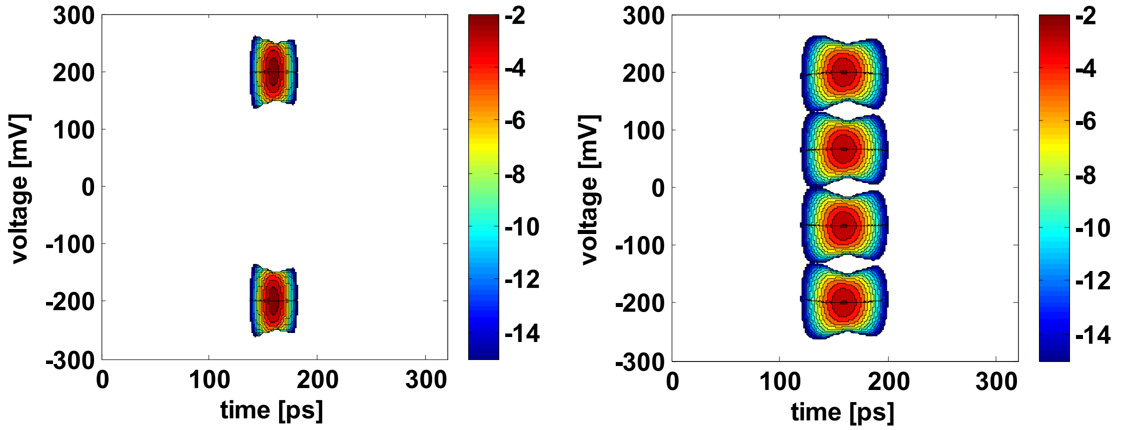


Figure 4.6: CDR loop phase probabilities, FR4 baseline case (a) PAM2 signal at 3.125 Gb/s (b) PAM4 signal at 6.25 Gb/s, CDR with minor crossings as in Section 5.3

By using the CDR phase probabilities as conditional probabilities for the receiver sampler, we can plot the distribution of the signal samples as seen by the receiver sampler triggered by the CDR loop. So, we multiply the signal distribution at each time-slice in Figure 4.5 by the probability of the CDR phase from Figure 4.6 that corresponds to that time. As a result, we get the distribution of the received signal samples as seen by the receiver slicers. In the ideal case, without noise and ISI, we would only have two constellation points in Figure 4.7 for PAM2 (at ±200 mV and 160 ps), and four for PAM4 (at ±200 and ±100 mV at 160 ps). However, due to ISI, the samples will deviate from their ideal positions with some probability, as shown in Figure 4.7. As in Figure 4.5, any signal points in this voltage-time space that are lower than 10^{-15} are shown as white space.



(a) (b)

Figure 4.7: Probability distributions of the signal sampled by the receiver with CDR loop, FR4 baseline channel, (a) PAM2 at 3.125 Gb/s, (b) PAM4 at 6.25 Gb/s.

We can now further manipulate these signal distributions by adding the noise sources and receiver sensitivity described in Chapter 2.

4.1.3 Computing BER Contours

By adding the noise sources to the signal distributions illustrated in Figure 4.5, and using the signal distributions as conditional probabilities for the error caused by the noise at each time slice, we can get the BER as a function of the time slice at which the receiver samples the incoming signal. This procedure is illustrated in Figure 4.8 where we first compute the BER as a function of sampling time within a symbol period. Then, the inner product of these BERs conditioned with time, and phase probabilities of the CDR loop, gives the total BER.

Given that we did not add any extra voltage margin in computing the conditional BERs or CDR distributions, the total BER corresponds to the voltage margin of zero, as indicated by a circle on the right of Figure 4.8. Had we added some extra voltage offset in receiver thresholds, for example 50 mV, when computing the conditional BERs and CDR phase distributions then we would have obtained the total BER that corresponds to a voltage margin of ±50 mV, depending on the sign of the added offset.

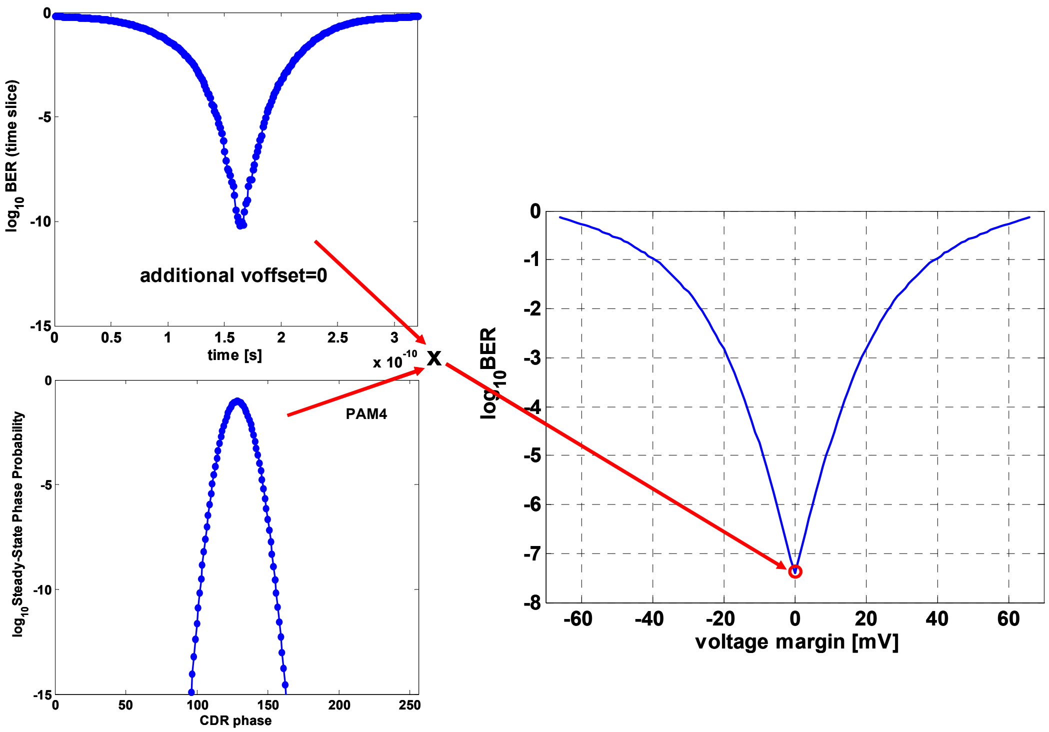
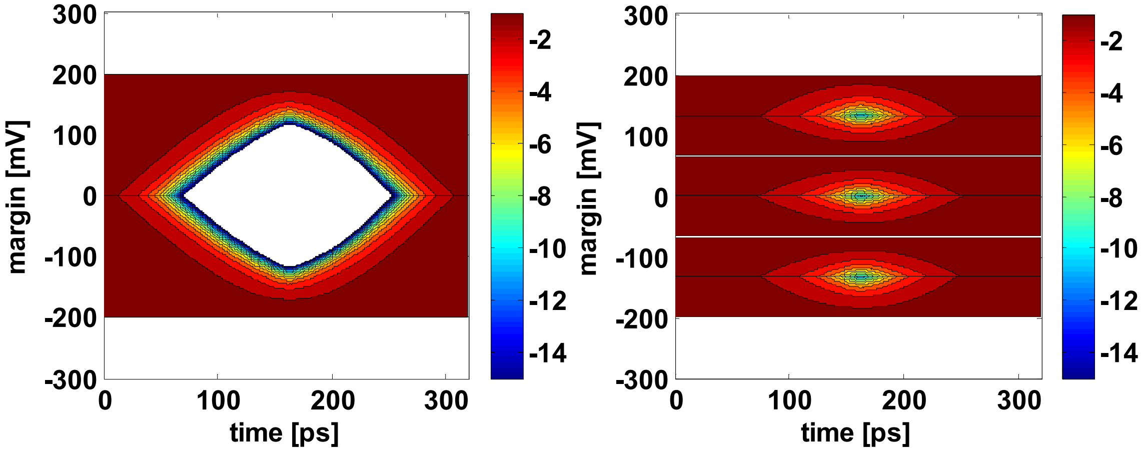


Figure 4.8: Computing the BER by combining the time conditioned BER (bathtub curve in upper left) and CDR loop phase probabilities (lower left), example for PAM4 at 6.25 Gb/s. Since conditional BERs are computed with receiver thresholds set at nominal levels with receiver sensitivity of 10 mV, as described in section 2.2.1 , the computed BER has no additional voltage margin. If in turn, we compute the conditional BERs assuming some additional voltage offset, for example 50 mV, then the computed aggregate BER would have an additional voltage margin of ±50 mV depending on the sign of the applied offset.

In estimating the system level performance, this approach is preferred, since we cannot always include all the possible noise sources, and even if we could it is always good to know how BER, as a measure of the system quality, behaves as a function of additional voltage margin. Note that contrary to common practice, the term “timing margin” is meaningless in the case where we use the CDR loop to position the sampling phase in the receiver. In this situation we have no control of the timing phase, and even if we offset the sampling phase externally, the CDR as a tracking loop would readjust and compensate for this disturbance[28].

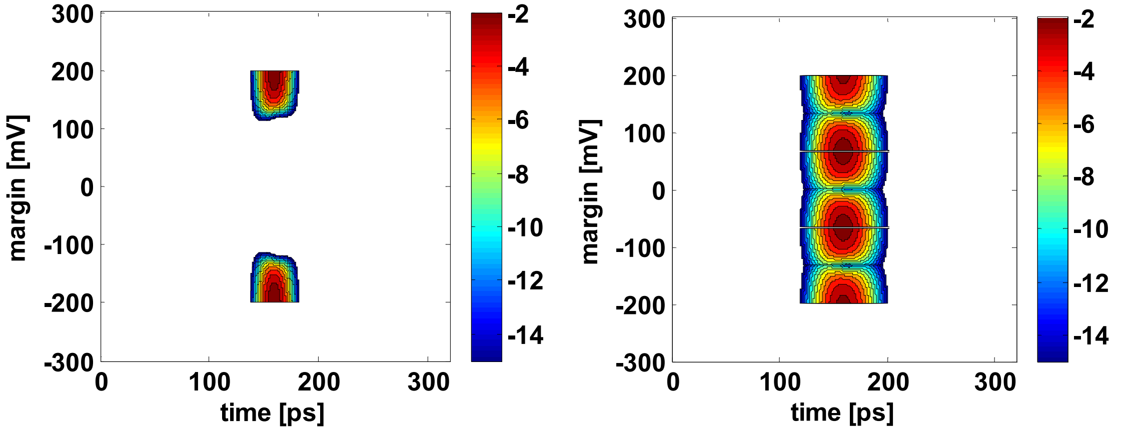
[28] Any mismatch between the phase detector timing and data sampler timing should be already included in the CDR model, since it cannot be corrected by the loop’s feedback.



(a) (b)

Figure 4.9: BER vs. voltage margin and sampling time, FR4 baseline channel, (a) PAM2 at 3.125 Gb/s and (b) PAM4 at 6.25 Gb/s

Another way to view the BER dependence on the voltage margin, is to plot the BER as a function of the time slice at which the signal is sampled and the additional voltage margin that is added. This plot gives us an idea how much voltage and timing margin we would have in a system where we can somehow determine the ideal sampling position in time. For example, in the PAM2 case shown on the left of Figure 4.9, the voltage margins are ±116 mV, and ±90 ps for BER of 10^{-15}. However, notice that the voltage and timing margins are mutually dependent since at a point where we lose the timing margin, 70 ps or 250 ps time index, we also lose the voltage margin.

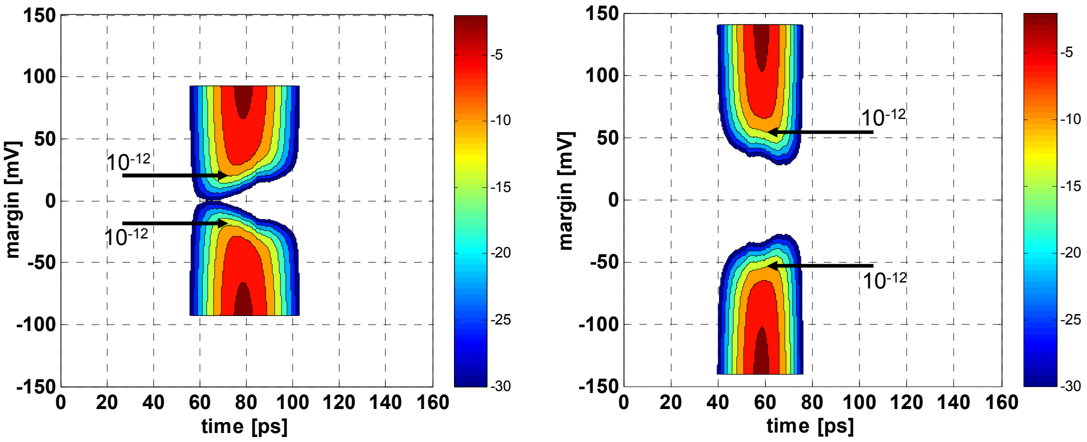


(a) (b)

Figure 4.10: BER vs. voltage margin and sampling time, with sampling time determined by the CDR loop, FR4 baseline channel, (a) PAM2 at 3.125 Gb/s and (b) PAM4 at 6.25 Gb/s

In order to take the CDR loop into account, we just need to condition each of the time slices in Figure 4.9 with the probability that the CDR loop could choose a phase that corresponds to that time slice, from Figure 4.6. This procedure is similar to the one we used in Figure 4.7. As a result, we can now observe at which point in time we have certain voltage margin with guaranteed BER, when sampling the incoming signal with a receiver phase chosen by the CDR loop, Figure 4.10.

As an example, we first compare the improvement in margins when one tap DFE using loop-unrolling is added to 5 taps of transmit pre-emphasis in PAM2, now with PAM2 running at 6.25 Gb/s. The voltage margin for a given BER is the minimum distance between the BER contours and the threshold at zero. For example, for BER of 10^{-12}, the voltage margin with transmit pre-emphasis only, Figure 4.11a, is about ±20 mV and it grows to ±50 mV when one tap DFE is added, Figure 4.11b.



(a)

(b)

Figure 4.11: Comparison of voltage margins from BER contours, PAM2 modulation at 6.25 Gb/s, (a) 5 tap transmit pre-emphasis and (b) 5 tap transmit pre-emphasis with one tap DFE with loop-unrolling.

The same data rate of 6.25 Gb/s can also be achieved by using PAM4 and reducing the signaling speed. In Figure 4.12 we compare the performance of PAM4 with 5 taps of transmit pre-emphasis with PAM2 with loop-unrolling from Figure 4.11b.

For this channel the margin is better for PAM4 than the PAM2 system with linear transmit pre-emphasis, but they are slightly worse than the margins for the DFE system. We can compare the margin of these three approaches for different channels formed by changing the length of the backplane trace. This data is shown in Table 4.1.

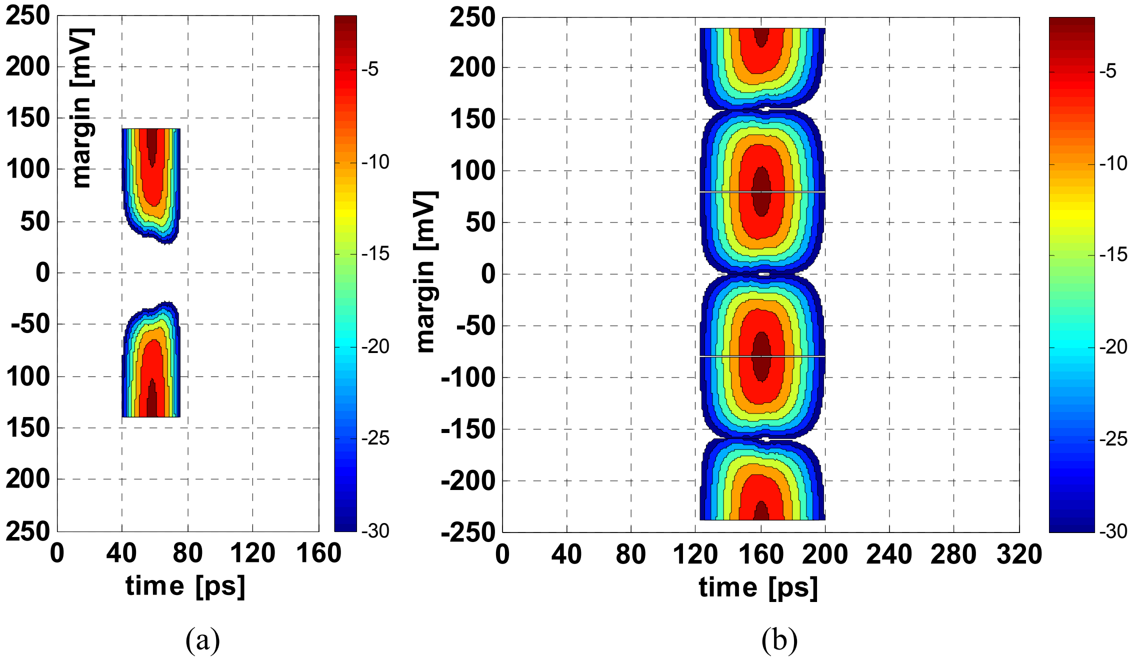


Figure 4.12: Comparison of voltage margins from BER contours, 6.25 Gb/s, (a) PAM2 modulation with 5-tap transmit pre-emphasis and one-tap DFE with loop-unrolling, (b) PAM4 with 5-tap transmit pre-emphasis.

Table 4.1: Voltage margins [mV] at target BER=10^{-12} at 6.25 Gb/s, for PAM2, PAM2 with DFE and PAM4 signaling, over 3, 10 and 20" backplanes. Transmitter peak output swing is ± 500 mV, and receiver sensitivity ± 10 mV.

Eq/Mod type vs. BP length	3"	10"	20"
PAM2	32	17	19
PAM2 w. DFE	79	49	44
PAM4	10	37	31

Short channels have less loss and more reflections, so not surprisingly PAM2 systems are much better than PAM4 systems in these situations. As the backplane gets longer, the attenuation increases, and now PAM4 is better than PAM2 with linear filters, but it is still not better than PAM2 with a simple DFE. In longer backplanes, the loss at PAM2 signaling rate becomes unbearable, and PAM4 is expected to be more favorable. Of course, this all depends on the intended signaling rate for each type of modulation and the noise sources. In the next section we evaluate the performance of equalization and multi-level modulation techniques with link-specific noise sources and hardware constraints.

4.2 Sensitivity Analysis

In this section we analyze the performance of high-speed link systems in two ways. First, we would like to understand the spectral behavior of different noise sources as seen by the receiver sampler, and then look at how each of the noise sources impacts the link data rate.

4.2.1 Spectral Properties of Noise Referred to the Receiver Input

We have already presented in Section 2.2.2 the spectral properties of jitter-induced voltage noise at the receiver. We extend this spectral analysis to other noise sources presented in Chapter 2, and now that we have covered the equalization in Chapter 3, we also analyze the effect of equalization on the noise in the system.

The model for jitter propagation in Figure 2.18 indicates that transmit pre-emphasis affects both the signal propagation path, marked as channel pulse response *p(nT)*, and jitter propagation path marked as channel impulse response delayed by a half of the symbol time *h(nT+T/2)*. We start our analysis by looking at the effect of transmit pre-emphasis on these two paths, shown in Figure 4.13.

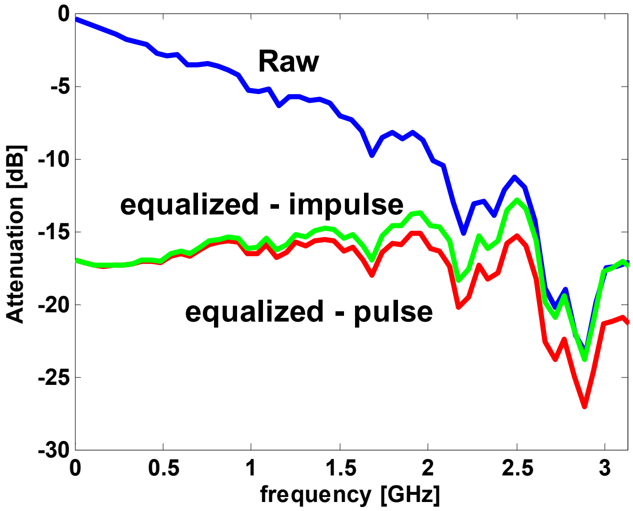


Figure 4.13: Channel frequency response (Raw), and frequency response of the equalized 160 ps pulse, and impulse.

As we said earlier, transmit pre-emphasis attenuates the low frequencies of the signal to match the highest attenuation in the channel, and as shown in red in Figure 4.13, the frequency response of the equalized channel is fairly flat. If we apply the same pre-emphasis filter to channel impulse response (the jitter path in Figure 2.18) we see that it similarly affects the transmitted impulse, i.e. attenuates the low-frequencies to match the attenuation at the Nyquist frequency.

Intuitively, if transmit jitter is uncorrelated, the pre-emphasis filter does not significantly attenuate the transmit jitter-induced voltage noise, since the pre-emphasis filter does not really attenuate the signals at high-frequencies. If, on the other hand, the transmitter jitter is heavily correlated, than its effective voltage noise will be significantly attenuated by the pre-emphasis filter, since the filter attenuates the low frequencies of the signals. Similarly, since the effective voltage noise from receiver jitter is at each sample a shift of the whole transmit sequence, it is seen by the transmit pre-emphasis filter as a DC signal component and is attenuated significantly.

In Figure 4.14, we illustrate these effects by first plotting the spectrums of jitter-induced voltage noise shown previously in Figure 2.17, and then indicating the shift in the power spectral densities (PSDs) due to transmit pre-empahsis. As we intuitively explained, the effective voltage noise due to white transmitter jitter is affected very little by the equalizer, and is the most dominant source of jitter-induced voltage noise.

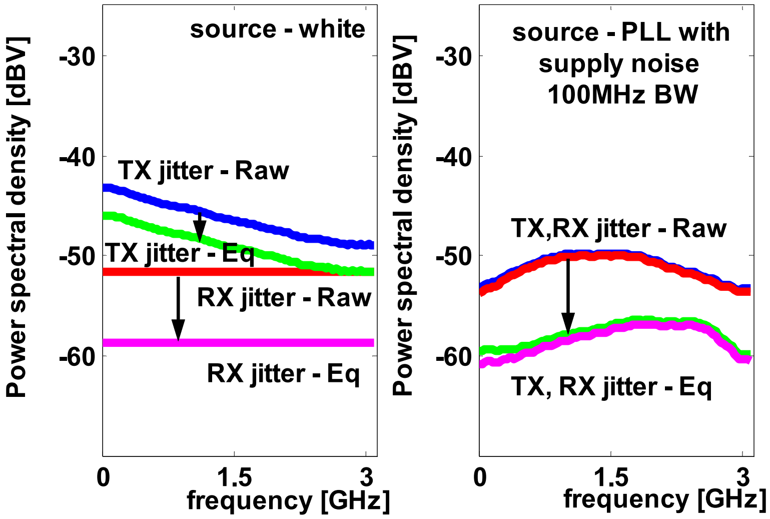


Figure 4.14: PSDs of transmitter and receiver jitter mapped to the data sample point through equalized and unequalized (Raw) channel, for cases of white jitter PSD and colored jitter PSD as output of the PLL with supply noise limited to 100 MHz.

The voltage noise induced by correlated transmitter jitter, which in our example originated from the PLL with supply noise limited to 100MHz, is significantly attenuated by the equalizer. It is interesting that regardless of the spectral properties of the receiver jitter, the effective voltage noise from receiver jitter is significantly attenuated by the pre-emphasis filter since it is treated as a DC signal component.

When we compare the jitter-induced voltage noise to other error sources, like equalizer quantization noise, tap value estimation error, and residual ISI of the equalized channel, we see that the jitter noise is comparable to some of these error sources. Using the analysis in Chapter 3 and Appendix A we plot in Figure 4.15 the spectral properties of the residual ISI, equalizer estimation and coefficient quantization errors, as seen by both the data slicers and the CDR phase detectors, and compare them with the spectral densities of the jitter-induced voltage noise from Figure 4.14.

The size of the residual ISI after equalization depends on the length of the equalizer. For the plot in Figure 4.15, we use a 5-tap linear equalizer (large for current high-speed links).

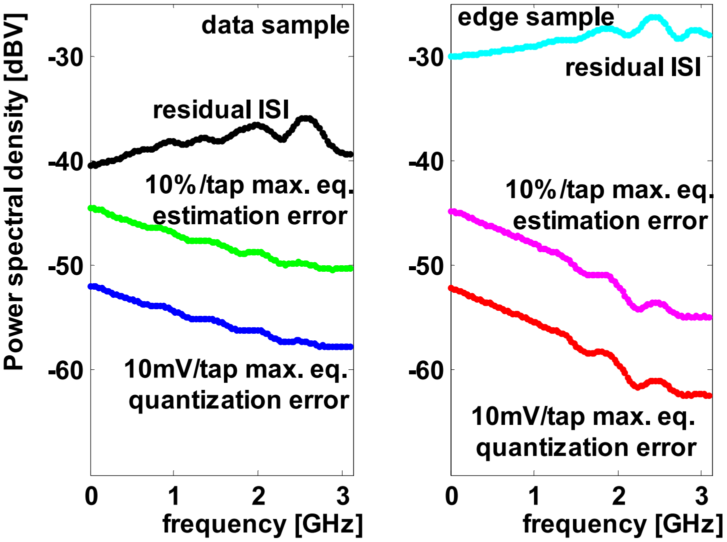
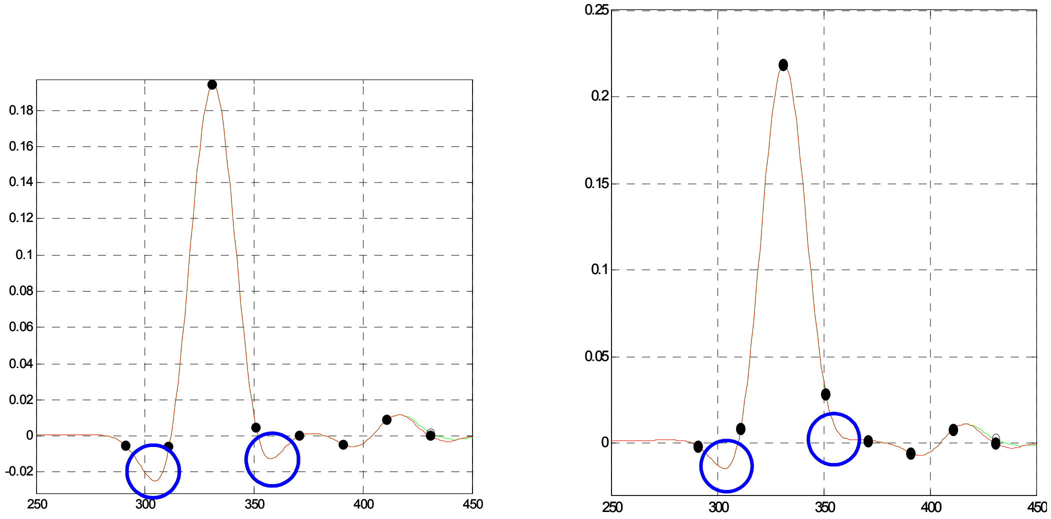


Figure 4.15: Power spectral densities of quantization, estimation and residual ISI errors, at data and edge samples for a 6.25 GHz NRZ signal using a 5-tap linear equalizer.

In Figure 4.15, we see that the residual ISI at the edge sample is much larger than that at data samples. This is expected, since the transmit pre-emphasis filter is symbol-spaced, i.e. its only goal is to reduce the ISI at the data samples (in the middle of the received eye). It not only disregards the ISI at the transitions, but it in fact introduces more ISI at the edges as we have seen from the bi-modal distribution of edge samples in Figure 4.2.

The main causes for this bi-modality can be located if we look at the equalized pulse response, in Figure 4.16a. The symbol spaced samples in Figure 4.16 are marked as black dots. We see that pre-emphasis creates two negative spikes between the sample times since it forces the ISI to zero at the sample times. These spikes, circled in Figure 4.16a, cause the predominantly bi-modal distribution at the edge samples.



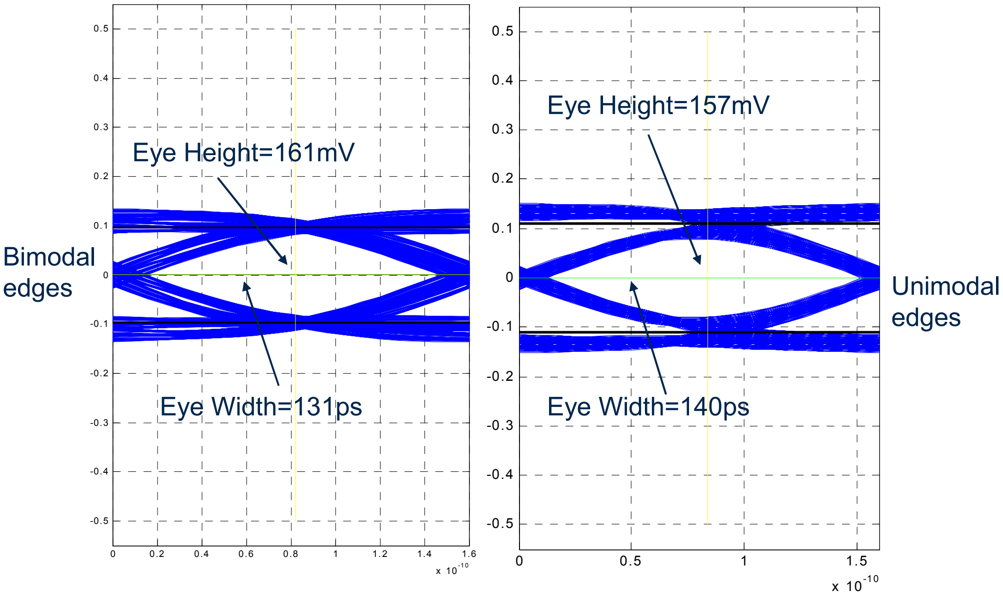
(a) (b)

Figure 4.16: Pulse responses equalized with transmit pre-emphasis, (a) using only error information from the data samples, (b) combining the error information from both data and edge samplers.

One way to get around this is to include the error information from the edge samplers when computing the equalizer coefficients from the data sampler error information. As an example, we modify the adaptive equalizer update loop, Equation (3.8), to include the information from signal transitions

$$\begin{aligned}\underline{w}_{n+1} = \underline{w}_n + step_{wd} sign(e_{dn}) sign(\underline{u}_{dn}) + \\ + step_{we} sign(e_{en}) sign(\underline{u}_{en})\end{aligned} \tag{4.1}$$

where e_{dn} is the error from the mean level of the received signal at data sample time, e_{en} is the error from the threshold level of the received signal at signal transitions, i.e. samples by CDR loop phase detector, $\underline{u}_{dn}$ is the vector of received data and $\underline{u}_{en}$ is the sum of the vectors of neighboring received symbols. By choosing the update weights $step_{wd}$ and $step_{we}$ we put more emphasis on ISI compensation at either edge or data samples.



(a) (b)

Figure 4.17: Eye diagrams of the received signal equalized with transmit pre-emphasis, (a) using only error information from the data samples, (b) combining the error information from both data and edge samplers.

Since we use symbol-spaced pre-emphasis we can only do one of the two things correctly. With different update steps, we actually make a compromise between the amount of ISI seen by the data and edge samplers. This is illustrated in Figure 4.16b, where by putting a small weight on the edge error we see that the edge peaks are reduced at the expense of increasing the ISI at data samples. In Figure 4.17, we illustrate this effect on the eye diagram of the received signal. For example, we see that voltage margins get reduced by 4 mV or 2.5%, while timing margins increase by 9 ps or 7%.

Another method of dealing with high ISI at signal transitions is to decrease the bandwidth of the CDR loop filter, or said differently use more samples to vote on the direction of the phase increment. This is more desirable in situations where we do not have enough voltage margin to trade off with timing. The only issue is that the latency of the CDR loop then increases and we need to add a frequency tracking loop [22,117], since the frequency tracking ability of the first order CDR loop diminishes with lower bandwidth.

Going back to Figure 4.14 and Figure 4.15 we see that the residual ISI is the largest error source, followed by the jitter-induced voltage noise. After these are resolved, we will have to start using equalizers with higher resolution and lower estimation error.

In this section we tried to qualitatively illustrate the impact of different link impairments by looking at the spectral properties of noise sources and residual interference for a fixed link configuration. Next we want to use the link system model that we have developed so far to explore the design space and find the most efficient equalization and modulation architectures with link-specific noise sources and hardware constraints.

4.2.2 Impact of Noise and Hardware Constraints on Link Data Rate

Before we fix the noise sources and start exploring the architectures, let us first take a look at the sensitivity of the link performance to the magnitude of thermal noise and jitter. This will give us a feeling of how much our final conclusions depend on the noise assumptions that we made.

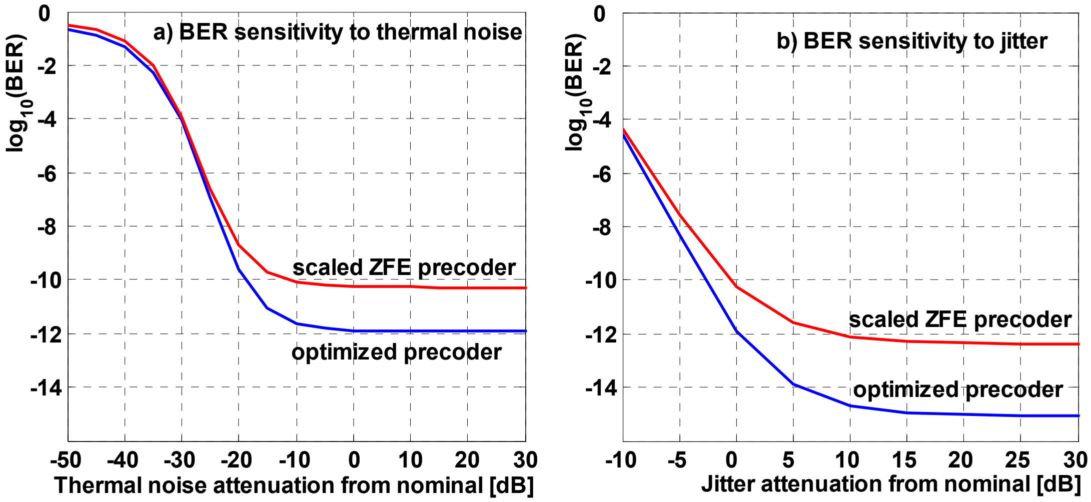


Figure 4.18: Sensitivity of BER to changes in: a) thermal noise, b) jitter, for a 5-tap precoder with coefficients from scaled ZFE and the optimization in (3.6). The system transmits PAM2 at 10 Gb/s, with the Nelco channel.

Figure 4.18 shows the sensitivity of a 10 Gb/s PAM2 Nelco backplane link to thermal noise and jitter. It also compares the link results from using scaled ZFE and the optimization results from the framework formulated in (3.6). The effect of changing the effective thermal noise is shown in Figure 4.18a, and effective jitter in Figure 4.18b. For Figure 4.18, and all the data given in this section, a noise figure of 7 dB is added to the (1 nV)^2/Hz thermal noise of the termination resistors, to account for thermal noise in the slicer. In addition, when noted we will also assume 10 mV of slicer resolution, and sampling jitter from a ring oscillator PLL with a standard deviation σ_ε=5° [39].

It is clear that noise from jitter is dominant in this link. Voltage noise due to jitter is especially harmful since it is proportional both to signal energy and jitter variance. This means that the only way to improve the system performance after system optimization is to minimize the jitter variance by careful circuit design.

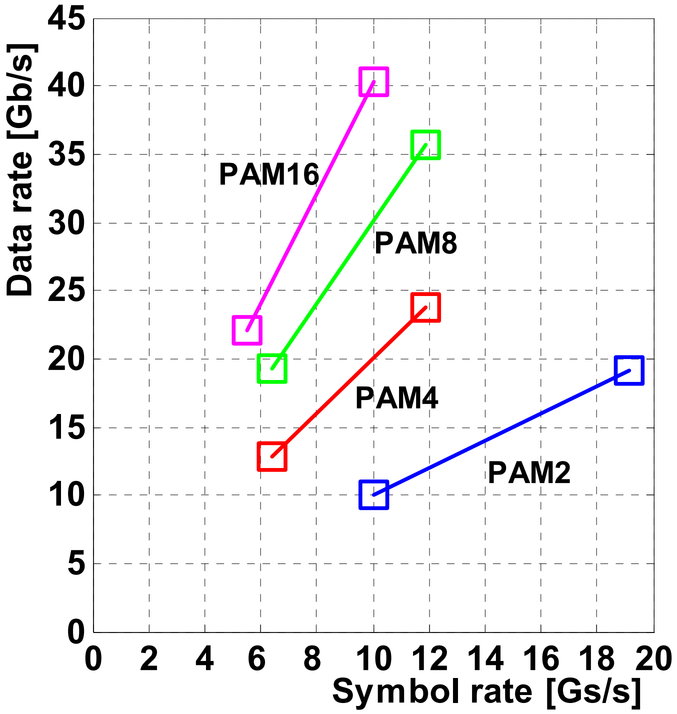


Figure 4.19: Data rates for 50-tap precoder with 80-tap feedback equalizer on best and worst channels with thermal noise, using different modulation levels. Two baseline channels in Figure 3.1 mark the beginning and the end of the range of achievable data rates with target BER of 10^{-15}.

Using our optimization framework we can now compare the expected performance of a number of different link architectures. Figure 4.19 gives the achievable data rates if there were no hardware complexity constraints – using the precoder as a feedforward filter and assuming perfect feedback equalization in the receiver – and the links were limited only by thermal noise. The plot illustrates the performance range between the best and worst channels (baseline channels from Figure 3.1) for different levels of modulation. The x-axis is the symbol rate and the y-axis is the data rate so the slope of PAM2, …, PAM16 curves is 1, …, 4 corresponding to the number of bits per symbol for each of the modulation formats.

Using higher levels of modulation in Figure 4.19, the system more efficiently utilizes the usable channel bandwidth (9-12GHz, from bit loading in Figure 3.4), and achieves very high data rates. We see that by using sixteen modulation levels in PAM16 we can more than double the data rates achievable with binary signaling. However, we need to look at other sources of noise in order to evaluate the efficiency of multi-level modulations.

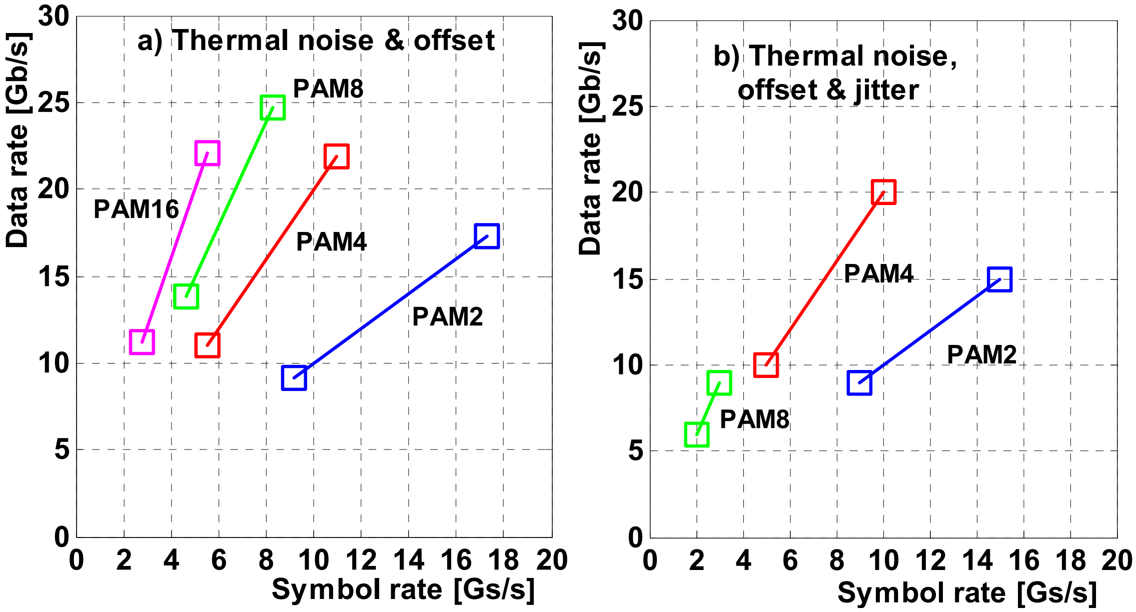


Figure 4.20: Data rates for 50-tap precoder with 80-tap feedback equalizer on best and worst channels, using different modulation levels in the presence of (a) thermal noise and sampling resolution, (b) thermal noise, sampling resolution and jitter. Two baseline channels in Figure 3.1 mark the beginning and the end of the range of achievable data rates with target BER of 10^{-15}.

In Figure 4.20a, we add the receiver sampling resolution requirement, and in Figure 4.20b, the sampling jitter. We see now that the high-data rates, provided by PAM with high number of levels, have decreased and that the highest data rates are provided by PAM4 and PAM2. PAM4 still almost doubles the PAM2 data rates covering the range from 10 to 20Gb/s. Figure 4.19 and Figure 4.20 indicate that in order to make better use of the available channel bandwidth, i.e. increase the spectral efficiency by using the multi-level signaling, we need to design better circuits. We need sampling circuits that are more precise and timing generation that is less noisy.

Sampling resolution imposes a constraint on the minimum distance between constellation points, so that one cannot add more constellation points within the peak power constraint without degrading system performance. This limits higher bandwidth utilization.

With good oscillator design, jitter noise is not dominant for small constellation sizes. However, since the energy of the jitter-induced voltage noise is proportional to signal energy, it becomes more detrimental as energy remains the same and minimum distance between constellation points decreases. Therefore, jitter also prohibits the use of large constellations.

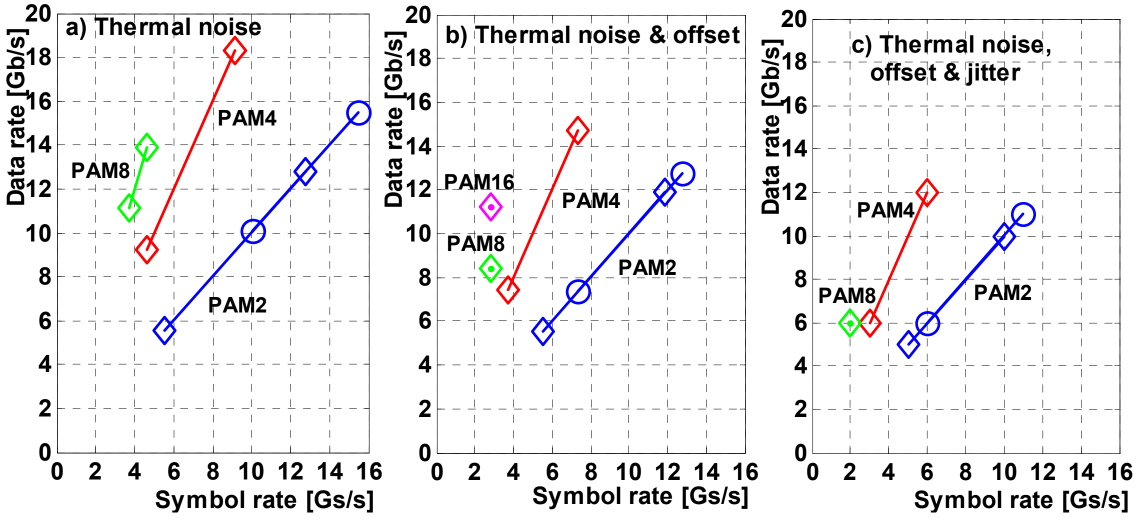


Figure 4.21: Achievable data rates with different noise sources for two architectures (◊) 5 taps of transmit precoding with 20 taps of windowed reflection cancellation, similar to [39], with different levels of modulation, (o) same architecture, with "loop unrolling" by one extra tap of feedback equalization with no latency [81] (only PAM2 modulation is practical due to exponential growth in complexity).

It is interesting to mention that a precoder filter alone has very poor performance, even without any constraints on complexity, due to the peak power constraint and large amount of ISI in the channels. Figure 4.21 shows the projected data rates of practical baseband link architectures, keeping the complexity/power within the power budget of the state-of-the-art links [39]. Since the large ISI cannot be completely compensated with this hardware complexity, higher PAM modulations start to fail. In fact residual ISI literally halves the data rates from those in Figure 4.19 and Figure 4.20. Today's links simply cannot afford to compensate all the ISI and that is what currently limits their data rates.

Adding the one tap of immediate feedback equalization using loop-unrolling improves the performance of PAM2 and pushes it deeper into the overlap with PAM4. On these channels PAM2 or PAM4 work better depending upon the accurate ISI distributions, as we indicated earlier in this chapter.

4.3 Summary

From the previous three figures, we see that both the receiver resolution and sampling jitter are limiting factors for the application of multi-level signaling techniques (higher than PAM4). Any form of feedback equalization applied to dispersion ISI taps improves the performance, as shown in the PAM2 example where loop unrolling is used to cancel the first causal ISI tap. In order to achieve very low BERs, it is also essential to remove the long-latency reflections with tap-selective feedback equalizer as we explained in Section 3.2.1 .

Our results clearly show that multi-level modulation together with precoding and feedback equalization with no latency is essential to achieving high data rates. In fact, the data rates of infinite length precoders and feedback equalizers are achievable with about 50 precoder taps and 80 feedback taps with no latency gaps. These rates, although high, are still not very close to the data rates projected in Figure 3.3, for integer uncoded multi-tone constellations with thermal and phase noise. While improving the performance of baseband techniques is challenging, to achieve the rates projected in Figure 3.3 will require the implementation of a practical multi-tone system that operates at these channel bandwidths.

This design space exploration also shows that for current circuit technology we can design the next generation of links operating at 5 to 12 Gb/s over a wide range of link channels, if we use around 5 taps of transmit pre-emphasis, 20 taps of reflection cancellation at the receiver, and we interchangeably use PAM4 and PAM2 modulation with one tap loop-unrolling. In the next chapter, we will describe an implementation of this link architecture.

Chapter 5

Experimental System

Having discussed the global issues in building efficient links, we can now focus on the practical implementation issues. As we said earlier, the system level optimization suggests that a link with dual-mode PAM2 and PAM4 modulation, transmit pre-emphasis and feedback equalization (both reflection cancellation and one-tap DFE with loop-unrolling) can achieve 5 to 12 Gb/s on a wide range of backplane channels.

The next generation link we describe in this chapter is designed to match this architecture with minimum complexity, by maximizing the number of components that are re-used in all modes of operation. The link features a dual-mode PAM2/PAM4 transceiver, a reconfigurable receiver with loop-unrolled DFE and reflection cancellation, and a reconfigurable CDR loop for multi-level and partial-response[29] input signals. We also present a version of the adaptive algorithm from Chapter 3, which enabled the efficient implementation of link adaptation and calibration algorithms, with minimum additional hardware at the receiver front-end.

The link was developed jointly with the RaserX group in Rambus Inc. Although we architected the link and worked on parts of the implementation and testing, this link would not have been possible without the implementation and design effort of many

[29] A partial-response channel has a controlled amount of ISI. In case of one-tap of DFE with loop-unrolling, the channel has one tap of controlled ISI.

people in the RaserX group, and we will mention them appropriately as we describe each of the blocks.

5.1 Link Architecture

Over the years, links have grown to be relatively complex systems, especially with the recent addition of various signal processing blocks [39, 81, 107, 118]. In Figure 5.1 we show a block diagram of our high-speed link [39]. The transmitter consists of a parallel-to-serial converter and pre-emphasis filter. The PLL in the clocking section generates the high-speed link clock from a slower reference clock. The CDR loop selects the right phase from the phase mixer via a phase control mechanism which depends on the type of modulation or equalization that is used in the system. The receiver receives and de-serializes the data, but also feeds data back through a scaled replica of the transmit pre-emphasis filter, implementing the tap-selective reflection cancellation loop.

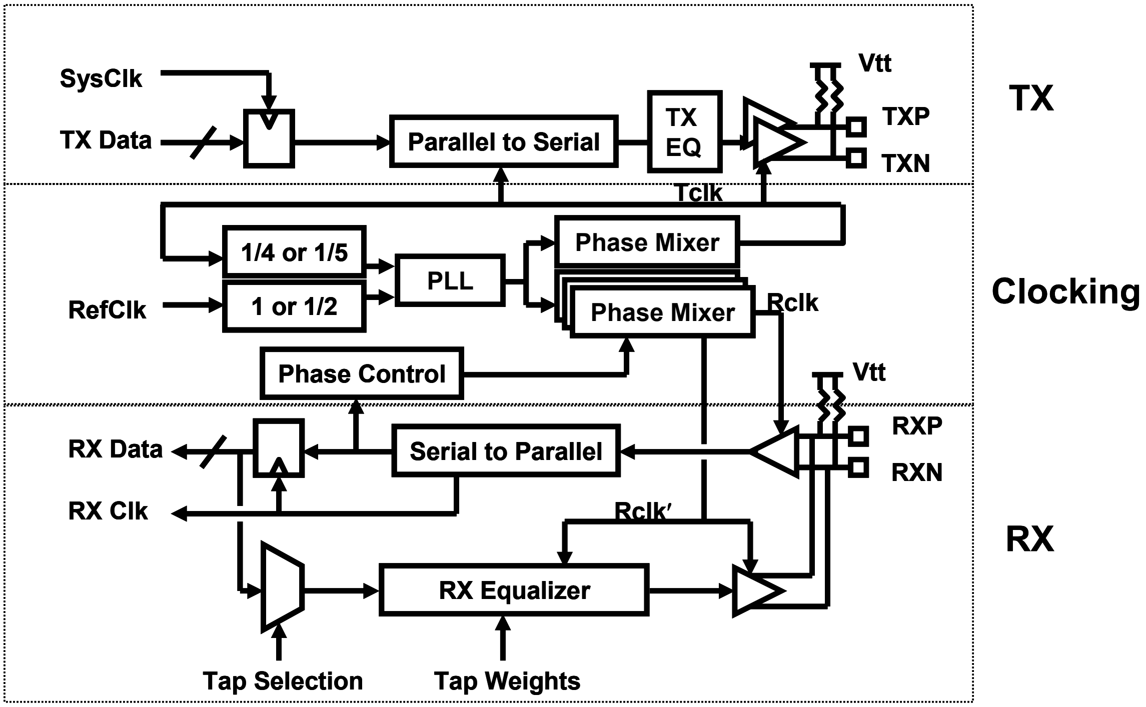


Figure 5.1: Link block diagram – transmitter with pre-emphasis, clocking infrastructure and the receiver with feedback equalizer

The link is a double data rate (DDR) system, which means that a data symbol is transmitted and received on each phase of the clock. We use the input multiplexed scheme [119] to minimize the output parasitic capacitance of the transmitter. For higher multiplexing ratios output multiplexing [13] is preferred.

Figure 5.2 shows more details on the transmitter and multi-level receiver implementation, which includes additional hardware for link adaptation.

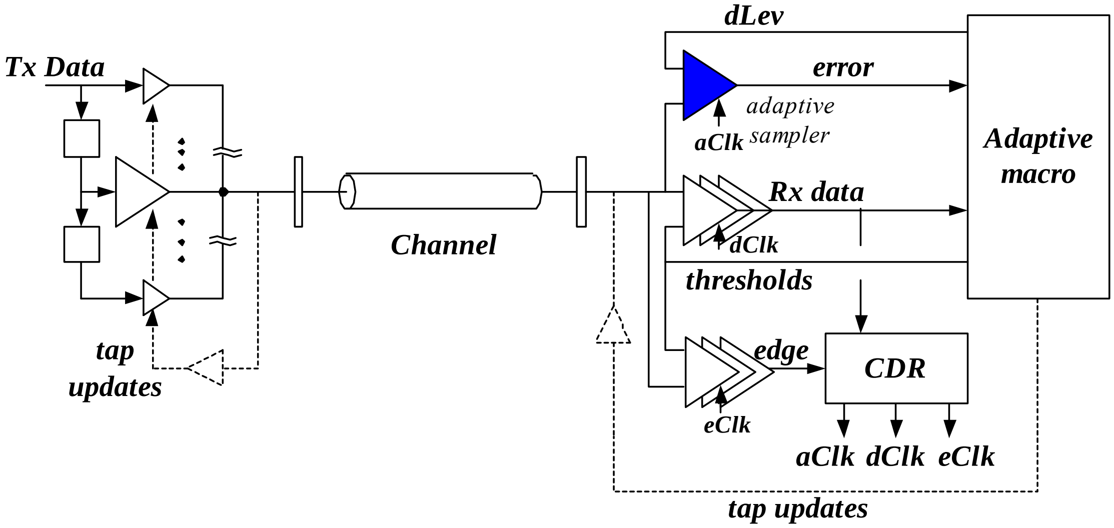


Figure 5.2: Adaptive multi-level equalizing link architecture [81]

The transmitter is an analog FIR filter, implemented as a bank of weighted current-mode drivers (segmented DAC) driven by delayed data. The receiver has data and edge sampler banks for a two-times oversampled bang-bang CDR loop. Each bank has three samplers per clock phase (twelve samplers total per link), for DDR and dual PAM4/PAM2 operation.

Using this generic link infrastructure as a base, we add an extra sampler to provide the error signal to the adaptive macro block. With variable timing and voltage references, this adaptive sampler can also serve as an on-chip sampling scope [109, 118, 81] that samples the received signal and monitors the performance of the link. In addition to this, we architected the receiver front-end such that the adaptive sampler can take the role of any other sampler in the receiver while that sampler is being calibrated off-line, without interrupting the flow of the data through the link. This calibration with uninterrupted data flow is very important in backplane links that are used in router or blade-server applications.

The adaptive system with transmit pre-emphasis requires a back-channel link from the receiver to the transmitter, to communicate the pre-emphasis updates. Andrew Ho proposed in [58] that common-mode signaling [120] be used in the back-channel since the forward link is differential and robust to common-mode variation. The link in [58, 81] is architected with differential high-speed forward channel and slow, common-mode back channel, to enable the adaptation of transmit pre-emphasis and other control functions that need to be communicated back to the transmitter. In the sections to follow, we will describe each of the link blocks in more detail.

5.1.1 Transmitter

The transmitter that we describe in this section has two important features. It uses tap-sharing in the pre-emphasis driver/filter to minimize the parasitic output capacitance of the transmitter and uses Gray coding to implement the dual PAM2/PAM4 operation with minimum complexity [39].

As we said earlier, most high-speed link transmitters incorporate some sort of pre-emphasis filter. High throughput and a very small power budget (<40 mW/Gb/s) rule out the implementation with a digital FIR and a DAC. Most link designs use either analog FIR filters [27, 39] with programmable weighted drivers, or RAM DACs [89, 121] where pre-emphasized symbol values are already computed and stored in a fast SRAM block in the transmitter and addressed by a window of transmit data that corresponds to the desired number of taps.

One problem with the RAM DAC approach is that it is not easy to update the whole RAM with new equalized symbols at each update of transmit pre-emphasis coefficients. This requires relatively complex hardware, essentially a digital FIR filter that can operate at the rate of adaptive equalization multiplied by the number of equalized symbols, and not the rate of the data flow through the link.

To avoid this, we use an analog FIR filter as a transmit pre-emphasis driver [39]. As we mentioned earlier, in a straightforward realization this filter is implemented as a bank of weighted drivers driven by delayed data. To allow for the full programmability of the filter, each of the drivers has to be sized the same, in order to be able to carry the same amount of current, i.e. tap weight.

Figure 5.3 shows a thermometer coded driver (for PAM4/PAM2 operation), where each of the segments is implemented as a 5-tap analog FIR filter with identical subsegments driven by delayed data (e.g. B[0], …, E[0] are delayed versions of A[0]).

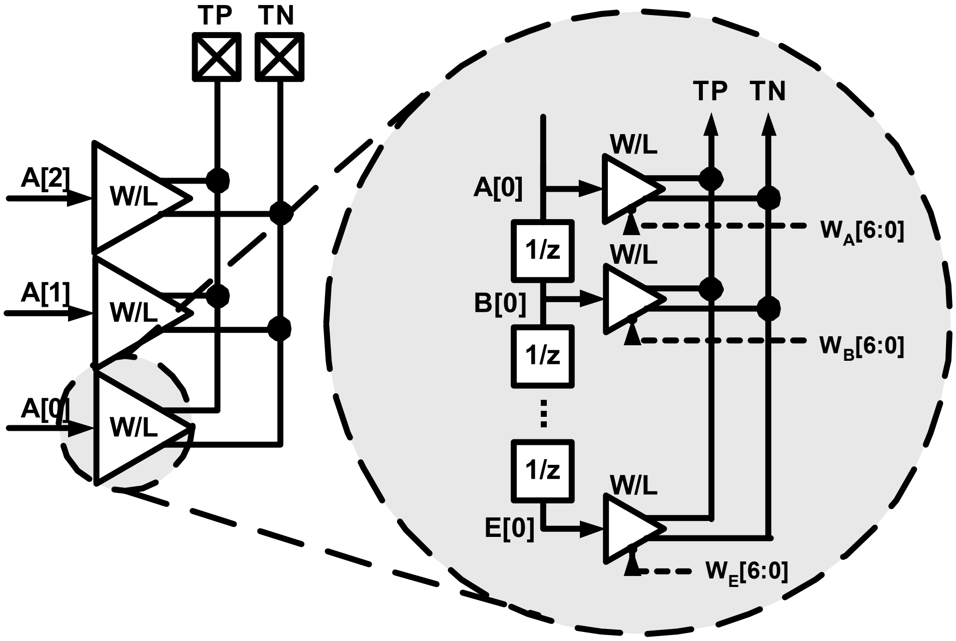


Figure 5.3: Straightforward implementation of transmit pre-emphasis via analog FIR filter, thermometer coded drivers for PAM2/PAM4. Simple PAM2/PAM4 transmitter has total gate size 3W/L, while with straightforward implementation of transmit pre-emphasis total gate size is 15W/L.

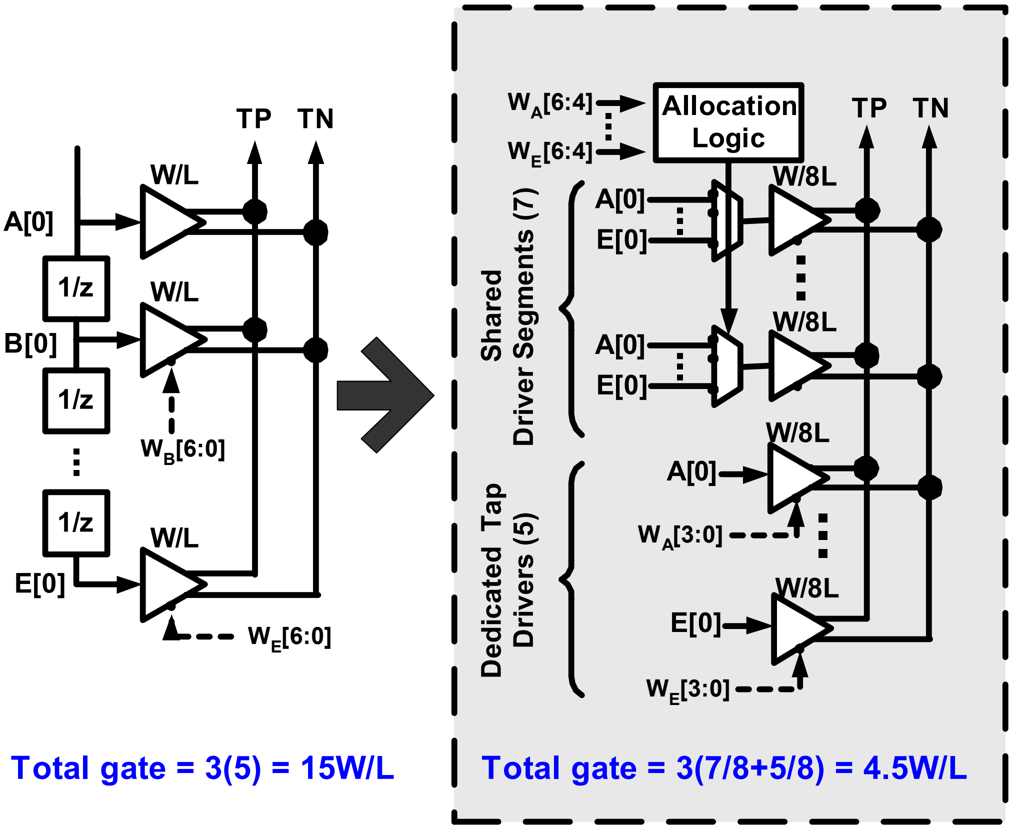
While straightforward to implement, such a driver has 5 times more output parasitic capacitance than if implemented as a RAM DAC. We saw in Chapter 2 that this capacitance forms a parasitic low-pass filter[30] at the output of the transmitter and can significantly degrade the performance of the link.

In order to minimize this parasitic capacitance and preserve the programmability of the analog FIR filter, a tap-sharing transmitter concept was developed jointly with Fred Chen (who also did the design and implementation of this transmitter [39, 81]). Figure 5.4 compares the straightforward and tap-sharing implementation.

The tap-shared architecture in Figure 5.4b leverages the fact that the transmitter is peak-power constrained due to output differential pair saturation margin. In a straightforward 5-tap implementation, this means that only 1/5th of the transmitter (or

[30] Output capacitance and 25 Ω impedance from the parallel connection of transmitter termination and output transmission line form a parasitic low-pass filter.

total gate width equal to the single-tap transmitter) will be active at a given time. The other 4/5ths of transistors will be off, contributing only to the parasitic capacitance.



(a) (b)

Figure 5.4: Implementation of shared transmit pre-emphasis filter, (a) straightforward implementation, (b) shared driver

Rather than keeping this device overhead, we divide a single transmitter into segments that can be shared by any of the taps, like regular DACs. We need to be careful here since this limits the resolution of the output driver to be the inverse of the number of segments into which we split the transmitter. For example, for 16 segments, the transmitter would only have a resolution of 4 bits. In order to use this as a pre-emphasis filter, we would need to add a five-tap 4-bit digital finite-impulse response (FIR) filter. As we discussed earlier, this would consume an unacceptable amount of power.

Instead of just segmenting the driver and allowing each tap to access any segment, the equalizer is partitioned into a shared section and a dedicated section. The shared section consists of seven large sub-drivers, each with current strength of 1/8th of the total allowed current. Each shared sub-driver can select from any of the five equalization tap streams A–E. The dedicated portion consists of five binary weighted drivers, one for each equalization tap, and each capable of driving up to 1/8th of the total allowed current. This combination of shared and dedicated drivers allows each equalization tap to have the same current range, for example 127 and resolution 1, of a non-equalizing 7-bit transmitter with only 50% additional parasitic overhead.

In addition to tap-sharing to minimize the overhead, this transmitter is capable of driving both PAM2 and PAM4 symbols with minimum increase in complexity. By encoding the PAM4 symbols using the Gray code, we can easily switch over from PAM4 to PAM2 mode by just injecting zeros into the LSB transmitter pipe. This coding scheme is shown in Figure 5.5. In addition to the Gray code, the transmitter also performs the Gray code to thermometer code conversion for better linearity, as was indicated in Figure 5.3.

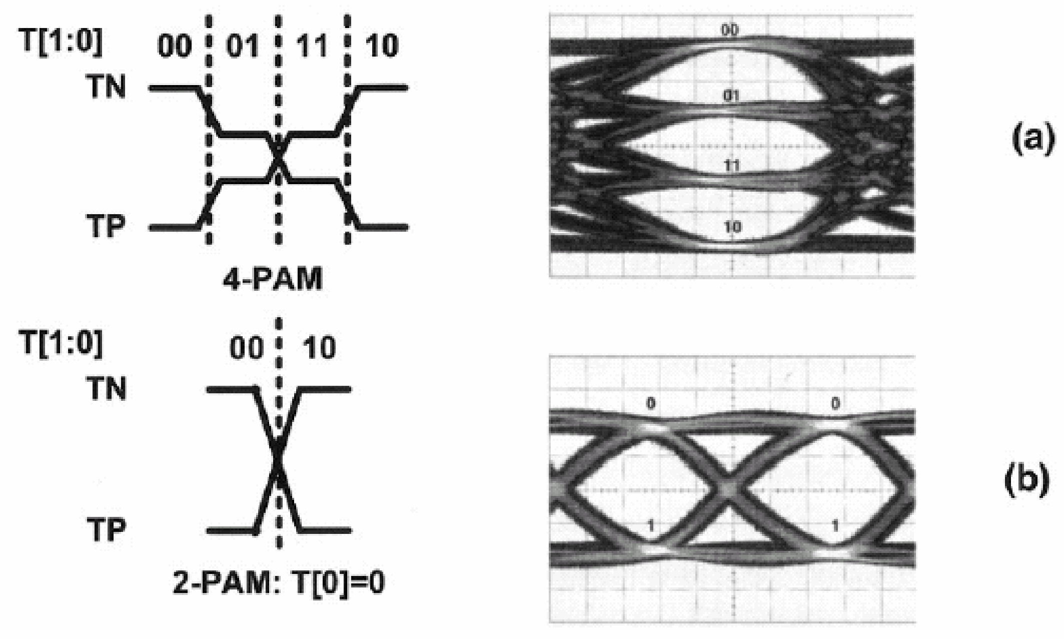


Figure 5.5: Gray coded levels provide compatibility between (a) PAM4 and (b) PAM2 signaling. In PAM2 mode, LSB=0 and data is only streamed into the MSB, [39].

In the next section, we focus on the receiver side and follow a similar principle of supporting dual PAM4/PAM2 operation by using reconfigurable architectures with little hardware overhead.

5.1.2 Receiver Front-End

Earlier link designs, with no equalization, and newer designs with transmit pre-emphasis use just data slicers in the receiver. Recently, some forms of analog (with inductive peaking [26] or capacitive tail degeneration [89]) and linear discrete time [108] receive equalization were introduced. Due to complexity reasons stated earlier, links use just the number of comparators necessary to make decisions on the data, so the resolution of this flash ADC is determined by the number of signal levels.

In PAM4 mode, our receiver has three data slicers per clock phase, with thresholds positioned between four distinct signal levels, as shown in Figure 5.6. The slicing thresholds are controlled by an 8-bit static DAC with a step size of about 2 mV. As shown in Figure 5.7, each of the slicers consists of a pre-amp and a comparator stage.

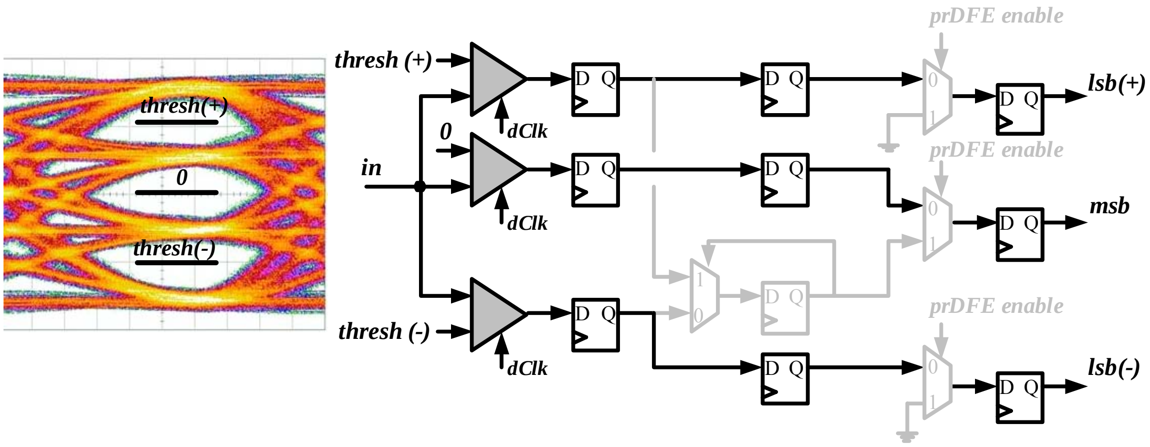


Figure 5.6: Receiver in PAM4 mode, three data slicers are needed to make a decision on four possible signal levels [81].

In differential slicers, it is very hard to introduce an additional threshold port and change the switching point of the slicer independently from the common-mode of the input signal. The threshold port in the pre-amp in Figure 5.7 is implemented through a programmable current DAC which changes the bias current of the two differential pairs.

These differential pairs have shifted switching points due to the imbalance in the transistor sizes. Increasing the current I_{thresh} steers more current into the differential pair on the left, which has a positive switching threshold[31], and blends that extreme positive switching point with the extremely negative switching point of the other differential pair. This blending results in very precise threshold control of the pre-amp that is to the first order independent of the common-mode voltage of the input signal. This scheme was designed by Bruno Garlepp, who extended the work of Casper [118], and also designed the rest of the receiver front-end. The second part of the slicer is a standard regenerative comparator, consisting of a sense-amplifier [122] and a balanced dynamic S-R latch.

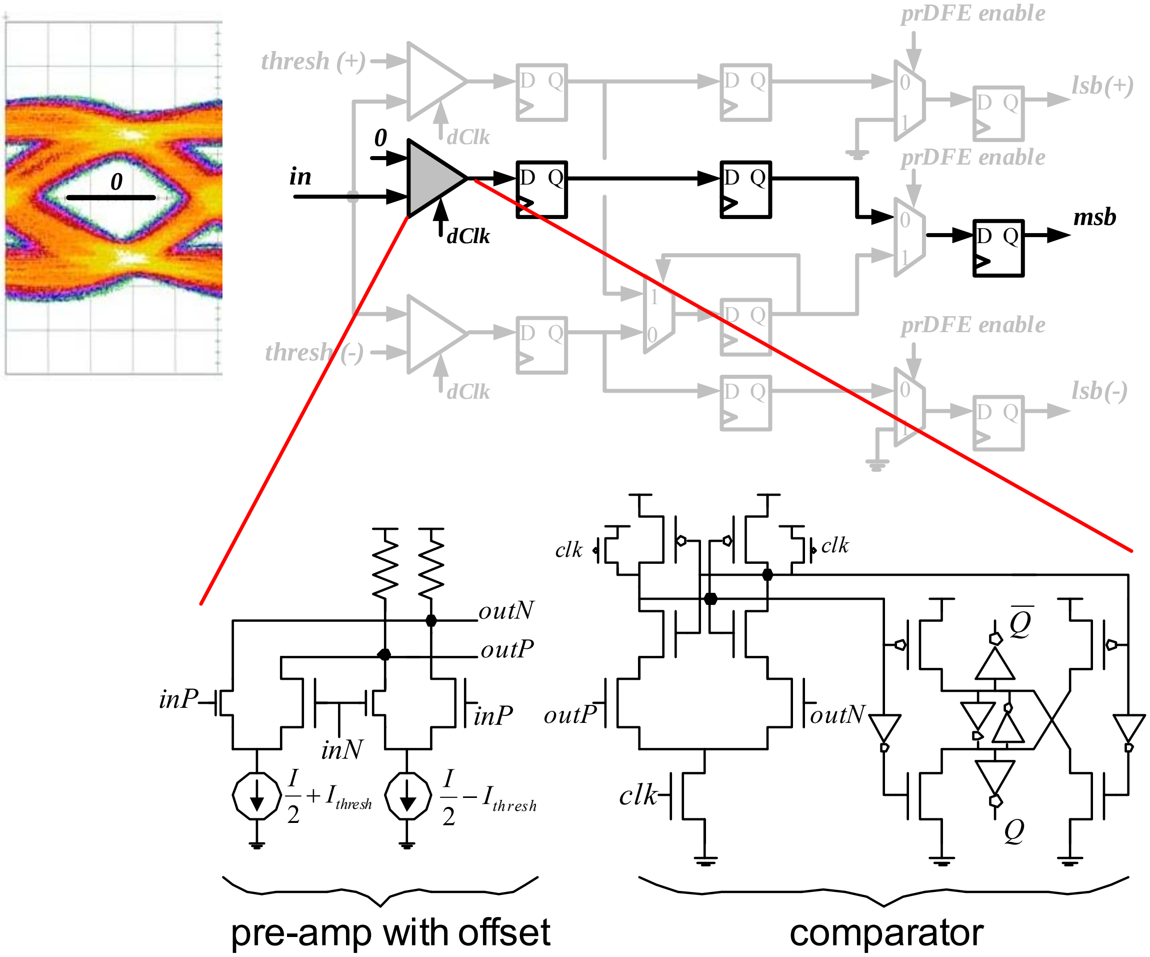


Figure 5.7: Receiver in standard PAM2 mode, with data slicer detail. Each data slicer consists of pre-amp with tunable switching threshold and a comparator based on a regenerative sense-amplifier with dynamic balanced S-R latch.

[31] Input *inP* drives a smaller input transistor that needs a larger overdrive voltage to switch the same current as the transistor driven by *inN*. Hence a differential pair where the transistor driven by *inP* is smaller than that driven by *inN* has a positive switching point.

As we have seen in Figure 5.7, in standard PAM2 mode, the link uses the middle slicer with zero threshold. In order to improve the performance of the link in PAM2 mode, we can use the other two slicers instead, with thresholds offset by the magnitude of the first post-cursor ISI tap, as shown in Figure 5.8. We can then choose the output of each of the samplers dynamically, by adding the multiplexer loop. In this way, we have embedded the one-tap loop-unrolled DFE for PAM2, in a PAM4 receiver with very little hardware overhead.

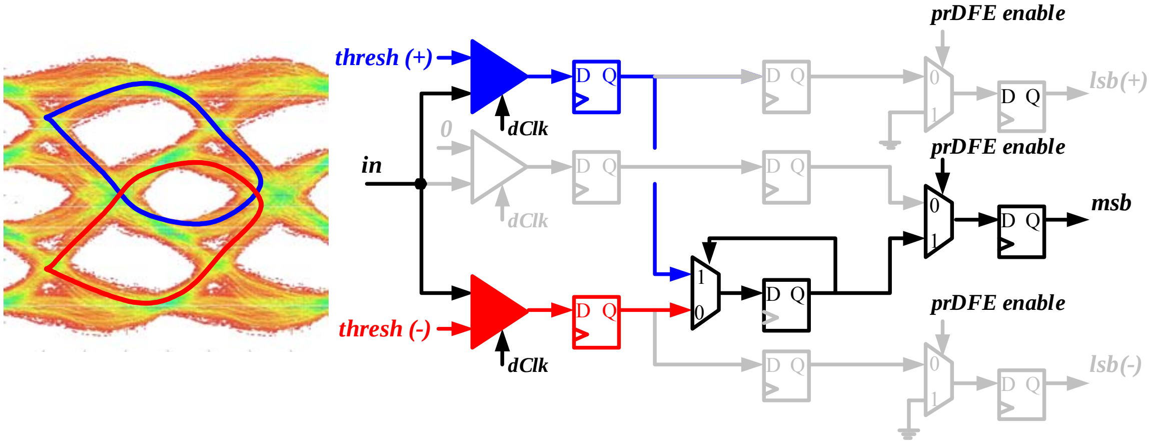


Figure 5.8: Integration of PAM2 partial response DFE receiver with loop unrolling into PAM4 receiver by re-use of PAM4 *lsb* slicers.

Each of the two slicers now sees either a positive or negative conditioned eye, centered around *thresh(+)* or *thresh(-)*. The PAM2 signal at the receiver is a partial response signal (since we tolerate one tap of ISI), and is in fact a multi-level signal. We can further leverage these multi-level properties and re-use the PAM4 CDR logic with minimum overhead, as we will explain later in the chapter. We will also describe later in the chapter how to extend the adaptive algorithm from Chapter 3 to adapt the transmit pre-emphasis and reflection cancellation equalizer to the partial response signal used in one-tap DFE with loop unrolling. Before we do that, let us give some details about the implementation of the feedback reflection canceller and additional hardware for link adaptation and calibration.

5.1.3 Feedback Equalization

As discussed in Section 3.2.1 , we use a current-mode analog FIR filter as a feedback filter, taking advantage of the linearity and high bandwidth of the transmission line termination at the input to the receiver and launching the feedback waveform directly into the transmission line termination. By adding and subtracting currents directly at the input pads, as shown in Figure 5.9, we get the additional benefit of canceling all the secondary reflections that would otherwise occur since a portion of the incoming signal is reflected off the parasitic capacitance of the receiver termination. An additional benefit is that we can simply re-use the transmit pre-emphasis filter and scale it down to minimize the added parasitic capacitance, since the magnitude of correction from reflections is smaller than that required for ISI. Our design reuses the transmit filter design, scaled to a 1/5th of the transmitter size.

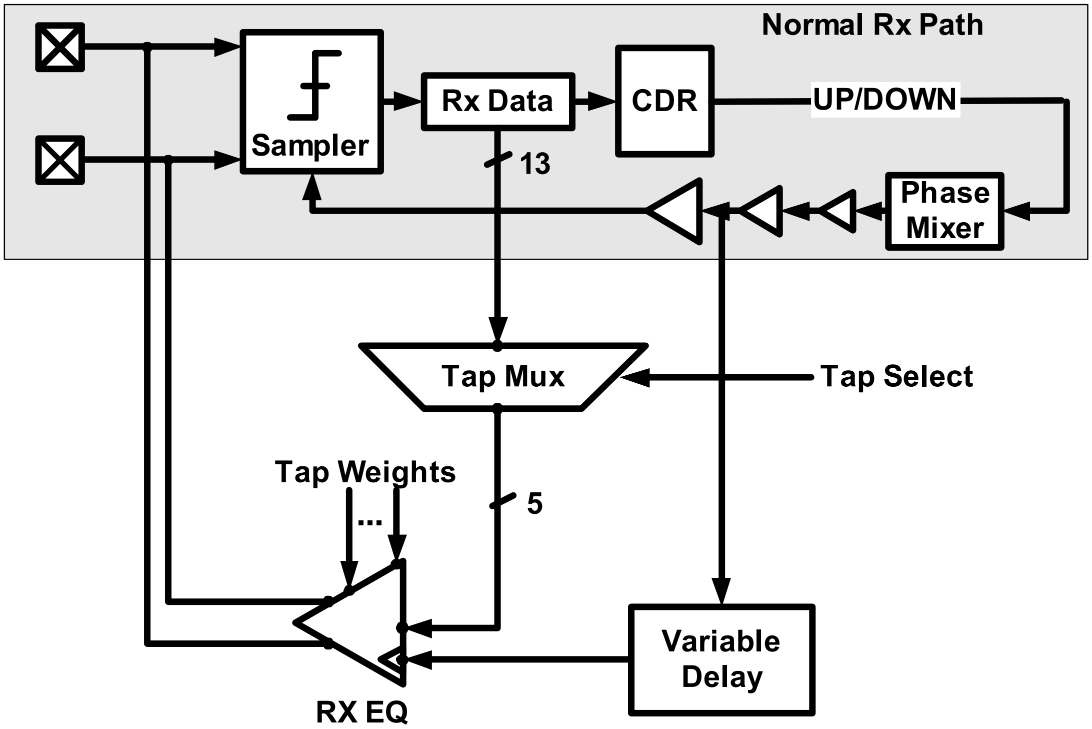


Figure 5.9: Receiver feedback equalizer with adjustable timing to compensate output driver clock-to-output delay. Designed and implemented by Fred Chen [39].

One difficulty with this type of receive equalizer is the timing alignment of the equalizer outputs to the incoming receive data, as the equalizer has a clock-to-output delay which varies over process, voltage, and temperature, and must be compensated for in order to make the most out of the equalizer's current drive. This delay is compensated by adding a limited-range variable delay element in the equalizer clock path. The delay element is adjusted by a training sequence where the receive equalizer sends a 0101 (clock) pattern which is received by the data path. During training, the clock data recovery (CDR) outputs are used to adjust the variable delay element while the normal receive phase value is kept fixed.

As reflections vary in both location and intensity between channels, in our first implementation [39] the receive feedback equalizer was designed to be very flexible, allowing for selection of any five taps within a window of 5–17 symbols after the main received symbol. The selection of the position is based on the magnitude of the reflections at each sample point, which can be obtained either from the received pulse response by using the adaptive sampler as a sampling scope or by recording the adapted tap magnitudes at each of twelve possible locations. In this way, the tap select multiplexer and tap weights are separately configured and optimized for each backplane channel.

In our recent design [81], we needed more reflection cancellation taps, to further improve the performance of the system. Since the tap selection multiplexer becomes very cumbersome for any to any tap selection, we decided to use a block of 10 taps, 6-15 symbols in PAM4 mode, and extend the range to 20 taps in PAM2 mode, by reusing the unused *lsb* pipe. In this way, the feedback equalizer covers the same time window in both the PAM2 and PAM4 cases.[32]

By analyzing the reflection patterns in Figure 2.9, we were able to position this feedback cancellation window around the first group of reflections so that most of the taps in the window are highly utilized. In this way, all the taps can be adapted by directly applying the adaptive algorithm in Chapter 3. Next we describe the additional hardware in the receiver front-end that enables the link calibration and adaptation.

5.1.4 Hardware for Link Adaptation and Calibration

As we mentioned earlier, we add one data sampler to the receiver, in order to provide the error information to the adaptive loop.

[32] This is important since the positions of the reflections are only set by their travel times and positions of discontinuities and they are independent of the signaling or modulation rate.

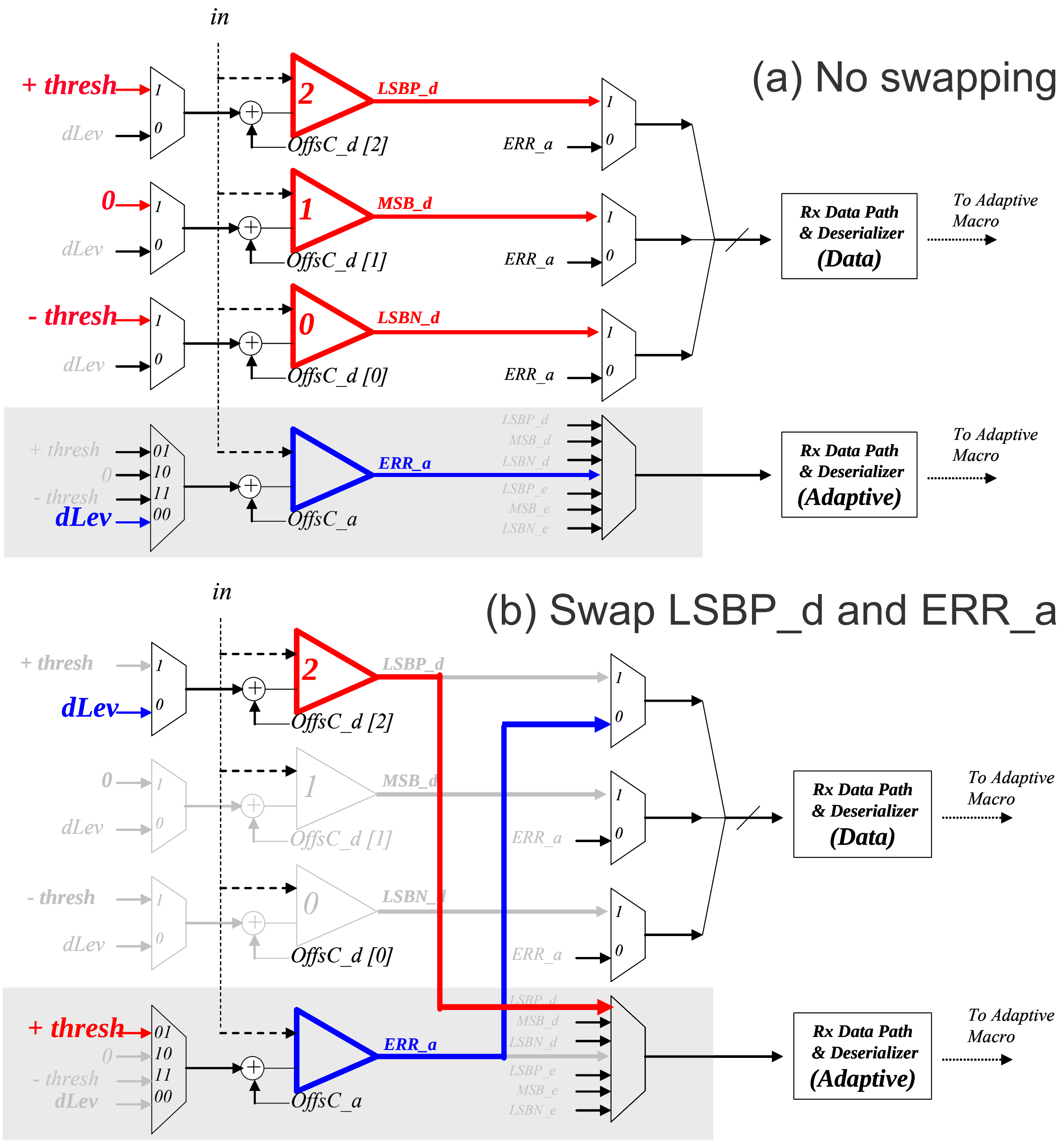


Figure 5.10: Sampler swapping capability of the receiver front-end, (a) No swapping enabled (each sampler drives its dedicated pipe), (b) Sampler that gives LSBP_d is taken off line and feeds the adaptive (error) pipe, while the adaptive sampler's output ERR_a feeds the LSBP data pipe.

In Figure 5.10, this adaptive sampler is shown in shaded area with the output ERR_a. The other samplers on the plot are data samplers with outputs LSBP_d, MSB_d and LSBN_d that correspond to the positive *lsb*, *msb* and negative *lsb* bits from the Gray coded PAM4 scheme in Figure 5.5. The pre-amp of each sampler has a threshold that is determined by a dedicated 5bit offset DAC and either a common 8-bit threshold value,

±*thresh*, or 9-bit adaptive reference level, *dLev*. Since the pre-amp switching point is determined by the bias current, all the offset, *thresh* and *dLev* DACs are implemented as current-mode and easy to multiplex. The multiplexers at the samplers' outputs select whether the output goes to the data pipe or the adaptive (error) pipe.

With this double multiplexing arrangement, the adaptive sampler can take the role of any other sampler[33] while that other sampler is taken off-line and calibrated. In Figure 5.10b, the adaptive sampler takes the role of the LSBP_d sampler. Now, the LSBP_d sampler has a threshold set to *dLev* for calibration and its output feeds the adaptive (error) pipe, while the adaptive sampler's threshold is set by *+thresh* in order to slice the incoming signal and feed the ERR_a output into the positive *lsb* data pipe.

5.2 Adaptive Equalization

Using only one adaptive sampler may seem to be insufficient from the perspective of the adaptive algorithm that we presented in Chapter 3. However, rather than having 4 error samplers, one for each level in PAM4, as proposed by Stonick *et al* [107], we use only one adaptive sampler and perform updates only when data is received that corresponds to the signal level at which the adaptive sampler is located. By doing this, we trade-off convergence time for receiver simplicity since convergence is not a problem with multi-Gb/s data rates and slow channel changes [36].

This revised tap update may be formally written using an indicator function I_{LMS}

$$w_{n+1}^{k} = w_{n}^{k} + I_{LMS,n} \cdot \Delta_{w} sign(d_{n-k}) sign(e_{n}) \tag{5.1}$$

$$I_{LMS,n} = \begin{Bmatrix} 1, & d_n = d_{\text{target}} \\ 0, & otherwise \end{Bmatrix}$$

[33] The adaptive sampler can swap with both data and edge samplers. In Figure 5.10 we omit the edge samplers for clarity, but their outputs are visible as the inputs to the multiplexer at the output of the adaptive sampler.

which is equal to one when the received symbol is equal to the target symbol of the signal level at which the adaptive sampler is located. The reference level loop can be updated as well using the same indicator function I_{LMS}:

$$dLev_{n+1} = dLev_n - I_{LMS,n} \cdot \Delta_{dLev} sign(e_n) \tag{5.2}$$

Now that we have enhanced the adaptive algorithm to work with only one sampler, we can extend it in order to adapt the transmit pre-emphasis and feedback equalizers when the link is set to PAM2 mode with one-tap loop-unrolled DFE. Our goal is to adaptively find the transmit pre-emphasis and feedback equalizer settings as well as the threshold of the two receiver slicers, which corresponds to the magnitude of the first post-cursor ISI tap, as in Figure 5.8.

Instead of forming the indicator function and filtering the error signal and loop updates (for both *dLev* and equalizer taps) with bit values that form the current received symbol, we can apply data filtering with the current and past bit in order to lock the *dLev* to one of the four signal levels (±1±α), present in a one tap binary DFE system, see Figure 3.8. This filter is very similar to data filtering for PAM4 equalization. A similar algorithm, but without data-based update filtering, was proposed for one-tap DFE by Winters and Kasturia [123] and incurs significant sampler overhead.

Using just one adaptive sampler and data based update filtering we estimate the size of the trailing ISI in an iterative manner. In the first phase, loop updates are filtered by the $(d_{n-1},d_n)=(1,1)$ criterion to lock *dLev* to the 1+α level, and in the second phase, updates are filtered by $(d_{n-1},d_n)=(0,1)$ to lock to the 1-α level. During these two phases, the equalizer only compensates for the error caused by ISI taps other than the first trailing tap, as shown in Figure 5.11.

In order to find the required threshold value, the first step is shown in Figure 5.11a, where the reference loop locks *dLev* to the (1,1) level, and in Figure 5.11b, where *dLev* locks to the (0,1) level. Sampler thresholds are then offset by the extracted magnitude of the trailing ISI, 0.5*(*dLev*(1,1)-*dLev*(0,1)).

Equalization and locking phases one and two are interleaved such that the optimal value of trailing ISI is found at the point when all other ISI has been minimized by the transmit equalizer and long-latency feedback equalizer (reflection canceller). This is necessary since the absolute magnitudes of the main and trailing ISI tap change due to rescaling which maintains the peak power constraint in the transmitter. The final sampler threshold is extracted at the end, when equalizer taps have converged, as illustrated in Figure 5.11c,d.

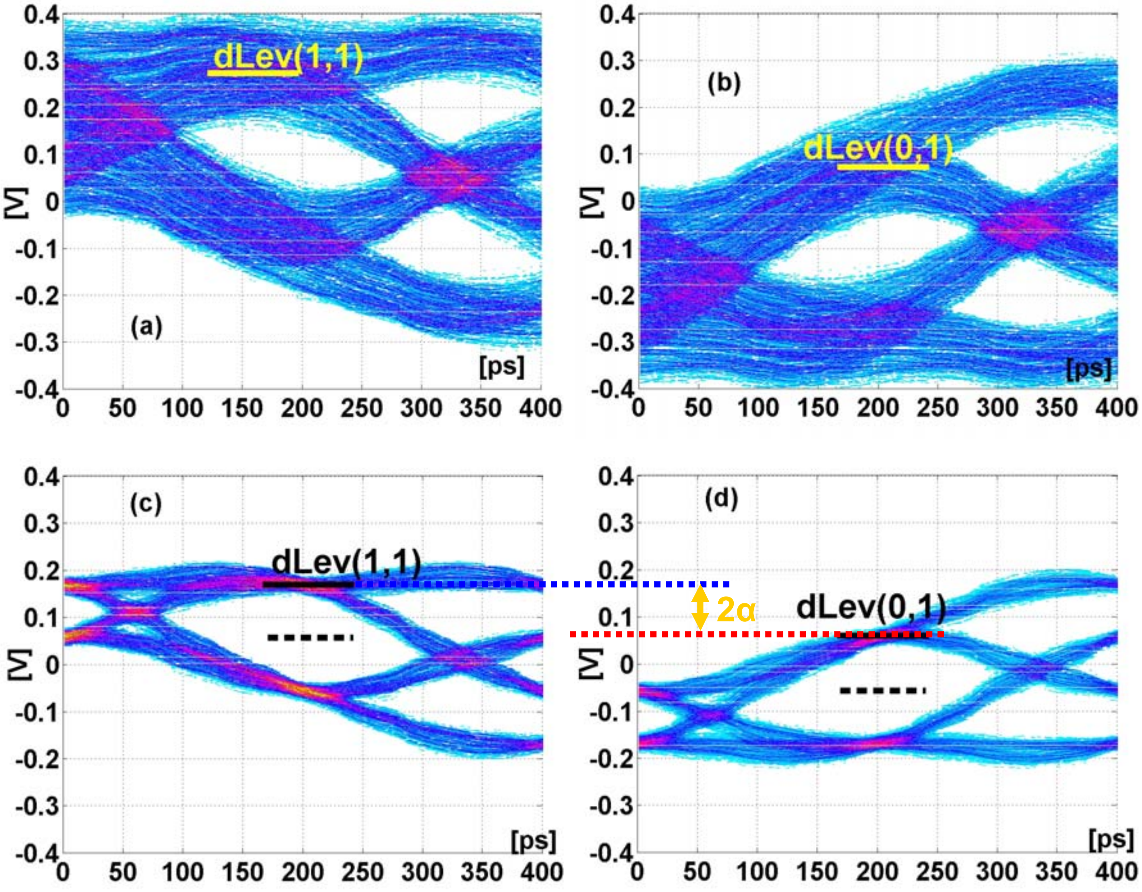


Figure 5.11: Joint equalization and extraction of the trailing tap magnitude. Plots are based on simulation using the measured pulse response, Figure 5.16a, obtained with the adaptive sampler; symbol time is 200 ps. a) Locking of *dLev* to (1,1) level – eye as seen by the upper sampler in Figure 5.8, b) Locking to (0,1) level – eye as seen by the lower sampler in Figure 5.8, c) Final locking point of *dLev* to (1,1) level after equalization, d) Final locking point of *dLev* to (0,1) level after equalization. Sampler thresholds are offset by the extracted final magnitude of the trailing ISI 0.5*(*dLev*(1,1)-*dLev*(0,1)) (dashed line).

In addition to the adaptive equalization of the loop-unrolled partial-response signal, synchronizing the receiver with such a multi-level signal is challenge that must be met. In the next section we first describe the CDR algorithms for classical multi-level signaling, like PAM4, and then we show how to reconfigure this CDR for partial-response one-tap loop-unrolled systems.

5.3 CDR Techniques for Multi-Level Modulation / Loop-Unrolled DFE

In a link with dual-mode PAM2/PAM4 operation, we need to design a flexible CDR that uses the optimal transitions available for clock recovery in either PAM2 or PAM4 mode. The complete set of PAM4 transitions, shown in Figure 5.12, consists of three minor transitions (smallest change in voltage level possible), one major transition (largest change possible), and two intermediate transitions for a total of six different transition types.

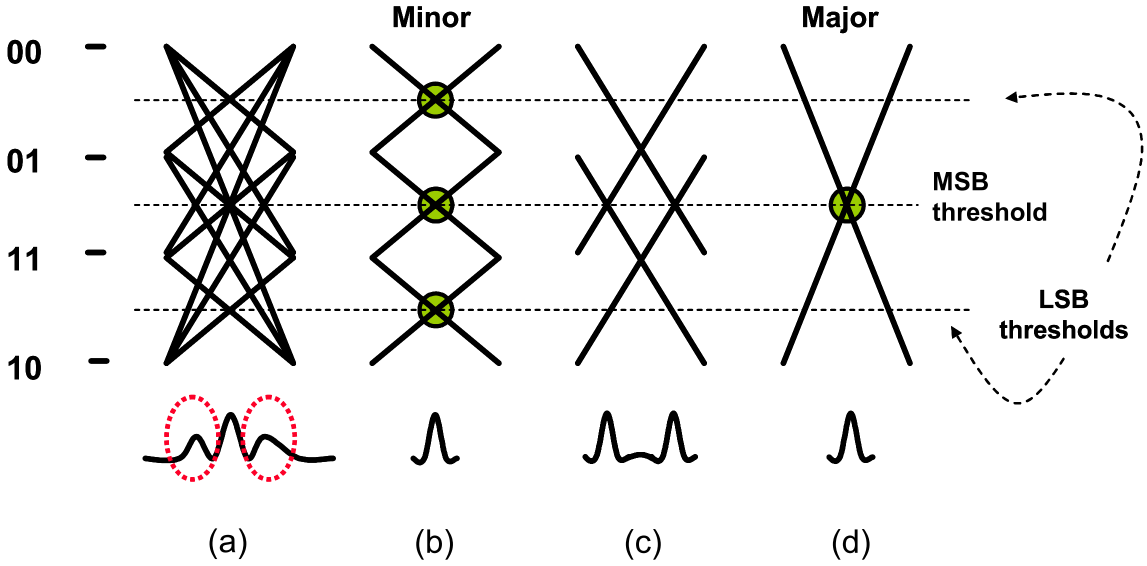


Figure 5.12: Optimal PAM4 and PAM2 transitions for two times oversampled CDR. (a) All possible transitions, (b) minor transitions, (c) simultaneous LSB/MSB transitions, and (d) the major transitions. Group (c) has undesirable timing distributions at the LSB slicer thresholds and its timing is ignored in PAM4 mode [39].

If a conventional two-times oversampled zero-crossing CDR [64] is used to recover the clock on an uncoded PAM4 signal, the CDR loop can have very large dither because the distribution of edges is multi-modal at the zero crossing (i.e. MSB sampler threshold). There are three distinct zero-crossing regions, as shown in Figure 5.12a. Similarly, the offset LSB sampler thresholds also contain three distinct crossing regions. Such distributions can cause increase in CDR dither jitter, or worse, static phase offsets, if the data pattern exhibits a predominance of one transition type over another.

In this design, implemented by Jason Wei, the optimal transitions, those in Figure 5.12b and d, are used for clock recovery depending on the link mode. In PAM2 mode, the MSB major transition, Figure 5.12d, is used. In PAM4 mode, the minor transitions of either the MSB or LSB, Figure 5.12b, are also included, while the transitions with skewed crossings, Figure 5.12c, are ignored. By eliminating the transitions that cause bi-modal threshold crossing distributions we minimize both clock jitter and phase offset.

The CDR logic that eliminates the unwanted transitions edge exclusion is shown in Figure 5.13. All MSB and LSB edge and data samplers are used. Adequate density of optimal transitions is assured through means of scrambling or coding [128].

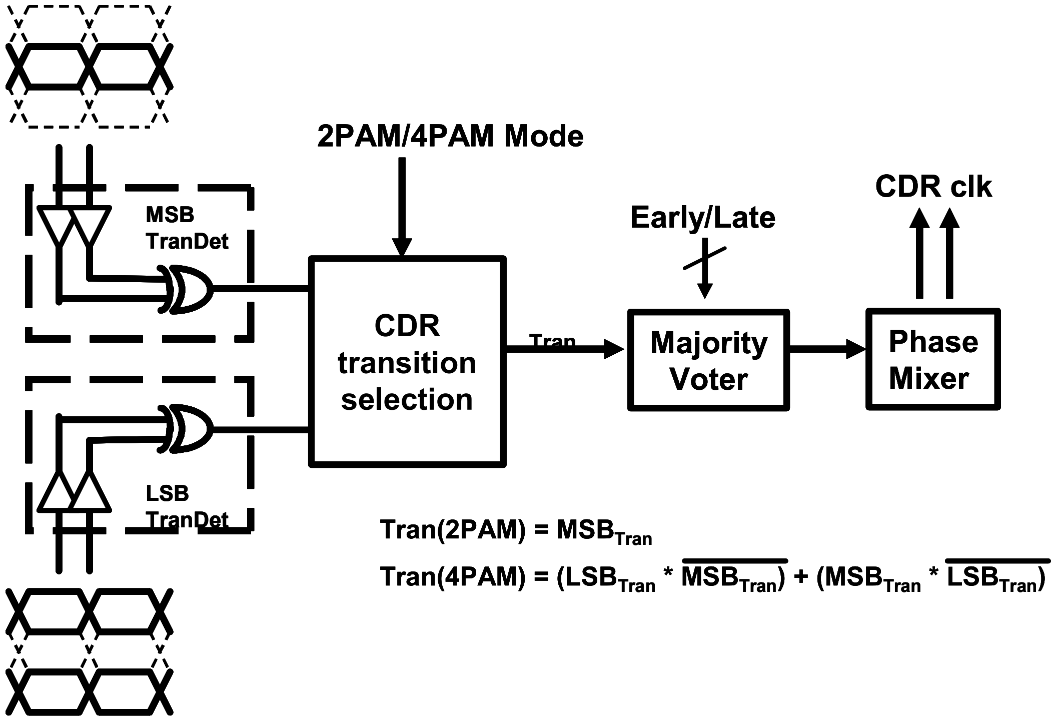

2PAM/4PAM Mode
CDR clk
Early/Late
MSB
TranDet
CDR
transition
selection
Tran
Majority
Voter
Phase
Mixer
LSB
TranDet
Tran(2PAM) = MSB_{Tran}
Tran(4PAM) = $(LSB_{Tran} * \overline{MSB_{Tran}}) + (MSB_{Tran} * \overline{LSB_{Tran}})$

Figure 5.13: Dual-mode PAM2/PAM4 CDR logic eliminates transitions with poor timing information [39].

In the most recent version of our link [81], we extend the transition filtering CDR for PAM4, to PAM2 mode with loop-unrolled one-tap DFE. By noting that in PAM4 mode transition filtering is done on two-bit symbols, we can reconfigure the PAM4 CDR to partial-response CDR by filtering the transitions based on the pairs of current and preceding bits.

We have already seen in Figure 5.8 and Figure 5.11 that the presence of the trailing tap of ISI causes the received signal to have four levels, similar to PAM4 albeit non-uniformly separated. The transitions from one level to another are guided by the values of the future, current and immediately preceding data bits, as shown in Figure 5.14. These transitions form two distinct modes or principal zero crossings, denoted by arrows in Figure 5.14. In order to avoid this bi-modal behavior, we could filter out one type of transition by filtering the edge crossings in the clock and data recovery (CDR) block.

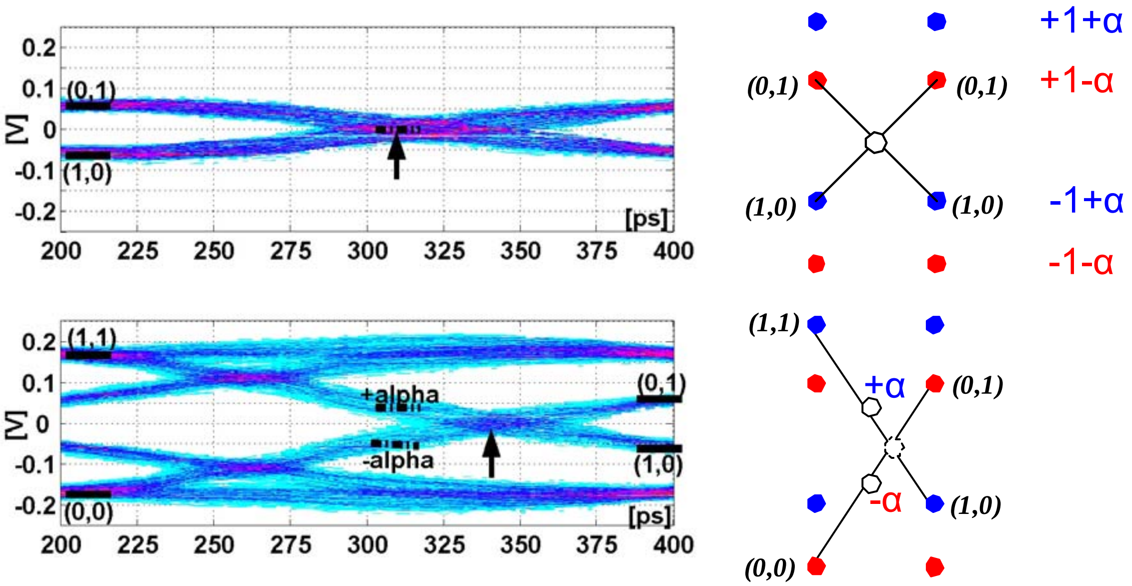
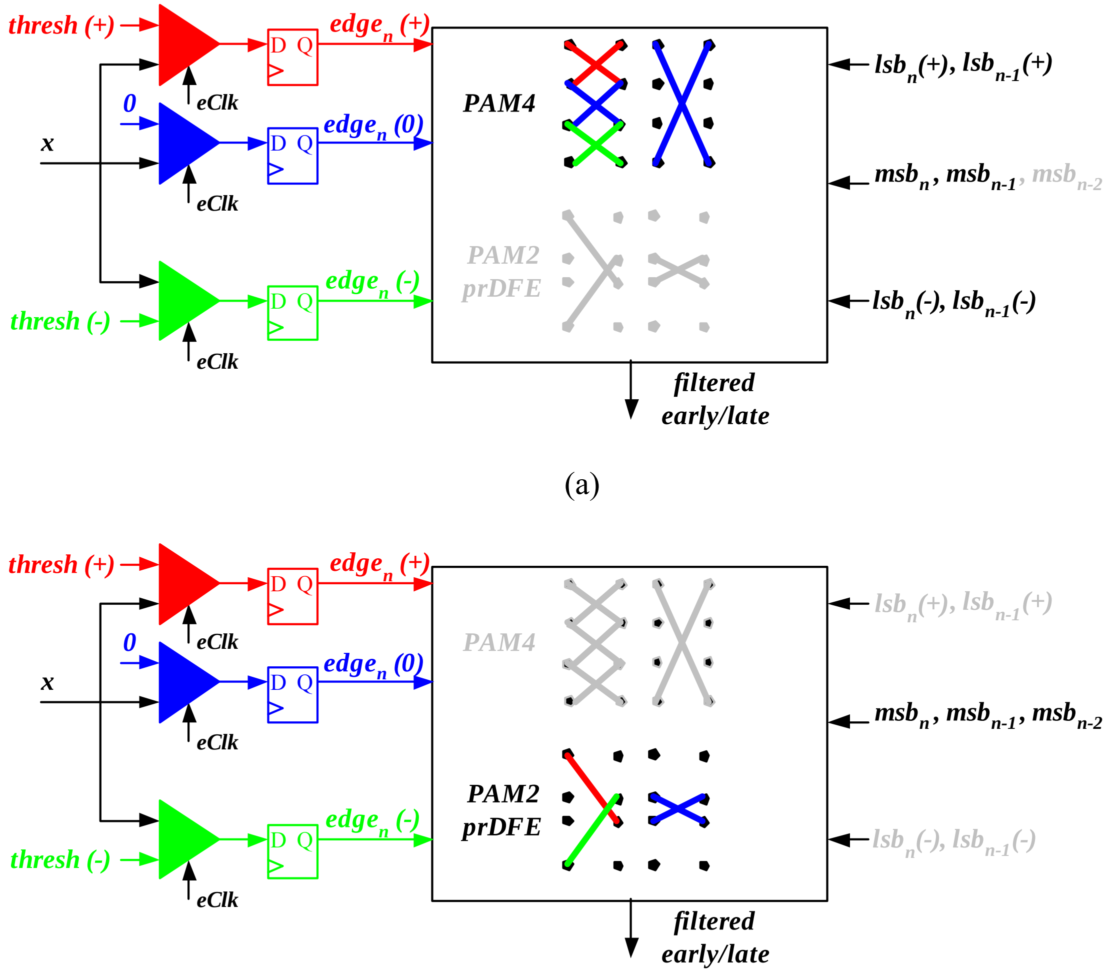


Figure 5.14: Bi-modal transitions in 1+αD channel: first mode (1,1)→(1,0) and (0,0)→(0,1), second mode (0,1)→(1,0) and (1,0)→(0,1).

Since edge filtering decreases the probability of CDR updates and puts additional constraints on the first-order CDR loops in plesiochronous systems, as we said earlier, we use *lsb* edge samplers to make use of minor transitions in PAM4 mode. In the partial response mode of operation, we make use of these *lsb* edge samplers, offsetting them by the magnitude of the trailing ISI and align the edge slicing timing as shown by the left arrow and three dotted levels in Figure 5.14. In this way, no transitions are lost and the rate of CDR updates is maximized.

The clock and data recovery front-end remains the same as in the PAM4 case. Three edge samplers provide tentative early/late information, while the transition filtering section either uses $lsb_n(+/-)$, msb_n, $lsb_{n-1}(+/-)$ and msb_{n-1} data in PAM4 mode, as shown in Figure 5.15a, or msb_n, msb_{n-1} and msb_{n-2} in PAM2 partial response mode, as shown in Figure 5.15b.



(a)

(b)

Figure 5.15: Generation of early/late updates in 2x oversampling CDR loop, in: a) PAM4 mode and b) PAM2 mode with partial response DFE.

5.4 Experimental results

In order to understand the quality of the channel, we can use the adaptive sampler to scan out the pulse response of the whole channel as seen by the receiver, including any bandwidth limitations in the receiver. In Figure 5.16a, we show the pulse response before equalization with 200 ps symbol time samples denoted as dots. We see that both pre and post-cursor dispersion ISI components are very big, but we also have some reflections at longer latencies.

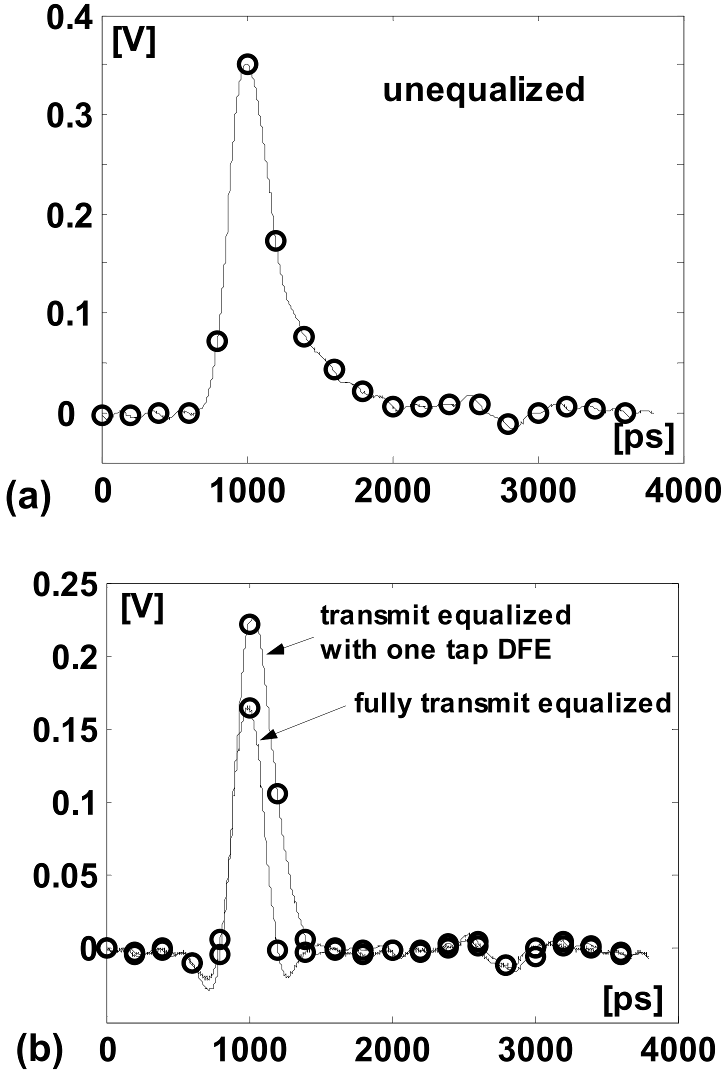


Figure 5.16: *E-scope*, [109], of the pulse response: a) unequalized, b) Comparison of transmit equalized pulse for one-tap DFE and a fully transmit equalized pulse. Dots indicate symbol spaced sample points (symbol time is 200 ps).

In Figure 5.16 we compare the pulse response equalized with transmit pre-emphasis with the pulse response equalized by jointly using transmit pre-emphasis and one-tap DFE with loop-unrolling. As we mentioned earlier, transmit pre-emphasis only creates the partial-response signal by leaving out the first post-cursor tap, which is then compensated for at the receiver. Not using pre-emphasis to cancel the first post-cursor tap saves some signal energy at lower frequencies and results in a higher main sample, as shown in Figure 5.16. The pulse response equalized for one tap DFE at 5 Gb/s, 26” FR4 channel, is about 60 mV (40%) larger than the fully equalized pulse, due to the peak output power constraint in the transmitter.

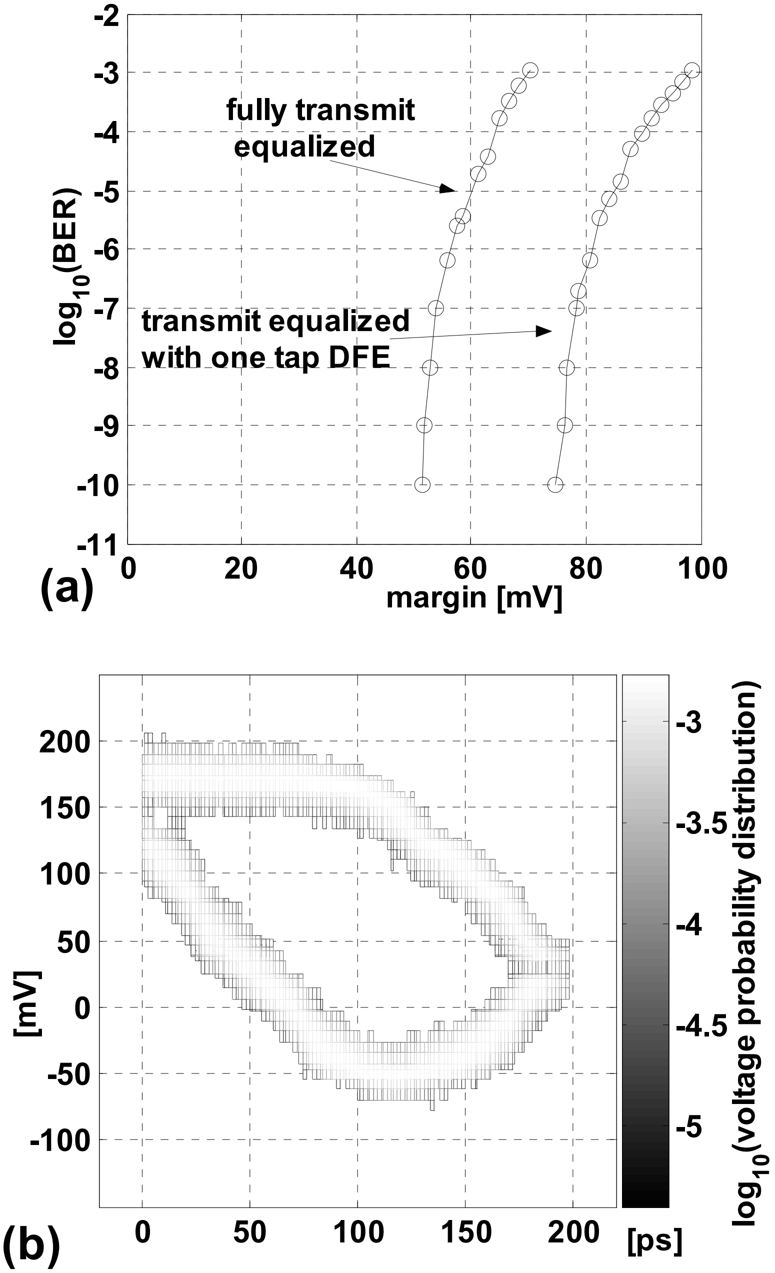


Figure 5.17: Comparison of bit error rate (BER) vs. receiver noise margin for fully transmit-equalized link and transmit-equalized with one-tap DFE, b) Statistical shmoo of the eye diagram as presented to the positive *lsb* sampler for one tap DFE.

In Figure 5.17a we compare the BER versus signal margin for the two equalization schemes. The transmit pre-emphasis with one-tap DFE has around 25 mV better voltage margin at BER of 10^{-10}. The steep slope of the BER vs. noise margin curves suggests that random noise components (jitter and voltage thermal noise) are relatively small and that ISI is still the most dominant error term.

It is interesting to observe the shape of the equalized eye in a loop-unrolled DFE scheme, Figure 5.17b[34]. While not as symmetric as a fully equalized PAM2 eye, it is actually slightly more robust to jitter. Measured peak-to-peak jitter from the 2.5 GHz recovered clock shows that CDR dither decreases from 14 ps to 5 ps when one-tap DFE is used instead of full transmit pre-emphasis. The tri-modal edge distribution present in fully-transmit-equalized PAM2 is partially avoided in the one-tap DFE scheme since the first post tap of the transmit pre-emphasis is not significantly engaged. Inherent PLL jitter was 26 ps peak-to-peak.

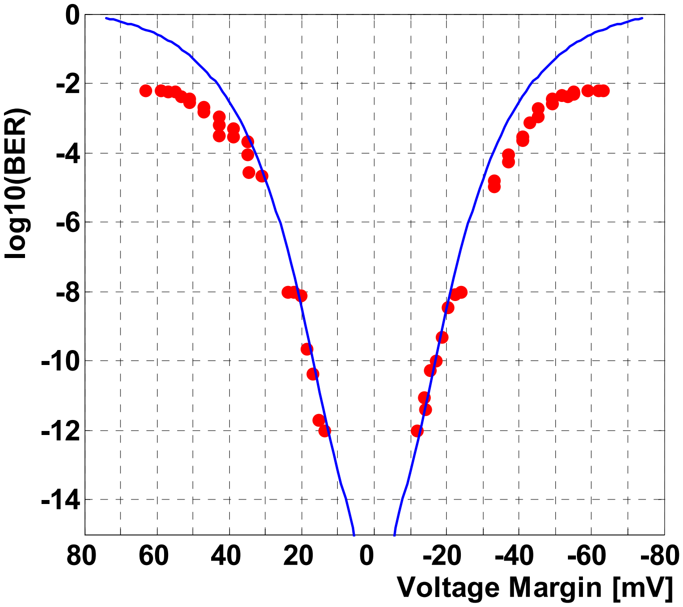


Figure 5.18: Comparison of the measured and predicted link BER as a function of voltage margin. The predicted BER was calculated by using a link system model and analysis in Chapter 4.

We used these jitter and CDR dither numbers to predict the BER of the link as a function of voltage margin by using link models and analysis in Chapter 4. Then, we

[34] This is only a half of the whole eye, i.e. a positive conditioned eye centered around +*thresh*, refering to Figure 5.8.

compare these predictions with the BER measured for different voltage margins and confirm the accuracy of the link models that we created. Figure 5.20 shows the measured and modeled bit-error rate for a given voltage margin of the link. The red dots indicate measured data and blue curve is the link model. The link was set to PAM4 mode with 3 taps of transmit equalization active, with a data rate of 5 Gb/s.

We see that there is a good agreement at low BERs, which is desirable, since links normally operate in this region. The reason why we have small disagreement for low BERs is due to fundamental limitation of our measurement procedure – we could not read off errors from time intervals that are shorter than those corresponding to BERs of about 10^{-3}, hence we see the saturation of measured results at BER of 10^{-2}.

In addition to these results, we also measured the convergence properties of the implemented dual-loop adaptive algorithm. Figure 5.19a shows the learning curve of the reference level (*dLev*) loop, and Figure 5.19b shows learning curves of the four transmit pre-emphasis taps.

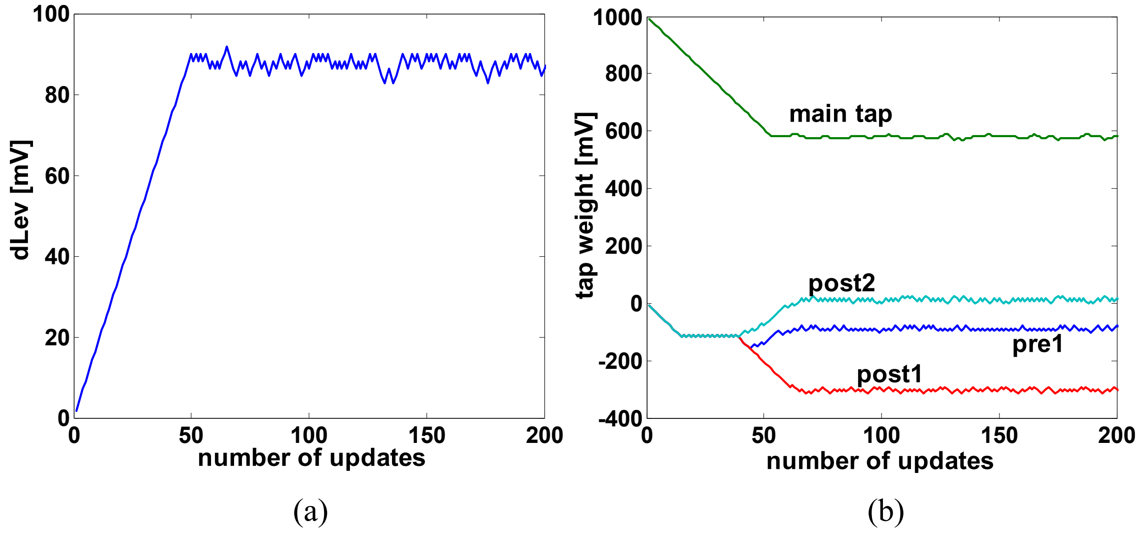
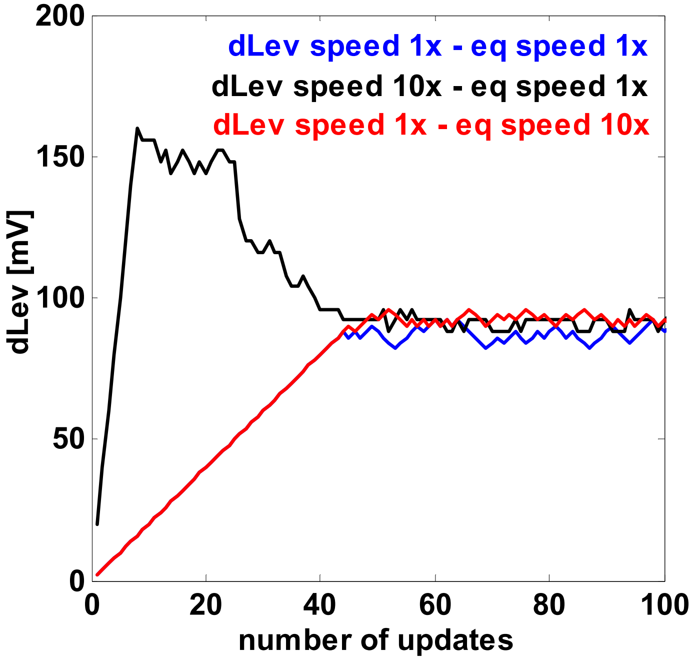
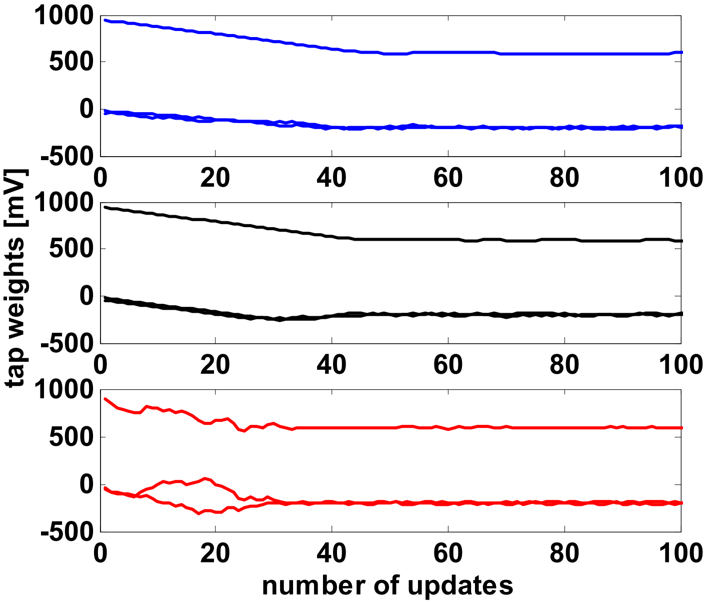


Figure 5.19: Measured learning curves of the dual-loop adaptive algorithm: (a) Reference level *dLev* loop, (b) Four taps of transmit pre-emphasis.

The measurements in Figure 5.20 show that the equalization algorithm is stable for a relatively wide range of update speeds of one loop with respect to another. In Figure 5.20a, we show the learning curves of the reference level loop, while changing the speed of that loop to be an order of magnitude slower or faster than the pre-emphasis loop.



(a)



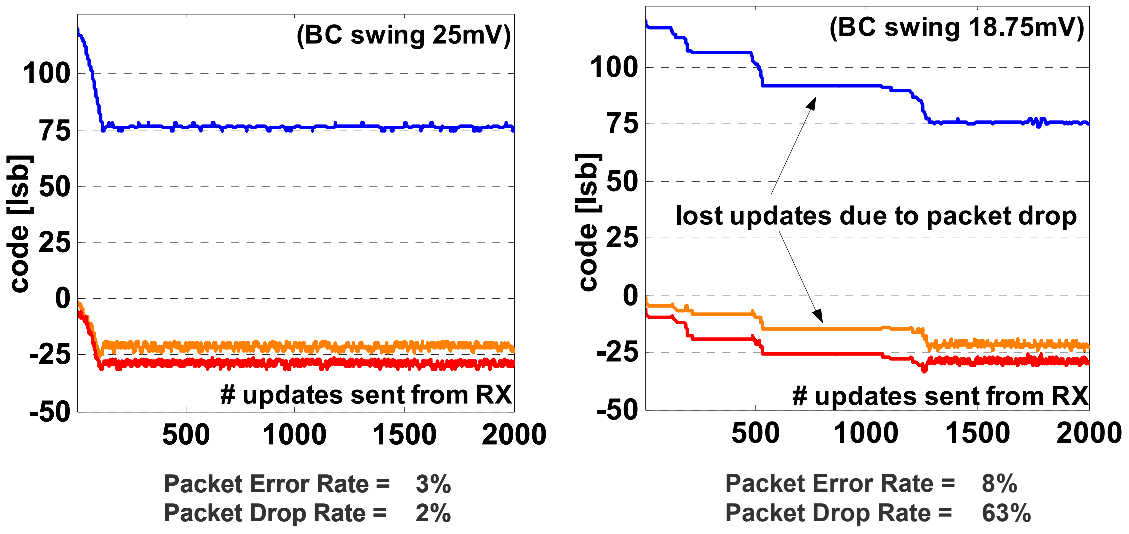
(b)

Figure 5.20: Dual-loop adaptive learning curves for different speeds of the *dLev* and equalizer tap loops, PAM2 at 5 Gb/s over 20” FR4. Updates are filtered on received data being high (since the adaptive sampler tracks the positive signal level) and then block averaged by 127 to smooth the sign-sign gradient estimate.

With a faster reference level loop, we see that the loop catches up with the signal faster, but then descends following the attenuation of the signal due to convergence in the transmit pre-emphasis loop. The corresponding convergence of the transmit pre-emphasis is shown in Figure 5.20b.

To complete the link analysis, we also experimented with the full link adaptation using a back-channel [58]. The common-mode back-channel swing can be adjusted to provide a tradeoff between forward-channel signal integrity and back-channel noise immunity. In this back-channel design, a packet is dropped when an error in synchronization or a parity error is detected.

While packet error and drop rates are important for reliable transmission of various configuration and link information data, it is interesting to note that theoretically for adaptive algorithm convergence it is only necessary to have an error rate of less than 50%. The packet drop rate only influences the total adaptation time.



(a) (b)

Figure 5.21: Transmit pre-emphasis tap convergence: a) Back-channel swing of 25 mV with packet drop rate of 2% and packet error rate of 3%, b) Back-channel swing of 18.75 mV with packet error rate of 8% and packet drop rate of 63%.

For example, the packet drop rate decreases nearly two orders of magnitude when back-channel swing is increased from 18.75 mV to 25 mV, so adaptation time improves rapidly. To illustrate this effect, we plot the convergence of transmit pre-emphasis taps as a function of the number of sent updates for these two backchannel swing amplitudes in Figure 5.21.

These plots indicate that efficient adaptation is still possible even at very low back-channel amplitude. Very reliable back-channel communication can be achieved for back-channel swings of 50 mV and higher.

Even at the largest available back-channel swing of 100 mV, there was no measurable impact on the noise margin of the forward channel at a target BER of 10^{-12}. This indicates that the noise induced by the back-channel on the forward channel is less than 2 mV – the minimum change in margin detectable by our measurement setup.

5.5 Summary

In this chapter we focused on the practical issues in building a next generation high-speed link that attacks the band-limited communication problem from a system perspective. The link was built according to the specifications obtained from the system level optimization and analysis, where the design space was explored to find the link architecture that can achieve the required 5-12 Gb/s performance with the least complexity.

The link features dual-mode PAM2/PAM4 operation in order to achieve the desired performance over a wide range of backplane channels. Because of this variety of channel characteristics and slowly time-varying channel properties, the link was designed as a self-sufficient, fully adaptive system.

A modification of the adaptive equalization algorithm from Chapter 3 that includes data filtered updates enables algorithm implementation by adding just one additional sampler to the receiver's front-end. With a similar data filtering technique this modified algorithm is extended to work with the partial-response signals needed for one-tap loop-unrolled DFE.

Owing to the similarities between the PAM4 and one-tap partial-response binary signal, a new receiver architecture adds one-tap loop-unrolled DFE for PAM2 to the existing PAM4 receiver with minimum hardware overhead. This hardware reconfiguration approach is also used in the CDR loop that is re-designed to work with both PAM4 and PAM2 partial-response signals.

A very important feature is the ability of the link to self-configure/adapt itself to the channel without requiring the external communication channel, due to the large number of links within a router. For that reason, as we mentioned earlier, the adaptive link also contains a common-mode back-channel which uses the same pair of wires as the differential forward link, to communicate the update and control information back to the transmitter. The back-channel is designed to be low-speed and with more relaxed BER requirements than the forward channel in order to minimize its impact on the performance of the forward link and have negligible power and complexity overhead. We have seen that adaptive equalization works without convergence issues for back-channel swings down to 25 mV. With swings of 50 mV, the back-channel proves to be relatively reliable even for control information, with packet error and drop rates of less than 10^{-3}. No significant impact on the forward-channel voltage margins was observed even for back-channel swings of 100 mV.

Experimental results show that the adaptive algorithm converges nicely for a wide range of update speeds for both the reference level and equalizer loops. They also show the performance advantage of joint transmit pre-emphasis and one-tap loop-unrolled DFE when compared to transmit pre-emphasis alone. We used the chip results to verify and calibrate the link system model.

All the techniques used in this chapter allowed the implementation of a next generation link that efficiently deals with band-limited backplane channels, enabling data rates between 5 and 12 Gb/s. This design also proved that our models are capturing the most important link impairments. As we have seen in Chapter 3, these models also indicate that these baseband data rates are still very far from the capacity of backplane channels. In the next section we will try to conclude this work by addressing the power and complexity issues in further scaling of link data rates.

Chapter 6

Conclusions

During the 1990's, link performance scaled dramatically by treating I/O wires as transmission lines, and using on-chip parallelism (multiplexing transmitters and de-multiplexing receivers) and PLLs to rapidly scale the bit rate CMOS circuits could support. We have scaled links so well that we are now running into the intrinsic frequency limitations of the wires. In this new regime of operation, current and future link design requires a strong coupling between high-speed circuit design, digital communications, and optimization, and while it leverages results from all of these areas, its constraints are sufficiently different that new solutions need to be created.

Equalized baseband signaling is the current approach to addressing this bandwidth limitation. Groups have suggested different approaches like PAM2 with DFE, multi-level PAM modulations with 2-3 bits per symbol, transmit pre-emphasis etc. While this is similar to the standard communication problem of signaling over band-limited channels, conventional communication solutions cannot meet the added constraints of low BER, peak power and high symbol rates. Modifying those approaches to our problem is critical to creating an effective optimization framework.

In these systems, the most important parameter to track is the effective noise and interference that the receiver sees. Since most of the noise and interference in this system are not unbounded Gaussian noise, it is critical to characterize them accurately by looking at both statistical distributions and correlation. This dissertation presented one such link system model that includes the effects of timing, voltage noise and interference including the impact of the CDR loop.

Fortunately, the resulting system model is convex, allowing us to estimate both the capacity limits and the practical data rate limits for a number of backplane channels using a convex optimization framework. While the effective bandwidth of these channels is limited to less than 10 GHz, the capacity and integer constellation data rates are relatively high (40-100 Gb/s). The problem lies in achieving these high data rates in a practical system.

In these practical systems, the power/complexity constraints limit our ability to cancel ISI, preventing us from increasing the data rate either by increasing the symbol rate or by using more bits/symbol (since the required SINR grows rapidly with larger signal constellations). While timing jitter and slicer sensitivity are not currently limiting the link performance, they are not far behind ISI as limiting error sources. Even if we could cancel all ISI, our data indicates that both timing jitter and slicer sensitivity would limit the data rate of baseband links to less than ¼ of capacity.

A big problem is that even to achieve these bounds, we can no longer rely on technology scaling to overcome the power/complexity constraints. Normally we could count on technology scaling allowing us to continue to increase link complexity and build better equalization techniques with each scaling step. While this is still true to some extent, since supply scaling will be limited in the future, the power per function will scale much more slowly than it did previously. This means that the complexity of the links will scale more slowly, and if we only extend the baseband link approach, link data rate scaling will slow down.

The key to avoiding this innovation slow-down will involve the application of other digital communication ideas to high-speed links. One way is to limit the ISI by partitioning the channel into different frequency bands.

In Section 6.1 we first address the energy efficiency of data transmission in baseband links, while in Section 6.2 we look at multi-tone signaling as a possible way to continue data rate scaling of links by decreasing this energy cost per bit.

6.1 Data Rate Scaling in High-Speed Links

Energy cost of data transmission is a very important metric in high-speed links due to potential integration of a large number of links onto chips with a limited total power budget, such as switch chips in routers or microprocessors with high-speed I/O. Since scaling of link data rates is directly dependent on our ability to scale the complexity of the links, in this section we will first show the details of the link energy consumption per function, and then try to establish some rules of thumb about the scaling of data rates in baseband links.

The energy cost of data transmission is usually measured in mW/(Gb/s). Figure 6.1 shows the energy cost for different link components that were taken from the design in [39,81].

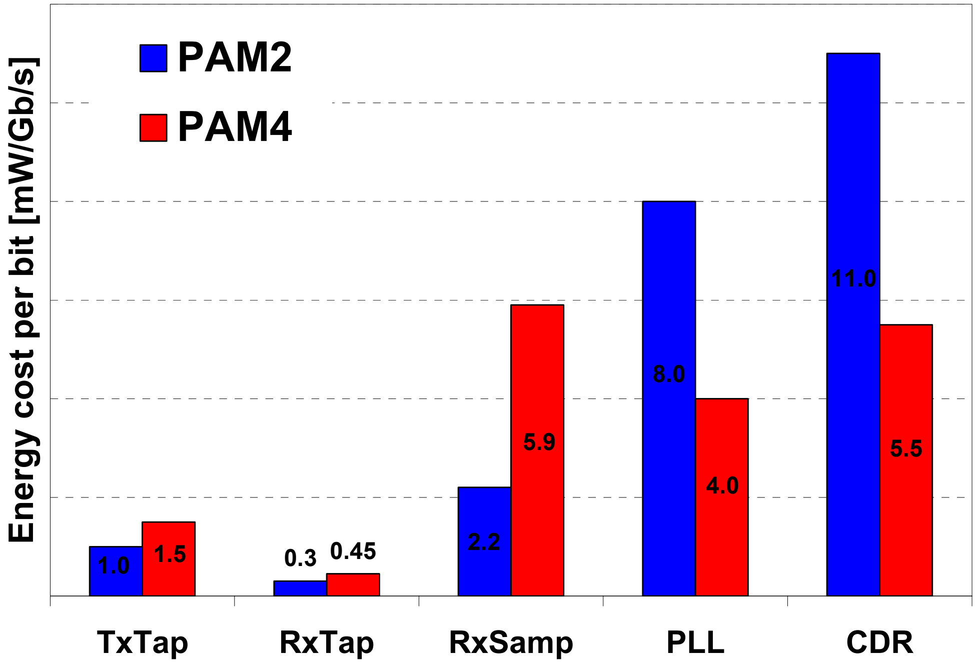


Figure 6.1: Energy cost of link components in mW/(Gb/s) for a 0.13 µm CMOS design running at 1 V. TxTap is cost per transmitter precoder tap, RxTap per feedback equalizer tap, RxSamp cost of sampling front-end, PLL cost of the phase-locked loop and CDR is the cost of the clock and data recovery loop.

A transmit pre-emphasis tap is more expensive than a feedback equalization tap due to the larger size of the transmitter devices required to drive the desired output power. At the receiver, the size of the feedback taps can be smaller since the channel already attenuates the received signal. It is also interesting to note in Figure 6.1 that the cost of the pre-emphasis and feedback equalizer taps increases with the number of levels of modulation (due to thermometer coding), while the cost of the supporting blocks like synchronization (phase-locked loops and clock and data recovery loops) drops due to lower symbol rate requirements for the same data rate. Another component of power dissipation is the transmitter output power component, which is in fact the signal power that the transmitter delivers to the line. With ±500 mV peak output swing in the transmitter, in a differential system the output power provided by a current-mode driver is fixed to 20 mW, regardless of data rate.

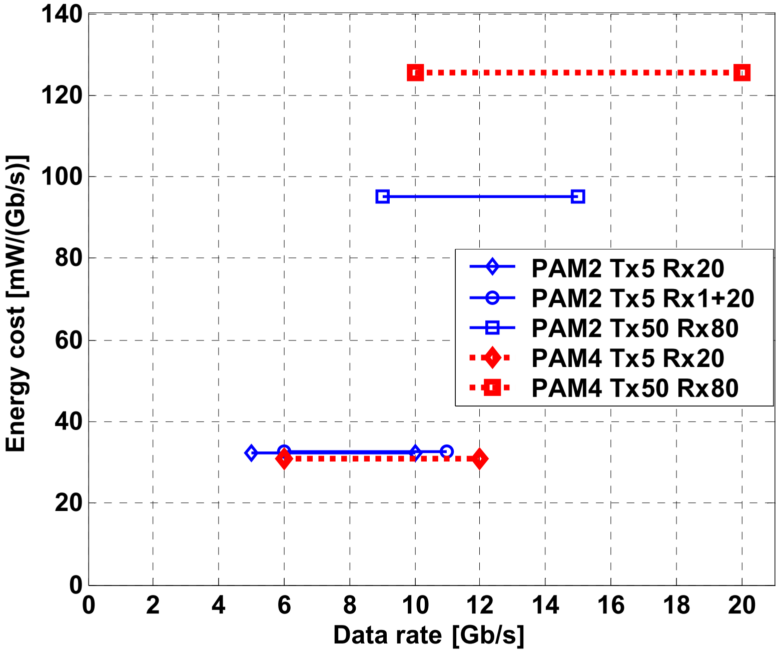


Figure 6.2: Energy cost of baseband architectures at different modulation levels. We start from implemented link architecture with PAM4 and 5 taps of transmit pre-emphasis and 10 taps of reflection cancellation in the receiver, and then extrapolate the power numbers to several different baseband architectures: PAM2 5 taps in transmitter, one-tap loop-unrolled DFE with twenty taps of reflection cancellation, PAM4 5 taps in transmitter and 20 taps of reflection cancellation in the receiver and also with 50 taps in the transmitter and 80 taps in the receiver.

Using the data from Figure 6.1, and achievable data rates for different architectures from Figure 4.20 and Figure 4.21, we plot in Figure 6.2 the energy-efficiency of different architectures vs. data rate, for PAM2 and PAM4 modulations. The data indicates that for architectures with large number of taps, multi-level techniques are less energy efficient, since multi-level taps are more costly (due to thermometer coding), while for architectures with a small number of taps, multi-level architectures are more efficient since they decrease the amount of energy that is consumed in the supporting part of the link (for synchronization and clock generation).

The curves in Figure 6.2 are given for current state-of-the-art 0.13 μm CMOS technology. Conventional CMOS technology scaling assumes cubic energy scaling (quadratic in supply voltage and linear in capacitance). However, supply voltage scaling is severely limited in future technologies by transistor leakage, hence energy will most likely scale linearly with feature size due only to capacitance scaling.

This scaling limitation means that complex high-speed links will be pushed further in the future depending on the scaling of the device feature size. For example, reading from Figure 6.2, the energy efficiency of the next generation links, which operate in the 5-12 Gb/s range, is about 30 mW/(Gb/s) in 0.13 μm CMOS technology. Moving the data rate up to 20 Gb/s in baseband links requires nearly fourfold increase in mW/(Gb/s). If voltage scaling stops it would require us to scale the technology to 30 nm to reach the same cost per bit as today’s 10 Gb/s links.

This means that baseband links will not be able to increase in complexity fast enough to keep up with desired increases in data rates. As energy is likely to stay a key constraint in the future, we need to design links using alternative digital communication techniques that are more energy-efficient in dealing with ISI in band-limited channels.

6.2 Future Work

From history lessons on modems and DSL, we know that multi-tone systems are the most efficient in mitigating the ISI. Our analysis of the capacity of backplane channels [124] indicated that usable signaling bandwidth in backplane links is up to 10 GHz with fairly wide (1-2 GHz) sub-channels on a variety of backplane channels. This fact, coupled with desire to avoid expensive DSL-like digital implementations, led us to consider the modification of one of the earliest multi-tone implementations [125], i.e. analog filter-bank multi-tone.

In this system, each baseband link is loaded with the maximum possible number of bits and upconverted to its corresponding passband using a local oscillator. By adapting our baseband link system models to this MIMO system, we were able to show in a preliminary analysis [124] that by using the enhanced analog filter-bank multi-tone system, it is possible to nearly double the data rates of today's baseband links without power penalty.

The biggest issue in this system is in balancing the effects of sub-channel ISI and inter-channel interference (ICI). This balancing involves complexity and performance tradeoff analysis between the hardware costs of transmit pre-emphasis, pulse shaping, and feedback equalization on one hand, and analog filters and RF components on the other. There are many possible configurations, and this direction in link design presents a very rich area for future research.

As we add more complexity to links to achieve higher data rates over the same bandlimited channels, scaling link performance becomes increasingly difficult, but it is not impossible. It is a perfect example of an area that will require combining sophisticated mathematical tools and analysis with a deep understanding of the capabilities of scaled CMOS circuits to create the needed innovative solutions. This thesis has taken the first step, by creating a model for baseband links, and showing how it can be optimally solved using convex optimization, and sub-optimally solved using a simple sign-sign LMS algorithm. Our challenge now is to extend these techniques to multi-tone systems to allow us to continue link data rate scaling.

Appendix A

Noise Correlations

In this section we briefly derive the autocorrelation functions for several error sources, such as residual ISI, equalizer quantization noise and error in equalizer settings from the channel estimation. Using the link model framework from Chapter 3, we first calculate the autocorrelation of the residual ISI after equalization, Equation (A.1).

$$R_{x_{ISI}}(m) = E_a \cdot \left[\underline{w}\,\mathbf{P}(\mathbf{I} - diag(\underline{1}_\Delta)) \quad \underline{0}_{1\times m}\right]\begin{bmatrix} \underline{0}_{m\times 1} \\ (\underline{w}\mathbf{P}(\mathbf{I} - diag(\underline{1}_\Delta)))^T \end{bmatrix} \tag{A.1}$$

Next we calculate expressions for equalizer quantization noise, assuming that the quantization error of each coefficient is uniformly distributed in the interval [-Δ_w/2, +Δ_w/2]. Since the quantization errors are independent and identically distributed, the autocorrelation of the quantization noise at the output of each of the equalizer taps is a delta pulse, as shown in Equation (A.2).

$$R_{qw}(m) = \frac{\Delta_w^2}{12}\delta(m) \tag{A.2}$$

After propagation through the channel, the autocorrelation of the received noise becomes as in Equation ((A.3))

$$R_{x_{qw}}(m) = L \cdot R_{qw}(0) \cdot E_a \cdot \left[\underline{p} \quad \underline{0}_{1\times m}\right] \begin{bmatrix} \underline{0}_{m\times 1} \\ \underline{p}^T \end{bmatrix} \tag{A.3}$$

where ***p*** is a sampled pulse response vector and ***L*** is the number of equalizer taps.

Similarly, if we assume that some proportional error may result from improperly set equalization coefficients in case there is an error in channel estimation, we can represent it as in Equation ((A.4)).

$$\tilde{w}_i = w_i \cdot (1 + \delta_i) \tag{A.4}$$

The resulting autocorrelation at the receiver is shown in Equation (A.5).

$$R_{x_{ew}}(m) = E_a \cdot \frac{\delta_w^2}{12} \cdot \underline{w}\underline{w}^T \left[\underline{p} \quad \underline{0}_{1\times m}\right] \begin{bmatrix} \underline{0}_{m\times 1} \\ \underline{p}^T \end{bmatrix} \tag{A.5}$$

With fixed and proportional noise from the transmit equalizer, represented in Equations (A.3) and (A.5) we can model the impact of any kind of quantization or estimation noise in the equalizer on the performance of the link.

Appendix B

Carrier Jitter (Phase Noise)

Carrier phase noise is present in a possible multi-tone implementation of a high-speed link and induces some crosstalk between the real and imaginary parts of the signal. We also use it in the capacity calculations in Chapter 3, as a phase noise of narrowly spaced tones in a capacity-achieving multi-tone scheme.

In narrow-bandwidth communication, the received signal can be represented as a function of the transmitted symbol, channel frequency response and carrier phase noise in transmitter and receiver

$$x = H(j\omega_c t)\, a_{TX} e^{j\varphi_{aTX}}\, e^{j(\theta_{noise}^{TX} - \theta_{noise}^{RX})} \tag{B.1}$$

where ω_c is the carrier frequency, $H(j\omega_c t)$ is the channel response at the carrier frequency, a_{TX} is the magnitude and φ_{aTX} the phase of the transmitted symbol, and $\theta^{TX}{}_{noise}$ and $\theta^{RX}{}_{noise}$ transmitter and receiver carrier phase noise, respectively.

The phase noise term in Equation (B.1) results in mixing of the real and imaginary parts of the signal, which causes signal proportional noise with autocorrelation:

$$R_{x^{pn}}(m) = E_a \left\| H(j\omega_c t) \right\|^2 \left(R_{\theta_{noise}^{TX}}(m) + R_{\theta_{noise}^{RX}}(m) \right) \tag{B.2}$$

where E_a is the average transmit alphabet energy and $R_{\theta}{}^{TX}{}_{noise}(m)$ and $R_{\theta}{}^{RX}{}_{noise}(m)$ are the autocorrelation functions of transmitter and receiver carrier phase noise.

Appendix C

General MIMO System Formulation

In this appendix we will describe a general formulation for a link system with transmit pre-emphasis. Although we will present here the derivation from our previous work [45], where this general formulation was used in the context of a time-multiplexed link (TDM), the same multiple-input multiple-output (MIMO) structure can be used for FEXT cancellation and frequency-multiplexed multi-tone links with transmit pre-coders (FDM) [124]. While system structures remain the same among these different cases, care has to be taken to properly identify all the elements of the channel matrix, depending on the type of the problem. In this section we will focus first on the channel matrix formulation for a TDM link system, but once the channel matrix is formulated, the pre-emphasis derivations are applicable to any kind of MIMO system.

Before we engage in a detailed system model, let us first describe the link [28] for which the model was created. As shown in Figure C.1, the transmitter consists of eight time-multiplexed 8-bit DACs, in groups of two, clocked from the transmitter PLL through the phase adjusters. The receiver consists of eight time-multiplexed 4-bit ADCs. The phase adjusters in the receiver synchronize each ADC (with a sampling rate of 1 GS/s) with one of the transmitting DACs, thus achieving an aggregate sampling rate of 8 GS/s.

As we have seen in Chapter 1, the fundamental reason for performing TDM is to avoid the on-chip clock frequency limit in current CMOS technology. However, this significantly increases the parasitic RC filtering at the transmitter output/receiver input, as we discussed in Chapter 2, thus reducing the useful bandwidth of the link. In link implementation [28], an attempt is made to "distribute" the parasitic capacitance by insertion of inductors between each pair of transmitters and receivers in order to form a lumped LC transmission line in a manner analogous to distributed amplification [126]. This form of analog equalization extends the useful bandwidth of the link, effectively without any noise enhancement penalty.

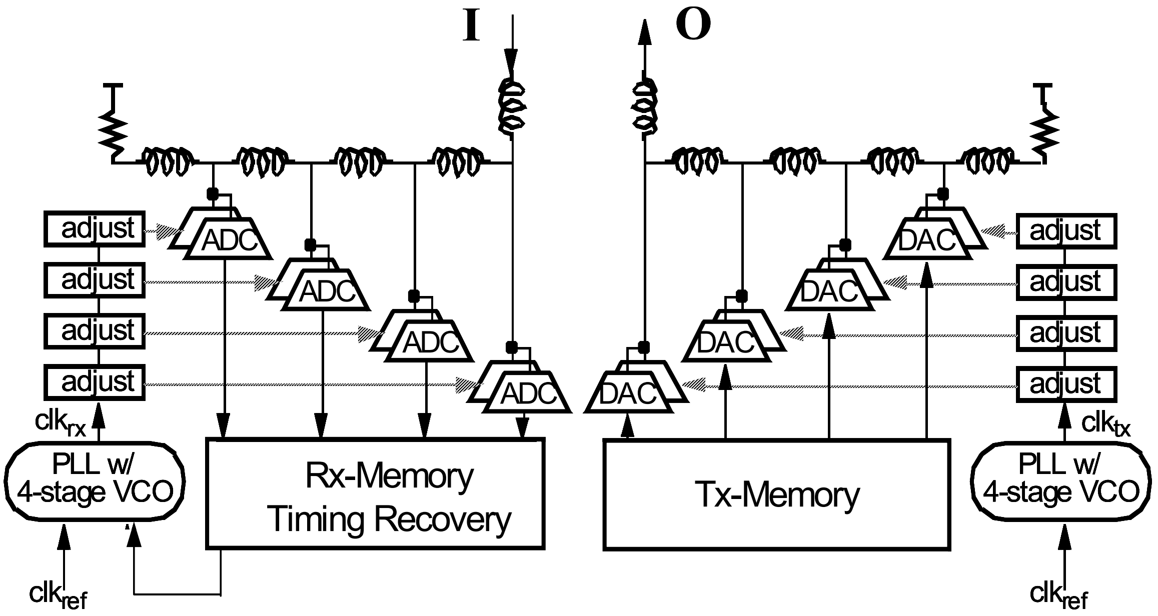


Figure C.1: Transceiver block diagram.

A comparison of the lower and upper bounds of the "distributed" vs. the "non-distributed" transmitter frequency response is given in Figure C.2. The data represents the FFT of the pulse response captured by a sampling oscilloscope after one meter of coaxial cable from point **O**, Figure C.1, in the experimental setup [28]. Large variations in the bondwire inductors result in significant differences between the TDM sub-channels. Instead of exhibiting the behavior of a lumped LC transmission line, the frequency response is dominated by second-order peaking. Evidently, the variation between transmitter responses is much smaller in the case without inductors than in the case with inductors. However, the useful bandwidth increase in the "distributed" case is apparent.

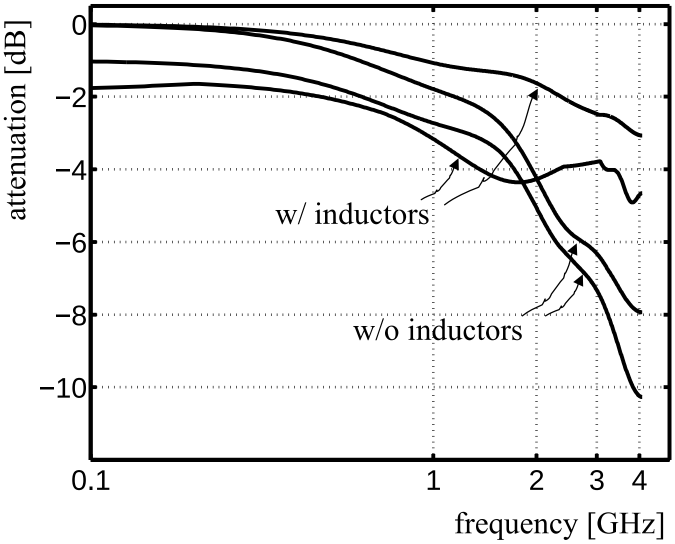


Figure C.2: Measured transmitter frequency response (envelopes of all 8 DACs) with and without inductors.

Having seen the physical example of a *distributed*-TDM link, we can now first formulate the channel model for such a system, and then find the closed form and adaptive solutions for the MIMO transmit pre-emphasis filter, using the peak-power constraint at the output of every transmitter, as in Chapter 3.

C.1 Mapping of the Distributed-TDM System to a MIMO System

In order to facilitate the derivation of a general solution, the above presented system is mapped to an *NxN* MIMO system, as illustrated in Figure C.3. Note that *N* is the number of TDM sub-channels that equals the number of ADCs and DACs. As shown earlier, the TDM sub-channels obtained from *distributed-TDM* may have considerably different characteristics. A TDM block consists of symbols $x_1(n)$ ... $x_N(n)$ that are transmitted sequentially in time via transmitters $T_1(z)$ ... $T_N(z)$. Receivers $R_1(z)$ ... $R_N(z)$ output the samples of the TDM block $y_1(n)$ ... $y_N(n)$.

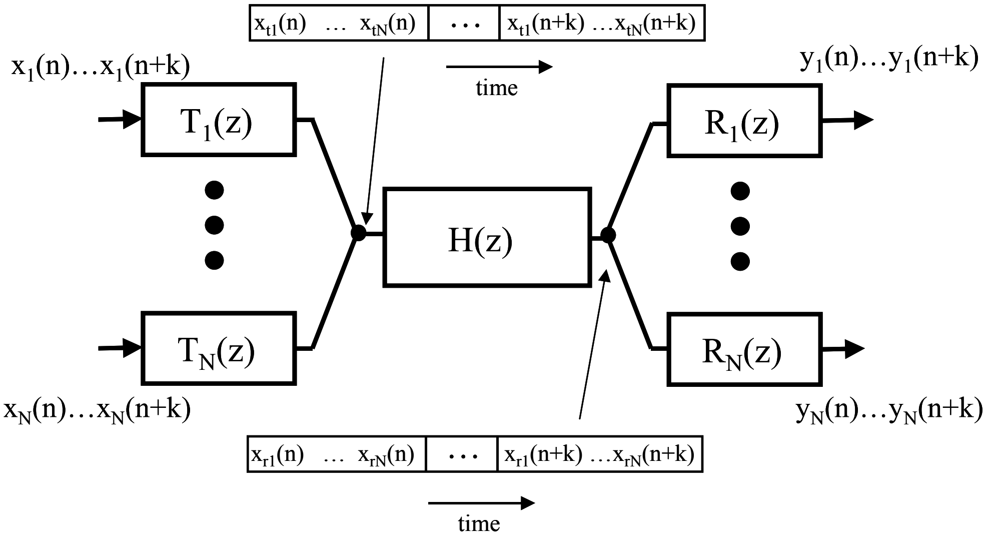


Figure C.3: NxN *distributed-TDM* system.

To illustrate the concept, a 2x2 case is shown in Figure C.4. The transmitter-channel-receiver MIMO model is characterized by the filters $\underline{\mathbf{p}}_{11}, \underline{\mathbf{p}}_{12}, \underline{\mathbf{p}}_{21}, \underline{\mathbf{p}}_{22}$. Assuming that a pulse is transmitted on the first input at time *n*, the noisy measured responses are $\underline{\tilde{\mathbf{p}}}_{11}$ at the first output and $\underline{\tilde{\mathbf{p}}}_{21}$ at the second output. The samples transmitted/received by a particular transmitter/receiver are shown in black (black dots) while the ones that are "skipped" are shown in grey (white dots).

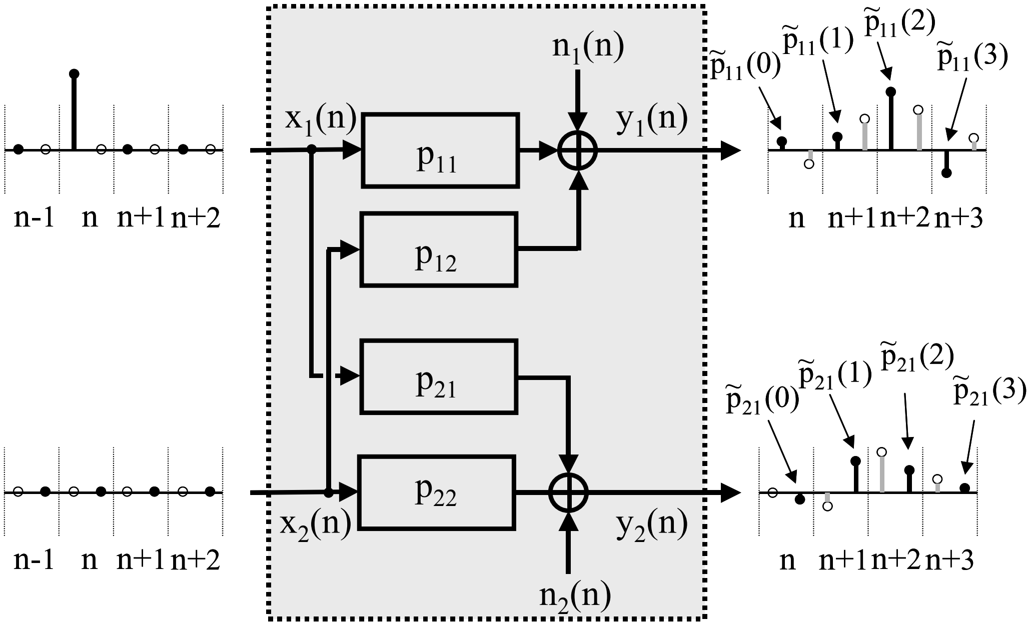


Figure C.4: 2x2 multi-channel system with example channel response.

Another way to view this mapping is that the samples of each TDM block are considered as a vector. This has the advantage of transforming a Single-Input-Single-Output cyclo-stationary channel into a MIMO time-invariant channel.

C.2 Equalization

C.2.1 Problem Formulation

The link described in previous section can be modeled as the MIMO system shown in Figure C.5, which includes transmit pre-emphasis filter, the channel, and receiver scaling necessary to restore the signal to known target levels, as we discussed in Chapter 3.

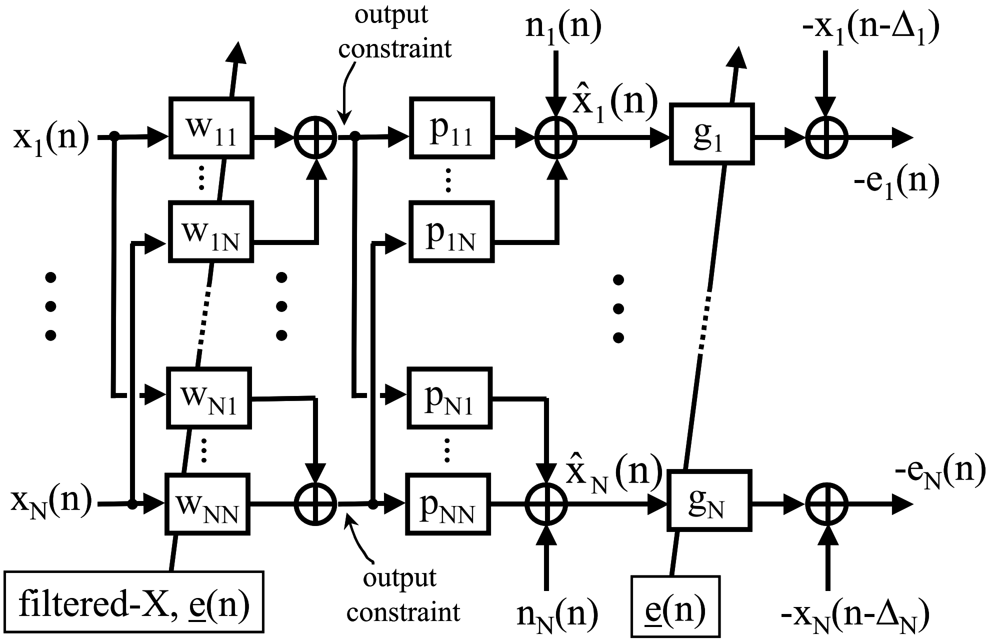


Figure C.5: NxN MIMO system with transmit pre-emphasis filter and receiver scaling.

The formulation used here is based on the *NxN* case of a system presented in [44]. A Minimum Mean-Square Error (MMSE) criterion is initially employed, and filters are assumed to have finite length. The response of the system from input *k* to output *l* at sample time *n* is given by:

$$\hat{x}_{lk}(n) = \underline{\mathbf{x}}_k^{\mathrm{T}}(n)\left(\mathbf{P}_{l1}\underline{\mathbf{w}}_{1k} + \mathbf{...} + \mathbf{P}_{lj}\underline{\mathbf{w}}_{jk} + \mathbf{...} + \mathbf{P}_{lN}\underline{\mathbf{w}}_{Nk}\right) \tag{C.1}$$

where $\underline{\mathbf{x}}_k(n) = \left[x_k(n) \quad \ldots \quad x_k(n-L-v+1)\right]^{\mathbf{T}}_{\mathbf{(L+v)\times 1}}$ is the data vector at input *k*, $\underline{\mathbf{w}}_{jk}(n) = \left[w_{jk}(0) \quad \ldots \quad w_{jk}(L-1)\right]^{\mathbf{T}}_{\mathbf{L\times 1}}$ is the pre-emphasis filter from data input *k* to transmitter output *j*, and the channel convolution matrix $\mathbf{P}_{lj}$ is defined as:

$$\mathbf{P}_{lj} = \begin{bmatrix} p_{lj}(0) & \ldots & p_{lj}(v) & 0 & 0 & 0 \\ 0 & p_{lj}(0) & \ldots & p_{lj}(v) & 0 & 0 \\ 0 & 0 & \ldots & \ldots & \ldots & 0 \\ 0 & 0 & 0 & p_{lj}(0) & \ldots & p_{lj}(v) \end{bmatrix}^{\mathbf{T}}_{\mathbf{(L+v)\times L}} \tag{C.2}$$

where the maximum ISI spread (measured in number of TDM blocks) for all $\mathbf{P}_{lj}$ is ν, and the pre-emphasis filter length is L. The system is fully described by:

$$\hat{\underline{\mathbf{x}}}(n)_{\mathbf{N\times 1}} = \mathbf{X}(n)_{\mathbf{N\times N^2(L+v)}}\,\mathbf{\Psi}_{\mathbf{N^2(L+v)\times N^2L}}\,\underline{\mathbf{W}}_{\mathbf{N^2L\times 1}} + \underline{\mathbf{N}}(n)_{\mathbf{N\times 1}} \tag{C.3}$$

where the channel matrix $\mathbf{\Psi}$ is defined as:

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{P} & & \mathbf{0} \\ & \ldots & \\ \mathbf{0} & & \mathbf{P} \end{bmatrix}_{\mathbf{N^2(L+v)\times N^2L}}, \quad \mathbf{P} = \left[\mathbf{P}_1^{\mathbf{T}} \ldots \mathbf{P}_N^{\mathbf{T}}\right]^{\mathbf{T}}_{\mathbf{N(L+v)\times NL}} \tag{C.4}$$

with $\mathbf{P}_k = \left[\mathbf{P}_{k1} \cdots \mathbf{P}_{kN}\right], \forall k = 1 \ldots N$, and the pre-emphasis filter $\underline{\mathbf{W}}$ is defined as:

$$\underline{\mathbf{W}} = \left[\underline{\mathbf{w}}_1^{\mathbf{T}} \quad \ldots \quad \underline{\mathbf{w}}_N^{\mathbf{T}}\right]^{\mathbf{T}}_{\mathbf{N^2L\times 1}}, \quad \underline{\mathbf{w}}_k = \left[\underline{\mathbf{w}}_{1k}^{\mathbf{T}} \quad \ldots \quad \underline{\mathbf{w}}_{Nk}^{\mathbf{T}}\right]^{\mathbf{T}}_{\mathbf{NL\times 1}} \tag{C.5}$$

and the input data matrix $\mathbf{X}(n)$ is defined as:

$$\mathbf{X}(n) = \left[\mathbf{X}_1(n) \quad \ldots \quad \mathbf{X}_N(n)\right]_{\mathbf{N \times N^2 (L+\nu)}}, \ \mathbf{X}_k(n) = \begin{bmatrix} \underline{\mathbf{x}}_k^{\mathbf{T}}(n) & & \underline{\mathbf{0}} \\ & \cdots & \\ \underline{\mathbf{0}} & & \underline{\mathbf{x}}_k^{\mathbf{T}}(n) \end{bmatrix}_{\mathbf{N \times N(L+\nu)}} \tag{C.6}$$

The noise vector $\underline{\mathbf{N}}(n)$ represents static and dynamic voltage domain noise as well as static and dynamic time domain noise mapped to the voltage domain, as modeled in Chapter 2. The *unbiased error* is defined by

$$\underline{\mathbf{e}}(n)_{\mathbf{N \times 1}} = \frac{1}{\sqrt{\overline{E}_x}} \left(\underline{\mathbf{x}}(n - \underline{\mathbf{\Delta}})_{\mathbf{N \times 1}} - \mathbf{g}_{\mathbf{N \times N}} \hat{\underline{\mathbf{x}}}(n)_{\mathbf{N \times 1}} \right) \tag{C.7}$$

where ***g*** is an *NxN* diagonal matrix with diagonal elements equal to $g_i, \forall i = 1 \ldots N$, where g_i is the scaling required at receiver *i*, so that the decisions include no bias. $\overline{E}_x$ is the average symbol energy, the vector of *delayed* transmitted data is $\underline{\mathbf{x}}(n - \underline{\mathbf{\Delta}}) = \left[x_1(n - \Delta_1) \quad \ldots \quad x_N(n - \Delta_N) \right]_{\mathbf{1 \times N}}^{\mathbf{T}}$, and $\underline{\mathbf{\Delta}}$ is a vector containing the decision delays of the corresponding inputs.

The total mean square error, Equation (C.8), represents the sum of the inverse of Signal-to-Noise Ratios (SNRs) and hence directly reflects the performance of the system.[35]

$$\xi = E\left(\underline{\mathbf{e}}^{\mathbf{T}}(n)_{\mathbf{1 \times N}} \underline{\mathbf{e}}(n)_{\mathbf{N \times 1}} \right) \tag{C.8}$$

At this point, it is useful to observe the duality between receiver equalization [12] and transmit pre-emphasis. While receiver equalization attempts to flatten the frequency response with an effective gain of one, transmit pre-emphasis with an output range constraint attempts to flatten the frequency response at a level below the biggest channel

[35] Summing the elements of the vector objective is just one of the ways to regularize the vector objective function [84] such as the vector MSE function at the outputs of the MIMO system. Given the tight BER constraints on link systems, one of the other regularization methods would be to minimize the maximum MSE across channels.

attenuation. Similar to the well-known noise enhancement problem related to receiver equalization, the noise at the receiver is relatively amplified due to the signal attenuation.

Retaining only the part of the noise that is independent of the data signal, having variance σ_{total}^2, Equation (C.8) is expanded to:

$$\xi = N - 2\underline{\mathbf{W}}^{\mathbf{T}}\mathbf{\Psi}^{\mathbf{T}}\mathbf{G}^{\mathbf{T}}\underline{\mathbf{I}}_{\Delta}^{\mathbf{T}} + \underline{\mathbf{W}}^{\mathbf{T}}\mathbf{\Psi}^{\mathbf{T}}\mathbf{G}^{\mathbf{T}}\,\mathbf{G}\mathbf{\Psi}\underline{\mathbf{W}} + \frac{tr\{\mathbf{g}^2\}}{SNR_0} \tag{C.9}$$

where the scaling matrix G is defined through the expression $\mathbf{X}^{\mathbf{T}}(n)\,\mathbf{g}^{\mathbf{T}} = \mathbf{G}^{\mathbf{T}}\,\mathbf{X}^{\mathbf{T}}(\mathbf{n})$. Also, the vector $\underline{\mathbf{I}}_{\Delta} = \begin{bmatrix}\underline{\mathbf{1}}_{\Delta_1} & \underline{\mathbf{0}} & \ldots & \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{1}}_{\Delta_2} & \ldots & \underline{\mathbf{0}} & \ldots & \underline{\mathbf{0}} & \ldots & \underline{\mathbf{0}} & \underline{\mathbf{1}}_{\Delta_N}\end{bmatrix}_{\mathbf{1\times N^2(L+v)}}$ represents the system delay, where the vector $\underline{\mathbf{1}}_{\Delta_l} = \begin{bmatrix}0 & \ldots & 0 & 1 & 0 & \ldots & 0\end{bmatrix}_{\mathbf{1\times(L+v)}}$ has a 1 at position $\Delta_l + 1$, and $SNR_0 = \overline{E}_x / \sigma_{total}^2$.

It may seem that an alternative approach to obtaining a solution to this problem is to determine the well-known receiver equalizer, and subsequently place it at the transmitter (possibly combined with proper scaling). However, such a method ignores the basic fact that the channel and equalizer matrices do not in general commute. If the equalizer is equal to the inverse of the channel (perfect zero-forcing), then under some assumptions the matrices may commute. Finite-length implementations cannot generally achieve perfect zero-forcing. Moreover, MMSE Linear Equalizer (MMSE-LE) solutions have been shown to be more desirable than Zero-Forcing Linear-Equalizer (ZF-LE) in circumstances of high interference.

C.2.2 Optimal Solution

The equalizer design may be formulated as an optimization problem, where the objective is the minimization of Equation (C.9). This problem is subject to two sets of constraints. First, the receiver scaling must be such that the decisions are unbiased. Therefore, the following must hold:

$$g_k \underline{\mathbf{w}}_k^{\mathbf{T}} \mathbf{P}_k^{\mathbf{T}} \underline{\mathbf{1}}_{\Delta_k}^{T} = 1, \forall k = 1 \ldots N \tag{C.10}$$

which implies that:

$$\underline{\mathbf{W}}^{\mathbf{T}} \mathbf{\Psi}^{\mathbf{T}} \mathbf{G}^{\mathbf{T}} \underline{\mathbf{I}}_{\Delta}^{\mathbf{T}} = N \, . \tag{C.11}$$

Then, Equation (C.9) becomes:

$$\xi = -N + \sum_{k=1}^{N} \underline{\mathbf{w}}_k^{\mathbf{T}} \left(\sum_{l=1}^{N} \frac{\mathbf{P}_l^{\mathbf{T}} \mathbf{P}_l}{\left(\underline{\mathbf{1}}_{\Delta_l} \mathbf{P}_l \underline{\mathbf{w}}_l \right)^2} \right) \underline{\mathbf{w}}_k + \frac{1}{SNR_0} \sum_{k=1}^{N} \left(\underline{\mathbf{1}}_{\Delta_k} \mathbf{P}_k \underline{\mathbf{w}}_k \right)^{-2} \, . \tag{C.12}$$

Additionally, the peak-transmitted signal on each of the outputs must be constrained:

$$\widetilde{h}_j \left(\underline{\mathbf{W}} \right) = \sum_{k=1}^{N} \sum_{i=0}^{L-1} \left| w_{jk}(i) \right| \le 1, \; \forall j = 1 \ldots N \, . \tag{C.13}$$

The filter design is expressed as an optimization problem involving the minimization of Equation (C.12) subject to the constraints of Equation (C.13). It appears that optimal closed-form solutions are impossible to obtain. In Chapter 3 we have already seen that this kind of optimization problem, although appears as a standard non-convex fractional quadratic programming problem, is really quasi-concave (in SNR) and quasi-convex in the context of Equation (C.12), and has a globally optimal solution.

At this point, the optimization framework that we outlined in Chapter 3 can be used on this regularized MIMO cost function and constraints, including all link-specific noise sources. However, these optimization algorithms have very high computational complexity and we also want to investigate some sub-optimal solutions that can yield very simple adaptive structures. In the next section we show the derivation of the sub-optimal algorithms that eventually lead to the adaptive formulation for the MIMO transmit pre-emphasis filter whose simplified SISO solution we presented in Chapter 3.

C.2.3 Sub-Optimal Solutions

Two sub-optimal approaches are presented that are based on the above formulation and work well at moderate to high SNR. These allow the implementation of simple LMS-type [105] adaptive algorithms, which are proposed in the next section.

According to the first approach, which does not have a closed form solution, the inequality constraints in Equation (C.13) are substituted by equalities. This is equivalent to forcing the maximum output of every pre-emphasis filter to be equal to the maximum transmitter range, thus transmitting the maximum available power into the channel and putting more strain on the pre-emphasis.

With the second approach, one first finds the unconstrained ZF-LE (ZFEU) solution, and then scales all the transmit filters by the same amount obtained from the unconstrained pre-emphasis filter with largest peak output, as shown below:

$$\underline{\mathbf{W}} = \frac{\underline{\mathbf{W}}_{ZFEU}}{\max\limits_{j=1\ldots N}(\widetilde{h}_j(\underline{\mathbf{W}}_{ZFEU}))}, \; \underline{\mathbf{W}}_{ZFEU} = \left(\mathbf{\Psi}^{\mathbf{T}}\mathbf{\Psi}\right)^{-1}\mathbf{\Psi}^{\mathbf{T}}\mathbf{G}^{\mathbf{T}}\underline{\mathbf{I}}_{\mathbf{\Delta}}^{\mathbf{T}} \tag{C.14}$$

The optimal delay $\underline{\mathbf{I}}_{\mathbf{\Delta}}$ vector in Equation (C.14) is determined from the set of delays $\underline{\Delta} = [\Delta_1 \quad \ldots \quad \Delta_N]$ that result in minimum total square error on each channel:

$$\Delta_j = \arg\max\left(diag_{j(L+\nu):(j+1)(L+\nu)-1}\left(\mathbf{\Psi}\left(\mathbf{\Psi}^{\mathbf{T}}\mathbf{\Psi}\right)^{\mathbf{-1}}\mathbf{\Psi}^{\mathbf{T}}\right)\right) \tag{C.15}$$

C.2.4 Adaptive Solution

While these sub-optimal closed form solutions have theoretical value, their mathematical complexity still makes them impractical for the implementation of a real high-speed system. However, both sub-optimal approaches, individual scaling and maximal scaling, Equations (C.13) and (C.15), can be implemented adaptively in a very simple manner using a modification of the multi-channel, multiple-error filtered-X LMS algorithm [106] as shown in Figure C.1. The pre-emphasis filter tap adaptation is described below:

$$SE_n = \underline{\mathbf{e}}(n)^{\mathbf{T}}\,\underline{\mathbf{e}}(n) \tag{C.16}$$

$$\hat{\nabla}_n\left(\underline{\mathbf{W}}\right) = \frac{\partial SE_n}{\partial \underline{\mathbf{W}}} = -\frac{2}{\sqrt{E_x}}\,\underline{\mathbf{U}}(n)^{\mathbf{T}}\mathbf{g}^{\mathbf{T}}\,\underline{\mathbf{e}}(n)\,,\ \underline{\mathbf{U}}(n) = \mathbf{X}(n)\boldsymbol{\Psi} \tag{C.17}$$

$$\underline{\tilde{\mathbf{W}}}_{n+1} = \underline{\mathbf{W}}_n - \mu_w \hat{\nabla}_n\left(\underline{\mathbf{W}}\right) = \underline{\mathbf{W}}_n + \frac{2\mu_w}{\sqrt{E_x}}\,\underline{\mathbf{U}}(n)^{\mathbf{T}}\mathbf{g}^{\mathbf{T}}\,\underline{\mathbf{e}}(n) \tag{C.18}$$

$$\underline{\mathbf{W}}_{n+1} = \begin{bmatrix} \dfrac{\underline{\tilde{\mathbf{w}}}_{11}^{\mathbf{T}}(n+1)}{\tilde{h}_1(\underline{\tilde{\mathbf{W}}}_{n+1})}_{\mathbf{1\times L}} & \mathbf{\cdots} & \dfrac{\underline{\tilde{\mathbf{w}}}_{NN}^{\mathbf{T}}(n+1)}{\tilde{h}_N(\underline{\tilde{\mathbf{W}}}_{n+1})}_{\mathbf{1\times L}} \end{bmatrix}^{\mathbf{T}}_{\mathbf{N^2L\times 1}} \tag{C.19}$$

$$\underline{\mathbf{W}}_{n+1} = \frac{1}{\max\limits_{j=1\ldots N}(\tilde{h}_j(\underline{\tilde{\mathbf{W}}}_{n+1}))}\,\underline{\tilde{\mathbf{W}}}_{n+1} \tag{C.20}$$

In the equations above, $\underline{\mathbf{U}}(n)$ is a filtered-X signal available in the transmitter, i.e. a noiseless copy of the signal without pre-emphasis that is received at the output of the channel[36]. The difference between the two approaches lies in the scaling of the filter coefficients after each iteration, which is shown in Equation (C.19) for individual scaling and in Equation (C.20) for maximal scaling.

The scaling values are adapted as shown below:

$$\hat{\nabla}_n\left(\mathbf{g}\right) = \frac{\partial SE_n}{\partial diag(\mathbf{g}_n)} = -\frac{2}{\sqrt{E_x}}\,diag(\,\hat{\underline{\mathbf{x}}}(n)\underline{\mathbf{e}}^{\mathbf{T}}(n)) \tag{C.21}$$

[36] This copy is called filtered-X since it is obtained in the transmitter either by convolving the symbols with estimated replica of the channel, or by alternating the equalized and un-equalized transmission and sending back to the transmitter the estimate of the un-equalized received signal.

$$diag(\mathbf{g}_{n+1}) = diag(\mathbf{g}_n) + \frac{2\mu_g}{\sqrt{E_x}} diag(\underline{\hat{\mathbf{x}}}(n)\underline{\mathbf{e}}^{\mathbf{T}}(n)) \tag{C.22}$$

The scaling loop has to converge much faster than the equalizer loop, since it provides the reference for the equalizer update.

In high-speed links, the algorithm is only intended to run when the configuration of the system has changed, and occasionally to adapt to variations of the transmission environment. Therefore, convergence time is not a huge issue and the implementation of a delayed version of the above algorithm is possible. The delayed version of the algorithm updates the equalizer coefficients only when new error information is available from the receiver on the back-channel. As we have seen in Chapter 3, we can further simplify this MMSE algorithm into a ZFE algorithm by using the sign of the received equalized symbol as a component of the update equation, instead of the sign of the filtered-X signal.

C.3 Experimental and Simulation Results

This section presents the results of performance simulations with the channel data obtained from the experimental test-bed presented in [28]. The original system employed several algorithms for static noise correction and channel equalization, and operated at 8 GSa/s with PAM2 and PAM4 modulation.

The simulation environment obtains the coefficients of the pre-emphasis filters using the adaptive algorithms from the previous section. The setup uses PAM4 modulation and the peak transmitter output range is 750 mV, with L=1 to 4 taps per pre-emphasis filter and ν=4.

To illustrate the channel-to-channel variations, pulse responses from all eight channels obtained in [28] are shown in Figure C.6. Clearly, not only are the ISI “profiles” distinct, but also the channel attenuation differs.

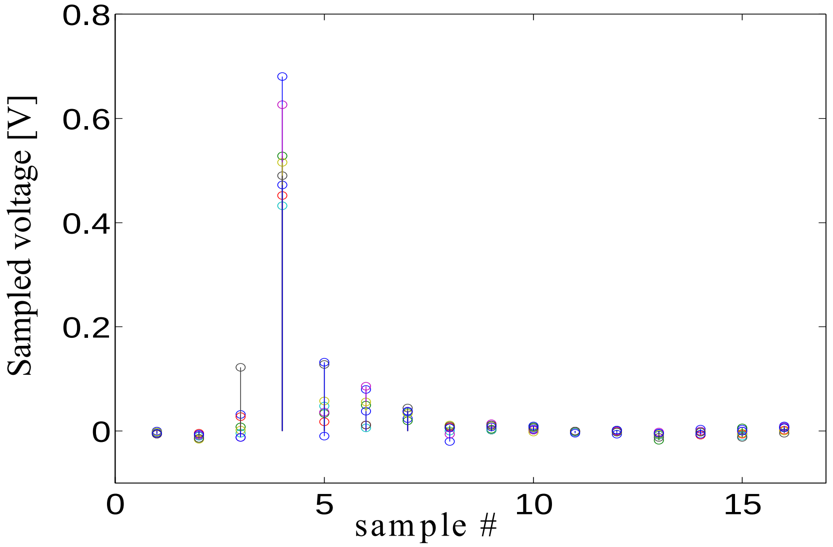


Figure C.6: Overlaid pulse responses of the 8 TDM sub-channels.

Learning curves of the total cost function ξ are shown in Figure C.7 for both individual scaling per channel and maximal scaling per channel, together with the cost function of scaled ZFEU with σ_{total}=4 mV, σ_{jitter}=6 ps, L=4.

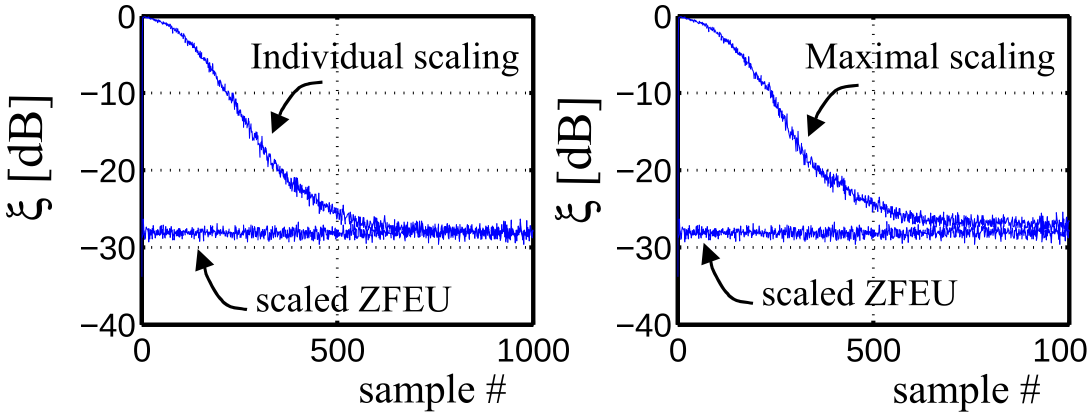


Figure C.7: Cost function learning curves.

Evidently, individual scaling performs slightly better than maximal. Adaptive maximal scaling converges to the scaled ZFEU solution with a misadjustment penalty, which depends on the convergence rate and the number of filter taps. The advantage of individual over maximal scaling is highly dependent on the type of the channel. In certain cases, the benefit of individual scaling is offset by the increase in residual ISI.

Learning curves for cost and scaling per channel are shown in Figure C.8. Notice that the scaling curves are a lot smoother and converge faster than the cost function. This is characteristic of the filtered-X algorithm, since the filtered-X value contains no noise information and hence is less robust to noise than the scaling loop at the receiver.

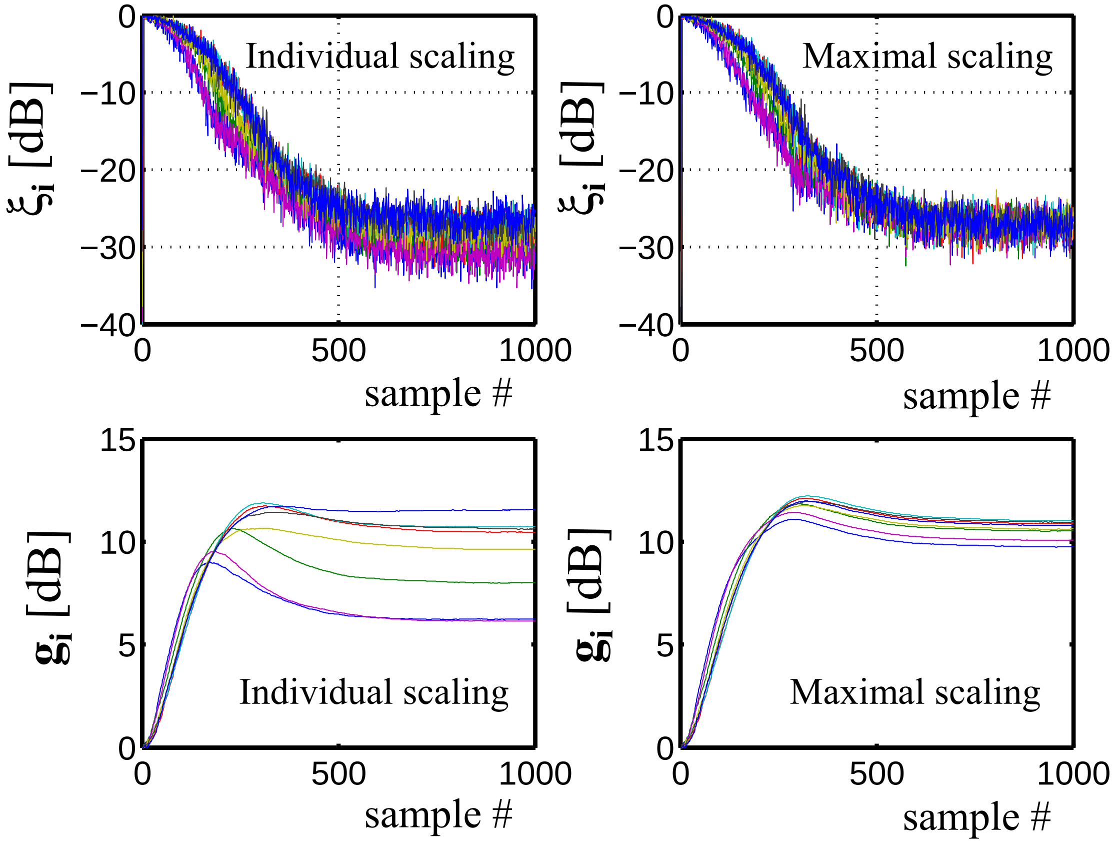


Figure C.8: Cost and scaling learning curves per channel.

The performance of the system is illustrated in Figure C.9, for different L and noise values. Voltage domain noise is composed of thermal noise and voltage reference noise with noise power $\sigma_{total.}$ Time domain noise consists of the clock jitter having standard deviation σ_{jitter}. After the taps are obtained from the simulation, the probability of error is estimated from the SNR value corresponding to the tap coefficients.

Different jitter noise values do not affect the filter coefficients, since these are averaged over multiple runs. At high noise setting, the point of noise-ISI tradeoff occurs at smaller number of taps (L=3-4), than at lower noise setting, indicating that the algorithms are trading noise for ISI.

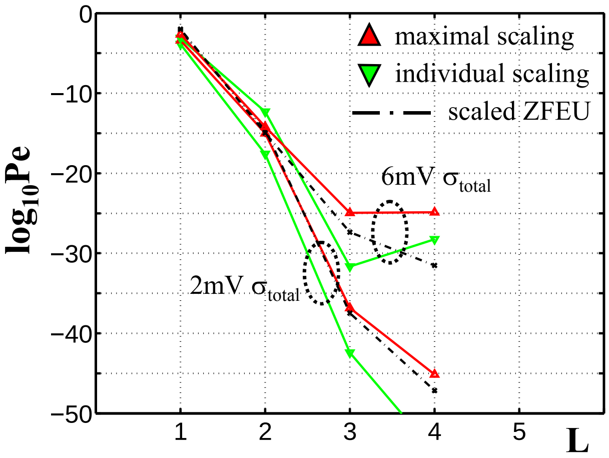


Figure C.9: Error probability (Pe) versus filter length, σ_{jitter}=6 ps.

Both individual scaling and maximal scaling have similar performance on the presented type of system, with some advantage obtained with individual scaling. However, both algorithms offer similar performance to closed form solutions with significantly less computation. The residual non-averaged jitter noise limits the accuracy of the probability of error calculation method, especially for very low probability of error values that occur when a large number of taps is used.

Another effect that was observed in the results is the non-monotonicity of the probability of error curve with the number of taps. This is attributed to symbol spaced equalization and jitter. As we have illustrated in Chapter 4, symbol spaced equalization does not have direct control over the width of the data eye. Hence, although more taps in the filter mean less residual ISI, wider data eye is not guaranteed. It is possible in cases with dominant jitter noise to improve voltage margins with more taps, but degrade timing margins at the same time, and thus degrade the probability of error.

Appendix D

Pre-Emphasis Scaling

In this appendix we present an approximation to pre-emphasis tap rescaling for the adaptive algorithm in Chapter 3. This approximation results in a very simple hardware implementation.

We can rewrite the update Equation (3.8), to reflect the re-scaling update with peak transmitter headroom constraint W_{max}:

$$\underline{w}_{n+1} = (\underline{w}_n + \underline{update}_n) \cdot \frac{W_{\max}}{\left\| \underline{w}_n + \underline{update}_n \right\|_1} \tag{D.1}$$

where $\underline{update}_n$ represents a vector of updates $sign(e_n)sign(\underline{d}_n)$.

The total required increase in headroom from the update is then:

$$W_{residual} = \left\| \underline{w}_n + \underline{update}_n \right\|_1 - W_{\max} = \operatorname{sgn}(\underline{w}_n)^T \cdot \underline{update}_n . \tag{D.2}$$

To see the approximation more easily we first rewrite the Equation (D.1):

$$\begin{aligned} \underline{w}_{n+1} &= (\underline{w}_n + \underline{update}_n) \cdot \frac{W_{\max}}{\left\| \underline{w}_n + \underline{update}_n \right\|_1} \\ &= (\underline{w}_n + \underline{update}_n) \cdot \frac{1}{1 + W_{residual} / W_{\max}} \end{aligned} \tag{D.3}$$

Now assuming that $W_{residual}$ is much smaller than W_{max} we can approximate Equation (D.3) with its first order Taylor series expansion:

$$\begin{aligned} \underline{w}_{n+1} &\approx (\underline{w}_n + \underline{update}_n) \cdot \left(1 - W_{residual} / W_{\max} \right) \\ &= (\underline{w}_n + \underline{update}_n) - (\underline{w}_n + \underline{update}_n) \cdot \left(W_{residual} / W_{\max} \right) . \end{aligned} \tag{D.4}$$

Equation (D.4) can now be implemented in hardware very efficiently. We use binary addition to compute the sum $\underline{w}_n$+$\underline{update}_n$, and simple XORs and adds to compute $W_{residual}$ from the sign of tap coefficients and update information. The ratio of $W_{residual}$ and W_{max} can be approximated as right shift by $log_2(W_{max})$, i.e. a total number of bits in the transmit DAC, and a left shift by $log_2(W_{residual})$. Since $W_{residual}$ can be anywhere between $–L$ and L, where L is the number of taps, we can pre-compute the required left-shifts, and even perform dithering to better approximate the left-shift values. For example if $W_{residual}$ is 3, we can alternate the left-shift between 2 and 4 during adaptation.

An alternative approach to tap scaling developed by Ho in [81] is to first update only the non-main taps and then re-adjust the main tap to maintain the total transmit headroom.

Bibliography

[1] G.D. Forney Jr., L. Brown, M.V. Eyuboglu, J.L. Moran, “The V.34 high speed modem standard,” IEEE Communications Magazine, vol. 34, no. 12, Dec. 1996, pp. 28–33.

[2] D-Y. Kim, P.A. Humblet, M.V. Eyuboglu, L. Brown, G.D. Forney Jr., S. Mehrabanzad, ”V.92: the last dial-up modem?,” *IEEE Transactions on Communications*, vol. 52, no. 1, Jan. 2004, pp. 54–61.

[3] J.S**.** Chow, J.C. Tu, J. M. Cioffi, “A Discrete Multitone Transceiver System for HDSL Applications,” *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 6, Aug. 1991, pp**.** 895-908.

[4] L.G. Roberts, “Beyond Moore's law: Internet growth trends,” *Computer*, vol. 33, no. 1, Jan. 2000, pp. 117-119.

[5] C. Metz, “IP routers: new tool for gigabit networking*,*” *IEEE Internet Computing*, vol. 2, no. 6, Nov.-Dec. 1998, pp. 14-18.

[6] A. Mu, J. Larson, R. Sastry, T. Wicki, W.W. Wilcke, “A 9.6 GigaByte/s throughput plesiochronous routing chip,” *Compcon* '96, Feb. 1996, pp. 261-266.

[7] K-Y.K. Chang, S-T. Chuang, N. McKeown and M. Horowitz, “A 50 Gb/s 32×32 CMOS crossbar chip using asymmetric serial links,” *IEEE Symposium on VLSI Circuits,* June 1999, pp.19-22.

[8] M.M Griffin *et al*, “A process-independent, 800-MB/s, DRAM byte-wide interface featuring command interleaving and concurrent memory operation,” *IEEE Journal of Solid-State Circuits*, vol: 33, no. 11, Nov. 1998, pp. 1741 –1751.

[9] H. Yoon *et al*, “A 2.5-V, 333-Mb/s/pin, 1-Gbit, double-data-rate synchronous DRAM,” *IEEE J. Solid-State Circuits*, vol. 34, no. 11, Nov. 1999, pp. 1589 –1599.

[10] T.P. Thomas, I.A. Young, “Four-way processor 800 MT/s front side bus with ground referenced voltage source I/O,” *IEEE Symposium on VLSI Circuits,* June 2002, pp. 70 –71

[11] S. Sidiropoulos, M. Horowitz, “A 700 Mbps/pin CMOS signalling interface using current integrating receivers,” *IEEE Journal of Solid-State Circuits*, vol 32, no. 5, May 1997, pp. 681-690.

[12] J. Proakis, M. Salehi, *Communication Systems Engineering*, Prentice Hall 1994.

[13] C.-K. Yang, M. Horowitz, “A 0.8um CMOS 2.5Gb/s oversampling receiver and transmitter for serial links,” *IEEE Journal of Solid-State Circuits*, vol. 31, no. 12, Dec. 1996, pp. 2015-2023.

[14] C.-K. Yang, R. Farjad-Rad, M. Horowitz, “A 0.5-µm CMOS 4.0-Gbit/s serial link transceiver with data recovery using oversampling,” *IEEE Journal of Solid-State Circuits*, vol. 33, no. 5, May. 1998, pp. 713-722.

[15] S. Sidiropoulos, M.A. Horowitz, “A semidigital dual delay-locked loop,” *IEEE Journal of Solid-State Circuits*, vol. 32, no. 11, Nov. 1998, pp. 1683-1692.

[16] S. Sidiropoulos, D. Liu, J. Kim, G. Wei, M. Horowitz,”Adaptive Bandwidth DLLs and PLLs using Regulated Supply CMOS Buffers,” *IEEE Symposium on VLSI Circuits*, June 2000, pp. 124-127.

[17] I.A. Young, J.K. Greason, J.E. Smith, K.L. Wong, “A PLL clock generator with 5 to 110 MHz lock range for microprocessors,” *IEEE International Solid-State Circuits Conference,* Feb. 1992, pp. 50-51, 240.

[18] M. Horowitz, A. Chan, J. Conbrunson, J. Gasbarro, T. Lee, W. Leung, W. Richardson, T. Thrush, Y. Fujii, "PLL design for a 500 MB/s interface," *IEEE International Solid-State Circuits Conference,* Feb. 1993, pp. 160-161.

[19] J.G. Maneatis, "Low-jitter process-independent DLL and PLL based on self-biased techniques," *IEEE Journal of Solid-State Circuits,* vol 31, no. 11, Nov. 1996, pp. 1723-1732.

[20] T.H. Lee, J.F. Bulzacchelli, "A 155-MHz clock recovery delay- and phase-locked loop," *IEEE Journal of Solid-State Circuits,* vol. 27, no. 12, Dec. 1992, pp. 1736-1746.

[21] L. DeVito, J. Newton, R. Croughwell, J. Bulzacchelli, F. Benkley, "A 52MHz And 155MHz Clock-recovery PLL," *IEEE International Solid-State Circuits Conference,* Feb. 1991, pp. 142, 306.

[22] M.-J.E. Lee, W.J. Dally, J. Poulton, T. Greer, J. Edmondson, R. Farjad-Rad, N. Tiaq, R. Rathi, R. Senthinathan, "A second-order semi-digital clock recovery circuit based on injection locking," *IEEE Solid-State Circuits Conference,* Feb. 2003, pp. 74-75

[23] R. Walker *et al*, "A 2.488Gb/s Si-Bipolar Clock and Data Recovery IC with Robust Loss of Signal Detection," *IEEE International Solid-State Circuits Conference*, Feb.1997.

[24] W.J. Dally, J. Poulton, "Transmitter equalization for 4-Gbps signaling," *IEEE Micro*, vol. 17, no. 1, Jan.-Feb. 1997, pp. 48-56.

[25] A. Fiedler, R. Mactaggart, J. Welch, S. Krishnan, "A 1.0625 Gbps transceiver with 2x-oversampling and transmit signal pre-emphasis," *IEEE International Solid-State Circuits Conference*, Feb. 1997, pp. 238-239, 464.

[26] H. Partovi *et al*, "A 62.5 Gb/s multi-standard SerDes IC," *IEEE Custom Integrated Circuits Conference*, Sept. 2003, pp. 585-588.

[27] R. Farjad-Rad, C.-K. Yang, M. Horowitz, T. Lee, "A 0.3-um CMOS 8-Gb/s 4-PAM Serial Link Transceiver," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 5, May 2000, pp. 757-764.

[28] C.-K.K. Yang, V. Stojanović, S. Modjtahedi, M.A. Horowitz, W.F. Ellersick, "A Serial-Link Transceiver Based on 8GSample/s A/D and D/A Converters in 0.25µm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 11, November 2001, pp. 1684-1692.

[29] Cisco 12000 series router (http://www.cisco.com/en/US/products/hw/routers/ps167/index.html)

[30] Juniper Networks T-series router (http://www.juniper.net/products/tseries/)

[31] IBM Blade Servers (http://www-1.ibm.com/servers/eserver/bladecenter/blade_servers/index.html)

[32] Cisco CRS-1 router (http://www.cisco.com/en/US/products/ps5763/)

[33] Teradyne GbX connector (http://www.teradyne.com/prods/tcs/products/connectors/backplane/gbx/)

[34] R. Kollipara, G-J Yeh, B. Chia, A. Agarwal, "Design, Modeling and Characterization of High-Speed Backplane Interconnects," *DesignCon 2003*.

[35] H. Johnson, M. Graham, *High-Speed Digital Design: A Handbook of Black Magic*, Prentice Hall 1993.

[36] G. Sheets, J. D'Ambrosia, "The Impact of Environmental Conditions on Channel Performance," *DesignCon 2004*.

[37] V. Stojanović, M. Horowitz, "Modeling and Analysis of High-Speed Links," *IEEE Custom Integrated Circuits Conference*, September 2003.

[38] J. Zerbe, C. Werner, R. Kollipara, V. Stojanović, "A Flexible Serial Link for 5-10Gb/s in Realistic Backplane Environments," *DesignCon 2004*.

[39] J. Zerbe, C. Werner, V. Stojanović, F. Chen, J. Wei, G. Tsang, D. Kim, W. Stonecypher, A. Ho, T. Thrush, R. Kollipara , M. Horowitz, K. Donnelly, "Equalization and Clock Recovery for a 2.5-10Gb/s 2-PAM/4-PAM Backplane Transceiver Cell," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 12, Dec. 2003, pp. 2121-2130.

[40] S. Sercu, J. De Geest, "BER Link Simulations," *DesignCon 2003.*

[41] G. Ginis, J.M. Cioffi, "A multi-user precoding scheme achieving crosstalk cancellation with application to DSL systems," *Asilomar Conference on Signals, Systems and Computers*, Nov. 2000. pp. 1627-1631.

[42] A. Maleki-Tehrani, B. Hassibi, J.M. Cioffi, "Adaptive equalization of multiple-input multiple-output (MIMO) channels," *IEEE International Conference on* Communications, June 2000, pp. 1670-1674.

[43] D.P. Palomar, J.M. Cioffi, M.A. Lagunas, "Joint Tx-Rx beamforming design for multicarrier MIMO channels: a unified framework for convex optimization,"*IEEE Transactions on Signal Processing*, vol. 51, no. 9, Sept. 2003, pp. 2381-2401.

[44] P. A. Nelson, F. O-Bustamante and H. Hamada, "Inverse Filter Design and Equalization Zones in Multichannel Sound Reproduction," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 3, May 1995, pp.185-92.

[45] V. Stojanović, G. Ginis, M.A. Horowitz, "Transmit Pre-emphasis for High-Speed Time-Division-Multiplexed Serial Link Transceiver," *IEEE International Conference on Communications,* May 2002, pp. 1934 -1939.

[46] T.H. Lee, *The Design of CMOS Radio-Frequency Integrated Circuits*, Cambridge University Press, December 2003.

[47] M. Mansuri, C-K.K. Yang, "Jitter optimization based on phase-locked loop design parameters," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, Nov. 2002, pp. 1375 –1382.

[48] K-L.J. Wong, C-K.K. Yang, “Offset Compensation in Comparators with Minimum Input-Referred Supply Noise,” *IEEE Journal of Solid-State Circuits*, vol. 39, no. 5, May 2004, pp. 837-840.

[49] A. Muhtaroglu, G. Taylor, T. Rahal-Arabi, “On-die droop detector for analog sensing of power supply noise,” *IEEE Journal of Solid-State Circuits*, vol. 39, no. 4, April 2004, pp.651-660.

[50] P. Larsson, “Power supply noise in future IC's: a crystal ball reading,” *IEEE Custom Integrated Circuits* Conference, May 1999, pp. 467-474.

[51] E. Alon, V. Stojanović, M.A. Horowitz, “Circuits and Techniques for High-Resolution Measurement of On-Chip Power Supply Noise,” *IEEE Symposium on VLSI Circuits*, June 2004, pp. 102-105.

[52] W. Gardner, L. Franks, “Characterization of cyclostationary random signal processes,” *IEEE Transactions on Information Theory*, vol. 21, no. 1, Jan. 1975, pp. 4-14.

[53] B.J. Skinner, F.M. Ingels, and J.P. Donohoe, “Stationary and cyclostationary random process models,” in *Proceedings of the IEEE Southeastcon ’94*, April 1994, pp. 450-454.

[54] W. Gardner, “Stationarizable random processes,” *IEEE Transactions on Information Theory*, vol. 24, no. 1, Jan 1978, pp. 8–22.

[55] A. Yukawa, “A CMOS 8-Bit High-Speed A/D Converter IC,” *IEEE Journal of Solid-State Circuits*; vol. 20, no.3, Jun. 1985, pp.775-79.

[56] M.J. Pelgrom, A.C.J. Duinmaijer, A.P.G. Welbers, “Matching properties of MOS transistors,” *IEEE Journal of Solid-State Circuits*, vol. 24, no. 5, Oct. 1989, pp. 1433-1439.

[57] R. Ho, K. Mai, M. Horowitz*,* "Applications of on-chip samplers for test and measurement of integrated circuits," *IEEE Symposium on VLSI Circuits*, June 1998, pp. 271-274.

[58] A. Ho, V. Stojanović, F. Chen, C. Werner, G. Tsang, E. Alon, R. Kollipara, J. Zerbe, M.A. Horowitz, "Common-mode Backchannel Signaling System for Differential High-Speed Links," *IEEE Symposium on VLSI Circuits*, June 2004, pp. 352-355.

[59] M. Shoji, *Theory of CMOS Digital Circuits and Circuit Failure*s, Princeton University Press, 1992.

[60] V. Stojanović, A. Amirkhany, M. Horowitz, "Optimal Linear Precoding with Theoretical and Practical Data Rates in High-Speed Serial-Link Backplane Communication," *IEEE International Conference on Communications*, June 2004, pp. 2799-2806.

[61] A. Hajimiri, "Noise in phase-locked loops," *Southwest Symposium on Mixed-Signal Design*, 2001. pp. 1–6

[62] A. Demir, *et al*, "Phase noise in oscillators: a unifying theory and numerical methods for characterization," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 5, May 2000, pp. 655 –674.

[63] B. Razavi, *Monolithic Phase-Locked Loops and Clock Recovery Circuits, Theory and Design*, IEEE Press 1996.

[64] J.D.H. Alexander, "Clock Recovery from Random Binary Signals", *Electronic Letters*, vol. 11, October 1975, pp. 541-542.

[65] J.K. Holmes, "Performance of a First-Order Transition Sampling Digital Phase-Locked Loop Using Random-Walk Models," *IEEE Transactions on Communications*, vol.com-20, no. 2, April 1972.

[66] A.E. Payzin, “Analysis of a Digital Bit Synchronizer,” *IEEE Transactions on Communications*, vol.com-31, no. 4, April 1983.

[67] D.W. Paranchych and N.C. Bealieu, “Use of second order Markov chains to model digital symbol synchronizer performance,” *IEE Proceedings – Communications*, vol. 143, no. 5, October 1996, pp. 250-258.

[68] I. Galton, “Analog-input digital phase-locked loops for precise frequency and phase demodulation,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 42, no. 10, October 1995, pp. 621-630.

[69] J.R. Cessna and D.M. Levy, “Phase Noise and Transient Times for a Binary Quantized Digital Phase-Locked Loop in White Gaussian Noise,” *IEEE Transactions on Communications*, vol.com-20, no. 2, April 1972.

[70] J. Kim, *Design of CMOS Adaptive-Supply Serial Links, Ph.D. Thesis*, Stanford University, December 2002.

[71] K.H. Mueller, M. Muller, “Timing Recovery in Digital Synchronous Data Receivers,” *IEEE Transactions on Communications*, vol. com-24, no. 5, May 1976, pp. 516-531.

[72] J. Armstrong, “Symbol Synchronization Using Baud-Rate Sampling and Data-Sequence-Dependent Signal Processing,” *IEEE Transactions on Communications*, vol. 39, no. 1, Jan. 1991, pp. 127-132.

[73] T. Aboulnasr, M. Hage, B. Sayar, S. Aly, “Characterization of Symbol Rate Timing Recovery Technique for a 2B1Q Digital Receiver,” *IEEE Transactions on Communications*, vol. 42, no.2/3/4, February/March/April 1994.

[74] S. Hongwei, B.V.K.V Kumar, E. Kurtas, Y. Yifei, L.L. McPheters, S.W. McLaughlin, “Iterative decoding for partial response (PR), equalized, magneto-optical (MO) data storage channels,” *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 4, April 2001, pp. 774–782.

[75] F. Cali, M. Conti, E. Gregori, "IEEE 802.11 wireless LAN: capacity analysis and protocol enhancement," *in Proceedings INFOCOM '98*, April 1998, pp. 142-149.

[76] G.G. Raleigh, J.M. Cioffi, "Spatio-temporal coding for wireless communication," *IEEE Transactions on Communications*, vol. 46, no. 3, March 1998, pp. 357-366.

[77] C. Berrou, A. Glavieux, P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," *IEEE International Conference on Communications*, May 1993, pp. 1064-1070.

[78] Reed, I., G. Solomon, "Polynomial codes over certain finite fields," *Journal SIAM*, vol. 8, June 1960, pp. 300-304.

[79] E. R. Berlekamp, "Bit-serial Reed-Solomon encoders," *IEEE Transactions on Information Theory,* vol. IT-28, no. 6, Nov. 1982.

[80] L. Song, M-L. Yu, M.S. Shaffer, "10- and 40-Gb/s forward error correction devices for optical communications," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, Nov. 2002, pp. 1565-1573.

[81] V. Stojanović, A. Ho, B. Garlepp, F. Chen, J. Wei, E. Alon, C. Werner, J. Zerbe, M.A. Horowitz, "Adaptive Equalization and Data Recovery in a Dual-Mode (PAM2/4) Serial Link Transceiver," *IEEE Symposium on VLSI Circuits*, June 2004, pp. 348-351.

[82] R.G. Gallager, *Information Theory and Reliable Computation*, John Wiley & Sons, Inc., USA, 1968.

[83] J.M. Cioffi, G.P. Dudevoir, M.V. Eyuboglu, G.D. Forney Jr., "MMSE decision-feedback equalizers and coding. II. Coding results," *IEEE Transactions on Communications*, vol. 43, no. 10, Oct. 1995, pp. 2595-2604.

[84] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press 2004. (http://www.stanford.edu/~boyd/cvxbook.html)

[85] J. Tellado, J.M. Cioffi, “Efficient algorithms for reducing PAR in multicarrier systems,” *IEEE International Symposium on Information Theory*, Aug. 1998, pp. 191.

[86] H. Rategh, H. Samavati, T. Lee, “A CMOS frequency synthesizer with an injection-locked frequency divider for a 5 GHz Wire LAN receiver,” *IEEE Journal of Solid-State Circuits*, vol. 35, May 2000, pp. 779--786.

[87] R.G. Gallagher, “Low density parity check codes,” *IRE Transactions on Information Theory*, vol. 8, 1962, pp. 21-28.

[88] J. Campello, “Practical bit loading for DMT,” *IEEE International Conference on Communications*, pp. 796-800, 1999.

[89] R. Farjad-Rad et al, “0.622-8.0Gbps 150mW Serial IO Macrocell with Fully Flexible Preemphasis and Equalization,” *IEEE Symposium on VLSI Circuits*, June 2003.

[90] C. A. Belfiore, J. H. Park, Jr., “Decision feedback equalization,” *Proceedings IEEE***,** August 1979, pp. 1143-1 156.

[91] J.M. Cioffi *et al*, “MMSE Decision-Feedback Equalizers and Coding-Part I: Equalization Results,” *IEEE Transactions on Communications*, vol. 43, no. 10, October 1995, pp. 2582-2594.

[92] M. Tomlinson, “New Automatic Equalizer Employing Modulo Arithmetic,” *Electronic Letters*, March 1971, pp. 138-139.

[93] H. Harashima, H. Miyakawa, “Matched–Transmission Technique for Channels with Intersymbol Interference,” *IEEE Transactions on Communications*, August 1972, pp. 774-780.

[94] R. Laroia, S.A. Tretter, N. Farvardin, “A simple and effective precoding scheme for noise whitening on intersymbol interference channels,” *IEEE Transactions on Communications*, vol. 41, no. 10, Oct. 1993, pp. 1460-1463.

[95] E.F. Haratsch, K. Azadet, “A pipelined 14-tap parallel decision-feedback decoder for 1000BASE-T Gigabit Ethernet,” *IEEE International Symposium on VLSI Technology, Systems, and Applications,* 2001, pp. 117-120.

[96] M.Q. Le, P.J. Hurst, K.C. Dyer, “An Analog DFE for Disk Drives Using a Mixed-Signal Integrator,” *IEEE J. Solid-State Circuits*, May 1999, pp. 592-598.

[97] K. Poulton *et al*, “A 20GS/s 8b ADC with a 1MB Memory in 0.18µm CMOS,” *IEEE International Solid-State Circuits Conference*, Feb. 2003, San Francisco.

[98] S. Kasturia, J.H. Winters, “Techniques for high-speed implementation of nonlinear cancellation,” *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 5, Jun 1991, pp. 711-717.

[99] K.K. Parhi, “High-Speed architectures for algorithms with quantizer loops,” *IEEE International Symposium on Circuits and Systems*, vol. 3, May 1990, pp. 2357-2360

[100] Y-S. Sohn *et al*, “A 2.2Gbps CMOS look-ahead DFE receiver for multidrop channel with pin-to-pin time skew compensation,” *IEEE Custom Integrated Circuits Conference*, September 2003.

[101] N. Al-Dhahir, J.M. Cioffi, “MMSE decision-feedback equalizers: finite-length results,” *IEEE Transactions on Information Theory*, vol. 41, no. 4, July 1995, pp. 961-975.

[102] B.R. Vojčić, W.M. Jang, “Transmitter Precoding in Synchronous Multiuser Communications,” *IEEE Transactions on Communications*, vol. 46, no. 10, October 1998, pp. 1346-55.

[103] S. Schaible, "Fractional programming," *Handbook of Global Optimization*, eds. Horst, R. and P.M. Pardalos, Nonconvex Optimization and its Applications, 2, Kluwer Academic Publishers, Dordrecht - Boston-London, 1995, pp. 495-608.

[104] B. Ahmad, "Performance Specification of Interconnects," *DesignCon 2003*.

[105] B. Widrow *et al*,, “Stationary and nonstationary learning characteristics of the LMS adaptive filter,” *Proc. IEEE,* vol. 64, no. 8, pp. 1151-1162, 1976.

[106] P. A. Nelson, H. Hamada, S. J. Elliot, “Adaptive Inverse Filters for Stereophonic Sound Reproduction,” *IEEE Transactions on Signal Processing*, vol. 40, no. 7, July 1992, pp. 1621-32.

[107] J.T. Stonick *et al*, "An adaptive pam-4 5-Gb/s backplane transceiver in 0.25-µm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 3, March 2003, pp. 436-443.

[108] J. Jaussi, G. Balamurugan, D. Johnson, B. Casper, A. Martin, J. Kennedy, R. Mooney, N. Shanbhag, “An 8Gb/s Source-Synchronous I/O Link with Adaptive Receiver Equalization, Offset Cancellation, and Clock Deskew,” *IEEE International Solid-State Circuits Conference*, Feb. 2004.

[109] J. L. Zerbe *et al*, “1.6Gb/s/pin 4-PAM signaling and circuits for a multidrop bus,” *IEEE J. Solid-State Circuits*, vol. 35, May 2001, pp. 752-760

[110] D. Secker, H-J Liaw, G. Pitner, “Design of a 3.2GB/s Memory Sub-System for PlayStation®2,” *DesignCon 2001.*

[111] W.J. Dally, J.W. Poulton, *Digital Systems Engineering*, Cambridge University Press, 1998.

[112] B. Ahmad, J. Cain, “Performance evaluation of high speed serial links,” *DesignCon 2001.*

[113] B.K. Casper *et al*, "An accurate and efficient analysis method for multi-Gb/s chip-to-chip signaling schemes," *IEEE Symposium on VLSI Circuits*, June 2002, pp. 54-57.

[114] G. Balamurugan, N. Shanghag, “Modeling and mitigation of jitter in multiGbps source-synchronous I/O links,” *IEEE International Conference on Computer Design*, Oct. 2003, pp. 254-260.

[115] P.K. Hanumolu, B. Casper, R. Mooney, G-Y. Wei, U-K. Moon, "Analysis of PLL clock jitter in high-speed serial links," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 50, no. 11, Nov. 2003, pp. 879-886.

[116] A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering*, Addison-Wesley Pub Co; 2nd edition, July 1993.

[117] H. Lee, C.H. Yue, S. Palermo, K.W. Mai, M. Horowitz, "Burst Mode Packet Receiver using a Second Order DLL," *IEEE Symposium on VLSI Circuits*, June 2004, pp. 264-267.

[118] B. Casper, A. Martin, J.E. Jaussi, J. Kennedy, R. Mooney, "An 8-Gb/s simultaneous bidirectional link with on-die waveform capture," *IEEE Journal of Solid-State* Circuits, vol. 38, no. 12, Dec. 2003, pp. 2111-2120.

[119] M.-J.E. Lee, W.J. Dally, P. Chiang, "Low-power area-efficient high-speed I/O circuit techniques," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 11, Nov. 2000, pp. 1591-1599

[120] T. Gabara, "Phantom Mode Signaling in VLSI Systems," *Conference on Advanced Research in VLSI*, Mar. 2001

[121] S. Wu *et al,* "Design of a 6.25-Gbps Backplane SerDes with Top-Down Design Methodology," *DesignCon 2004*.

[122] W. C. Madden, W. J. Bowhill, "High input impedance strobed CMOS differential sense amplifier," U.S. Patent 4 910 713, Mar. 1990.

[123] J.H. Winters, S. Kasturia, "Adaptive nonlinear cancellation for high-speed fiber-optic systems," *IEEE J. of Lightwave Technology*, vol 10, no. 7, July 1992, pp. 971-977.

[124] A. Amirkhany, V. Stojanovic, M.A. Horowitz, "Multi-tone Signaling for High-speed Backplane Electrical Links," *IEEE Global Telecommunications Conference*, Nov. 2004.

[125] B. Saltzberg, “Performance of an efficient parallel data transmission system,” *IEEE Transactions on Communications,* vol. 15, no. 6, pp 805-11, Dec. 1967.

[126] E. Ginzton, W. Hewlett, J. H. Jasberg, J. D. Noe, “Distributed Amplification,” *Proceedings of the IRE*, vol. 36, August 1948, pp. 956-69.

[127] A. X. Widmer, P.A. Franaszek, “A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code,“ *IBM Journal of Research and Development*, vol. 27, no. 5, Sep. 1983, pp. 440-451.

[128] A.G. Bessios, W.F. Stonecypher, A. Agarwal, J.L. Zerbe, “Transition-limiting codes for 4-PAM signaling in high speed serial links,” *IEEE Global Telecommunications Conference*, Dec. 2003, pp. 3747-3751.