ARCHITECTURAL & CIRCUIT LEVEL TECHNIQUES TO IMPROVE ENERGY EFFICIENCY OF HIGH SPEED SERIAL LINKS

BY

SAURABH SAXENA

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Doctoral Committee:

Associate Professor Pavan Kumar Hanumolu, Chair
Professor Jose E. Schutt-Aine
Professor Naresh R. Shanbhag
Professor Pramod Viswanath

Abstract

High performance computing and communication are two key aspects of all information processing systems. With aggressive scaling of silicon technology enabling integration of a large number of transistors in a small area, managing power and thermal reliability has become very challenging. While lowering the power needed for performing computation has been the prime focus for decades, energy consumed for data transfer has recently become a major bottleneck especially in high performance applications. The focus of this thesis is on improving energy efficiency of communication links by exploring design techniques at both the architectural and circuit levels.

In the first part of this work, we propose a time-based equalization scheme to implement transmit de-emphasis in voltage-mode output drivers. Using two-level pulse-width modulation, it overcomes the tradeoff between impedance matching, output swing, and de-emphasis resolution in conventional voltage-mode drivers. A prototype PWM-based 5 Gb/s voltage-mode transmitter was implemented in a 90 nm CMOS process and characterized across different channels and output swings to demonstrate the effectiveness of proposed techniques. The horizontal/vertical eye openings (BER=10^{-12}) at the ends of 60 inch and 96 inch stripline channels are 78 mV/0.6 UI and 8 mV/0.3 UI, respectively. This transmitter achieves an energy efficiency of 3.1 mW/Gb/s while compensating for 16-28 dB channel loss, which compares favorably with the state-of-the-art.

In the second part, techniques to improve energy efficiency of a complete transceiver are presented. The transmitter employs a novel partially segmented voltage-mode output driver to lower power consumption in pre-drivers during 2-tap FIR equalization. The receiver implements a low power half-rate clock and data recovery with the proposed ring PLL based multi-phase sampling clock generation in CDR loop and charge-based sampling and deseri-

alization. These techniques are verified using the measured results obtained from a 14Gb/s transceiver prototype. Transmitter achieves an energy efficiency of 0.89 mW/Gb/s while securing a 0.36 UI sampling time margin with BER = 10^{-12} at the end of the channel with 11 dB loss at Nyquist frequency. The receiver recovers sampling clock with 1.8 ps_{rms} long term absolute jitter while recovering 14 Gb/s data at BER = 10^{-12}. The receiver achieves an energy efficiency of 1.69 mW/Gb/s. Transmitter and receiver share an LC PLL, which achieves 0.605 ps_{rms} integrated jitter at 7 GHz output with an energy efficiency of 0.5 mW/GHz. The transceiver as a whole achieves an energy efficiency of 2.8 mW/Gb/s.

Acknowledgments

I thank Professor Pavan Hanumolu, my research advisor, for walking with me on this wonderful journey of research spread over six short years. I am grateful for the opportunity and freedom he has given me to explore and crystallize my research interests during the Ph.D. As a curious colleague and an enlightened advisor, he kept me inspired through all the seasons of learning. For a circuit designer, nothing could be more troubling than a dysfunctional chip designed over a long year, but rarely does any circuit work well at the first shot. In such awful moments of a non-working chip, he could easily pull me out with his insightful debugging schemes and focus on learning even with a failing chip. Learning from Professor Hanumolu seems a never ending curve whether it is research, teaching, writing, or presentation. According to him, my current drawings have improved 1000X over my first drawing, but there is a scope of 1000X more improvement. I hope I learn that too someday.

I still remember the first day of my graduate school and meeting with Professor Kartikeya Mayaram. The first thing he inquired was about my well being and settling in a new country. I would like to thank Professor Mayaram and Professor Terri Fiez for exposing me to different research fields and for the opportunity to work in their research group at Oregon State University. I am grateful to my professors for their invaluable suggestions and support, helping me pursue research in the field of my choice.

One thing which I should confess during my Ph.D. is my addiction to coffee with friends. Space and time were the only variables and friends for coffee seemed the rescue for every problem, personal or professional. I would like to elaborate on the coffee with friends because coffee trips deeply involved discussions of all kinds. In coffee trips, day-long problems could get solved, a new idea might spring from single discussion, a far-fetched idea could be materialized with inputs from everyone around, and round-the-globe information maintained

the sanity. I dare to confess the names of all accomplices for my coffee addiction and they include Romesh, Mrunmay, Woo-Seok, Guanghua, Seong-Joong, Ahmed, Charlie, Sachin, Reddy, Hariprasath, and Manideep. I would like to thank Brian for insightful discussions in the early morning hours and Tejasvi for sharing his research experience during health-conscious lunches. Apart from friends in my research group, I would like to thank Ankita and Deepak for being like family members throughout my Ph.D. Thank you all my friends for making my stay at school to be ever cherished in my life.

I would like to express my deepest gratitude to my parents, brother, and sister for their sacrifice, love, and support throughout this endeavor. Though I moved from a small town in India to a faraway place in United States for my Ph.D., my brother and sister-in-law residing in the US did not leave me feeling away from home. I would like to thank them for their love and friendship.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Computation and communication are fundamental functions in all modern day information processing systems. On one end, supercomputers which serve the high-end scientific community directly or large scale data centers which cater to the needs of millions of people worldwide, must possess enormous capability for computation and communication. On the other end of the performance spectrum, portable devices like laptops, mobile phones, etc., must provide adequate computing and communication performance with extreme power constraints. For several reasons, improved performance of all these devices at a lower and affordable cost is not only desirable but also the need of the hour to meet the growing demands of computation and communication. The primary bottleneck in improving performance of these devices is the power dissipation. As an example, it is worthwhile to examine the energy efficiency of the world's most powerful supercomputers.

Figure 1.1 shows the energy efficiency of the world's top supercomputers introduced over last seven years and which make it to the list of the top 500 Green Supercomputers [4]. Energy efficiency of a supercomputer is measured as energy spent in executing one floating point operation as shown along the y-axis. The computational capability of a supercomputer is measured as the number of floating-point operations per second (FLOPS) and shown along the x-axis. The plot illustrates three important things. First, computational capacity of supercomputers has increased over the years. Secondly, energy efficiency of supercomputers has become better over time. Last but not the least, large computational capability and low energy consumption is still a challenge as the highest performing supercomputers are not the most energy-efficient.

Roughly a decade back, when the petaFLOP supercomputer was introduced, it was estimated that one would need 1000X performance in the future, and efforts were made to

figure out what it would take to achieve a 1000X performance improvement [1]. To this end, the goal for high performance computing was set to achieve ExaFLOP operation with 20 MW power consumption, which leads to an energy efficiency of 20 pJ/FLOP as projected in Fig. 1.2. In a further step, advanced hardware and software techniques developed to achieve 20 pJ/FLOP energy efficiency would not be constrained to a single computational capability but energy efficiency should percolate up/down to benefit a wide range of energy-efficient operation. Figure 1.2 captures this wide range of energy-efficient operation which is the goal for exascale computing. Also, the energy-efficient operation should extend to a range of other applications not shown in the plot.

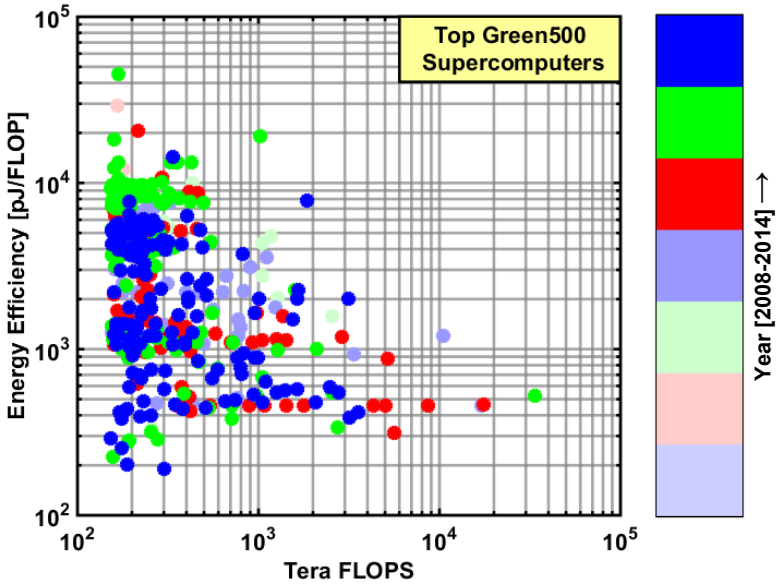


Figure 1.1: Energy efficiency of world's top 500 supercomputers versus their computational capability.

A feasibility study of such an exascale system was done after considering many technological advancements on both the hardware and software fronts [1]. Figure 1.3 shows the power breakdown of a hypothetical exascale system. The pie chart on the left shows that leakage power is going to be significant, amounting to 25% of the overall power consumption due to smaller dimensions in fine technology. FPU is the core computational floating point unit, which executes floating point operations and consumes 17% of the total power. A large memory is needed in such systems and it consumes 31% of the system power. A total of 31% memory power is distributed in L1 cache, L2/L3 cache, DRAM, disk, and register

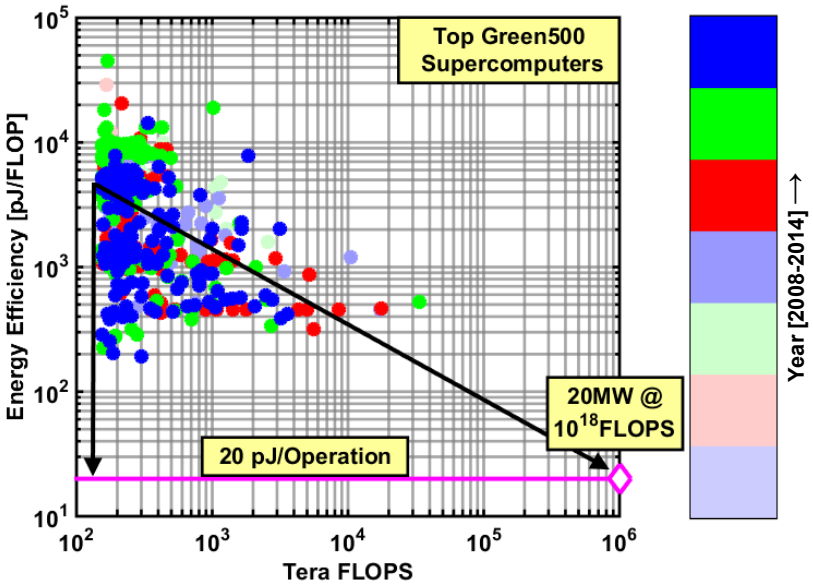


Figure 1.2: Goal for ubiquitous energy-efficient operation.

files with 39%, 16%, 3%, 14%, and 28%, respectively. The information or data which is to be processed is generated, stored, and used at different locations in a system. This data is transferred from one point to another through an interconnect. Depending on the location of source and destination, power dissipation in the data interconnect varies. In total, inter-

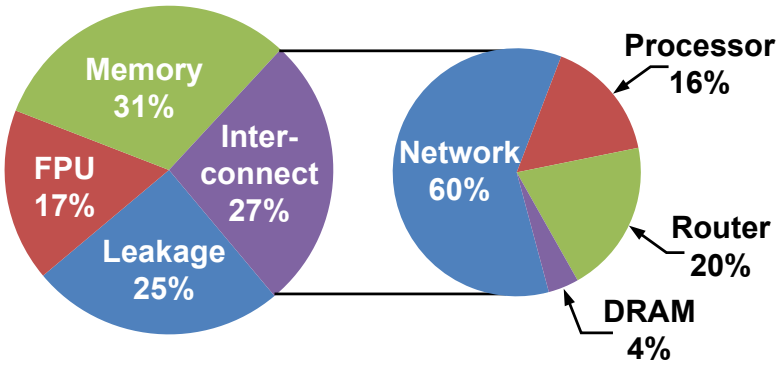


Figure 1.3: Power break down in a hypothetical exascale system [1].

connect power dissipation is expected to be 27% of total system power. Further breakdown of the interconnect power is shown in the pie chart on the right. It consists of power spent in data transfer on processor itself (16%), router (20%), DRAM (4%), and moving data from one rack to the other in a network. Here, the network includes data transfer between racks and has a dominant power consumption of 60% of the overall interconnect power. A data

interconnect is often called 'Serial Link' or 'SerDes' (serializer/deserializer). The focus of this work is on improving the interconnect energy-efficiency.

1.1 Basics of a Serial Link

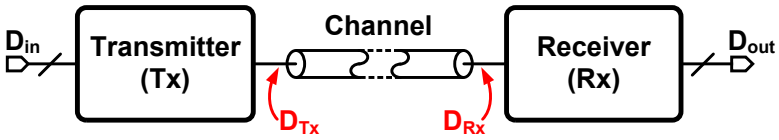


Figure 1.4: Simplified block diagram of a serial link.

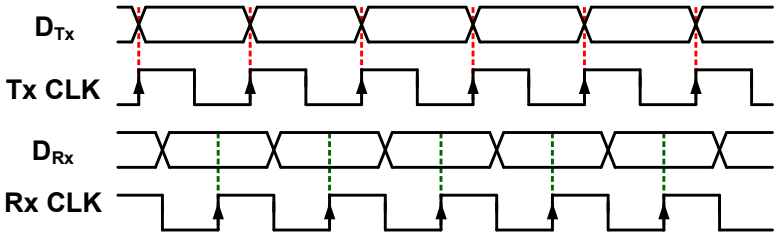


Figure 1.5: Data and clock signal waveforms in a serial link.

Figure 1.4 shows a simplified block diagram of a conventional serial link (data link). It consists of a transmitter (Tx) that modulates digital bits D_{in} with a pulse waveform into an analog signal D_{Tx} and couples it to the channel. The channel is a medium of transmission, which can be a metal trace, copper wire, an optical fiber, etc. On the other end of the channel is a receiver (Rx) block that samples the incoming signal D_{Rx} and recovers transmitted bits, ideally error-free. Figure 1.5 shows data/clock signal waveforms for the serial link. The transmitted data D_{Tx} is aligned with transmitter clock *Tx CLK*, and the receiver samples the received signal D_{Rx} in the center of each bit using clock *Rx CLK*. While the transmitter uses a local clock to send bits periodically, the receiver should know the data rate at which bits are transmitted, so that it can sample the received signal correctly. The clocking scheme on the Tx and Rx side in serial links distinguishes them from others in their usage.

Based on the clocking scheme on the Tx and Rx side, serial links can be classified into three categories: common clock I/O architecture, forwarded clock I/O architecture, and embedded clock I/O architecture. These clocking architectures are discussed in detail below.

1.1.1 Common Clock I/O Architecture

In common clock I/O architecture, Rx can use the same reference clock source for sampling as used during the transmission [5]. Figure 1.6 shows a block diagram of a common I/O clocking scheme. A single crystal is used as a reference for clock generation with phase locked loops (PLL) on both transmitter and receiver sides. A problem arises when on-chip and off-chip delays need to be matched for clock (*Tx CLK/Rx CLK*) and data paths (D_{Tx}/D_{Rx}) and it gets more critical for multi-Gb/s data interconnect. This scheme is limited to a few 100 Mb/s of data rate. Common I/O clocking has a problem in finding an optimum sampling clock phase on the Rx side given the mismatch in clock and data paths. It can be resolved if the clock used for transmitting data is also sent to Rx, which together formulate a forwarded clock architecture as discussed next.

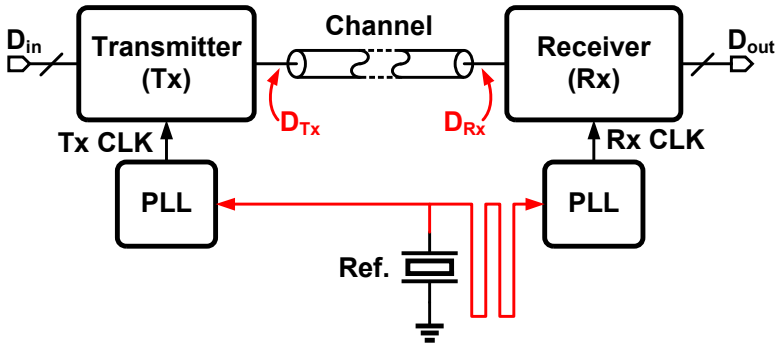


Figure 1.6: Block diagram of a serial link with common I/O clocking.

1.1.2 Forwarded Clock I/O Architecture

Figure 1.7 shows a block diagram of a transceiver with forwarded clock architecture that incorporates a dedicated channel to pass *Tx CLK* from Tx to Rx. The received clock is de-skewed adaptively to sample the received bit at an optimum location. The architecture comes with3 an extra cost of added channel for clock. However, this cost is negligible when amortized over multiple links operating in parallel at the same data rate. The forwarded clock technique is commonly used in processor-memory interfaces and multi-processor communication [6]. Forwarded clock architecture is limited to relatively low data rate short-reach

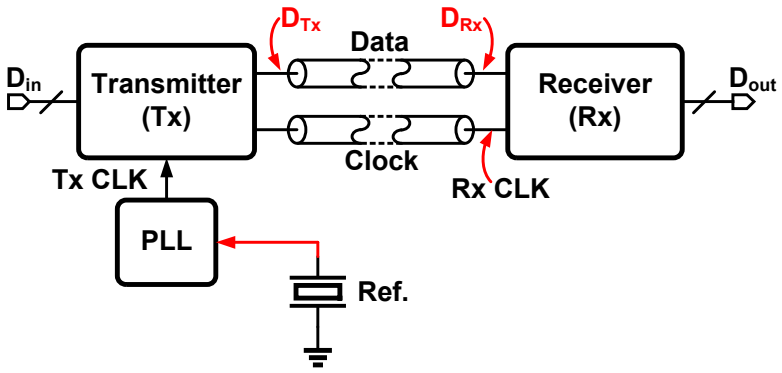


Figure 1.7: Block diagram of a serial link with forwarded clocking.

links like chip-to-chip communication on the same board. For long-reach links, embedded clocking is used as discussed next.

1.1.3 Embedded Clock I/O Architecture

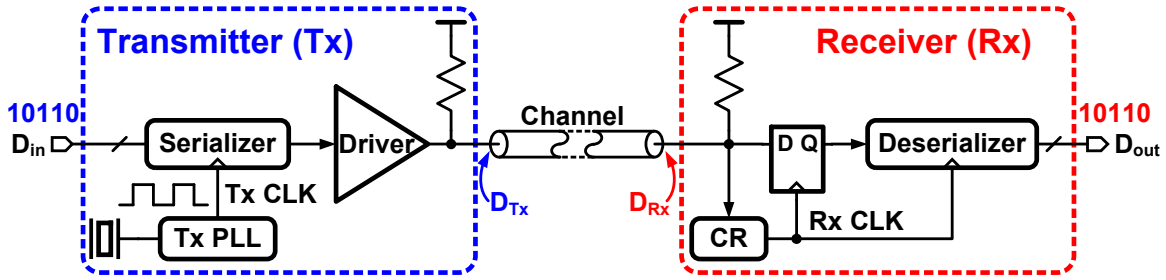


Figure 1.8: Block diagram of a serial link with embedded clocking.

Figure 1.8 shows a block diagram of a serial link with embedded clocking. Here, Tx uses a local clock generated by a phase locked loop (Tx PLL) to serialize parallel input bit streams (D_{in}) to a single high speed bit stream (D_{Tx}). On the Rx side, the sampling clock *Rx CLK* is recovered from the received data using a clock recovery (CR) loop, and the recovered clock samples the received signal D_{Rx} to recover transmitted bits correctly. The sampled data can be later deserialized for further use. The clock recovery loop needs extra power, but it recovers frequency and phase of the received data without any prior information and any additional channel. Embedded clocking is used in high speed mobile I/O (M-PHY), lossy backplane channels, optical fibers, etc.

1.2 Channel

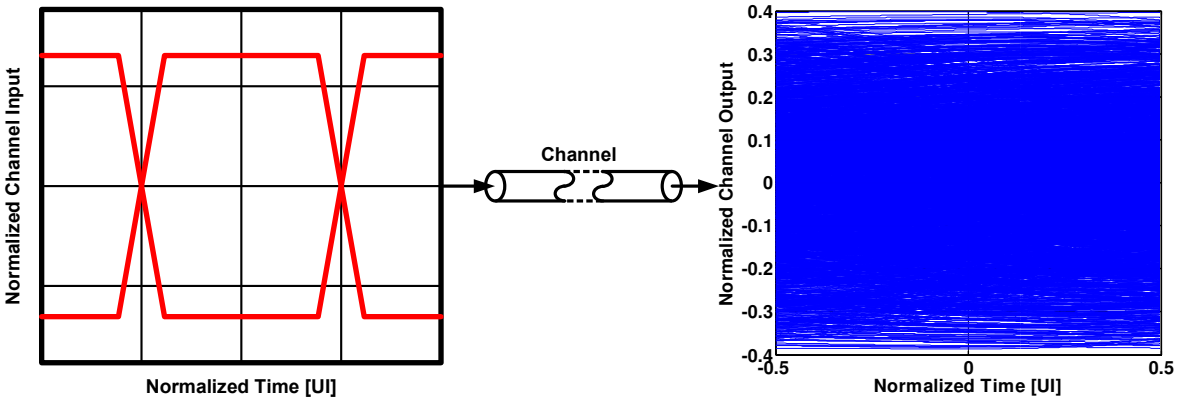


Figure 1.9: Eye diagrams at the input and output of a lossy channel.

It is not unfair to say that power consumption and complexity in serial links are very much dictated by the kind of channel. Starting from the transmitter output, everything which comes in the signal path till the Rx sampler input constitutes the channel. It includes bond wire, package parasitics, μ-stripline or stripline metal traces, SMA connectors, connecting cables, etc. A lossy channel attenuates, disperses, and distorts transmitted signal and makes it indistinguishable at the receiver end. Figure 1.9 shows typical eye diagrams at input and output of a lossy channel. A large channel loss leaves little or no timing margin to sample and recover data bits correctly. So, both Tx and Rx incorporate equalizers such as FFE (feed forward equalizer), CTLE (continuous time linear equalizer), and DFE (decision feedback equalizer) to compensate for channel anomalies.

1.3 Serial Link Performance Metrics

At the very top level, performance of serial links is compared with the following three metrics:

1. Data rate : Defined as number of bits transmitted/received per second.

2. Energy Efficiency : Defined as power consumed in Tx/Rx per unit data rate.

3. Channel Loss : Defined as power loss at Nyquist frequency. Though it is defined at a single frequency, channel characteristics throughout the transmit signal band are

important.

A high data rate per link or pin is beneficial to reduce board space. A lower power consumption per unit data transfer improves overall energy efficiency and enables more links in parallel for the same power budget. Channel loss plays a spoiler for high data rate and low energy efficiency. Since channel loss results in signal attenuation, distortion, and dispersion of transmitted signal, it demands additional energy to recover data bits in an error-free manner at the receiver.

Contrary to requirement, an abundant energy source for high data rate transmission across a lossy channel is not a solution. This is because thermal losses for interconnects become unbearable with the famous 'Power Wall' [7]. Rise in temperature is more detrimental for lower technology nodes, which are otherwise faster compared to slower and older technologies. One can try to dissipate heat at static processing centers with an unaffordable cost of cooling, but portable devices do not have that luxury either. Low power consumption per unit data rate (mW/Gb/s) is the only way forward to enhance computation and communication performance while keeping thermal dissipation within acceptable limits. Lower power consumption for higher data rate while compensating for higher loss has been a guiding factor for research in serial links in the past and will remain so in the future. The next section will review research in serial links over the last 15 years.

1.4 Research Trends in Serial Links

A research survey was carried out for serial link transceivers published by industry and academia in circuit conferences and journals. The survey includes 140 publications over the last 15 years. Figure 1.10 shows the data rate for transceivers with year of publication. In the last 15 years, data rates have steadily increased for both forwarded and embedded clock architectures. Advancement in silicon technology with finer dimensions is one big reason for the increase in data rates [8]. Figure 1.11 shows data rate versus CMOS process node. The figure shows a clear trend of increasing data rate over lower process nodes.

It is to be noted that the same peak data rate (e.g. data rate = 40Gb/s) in serial links is

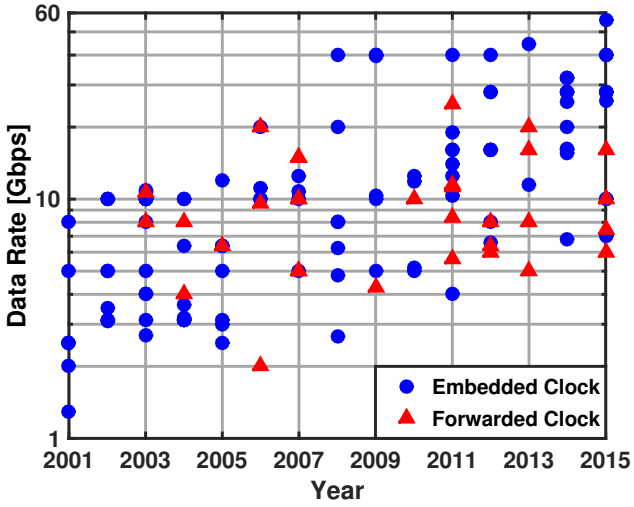


Figure 1.10: Data rate for serial link versus year of publication.

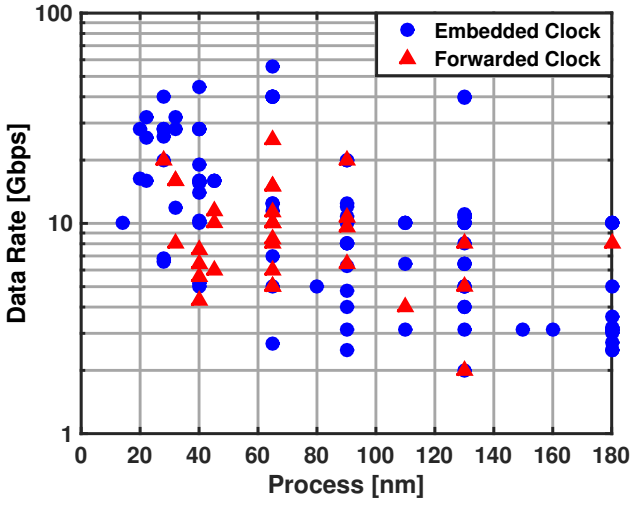


Figure 1.11: Data rate for serial link versus CMOS technology node.

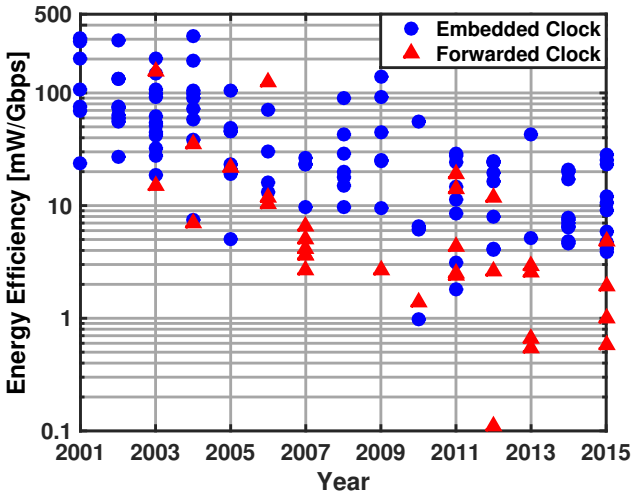


Figure 1.12: Energy efficiency for serial link versus year of publication.

demonstrated in different process nodes. This observation indicates that technology itself is not the limiting factor to achieve high data rate. And the differences between the two designs lie in the energy efficiency of such transceivers which benefit from architectural or circuit level innovations, finer technology, or lower channel loss. Figure 1.12 depicts energy efficiency of serial links versus year of publication. A clear trend of decreasing energy efficiency over the years is due to the combined effort at architecture and circuit level innovations and technological advancement. An important observation is that the energy efficiency of forwarded clock architectures is generally lower than that of embedded clock architectures. This is because of the difficulty in extracting clock from random data transmitted across lossy channels in embedded architectures.

Since the goal of research in serial links is to facilitate higher data rate and lower energy consumption, it is worthwhile to look at energy efficiency versus data rate as shown in Fig. 1.13. Data rates for forwarded clock architecture are limited, and the architecture achieves lower energy efficiency compared to embedded links. For embedded architectures, a roughly constant energy efficiency in the range of 4-30 pJ/bit, or energy efficiency increasing with data rates, is due to added power penalty for systems operating at high data rates with increased channel loss. Figure 1.14 shows energy efficiency for serial links versus channel loss at Nyquist frequency. Energy efficiency markers at 0 dB channel loss are publications reporting channel loss that is either negligible or unavailable. Most of the forwarded link

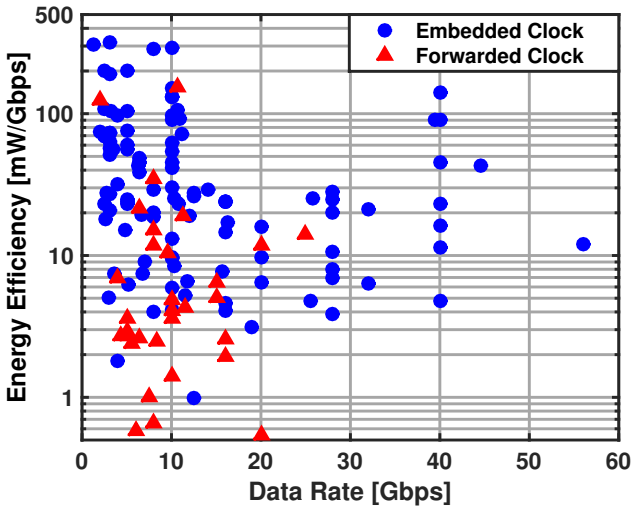


Figure 1.13: Energy efficiency versus data rate for serial links.

architectures stand out with low energy efficiency at a given channel loss when compared to embedded links because of easier data recovery with forwarded clock. Embedded links exhibit high energy efficiency at higher data rate with increased channel loss. While silicon technology is going to hit the limit of scaling soon, innovations at the architecture and circuit levels is the only way forward to improve energy efficiency at higher data rates.

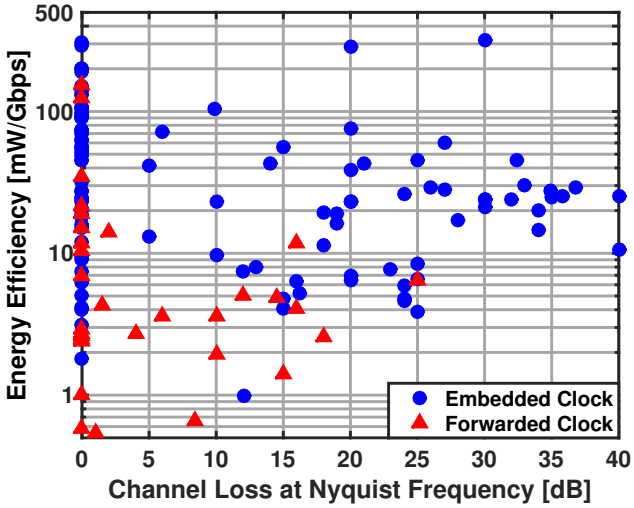


Figure 1.14: Energy efficiency versus channel loss for serial links.

The trends described above make a compelling case for research in embedded serial links to increase data rates and reduce energy consumption even in the presence of increased channel loss. This work focuses on innovating at the architectural and circuit levels to

improve energy efficiency of high speed serial links that employ embedded clock architectures. This thesis is organized as follows: In Chapter 2, basic building blocks of a serial link are reviewed, and potential problems and avenues for improving link energy efficiency are identified. In Chapter 3, a 5 Gb/s voltage-mode transmitter embedded with time-based de-emphasis is implemented in 90 nm CMOS process. Time-based de-emphasis overcomes the tradeoff between impedance matching, output swing, and de-emphasis resolution in conventional voltage-based de-emphasis. In Chapter 4, a 14 Gb/s transceiver prototype is implemented in a 65 nm CMOS process. The transmitter employs a novel partially segmented 2-tap FIR voltage-mode output driver to lower power consumption. The receiver implements a low power half-rate clock and data recovery using the proposed ring PLL based multi-phase sampling clock generation and charge-based sampling and deserialization. In Chapter 5, the above serial link projects are summarized and the thesis is concluded.

Chapter 2

Overview of Serial Links

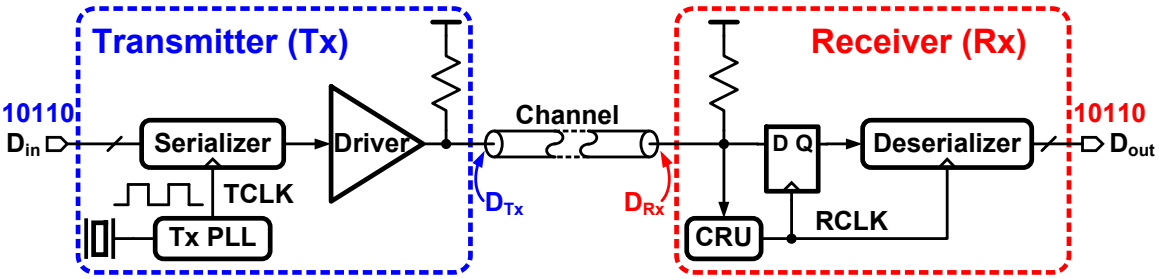


Figure 2.1: Block diagram of a serial link with embedded clocking.

Figure 2.1 shows block diagram of a serial link with embedded clocking. On the Tx side, a locally generated clock from Tx PLL is used to serialize N parallel low frequency bit streams (D_{in}) to full-rate data through serializer. An output driver performs equalization and couples full-rate data D_{Tx} to the channel. On the Rx side, the sampling clock is recovered from the received signal through a clock recovery unit (CRU) and data is recovered by sampling the received signal using the recovered clock *RCLK*. The recovered data is deserialized to N parallel low rate bit streams, D_{out}. Different building blocks of a conventional serial link are discussed below.

2.1 Transmitter

Figure 2.2 shows a simplified block diagram of a transmitter. It consists of a serializer, pre-driver, output (O/P) driver, and Tx PLL. The output driver, pre-driver, and last stage of the serializer process data signals at maximum speed and are power hungry blocks on Tx side. O/P driver consumes significant power which is proportional to channel loss in the serial link. Power dissipation in pre-drivers depends on O/P driver topology and can be

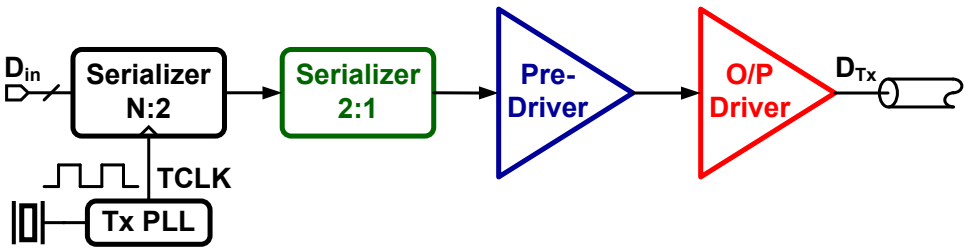


Figure 2.2: Block diagram of a transmitter.

a significant portion. Jitter requirements on the clock used for serialization guide Tx PLL design, and it can be sufficiently low power consuming compared to other blocks. In the end, it is the output driver design with embedded equalization which determines Tx design for energy-efficient operation. Characteristics of different output drivers are discussed below.

2.1.1 Transmitter Output Driver

The Tx output driver should possess three main features. First, it should have a fixed output impedance that is matched to the channel characteristic impedance to guarantee signal integrity. Impedance matching ideally should be independent of output signal swing. Second, output drivers must be amenable for embedding de-emphasis-based equalization without altering their output impedance. Finally, it is important to have control over the magnitude of output swing to optimize signaling power as a function of channel loss. Conventionally, output drivers can be categorized into current mode logic (CML) based output drivers and voltage mode (VM) output drivers. CML-based and VM O/P drivers are discussed in detail below.

2.1.1.1 Current Mode Logic (CML) based Output Driver

A current mode logic (CML) based output driver shown in Fig. 2.3 has all the desirable features as mentioned before. The output impedance is equal to R_l, which can be chosen to match the channel characteristic impedance, R_T. Large output impedance of the input pair makes impedance matching nearly independent of output signal swing. It is straightforward to control the output swing by varying the tail current source I_0. For feedforward

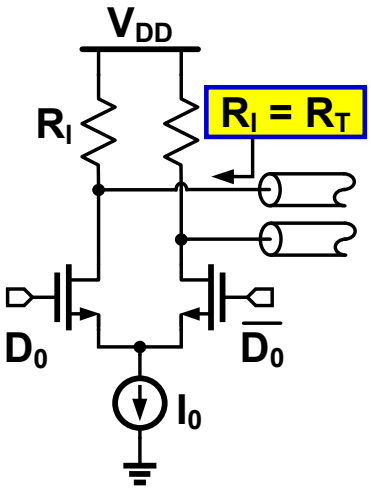


Figure 2.3: Current mode logic (CML) based output driver.

equalization, de-emphasis can be implemented by simply splitting the main current source according to equalizer weights as shown in Fig. 2.4. A major disadvantage of the CML driver is its large current consumption, which is equal to $V_{d,pp}/R_T$, where $V_{d,pp}$ is the peak-to-peak differential output swing. In contrast, voltage mode output drivers result in energy-efficient operation as discussed next.

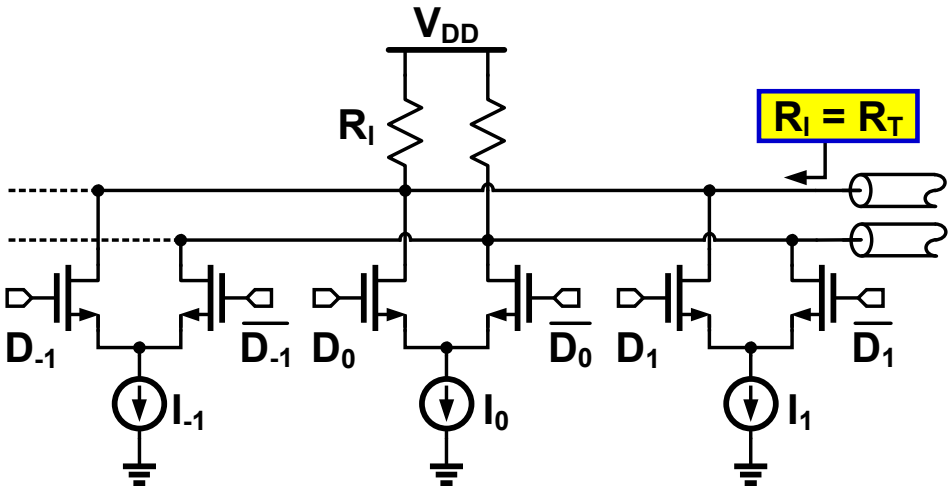


Figure 2.4: CML driver with de-emphasis.

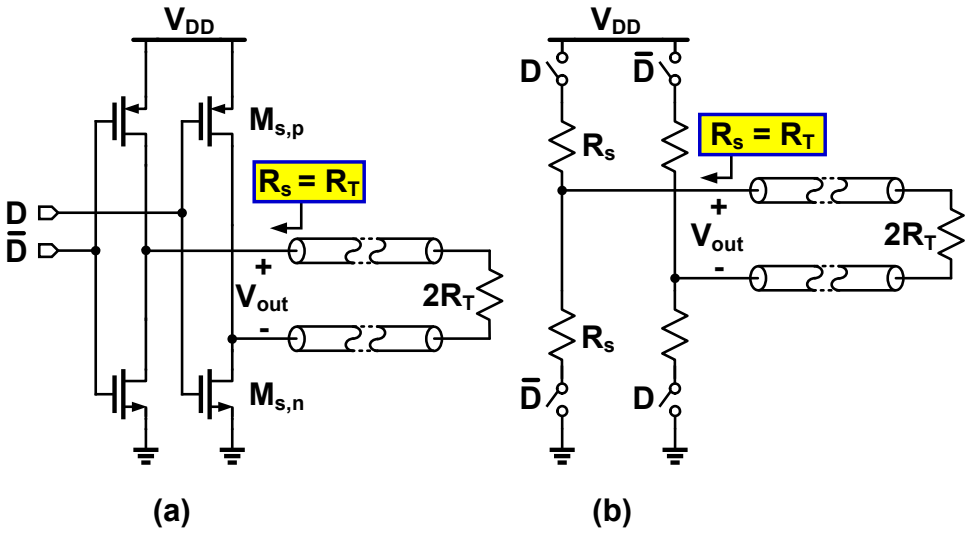


Figure 2.5: (a) Voltage mode O/P driver. (b) Equivalent model of an ideal VM O/P driver.

2.1.1.2 Voltage Mode (VM) Output Driver

A voltage mode (VM) output driver shown in Fig. 2.5(a) is an attractive alternative to the CML driver because it can be ideally 4X more power efficient [9]. A model of the VM O/P driver is shown in Fig. 2.5(b), where R_s models the ON state resistance of the driver switches, $M_{s,p}/M_{s,n}$. Transistors $M_{s,p}/M_{s,n}$ are biased and sized to match the channel characteristic impedance, R_T. Output swing is controlled by the driver supply voltage V_{DD} and it consumes current $V_{d,pp}/4R_T$ for differential peak-to-peak swing, $V_{d,pp}$. While the power efficiency benefit of the VM driver is appealing, it is not particularly suited for de-emphasis.

Figure 2.6(a) shows a conventional differential VM driver in which de-emphasis is implemented by changing the output voltage level using a resistive divider [10]. A resistive voltage division is created by switching resistors in branches R_s and R_d such that the modified driver output impedance, $R_s||R_d$, is equal to the channel characteristic impedance R_T. While resistive divider provides variable voltage for equalization, the current consumption increases ($\geq V_{DD}/4R_T$) due to low impedance path from supply to ground. Furthermore, the current consumption also depends on output swing and is maximum ($V_{d,pp}/2R_T$) at minimum output swing. In [11], the signaling current is held constant at $V_{DD}/4R_T$ by using a shunt resistance R_{shunt} (realized using transistor, M_{shunt}) across the differential output port, as shown in Fig. 2.6(a). However, if the resolution of equalization were not to be compromised,

the non-linear dependence of resistors R_s, R_d, and R_{shunt} on the output voltage incurs large power penalty in pre-drivers.

Combining the signaling power efficiency of the previous scheme with direct tuning of the driver output impedance [12], authors in [2] introduced a VM driver shown in Fig. 2.6(b). Here, the output driver impedance is tuned by controlling the gate voltage of the driver switches. A 2-tap equalization is implemented by switching R_s and R_{shunt} based on precoded data (D_0, D_{-1}). The signaling power scales up with the output swing unlike the previous architectures and it is minimum for minimum output swing. Nevertheless, in all the above cases, the output driver impedance realized by switching MOSFET is tightly coupled with the output voltage swing and the amount of de-emphasis. And the resolution of de-emphasis tap weight is inversely proportional to the power penalty in pre-drivers [2].

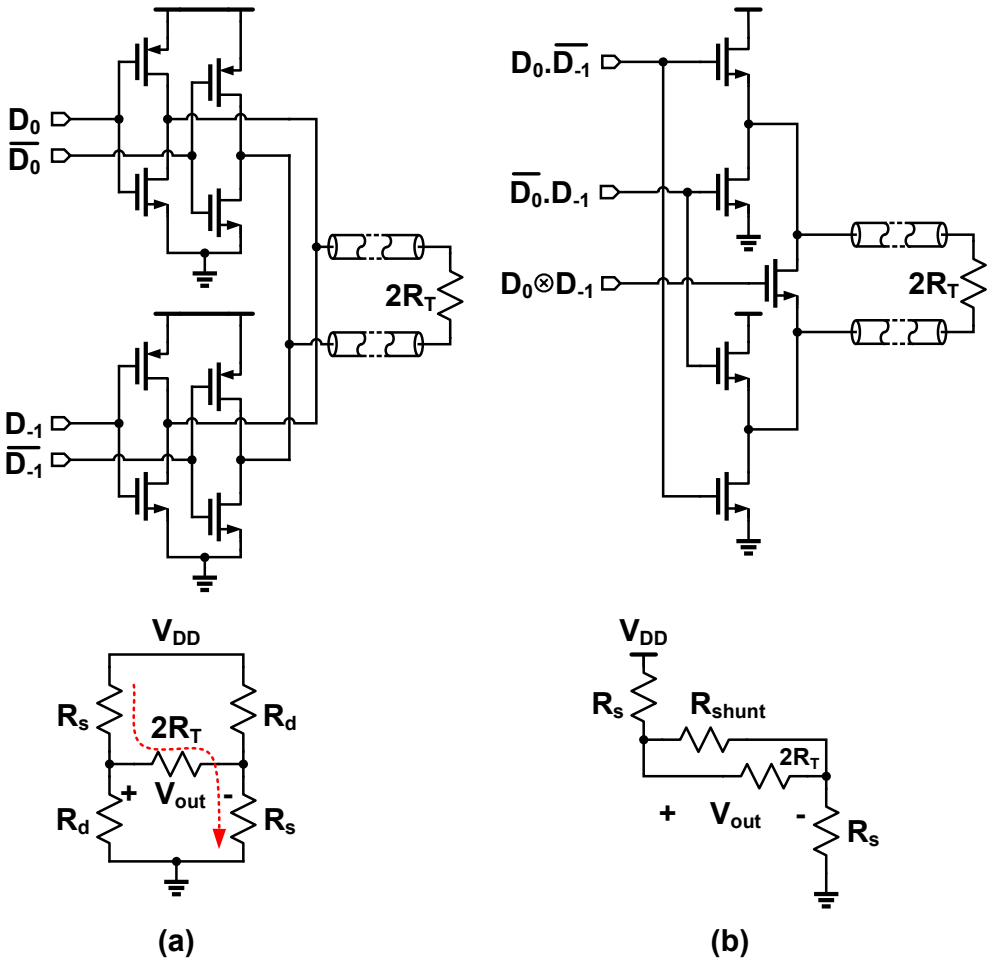


Figure 2.6: (a) Conventional VM driver with de-emphasis, and (b) VM driver in [2].

In view of the above conflicting tradeoffs, we seek to use time-based techniques to decouple de-emphasis from impedance control and output swing. To this end, we propose to drive a conventional VM driver with a two-level pulse width modulated (PWM) data stream to achieve de-emphasis without compromising the VM driver power efficiency. An overview of PWM-based de-emphasis is presented first in Chapter 3 and the impact of clock jitter with multiple transition edges during equalization is carefully analyzed. Later in the chapter, we implement a time-based 5 Gb/s voltage mode transmitter equalizing for channel loss > 15 dB.

2.2 Receiver

A receiver recovers data bits by sampling the received signal D_{Rx} using recovered clock *RCLK* (Fig. 2.1). The received signal is attenuated and distorted due to channel loss along the transmission path. Figure 2.7 shows typical eye diagram of a received signal D_{Rx}. Here, a non-return-to-zero (NRZ) pulse waveform is used to modulate transmit bits. The peak-to-peak received signal swing is V_{pp} and each bit is T period wide. A sampler in the Rx front-end compares the received signal with an ideal threshold voltage $V_{th} = 0$ V, for differential signals, and detects 1/0 for transmitted bits. However, any real comparator circuit has a lower limit on the minimum voltage level to be detected, which is defined as the sensitivity of the receiver. High sensitivity to received input at the Rx front-end enables lower transmit swing and less power consumption on the Tx side. The sampling circuit often suffers from process dependent statistical mismatch in differential paths, which adds a fixed voltage offset to the threshold level. Therefore, the effective voltage sampling margin of the sampler is reduced by V_m near the ideal threshold level. Voltage offset is usually compensated for in samplers but at the expense of increased power. Sensitivity of a sampler can be improved by using a high gain regenerative sense-amplifier based flip-flop [13]. Power consumption in data samplers is significant as they operate at maximum speed.

For sampling data, the recovered clock *RCLK* has to be aligned ideally with the center of the received signal, which has maximum voltage margin during sampling. Zero crossing variation (timing jitter) in received signal and sampling clock both reduce the sampling time

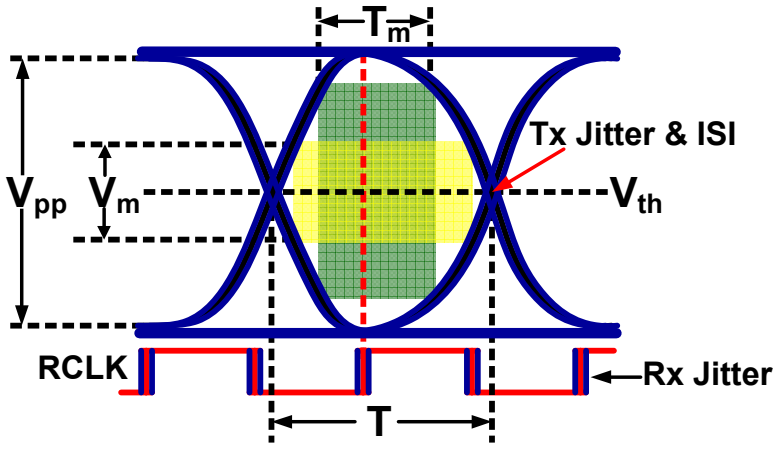


Figure 2.7: Typical eye diagram of a received signal.

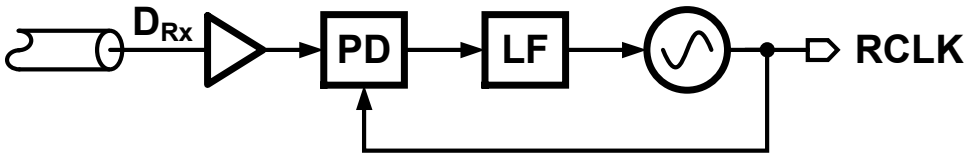


Figure 2.8: Block diagram of a clock recovery unit (CRU).

margin to $T_m \leq T$. Time domain jitter on *RCLK* and voltage sensitivity of the sampler decide correctness of detected bits. The correctness of sampled data is measured in terms of bit error rate (BER) and it requires a BER$< 10^{-12}$ for most serial link applications. Assuming a white Gaussian voltage noise ($N(0,\sigma_{\text{vnoise}})$) added to received voltage swing before sampling, a BER $= 10^{-12}$ requires an additional voltage margin of $\pm 7\sigma_{\text{vnoise}}$ at the input of sampler. Similarly, it requires a sampling time margin of $\pm 7\sigma_{\text{jitter}}$ for timing uncertainty σ_{jitter} in the sampling clock.

Sampling gets complicated when transitions in the received signal vary due to jitter on Tx side and intersymbol interference (ISI) introduced by channel. Hence, it is required that sampling clock *RCLK* should track the received signal in frequency and phase and this is referred to as clock recovery on the Rx side. Basics of clock recovery are discussed next.

2.2.1 Rx Clock Recovery

Figure 2.8 shows a block diagram of a typical clock recovery unit on the Rx side. It consists of phase error detector (PD), loop filter (LF), and an oscillator. PD detects phase error or

timing error between transitions in sampling clock and received data. Phase error, Φ_{err}, is filtered using analog or digital loop filter, LF. Loop filter output tunes the oscillator, thereby controlling the sampling phase and frequency. Analog or digital processing in the loop mostly depends on the phase error detector, whose output is analog or digital. Two commonly used phase detection methods between sampling clock and random data are: (a) linear (Hogge) phase detector, and (b) bang-bang (Alexander) phase detector. The two phase detection methods are discussed below in more detail.

2.2.1.1 Hogge (linear) Phase Detector

Figure 2.9(a) shows a block diagram of a Hogge phase detector. It consists of two D flip-flops (DFF) and two XOR gates. First DFF samples random data D at rising edge of clock Φ as shown in Fig. 2.9(b). Phase difference between data and clock is encoded as pulse width (PW) of XOR output, *Late(L)*. The second DFF generates a pulse (*Early(E)*) of half the clock period every time data makes a transition. The difference between pulse widths of two modulated signals, $PW_L - PW_E$ is the phase error between sampling clock and received data. In Fig. 2.9(b), the clock sampling edge is left of center w.r.t. random data. Hence, pulse width difference $\text{PW}_\text{E} - \text{PW}_\text{L} > 0$ and signifies that the sampling clock is early w.r.t. its ideal sampling location at the center. Similar phase detector operation holds for the sampling clock located right of center with $PW_L - PW_E < 0$. The evaluated phase error is later used to change the sampling clock phase in a feedback loop. Ideally, linear phase detectors quantify phase error exactly. However, they have a few drawbacks. At high data rates, the data bit period becomes comparable to circuit delays, and any delay mismatch in signal paths, like $\text{D} \rightarrow \text{L}$ and $\text{X} \rightarrow \text{L}$, is a cause of unintended phase error. Secondly, XOR gates suffer from limited circuit bandwidth while generating pulse width modulated output higher than data rates. In addition, the phase detector output is an analog signal and cannot benefit from low power and smaller area digital filtering in the feedback loop. A digital phase detector overcomes some of these issues as discussed next.

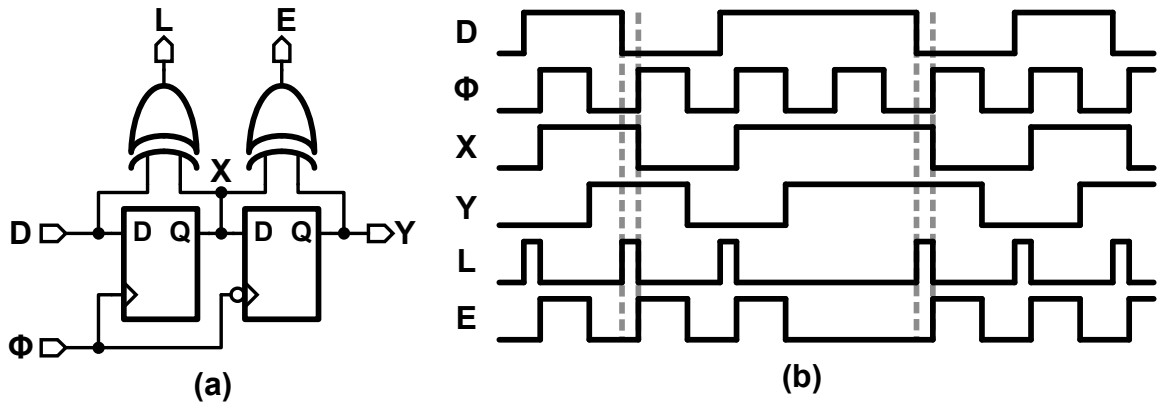


Figure 2.9: (a) Block diagram of Hogge (Linear) phase detector. (b) Signal waveforms in Hogge phase detector.

2.2.1.2 Alexander (bang-bang) Phase Detector (!!PD)

Unlike linear phase detectors, an Alexander phase detector (Fig. 2.10(a)) detects sign of phase error. The phase detector samples at both positive edge (data sample, D_n) and negative edge (edge sample, E_n) of clock. These data and edge samples are combined by a combinational logic to detect early or late. Figure 2.10(b) shows sampling waveforms with clock rising edge occurring earlier than the center of the data bit. Here, D_n and D_{n+1} are different but D_n and E_n are the same. After the combinational logic, *Early* signal evaluates HIGH and *Late* signal will be zero. *Early* and *Late* signals will be vice-versa for clock sampling edge placed to the right of the center of the data bit (Figure 2.10(c)). The phase detector outputs *Early/Late* are both zero for no data transitions. The signed output (HIGH/LOW) of the phase error detector is later processed digitally to correct sampling phase in a feedback loop. Bang-bang phase detectors are easy to use at high data rates and are not limited by pulses narrower than a bit period. However, !!PD output has a quantization error unlike the linear phase error detector.

In !!PD, data and edge samplers operate at maximum speed and consume a lot of power. Cascaded sense-amplifier based flip-flops with large non-linear gain and high voltage sensitivity are used for data/edge sampling. Power consumption in these samplers is dynamic in nature and scales linearly with data rate. Charge sharing based front-end sampling in [3] consumes lesser power compared to sense-amplifier based flip-flop. Chapter 4 elaborates on already existing sampler designs and introduces a novel energy-efficient charge-based Rx

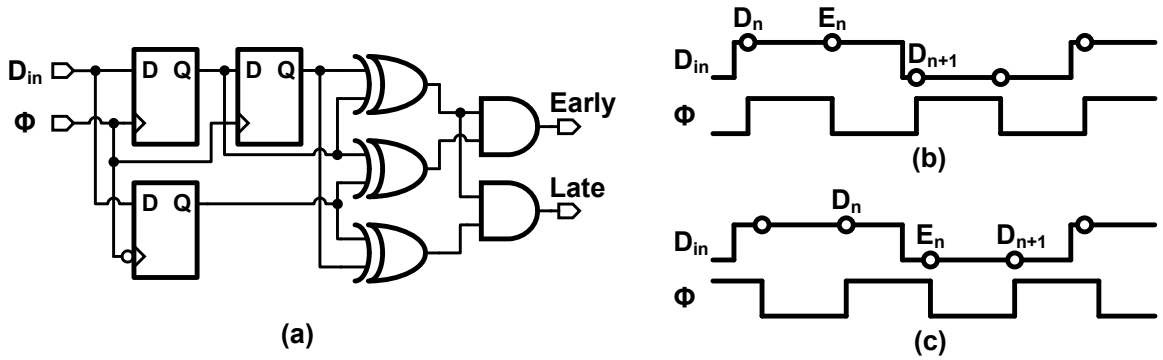


Figure 2.10: (a) Block diagram of Alexander (bang-bang) phase detector. Signal waveforms for sampling with (b) early clock, and (c) late clock.

front-end.

It is to be noted that the underlying principle of phase error detection is sampling each bit at bit rate in linear phase detectors and twice the bit rate in bang-bang phase detectors. Since power consumption in CMOS circuits is directly proportional to frequency of operation, sub-rate phase detector architectures operating at lower frequencies have often been used. The basic idea of sub-rate architectures is to sample input with multiple phases of a lower frequency clock in place of using a single-phase high-frequency clock. The number of samplers increases in proportion to additional clock phases. Reducing the clock speed and using sub-rate phase detectors is not always a winning option for two reasons. First, the speed of samplers is reduced but the number of samplers has increased. The total power consumption for more samplers operating at sub-rate may not be always less than that for fewer samplers operating at full-rate clock. Secondly, generating and routing more sampling clock phases at lower rate can be more power hungry than doing the same for a full-rate clock. Generating low jitter clock and tracking multiple clock phases for sampling are two different problems in clock and data recovery (CDR) loop. Based on the application, a low jitter clock and its sampling phases are generated and tracked in different kinds of clock and data recovery loops. Chapter 4 discusses power consumption in conventional clock and data recovery loops in more detail. Later in the chapter, we propose a low power and low jitter clock and data recovery loop for half-rate CDR architecture.

Chapter 3

An Energy-Efficient Voltage-Mode Tx Using PWM-Based De-Emphasis

In this chapter, we propose a time-based equalization technique to implement de-emphasis in voltage-mode drivers. In section 3.1, a time-based equalization scheme is examined analytically while compensating for loss in backplane channels. In Section 3.2, incorporating the time-based de-emphasis, a 5 Gb/s voltage-mode transmitter is implemented. Section 3.3 shows the efficacy of PWM-based equalization with measurement results for implemented VM Tx prototype. Finally, this project is concluded in Section 3.4.

3.1 Pulse Width Modulation Based De-emphasis

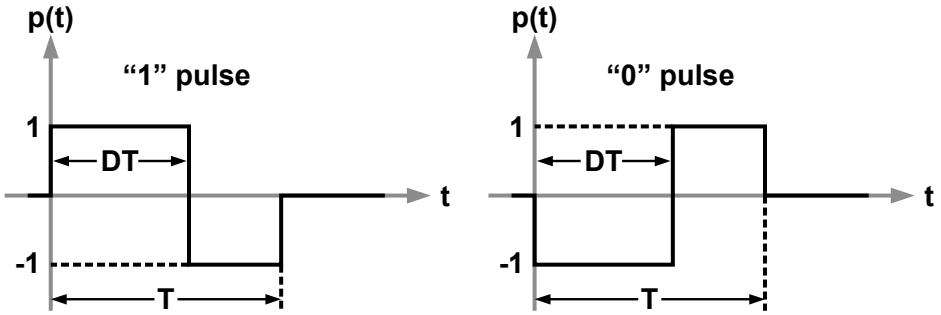


Figure 3.1: Representation of data bits "1" and "0" using PWM pulses with duty cycle D.

A pulse width modulated pulse (PWM) waveform (Fig. 3.1) with duty cycle, D, can be expressed as:

$$p_{pwm}(t) = u(t) - 2u(t - DT) + u(t - T) \tag{3.1}$$

where T is the bit period, $0.5 \leq \mathrm{D} \leq 1$ is the duty cycle of the pulse waveform, and $u(t)$ is a step function. The PWM pulse takes on only two discrete output levels ± 1 and reduces to a non-return-to-zero (NRZ) pulse and a Manchester encoded pulse when D=1 and D=0.5,

respectively.

It was shown in [14] that intersymbol interference (ISI) caused by channel loss can be suppressed by choosing D between 0.5 and 1 appropriately when PWM pulse is transmitted. The impact of duty cycle of the PWM pulse on ISI can be visualized both in the time and frequency domains. In time domain, consider the normalized channel response to a PWM pulse, shown in Fig. 3.2 for three different duty cycle conditions. The channel loss at Nyquist frequency is about 28 dB. Compared to D=1 (unequalized NRZ), D=0.6 pulse response exhibits significantly less ISI, thus illustrating that ISI can be minimized by choosing D optimally.

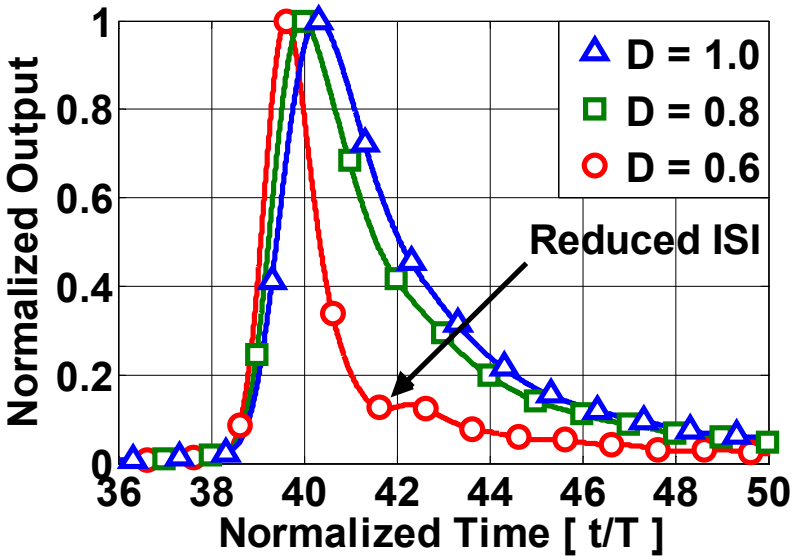


Figure 3.2: Normalized time domain response of a channel to PWM pulses.

This behavior can also be viewed in frequency domain. The power spectral density of the PWM pulse is calculated to be:

$$|S_{PWM}(f)| = \frac{1}{\pi f}\sqrt{1+\cos(\pi fT)\left[\cos(\pi fT) - 2\cos((2D-1)\pi fT)\right]} \tag{3.2}$$

Plotting the power spectral density $|S_{PWM}(f)|$ for different duty ratios, as shown in Fig. 3.3, illustrates frequency de-emphasis. Interestingly, the de-emphasis extends beyond the Nyquist frequency in contrast to a conventional n-tap finite impulse response (FIR) filter approach, which de-emphasizes frequencies only below the Nyquist frequency. As a result, PWM-

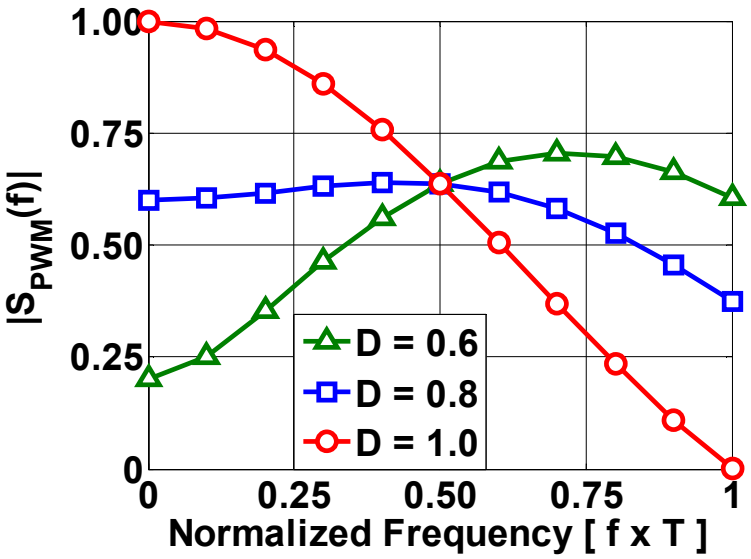


Figure 3.3: Power spectral density of PWM pulses with different duty cycles.

based de-emphasis exhibits superior ISI suppression. The equalization effectiveness can be quantified by evaluating its response to a bit sequence that generates the largest amount of ISI. The maximum distortion to a received pulse can be evaluated from a single pulse response with peak distortion analysis as discussed below.

3.1.1 Peak Distortion Analysis for PWM-based Equalization

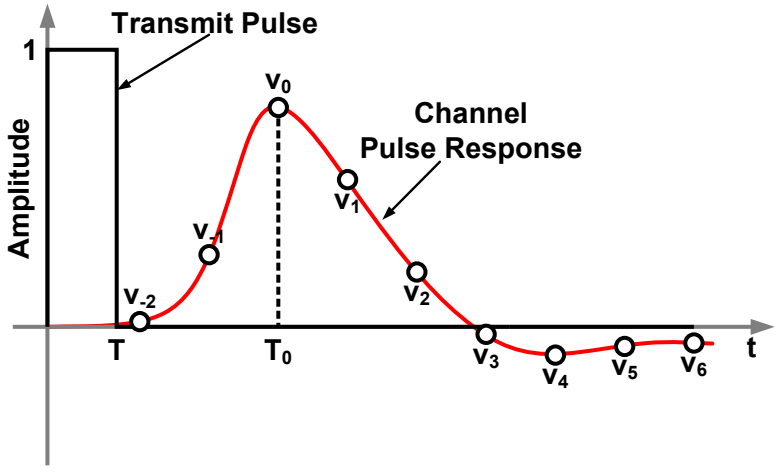


Figure 3.4: Pulse response of a channel for NRZ input pulse.

Figure 3.4 shows a typical pulse response of a channel for an NRZ input pulse. Sampling

instant t=T_0 with peak sampled voltage value v_0 is an ideal sampling location for the bit detection. Voltage values v_n sampled at $t = T_0 + nT$ where $n \in Z$ and $n \neq 0$, contribute to ISI. These samples can collectively reduce the peak sampled value with a bit sequence for the worst case ISI [15]. Similarly, for each time instant $t = T_0 + \Delta$ of one bit period where $-T/2 \leq \Delta < T/2$, the worst case bit sequence leads to the minimum horizontal eye opening due to ISI. The worst case vertical and horizontal eye openings serve as a metric for effectiveness in canceling ISI.

The worst case vertical and horizontal eye openings are calculated with PWM-based equalization for two different channel loss profiles having 24 dB and 28 dB loss at Nyquist frequency, respectively. Figure 3.5 and Figure 3.6 show the worst vertical and horizontal eye openings with varying duty cycle ratio for the two channels, respectively. It is to be noted that the eye was completely closed without de-emphasis (D=1). With an optimum duty cycle ratio, maximum vertical/horizontal eye openings of 5.9%/0.83 UI and 3.4%/0.69 UI can be obtained with channel losses of 24 dB and 28 dB, respectively.

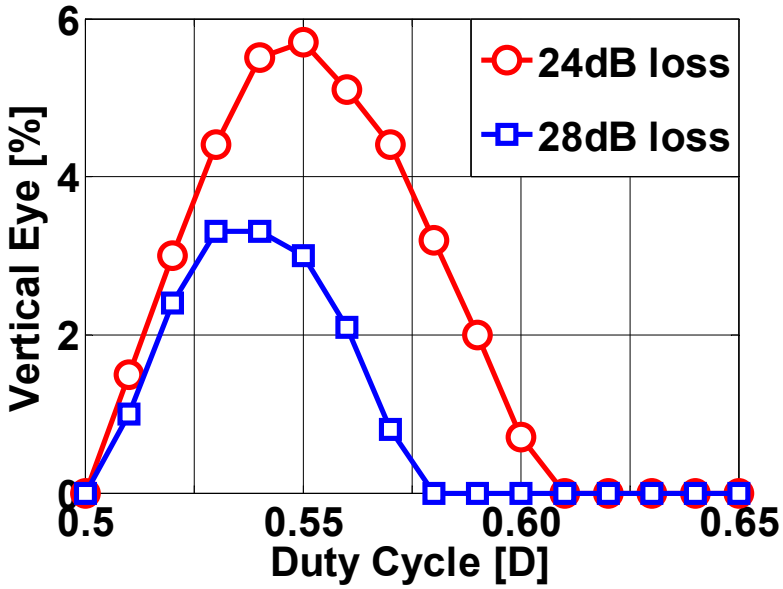


Figure 3.5: Worst case vertical eye opening with peak distortion analysis of PWM-based de-emphasis for two different channels.

These plots also illustrate a duty cycle tuning range to cover the desired channel loss. An optimum duty cycle for PWM-based equalization depends on the channel loss profile. As channel loss decreases, the duty cycle increases, which results in narrower pulses. This

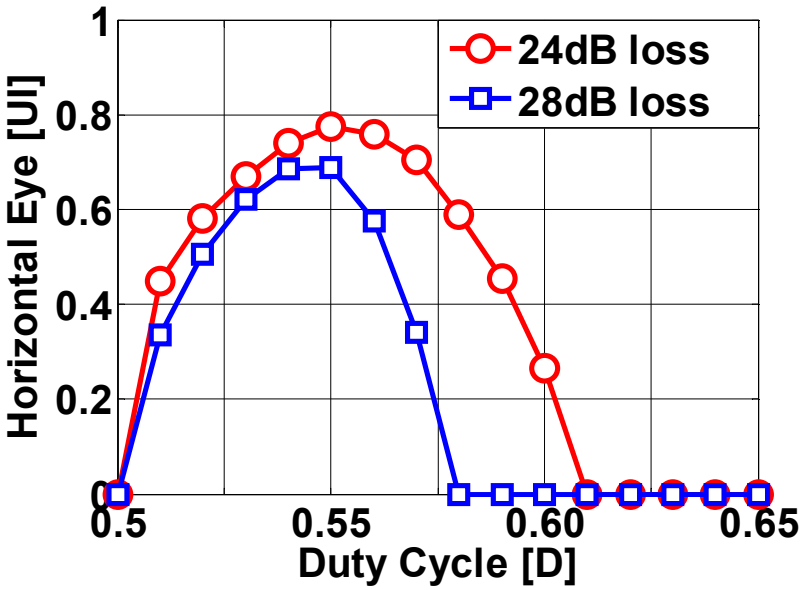


Figure 3.6: Worst case horizontal eye opening with peak distortion analysis of PWM-based de-emphasis for two different channels.

aggravates the pulse generation problem for higher data rates and low-loss channels. Effectiveness of PWM equalization at higher data rates will be technology limited due to the need for generating narrow pulses.

Unlike conventional voltage-based equalization where output voltage swing varies with the number of equalizer taps [16], PWM-based equalization transmits only two voltage levels (± 1), making it ideally suited for implementing de-emphasis in a voltage mode driver. Specifically, PWM-based de-emphasis offers the following advantages. First, it eliminates the non-linear dependence of driver output impedance on varying output swing present in voltage-mode de-emphasis. Second, it decouples termination impedance from both the amount of de-emphasis and output swing. In other words, driver output impedance can be set to match the channel characteristic impedance, independent of de-emphasis magnitude and resolution.

However, clock jitter has a detrimental impact on transmitter performance regardless of the signaling format (NRZ or PWM). Channel attenuation further exacerbates this due to the so-called jitter amplification effect [17]. To quantify the sensitivity to clock jitter, peak distortion analysis of PWM-based de-emphasis in the presence of jitter is performed in the next section.

3.1.2 PWM-based De-emphasis in presence of Jitter

The PWM pulse modulates the input bit sequence $...b_{k-1}, b_k, b_{k+1}, ...$ to transmit a signal given by

$$y(t) = \sum_k (b_k + b_{k-1})u(t - kT) - 2\sum_m b_m u(t - \overline{m + D}T) \tag{3.3}$$

There are two edge transitions every bit period and jitter affects both edges corrupting the final duty cycle D. In the presence of jitter $j(kT)$ at bit transition edge and $j(\overline{m + D}T)$ at duty cycled edge, the jittered transmit signal is $y_j(t)$ (eq (3.4)). Under Taylor series expansion and approximation $|j(kT)| \ll T$ and $|j(\overline{m + D}T)| \ll T$, $y_j(t)$ can be simplified as

$$\begin{aligned} y_j(t) &= \sum_k (b_k + b_{k-1})u(t - kT + j(kT)) - 2\sum_m b_m u(t - \overline{m + D}T + j(\overline{m + D}T)) \\ &\approx y(t) + \sum_k (b_k + b_{k-1})j(kT)\delta(t - kT) - 2\sum_m b_m j(\overline{m + D}T)\delta(t - \overline{m + D}T) \end{aligned} \tag{3.4}$$

where $\delta(t) = 1$ for $t = 0$ and $\delta(t) = 0$ for $t \neq 0$. The last two expressions in the above equation show the effect of jitter induced ISI in the transmit signal. The performance of PWM-based equalization in the presence of jitter with Gaussian distribution can be estimated with Matlab in the following manner [17]:

1. Evaluate the worst case bit sequence (d_k) for the channel in the presence of ISI and absence of jitter [15].

2. Construct the worst case transmit signal using the worst case bit sequence and eq. 3.4 with Gaussian distribution for jitter.

3. Convolve the worst case transmit signal with estimated channel impulse response $h(t)$ to measure the received eye in the presence of jitter.

4. It is to be noted that the number of jitter samples used above depends on the length of the worst case bit sequence (l_{wc}). So, the above steps are repeated n times with a

different seed for Gaussian jitter distribution each time to collect large enough jitter samples ($n \times l_{wc}$) in the end. Finally, the worst case received eye in the presence of jitter is obtained by superimposing received eye diagram for each noise seed.

When jitter $j(kT)$ and $j(\overline{m+D}T)$ are independent and identically distributed Gaussian random variables, the worst case vertical and horizontal eye openings obtained from the above algorithm are shown in Fig. 3.7. The worst case eye opening degrades with jitter for two different channel losses. For a channel with 28 dB loss at Nyquist frequency, RMS jitter

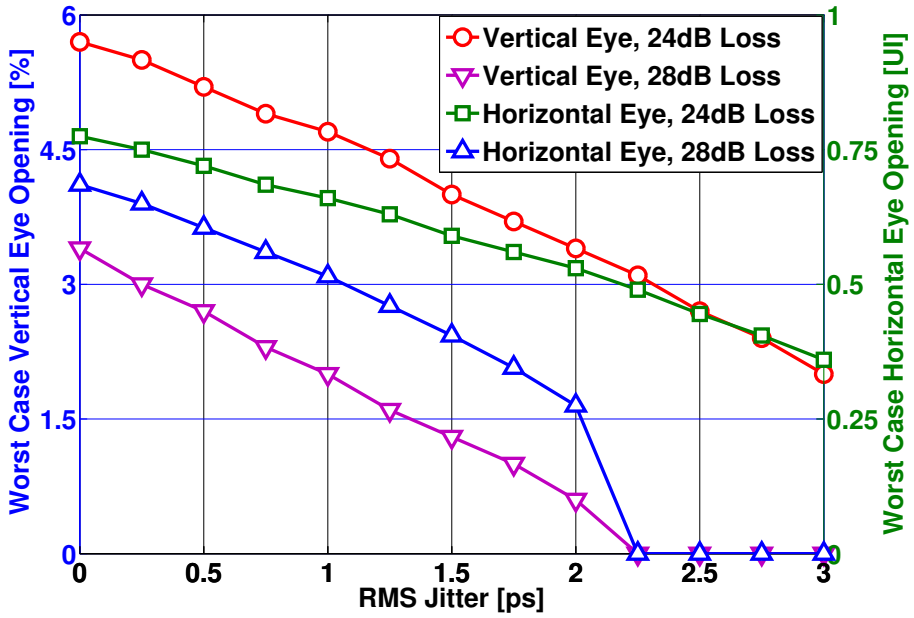


Figure 3.7: Worst case received eye opening vs r.m.s jitter amplitude.

in excess of 2 ps together with the worst case bit sequence can close the received eye opening completely. This is a very pessimistic upper bound of the jitter as it assumes worst case jitter and ISI to occur at the same time, but nevertheless provides a guideline for the target jitter specification for the design of clock generation.

Next we implement a voltage mode transmitter incorporating PWM-based equalization to reap the benefits of decoupled impedance matching, equalization, and output swing control.

3.2 Voltage-Mode Transmitter with PWM-based De-emphasis

An energy-efficient voltage mode (VM) transmitter has been realized while compensating for large channel loss with PWM-based equalization. PWM-based equalization helps in eliminating tradeoffs between impedance matching, high channel loss compensation, and variable output voltage swing in VM transmitters.

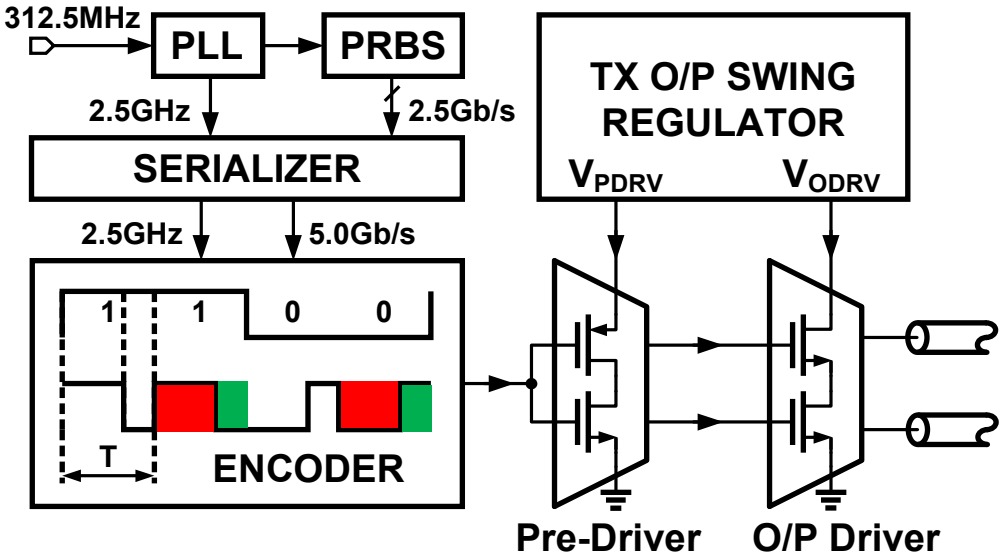


Figure 3.8: Simplified block diagram of the proposed transmitter.

Figure 3.8 shows the proposed voltage mode transmitter architecture. A digital PLL generates 2.5 GHz clock used to serialize half rate (2.5 Gb/s) PRBS data to 5 Gb/s NRZ data stream. At the heart of the TX is the ENCODER that converts 5 Gb/s NRZ data into pulse width modulated data. The duty cycle of the PWM output signal is adjusted in the encoder in accordance with the channel loss. The encoder output is buffered using a pre-driver and transmitted using a conventional N-over-N voltage mode driver. Driver output swing and impedance are independently controlled by properly setting supply voltage of the main driver (V_{ODRV}) and pre-driver (V_{PDRV}), respectively. On-chip low dropout regulators are used to generate V_{ODRV} and V_{PDRV}. Implementation details of the key building blocks of the transmitter are discussed below.

3.2.1 Encoder

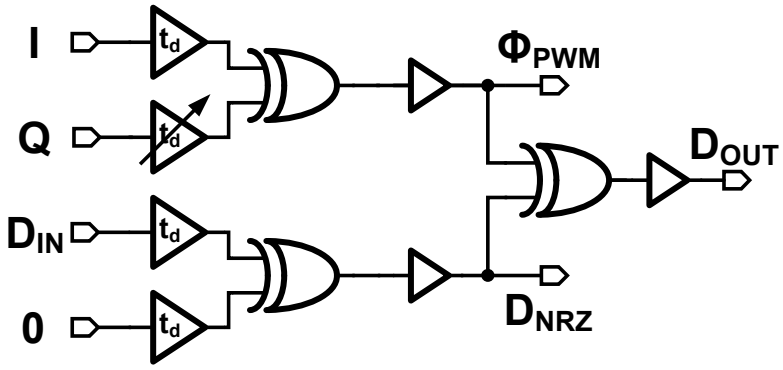


Figure 3.9: Simplified block diagram of encoder.

The simplified block diagram of the encoder is shown in Fig. 3.9. Using quadrature phases (I/Q) of a 2.5 GHz clock and 5 Gb/s non-return-to-zero data (D_{IN}), the encoder generates a pulse-width modulated output data stream, D_{OUT}. The top half of the encoder circuit operates on I/Q phases and generates a 5 GHz clock (Φ_{PWM}) with variable duty cycle. When all of the I/Q phases are delayed by the same amount, t_d, the output duty cycle is equal to 50%. By varying the delay of phase Q with respect to phase I, the duty cycle can be varied both above and below the nominal value of 50%. Because the delay can be tuned precisely, the de-emphasis resolution can be as high as desired. The bottom half of the encoder circuit is used to maintain the phase relationship between the I/Q phases and D_{IN} at the output. To this end, D_{IN} is passed through a delay path that is matched to that of the I/Q phases. The output 5 Gb/s PWM data is generated by XORing Φ_{PWM} with D_{NRZ} as depicted in the final stage in Fig. 3.9. The delay cells and CMOS XOR gate in the encoder are crucial to PWM-based equalization and are discussed next.

3.2.1.1 Current Controlled Delay Cell

The programmable delay is implemented by a cascade of current-starved inverters (delay cells) as shown in Fig. 3.10(a). An even number of delay cells are used to minimize duty cycle distortion otherwise caused by both asymmetric and unequal number of high-to-low and low-to-high transitions for odd number of cells. The delay is controlled using an external

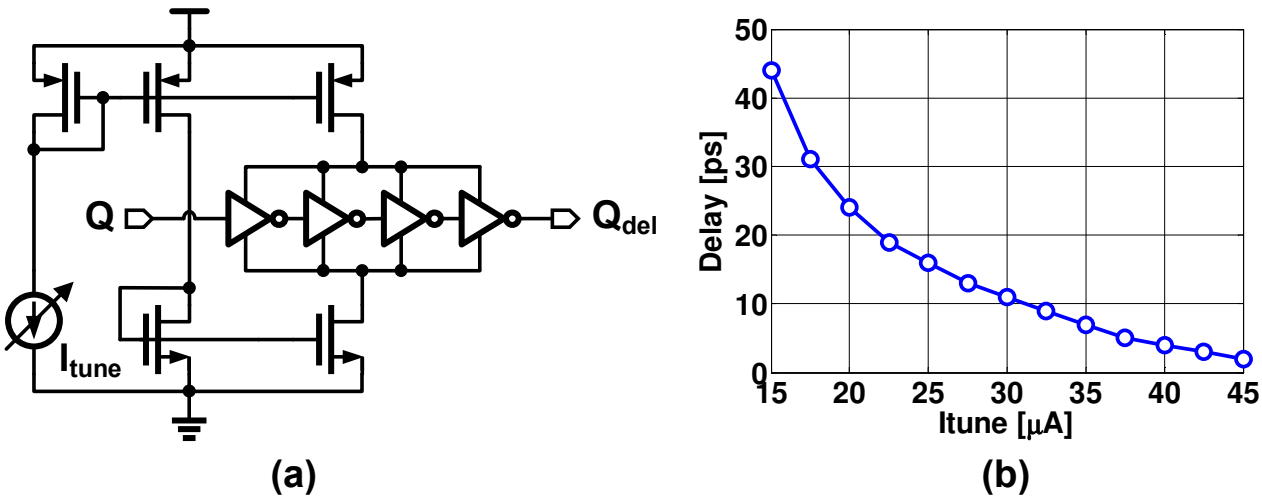


Figure 3.10: (a) Schematic of current controlled delay cell. (b) Tuning characteristic of the delay cell (output delay vs. tuning current).

current source (I_{tune}). The simulated transfer characteristic of a delay cell, including its parasitics extracted from the layout, is shown in Fig. 3.10(b). The output delay is offset from the nominal delay at I_{tune}=50 μA. A total delay tuning range in excess of 30 ps is achieved, which is adequate to compensate for channel loss ranging from 16 dB to 28 dB.

3.2.1.2 XOR-based Pulse Width Modulation

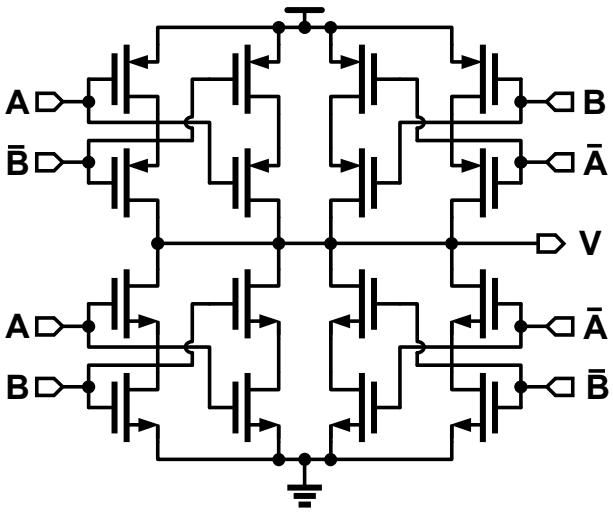


Figure 3.11: Pseudo-differential fully symmetric XOR gate.

The performance of the last stage CMOS XOR gate is critical to maintaining high fidelity

of the PWM output. Limited bandwidth of the static CMOS logic data causes intersymbol interference, which degrades equalization performance. Even an order of few fF of parasitic capacitance at internal nodes of the XOR gate (Fig. 3.11) can cause significant ISI.

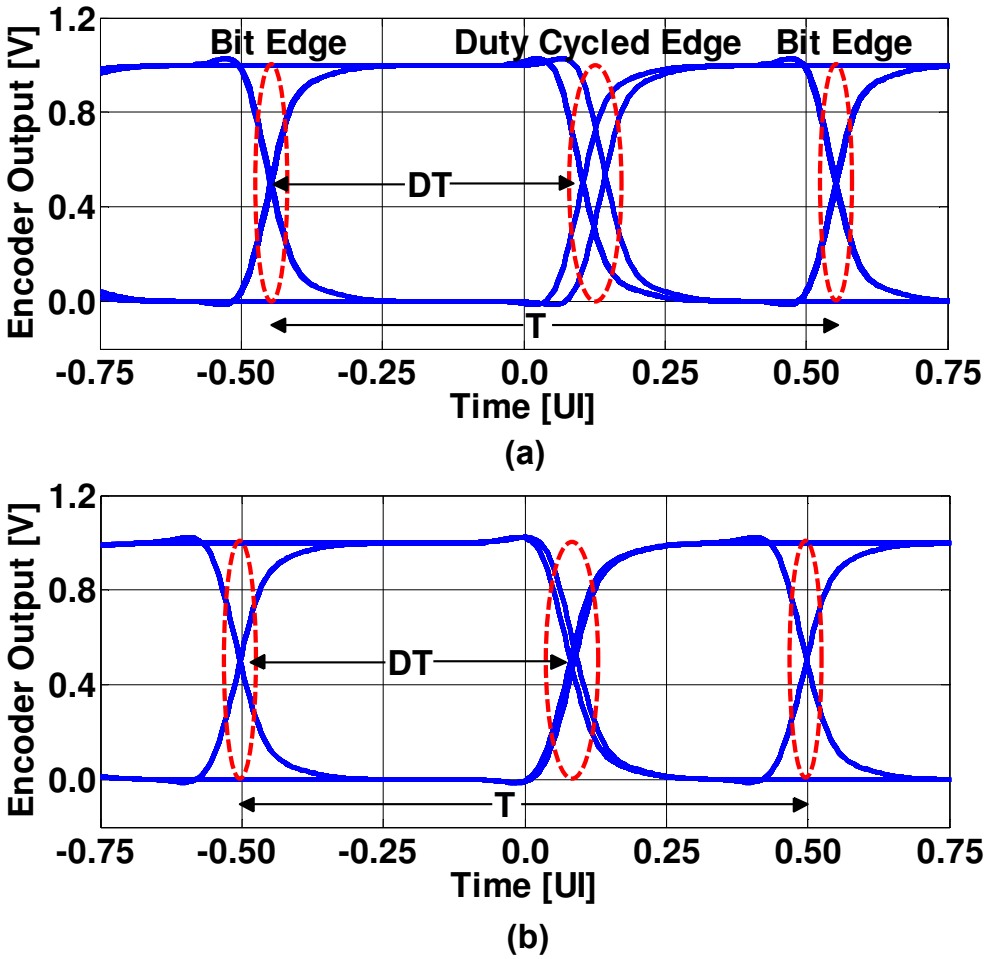


Figure 3.12: Eye diagram at the output of the encoder: (a) with bandlimited CMOS XOR gate (b) after optimizing transistor sizes and layout.

This ISI effect could be observed in the simulated output eye diagram shown in Fig. 3.12(a). The figure shows two transitions every bit period. 'Bit Edge' marks the beginning and end of each bit period. 'Duty Cycled Edge' marks the intermediate transition edge reflecting the duty cycle of the bit waveform. Bit edge has less jitter whereas the duty cycled edge shows two distinct streaks of jitter. This is due to two different kinds of transitions at the duty cycled edge: (i) a transition at the duty cycled edge for consecutive identical bits, and (ii) a transition at the duty cycle edge just after the bit transition 1-to-0 or vice versa.

This data-dependent jitter degrades the effectiveness of PWM equalization by corrupting the duty cycle ratio. For this reason, the CMOS XOR gate is optimized for minimal parasitic capacitance at all internal nodes. Figure 3.12(b) shows the simulated output of the encoder with extracted parasitic capacitance. The typical peak-to-peak jitter is about 4.2 ps. The XOR output is buffered to drive the output drivers. XOR gates used for just passing the data to emulate the delay in the clock modulation path are independently optimized as they operate with static inputs (either 1 or 0).

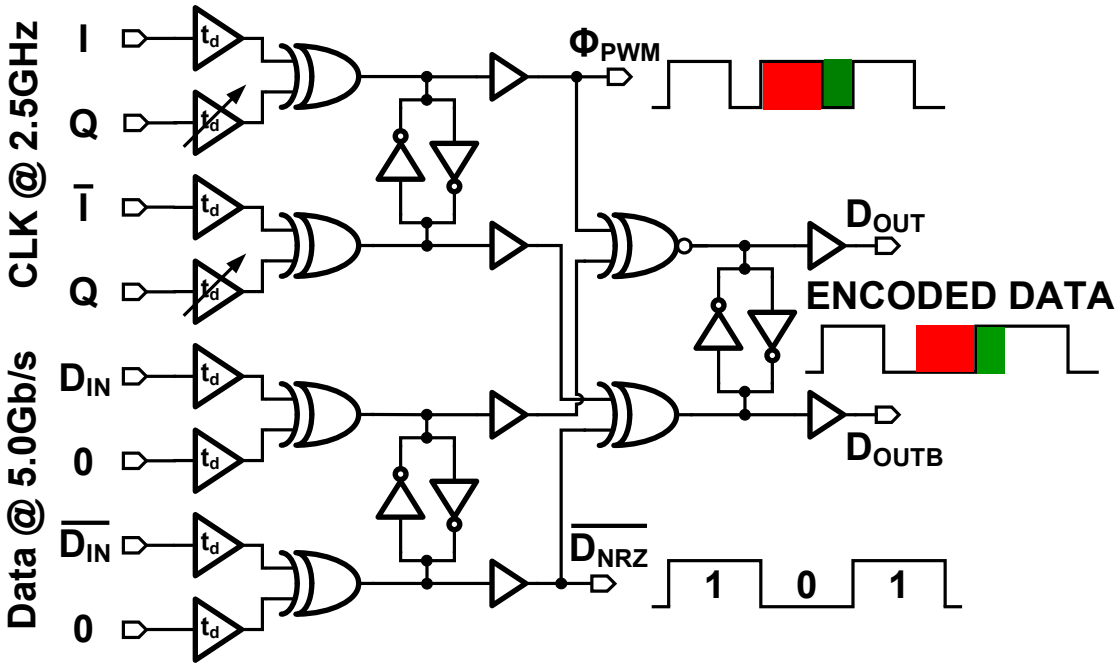


Figure 3.13: Complete block diagram of the pseudo-differential encoder.

The complete block diagram of the pseudo-differential encoder is shown in Fig. 3.13. The duty cycle modulated 5 GHz clock signal, Φ_{PWM}, and its compliment $\overline{\Phi_{\mathrm{PWM}}}$ are generated by XORing I with Q and $\bar{\mathrm{I}}$ with Q, respectively. $\mathrm{D_{NRZ}}$ and its compliment $\overline{\mathrm{D_{NRZ}}}$ are generated in a similar fashion. Differential PWM outputs are generated by XNORing Φ_{PWM} with $\mathrm{D_{NRZ}}$ and XORing $\overline{\Phi_{\mathrm{PWM}}}$ with $\overline{\mathrm{D_{NRZ}}}$. Cross-coupled inverters are inserted to maintain the differential nature of all the pseudo-differential signals.

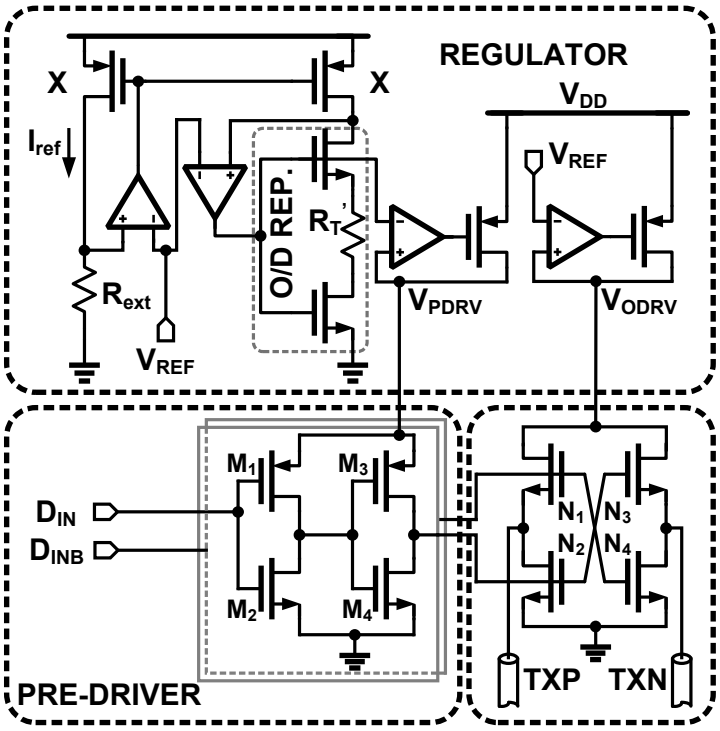


Figure 3.14: Schematic of voltage mode output driver.

3.2.2 Voltage Mode Output Driver

The block diagram of the voltage mode output driver is shown in Fig. 3.14. The output stage consisting of NMOS transistors N1-N4 operates with a regulated supply voltage of V_{ODRV}. A low dropout regulator sets V_{ODRV} to be equal to an external reference voltage V_{REF}, which makes the peak-to-peak differential output voltage swing also equal to $V_{ODRV} = V_{REF}$ and current consumption equal to $V_{REF}/4R_T$. Transistors N1-N4 are driven by a CMOS inverter based pre-driver, whose supply voltage is V_{PDRV} [18]. A bias circuit consisting of a replica of the output driver and a low dropout regulator generates V_{PDRV} so that the output impedance of the replica driver is matched to the reference resistor, R_{ext}. This replica-biasing ensures that the main driver output impedance is matched to the channel characteristic impedance. To save power in the replica biasing circuit, R_T' is scaled up by a factor of 8X and the output driver replica is scaled down by the same factor in comparison to the main output driver. Here, the output driver has minimum number of internal nodes as compared to the internal nodes in digitally controlled and segmented output drivers [9] or in output drivers with analog impedance control through transmission gate [19]. This helps to

suppress data dependent ISI while transmitting sub-bit rate pulses. The pre-driver consists of two inverters implemented using transistors M1/M2 and M3/M4 (Fig. 3.14). It acts as a level-shifting buffer and converts rail-to-rail swing encoder output to the lower swing input signals of the output drivers. The two inverters are separately optimized to minimize duty cycle distortion.

3.2.3 Digital Phase Locked Loop

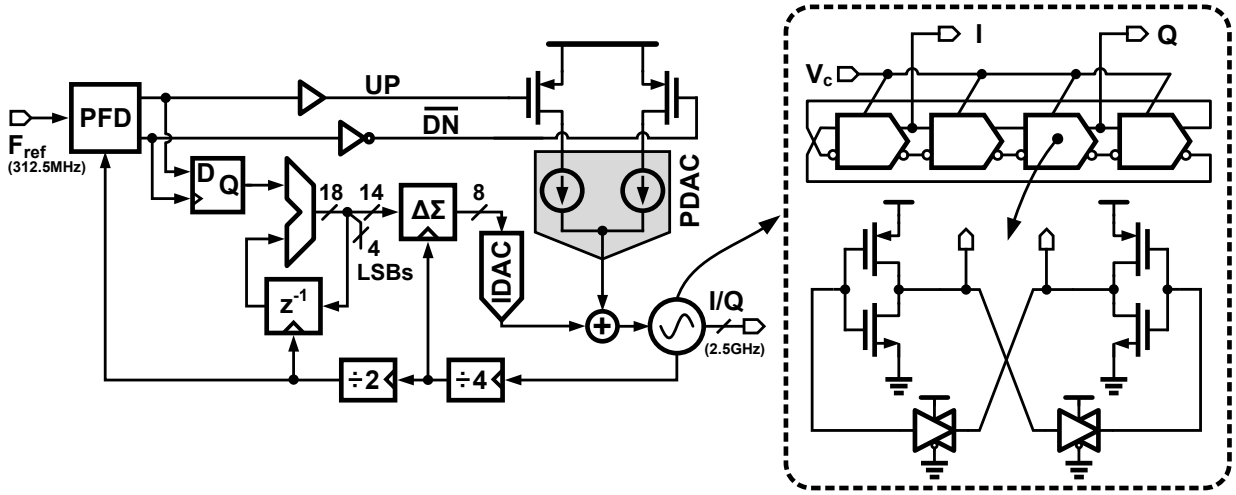


Figure 3.15: Digital phase-locked loop (DPLL) block diagram.

Low jitter I/Q clock phases at 2.5 GHz are generated from a 312.5 MHz reference clock using a type-II digital phase locked loop (DPLL) shown in Fig. 3.15 [20]. It consists of a linear proportional path, a digital integral path, a current controlled ring oscillator (CCO) with quadrature outputs, and a feedback divider. Proportional control is implemented by driving the CCO with the linear phase frequency detector (PFD) output through a three-level current DAC (PDAC). Because the pulse width of the PFD output is proportional to the input phase error, no quantization error is introduced into the loop. Integral control is implemented by accumulating the sign of the PFD output, measured using a bang-bang phase detector. The quantization error introduced by the bang-bang phase detector is suppressed by the 18 bit digital accumulator in the integral path. The lower 4 LSBs of the accumulator output are ignored to suppress the jitter caused by limit cycles and the rest

of the 14 MSBs are truncated to 8 bits using a second-order digital delta sigma modulator (DSM). The DSM output controls the oscillator through an 8 bit thermometer coded DAC (IDAC). The DSM is clocked at twice the frequency of the incoming digital input to reduce inband quantization error. The current controlled oscillator is implemented using a cascade of 4 pseudo-differential CMOS inverters coupled in a feedforward manner using transmission gates (see Fig. 3.15).

3.3 Measurement Results

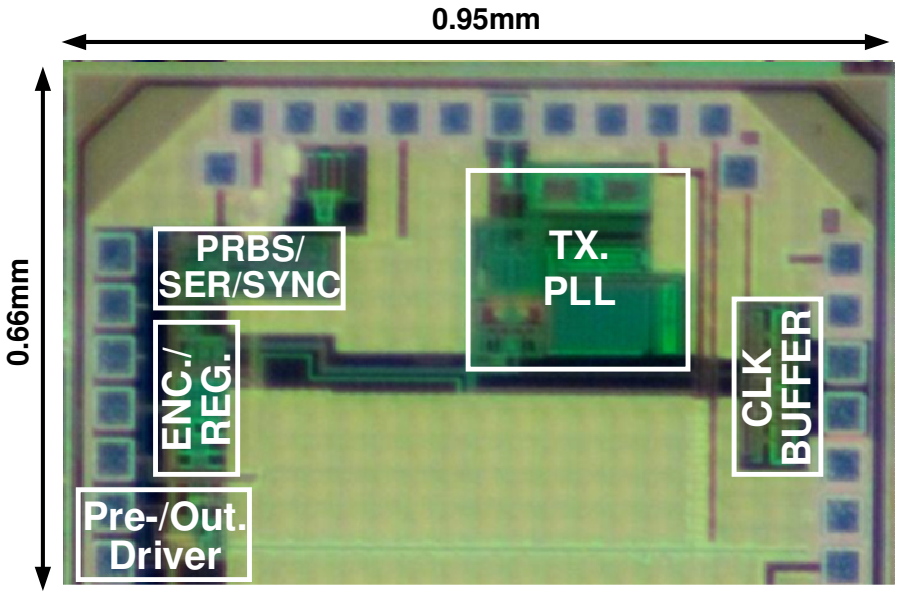


Figure 3.16: Die photo of the proposed transmitter.

The proposed transmitter was implemented in a 90 nm CMOS process and packaged in 60 pin QFN package. The transmitter occupies an active area of 0.13 mm^2 (Fig. 3.16).

The performance of the transmitter is characterized across three stripline channels of varying length (and loss) fabricated on a FR4 PCB board. Figure 3.17 shows the measured frequency response (S21) of the three channels, which indicates a loss of 16 dB (60 inch), 24 dB (86 inch), and 28 dB (96 inch) at the Nyquist frequency of 2.5 GHz. In addition to the stripline trace, the channel includes bondwires, package parasitics, 1.5 inch on-board microstrip line, SMA connectors, and SMA cables.

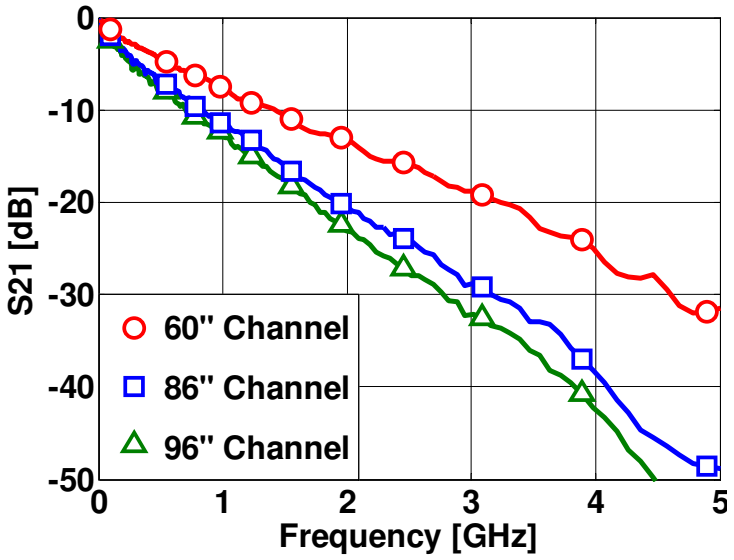


Figure 3.17: Measured frequency response of FR4 stripline channels.

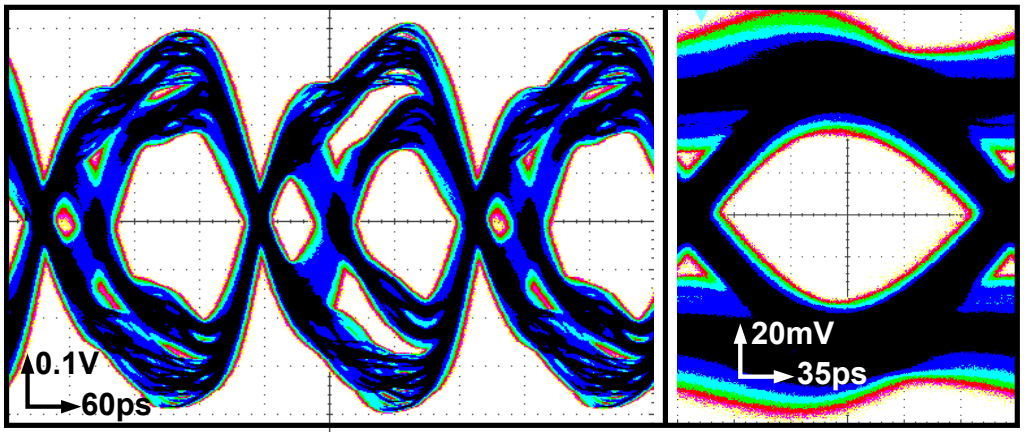


Figure 3.18: (a) TX eye diagram at the input of channel with 16 dB loss, and (b) received eye diagram at the output of the channel.

The differential output eye diagram when transmitting PRBS7 data across a 60 inch channel is shown in Fig. 3.18(a). An asymmetry between the odd and even bits is caused by duty cycle distortion of the clock phases. Loading mismatch on clock phases generated and routed on-chip resulted in the clock duty cycle distortion. The differential eye diagram at the output of the channel, shown in Fig. 3.18(b), indicates a vertical eye opening of 78 mV and horizontal eye opening of 0.6 UI. The duty cycle of the PWM signal to achieve this eye opening is roughly equal to 65%.

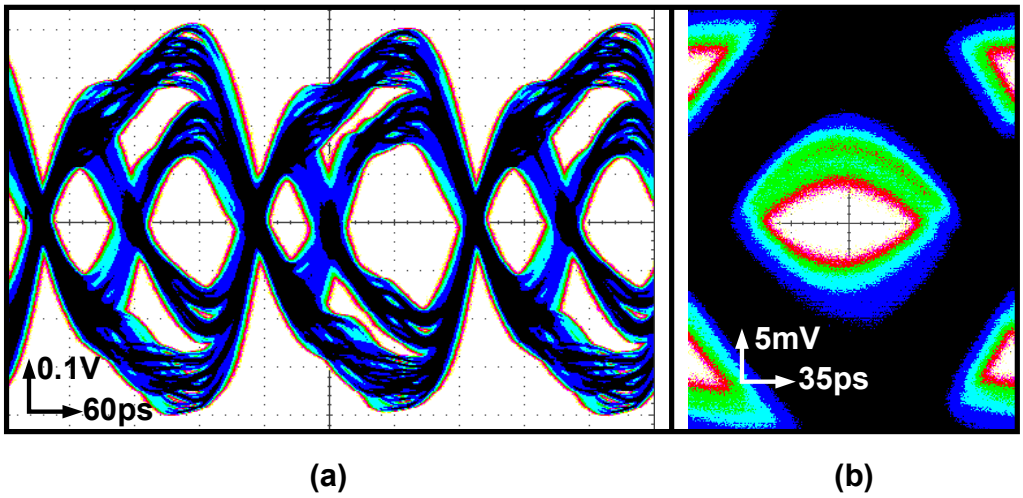


Figure 3.19: (a) TX eye diagram at the input of channel with 28 dB loss, and (b) received eye diagram at the output of the channel.

The experiment was repeated with a 96 inch channel and the resulting differential eye diagrams are shown in Fig. 3.19. Due to higher loss compared to the 60 inch channel, the optimal duty cycle of the PWM signal in this case is close to 58%. It can also be observed that the duty cycle distortion of clock phases has a greater impact on the received eye.

Figure 3.20 depicts that the vertical/horizontal eye openings of the odd and even bits are 8 mV/0.3 UI and 18 mV/0.75 UI, respectively. The wider eye opening of the even bits indicates that the performance can be further improved by correcting the clock duty cycle distortion.

The transmitter performance is also measured in terms of BER at the RX end. To this end, bathtub curves are measured using 80SJNB BER analysis software and Tektronix DSA8200 sampling oscilloscope and plotted in Fig. 3.21. The transmitter achieves BER $< 10^{-12}$ with

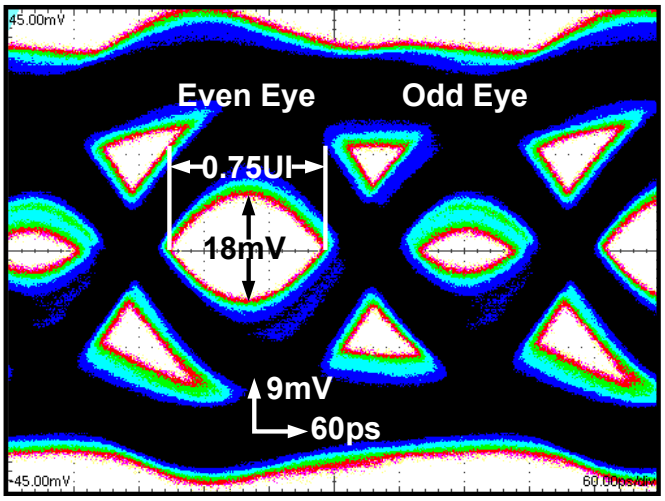


Figure 3.20: Received eye opening at the end of channel with 28 dB loss.

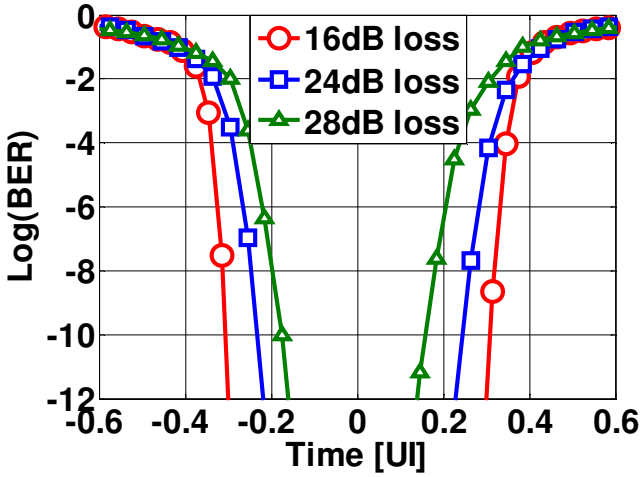


Figure 3.21: Bathtub plots for different channel loss.

a timing margin of 0.6 UI and 0.3 UI for channel loss of 16 dB and 28 dB, respectively.

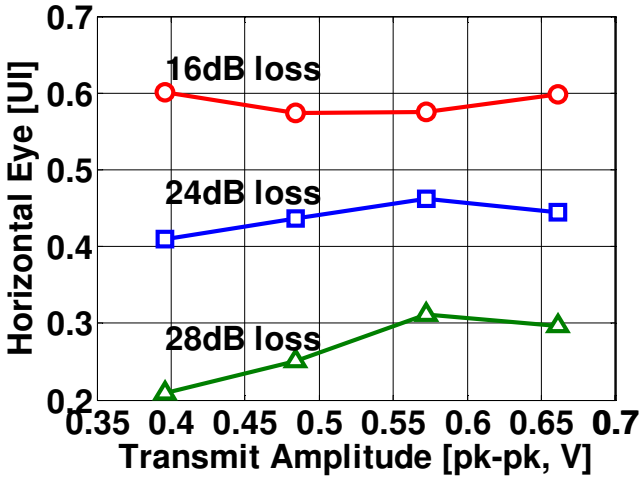


Figure 3.22: Horizontal eye opening vs. peak-to-peak transmit amplitude.

The performance of the transmitter was also evaluated across a range of amplitudes of the transmitted signal. The measured horizontal eye opening for BER = 10^{-12} with various output amplitudes shown in Fig. 3.22 illustrates that the horizontal eye opening remains fairly constant for the 16 dB loss channel and varies by about 0.1 UI with higher loss channels.

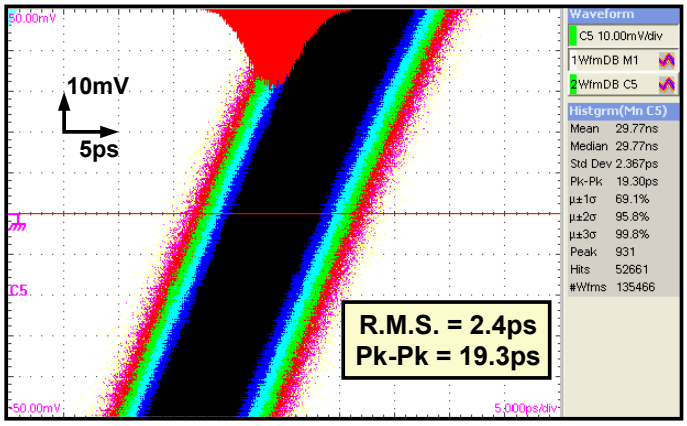


Figure 3.23: DPLL jitter histogram at 2.5 GHz output.

The measured long-term absolute jitter of the digital PLL at 2.5 GHz output frequency is shown in Fig. 3.23. It achieves 2.4 ps r.m.s. and 19.3 ps peak-to-peak jitter.

The taxonomy of the total power dissipated in the transmitter at 0.66 V output swing is provided in Table 3.1. The total power is 15.6 mW, which translates to a power efficiency of 3.1 mW/Gb/s. Increased frequency operation for pre-driver and output drivers while

Table 3.1: Power summary of 5 Gb/s VM Tx.

	Power [mW]
Serializer	0.84
Encoder	4.32
PLL	2.55
Pre-driver	1.31
Output driver	2.61
Regulator	3.92
Total Power	15.6

transmitting sub-bit rate pulses increases the power consumption compared to conventional NRZ-based pre-/output drivers. However, the power of the transmitter as a whole compares favorably while compensating for high channel loss (28dB). The performance of the transmitter is compared with other state-of-the-art transmitters in Table 3.2. This work achieves 7.1 times better power efficiency (mW/Gb/s) compared to [14]. A simple FOM1=mW/Gb/s does not signify the effort in equalizing the channel loss. So, a modified FOM2 defined below is also used to make the comparison [19].

$$\text{FOM2} = \text{DR(Gb/s)} \times 10^{(\text{Channel loss}/10)}/\text{Power(mW)} \tag{3.5}$$

where 'DR' is the data rate, and channel loss is in 'dB'. Larger FOM2 indicates better performance. This work achieves a maximum FOM2 of 202 which is 2.2X better than [14].

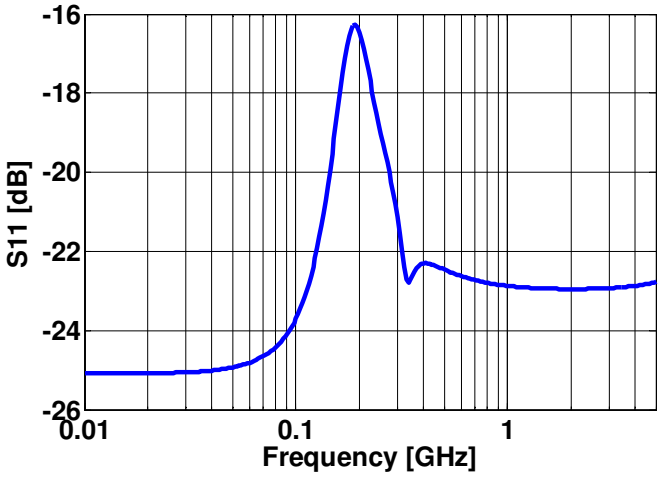


Figure 3.24: Simulated differential return loss (S11) for the TX.

Table 3.2: Performance summary for 5 Gb/s VM Tx and comparison to the state-of-the-art.

	JSSC06 [14]	ISSCC08 [19]	JSSC13 [2]	This Work		
Technology [nm]	90	90	65	90		
Supply Voltage [V]	1.2	1.2	1.2	1.0/1.1/1.25		
Data Rate [Gb/s]	5	8	10	5		
Channel Loss [dB]	33	37	13	16	24	28
Equalization Scheme	PWM (CML)	FFE-5/ DFE-2/CTLE	FFE-2 (VM)	PWM (VM)	PWM (VM)	PWM (VM)
Vertical Eye [mV]	15	-	200	78	18	8
Horizontal Eye [UI]	0.75	0.11	0.78	0.60	0.45	0.30
Power [mW]	110	232*	11	15.1	15.6	15.6
On chip PLL	NO	YES	NO	YES		
FOM1 [mW/Gb/s]	22	29	1.1	3.0	3.1	3.1
FOM2	91	165	18.1	13.3	80.0	202

* Total power for TX and RX.

Figure 3.24 shows the simulated differential return loss measurement for the TX including the parasitics extracted from the layout. Here, the TX is biased for 0.65 V peak-to-peak differential output swing. The differential return loss is $\leq$-16 dB over a frequency range 0-5 GHz.

3.4 Summary

Voltage-mode drivers are more power-efficient compared to CML drivers. But they are not well suited for incorporating de-emphasis to equalize channel loss. In particular, they suffer from conflicting design requirements for simultaneously achieving impedance matching, desired output swing, high de-emphasis resolution, and low power. In this paper, we proposed to use time-based de-emphasis, implemented using pulse width modulation, to overcome these tradeoffs. In contrast to conventional voltage-based approaches, PWM-based signaling requires transmitting only two voltage levels, independent of channel impedance, output swing, and amount of de-emphasis. These features make it best suited for implementing de-emphasis in voltage mode drivers.

A conventional N-over-N voltage mode driver is driven with the PWM signal to achieve

transmitter de-emphasis. Low dropout regulators were used to set the output swing and output impedance of the driver. The PWM encoder is implemented using CMOS digital logic. However, limited bandwidth of CMOS logic gates in the signal path can introduce ISI and degrade voltage and timing margins. Careful transistor sizing and layout design are shown to be effective in overcoming bandwidth limitations. Clock jitter requirements were also analyzed using peak distortion analysis and upper bounds were provided.

A prototype PWM-based 5 Gb/s voltage mode transmitter was implemented in a 90 nm CMOS process and characterized across different channels and output swings. The horizontal/vertical eye openings (BER=10^{-12}) at the ends of 60 inch and 96 inch stripline channels are 78 mV/0.6 UI and 8 mV/0.3 UI, respectively. The received signal swing with 28 dB channel loss appears quite low due to clock duty cycle distortion. Therefore, in an improved design, clock duty cycle error must be corrected to increase the magnitude of the received signal. The transmitter consumes a total power of 15.6 mW, of which 2.5 mW is consumed in the digital PLL and 7.4 mW in the pre-/output drivers and regulators. This translates to a power efficiency of 3.1 mW/Gb/s, which compares favorably to the state-of-the-art.

Chapter 4

A 14 Gb/s Wireline Transceiver

Multi-Gb/s wireline transceivers face challenges in performing their basic functions within a strict power budget. Maintaining constant energy efficiency while scaling data rate and power consumed, both by 2X, is difficult as stricter timing margins and limited circuit bandwidth with self-loading get more pronounced at higher data rates. Compensating for increased channel anomalies at higher data rates demands extended power budget. In such a scenario, achieving higher data rate with lower energy efficiency is more than challenging. Technology scaling does help to extend bandwidth [8] but low energy efficiency at higher data rates demands more than technology scaling. It requires newer techniques at the architecture and circuit levels to perform the same basic functions of sending and receiving data as in conventional designs.

Figure 4.1 shows a typical block diagram of a serial link with embedded clocking. On the Rx side, it requires data sampler and sampling clock *RCLK* with optimum voltage and timing margins, respectively, to recover data in low power budget. Since sampling time margins for *RCLK* decrease at higher data rates, power consumption in the clock recovery unit (CRU) increases non-linearly to maintain the same bit error rate at the output. Sub-rate CDR architectures reduce the maximum frequency of operation for samplers and CRU, and are common for multi-Gb/s links. But they have an added power penalty in generation

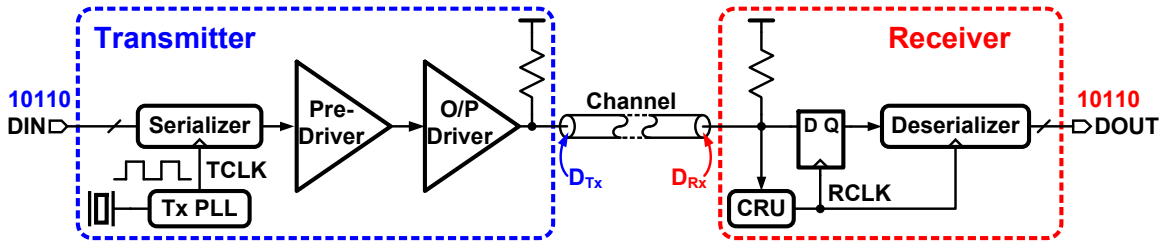


Figure 4.1: Block diagram of a serial link with embedded clocking.

and distribution of additional sampling phases for the same peak data rate. In this work, a new sub-rate CDR architecture is proposed to overcome problems with conventional digital clock and data recovery. Section 4.1 elaborates on current clock and data recovery issues and discusses the proposed clock and data recovery.

Next, samplers in the Rx front-end and data demultiplexer (DMUX) in the deserializer are blocks with large bandwidth requirement and power consumption proportional to frequency of operation. Limited-swing current mode logic (CML) and full-swing CMOS logic based circuits are dominant building blocks for front-end samplers and deserializers. The choice between the two logics depends on frequency of operation. With CMOS output levels, full-swing sense-amplifier flip-flop (SAFF) [13] benefits from its dynamic power consumption, but it suffers from rail-to-rail voltage swing at various circuit nodes which increases power consumption due to parasitic capacitance at all these nodes. In a frequency range where SAFF and CML based flip-flops both are functional within a reasonable power budget, the CML flip-flop consumes more power due to constant current requirement [3]. In [3], charge steered logic with limited swing operation and low dynamic power offers an attractive alternative to conventional CML or CMOS logic, but its implementations so far [3, 21–23] are not fit to replace standard CML or CMOS logic in samplers and deserializers without any significant power penalty in clocking. Section 4.2 reviews prior art in charge steered logic and discusses the proposed low power charge-based Rx front-end.

On the Tx side, the high speed signal chain including pre-driver and output driver with equalization embedded are the most power consuming blocks. While voltage mode (VM) O/P drivers are more energy-efficient compared to their current mode counterparts, embedded equalization in O/P drivers makes them power hungry. A host of VM transmitters with full/limited-swing outputs and embedded equalization [2,10,11,16,24] have demonstrated an improved performance with a newer technique every time. However, equalization with fine resolution and channel termination with Tx output impedance at the same time have led to increased power consumption in pre-drivers [2] and degraded energy efficiency of overall VM transmitters. Section 4.3 reviews prior work on uniformly segmented VM output drivers and proposes a partially segmented VM output driver while reducing overall power consumption in the voltage mode transmitter.

Incorporating the proposed design techniques, a 14 Gb/s wireline transceiver is proposed in Section 4.4. The effectiveness of the above methods is demonstrated with measurement results for the transceiver in Section 4.5 and this work concludes in Section 4.6.

4.1 Clock & Data Recovery

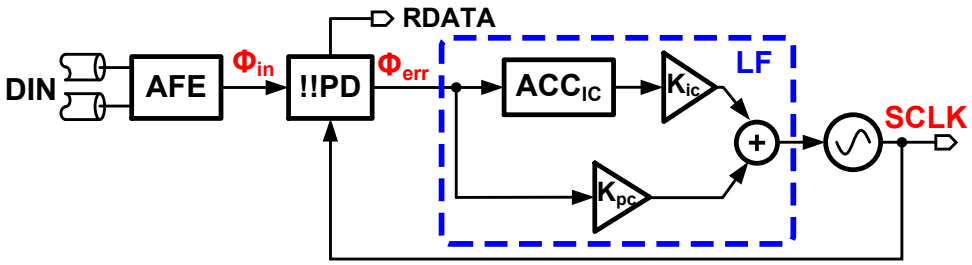


Figure 4.2: Block diagram of a conventional digital CDR.

Clock and data recovery is an essential part of embedded wireline receivers. Traditionally, sampling clock is recovered from incoming random data by processing phase and frequency error between random data and recovered sampling clock in analog fashion. But advancement of digital CMOS technology and difficulty in low power high speed analog processing of phase error have led to digitally processing the information during clock and data recovery.

Figure 4.2 shows a block diagram of a conventional second-order digital CDR. The received input DIN is terminated at the analog front-end (AFE), which often includes 50 Ω channel termination, a variable gain amplifier, and a continuous-time linear equalizer. AFE amplifies the received signal large enough to be resolved by samplers in a bang-bang phase detector (!!PD). The phase detector detects *Early/Late* arrival of the received signal w.r.t. the sampling clock (SCLK) and outputs a discrete error value Φ_{err}. Φ_{err} is processed by a first-order digital loop filter (LF) whose output later tunes frequency and phase of the oscillator as desired. Also, the incoming data is recovered (RDATA) as one of the signals within the !!PD. Figure 4.3 depicts a linearized small signal model of the above digital CDR [25]. Phase error Φ_{in} at the input of !!PD is processed along proportional (K_{pc}) and integral (K_{ic}) paths in the loop filter before controlling the oscillator. After substituting $z = e^{sT} = e^{s/f_{ref}}$ and $e^{sT} \approx 1 + sT$ for $|2\pi fT| \ll 1$, open loop gain for CDR loop, $LG_{CDR}(s)$ is given by eq. 4.1.

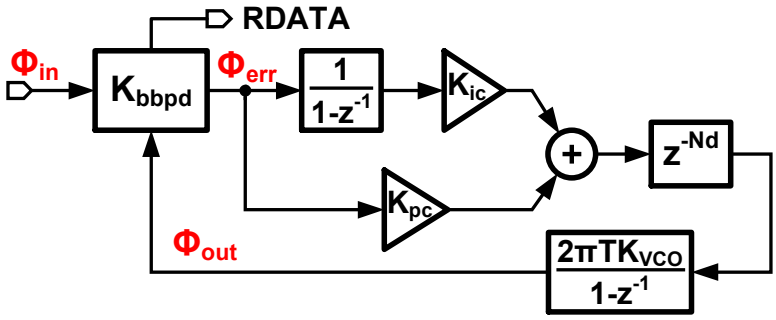


Figure 4.3: Linearized small signal model of a conventional digital CDR.

$$\mathrm{LG_{CDR}(s)} = \frac{\Phi_{\mathrm{out}}}{\Phi_{\mathrm{in}}} = (1 - \mathrm{sNd/f_{ref}}) \left[\frac{2\pi \mathrm{K_{bbpd} K_{pc} K_{vco}}}{\mathrm{s}} + \frac{2\pi \mathrm{K_{bbpd} K_{ic} f_{ref} K_{vco}}}{\mathrm{s^2}} \right] \tag{4.1}$$

Here, the first term accounts for delay in the loop which can be present in any of its building blocks. $\mathrm{K_{bbpd}}$ is gain of the bang-bang phase detector, $\mathrm{K_{pc}}$ is proportional path gain, $\mathrm{K_{ic}}$ is integral path gain, and $\mathrm{K_{vco}}$ is frequency gain of the oscillator. $\mathrm{K_{pc}}$ and $\mathrm{K_{ic}}$ terms in the loop gain expression compensate for phase and frequency errors, respectively. The second-order digital CDR overcomes problems in analog filters with resistors and capacitors. An LC oscillator or a ring oscillator is a basic clock source inside the CDR loop. It leads to a severe noise-versus-power tradeoff when !!PD quantization noise requires a lower CDR bandwidth, and oscillator phase noise suppression demands a larger loop bandwidth. While the LC oscillator has better phase noise performance compared to the ring oscillator in the same power budget, the problem gets critical when multiple sampling phases are needed in sub-rate clock and data recovery. Generating multiple sampling phases with an LC oscillator is power and area hungry, and with a ring oscillator only it is surely power intensive. The problem is generally addressed by using phase interpolator (PI) based digital CDR as discussed below.

4.1.1 Conventional PI-based Digital CDR

Figure 4.4 shows a block diagram of a commonly used phase interpolator based high speed clock and data recovery [26–28]. In this particular half-rate architecture, !!PD uses clock phases I/Q/IB/QB at half data rate to extract *DATA/EDGE* information from the received

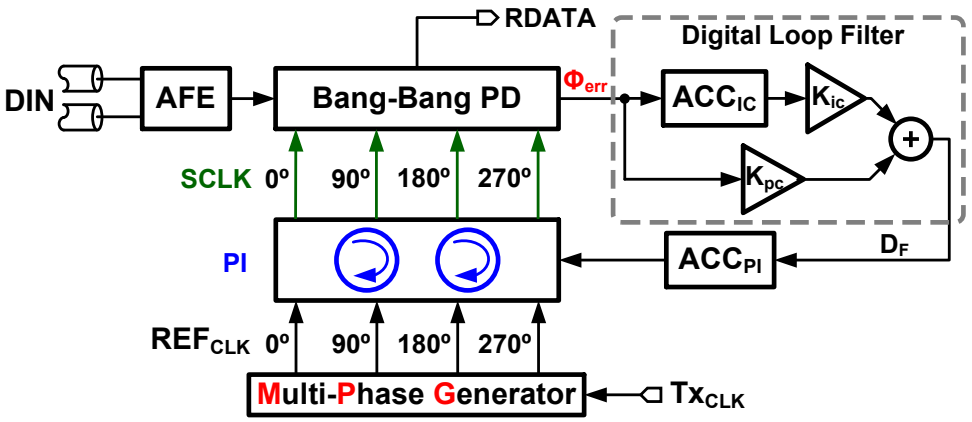


Figure 4.4: Block diagram of half-rate phase interpolator based digital CDR.

signal, thereby deducing *Early/Late* arrival of the received signal w.r.t. the sampling clock (SCLK). *Early/Late* information as Φ_{err} is later used to tune SCLK through digital loop filter (LF) and phase interpolator (PI). An accumulator ACC_{PI} plus PI acts as a digitally controlled oscillator which tunes frequency and phase of high speed reference input Tx_{CLK} depending on the loop filter output D_{F}. Multiple high speed reference clock phases (REF_{CLK}) are in turn generated by a multi-phase generator (MPG) which often employs PLLs, DLLs, clock mixer, etc. The PI-based architecture is quite common for high speed embedded clock and data recovery for a couple of reasons. First, an oscillator which is a significant noise source in clock generation is kept outside the CDR loop. Power and noise can be extensively optimized for multi-phase clock generation (REF_{CLK}) as it is not constrained by CDR bandwidth requirements. Secondly, multiple high speed reference clock phases (REF_{CLK}) can be shared among multiple transceivers operating in parallel. An independent multi-phase clock generation for each transceiver would surely be power intensive.

Despite the improvements over conventional analog/digital CDRs, the above architecture is still power hungry. There are two major power hungry operations involved. The first is power dissipation in high speed multi-phase clock distribution from MPG to PIs. While a number of clock phases required are constrained by power consumption in MPG and front-end samplers, a precise phase matching required at PI input makes the problem worse. Randomly skewed multi-phase PI input reduces the sampling time margin at PI output whereas a strict phase matching results in increased power consumption during clock

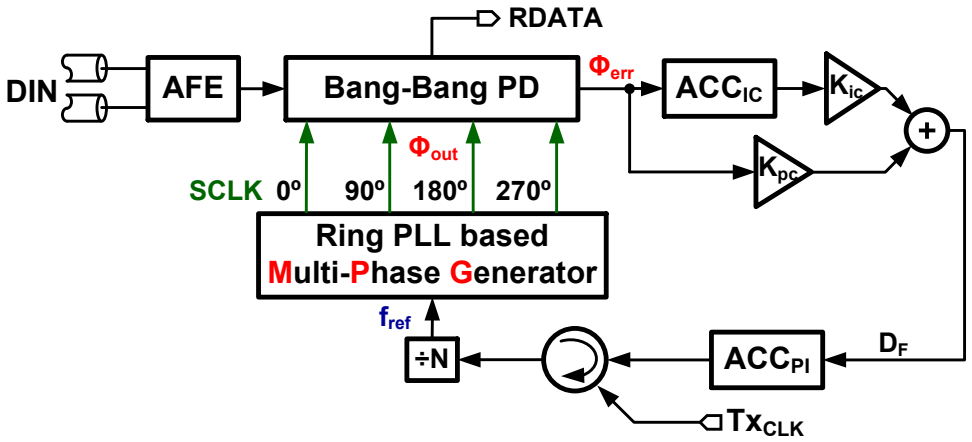


Figure 4.5: Block diagram of proposed ring PLL based multi-phase clock generation in CDR loop.

distribution and phase correction. The problem becomes severe when MPG provides multiple phases to several lanes running in parallel. Secondly, data rates $\geq 10\,\text{Gb/s}$ generally demand two or four PIs in CDR operating at quarter-rate [26] or half-rate [27]. The clock distribution in and around PI sinks in a significant power. In this work, a new clock and data recovery architecture is proposed which reduces power consumption in multi-phase sampling clock generation as compared to the commonly used PI-based architectures. The proposed CDR architecture is discussed below.

4.1.2 Proposed CDR Architecture

Figure 4.5 shows the proposed method of multi-phase sampling clock generation in digital CDR loop. The figure depicts two specific modifications to PI-based CDR. First, multiple sampling phases are directly derived from ring oscillator based PLL (RPLL) in contrast to multiple PIs. The sampling phases are matched to the accuracy of delay cells in the oscillator and more phases can be easily accommodated for sub-rate applications. Also, multi-phase clock distribution from ring PLL to front-end samplers is minimal when compared to sampling clock routing from multiple bulky PIs to samplers. Independently, ring oscillators face a severe noise-versus-power tradeoff and are not good for use in a CDR loop directly. But when the ring oscillator is embedded in the phase-locked loop with high frequency reference, output jitter is low enough for high speed applications. Here, the high frequency reference

for PLL, f_{ref}, is generated by a single PI which in turn is controlled by digital loop filter in the CDR loop. In contrast to PI-based conventional CDR loop, the digital loop filter tunes the sampling clock by controlling RPLL reference through an accumulator and PI. This particular multi-phase generation has three important benefits. First, it reduces the number of PIs, thereby reducing area and power. Second, it minimizes high frequency clock routing at input/output of the PI. Third, the clock frequency for interpolation is most probably lower as PI output is used as a reference for a clock multiplier (RPLL). Although the power consumption has decreased with this particular implementation, jitter performance of the CDR gets degraded due to increased loop delay in the CDR loop.

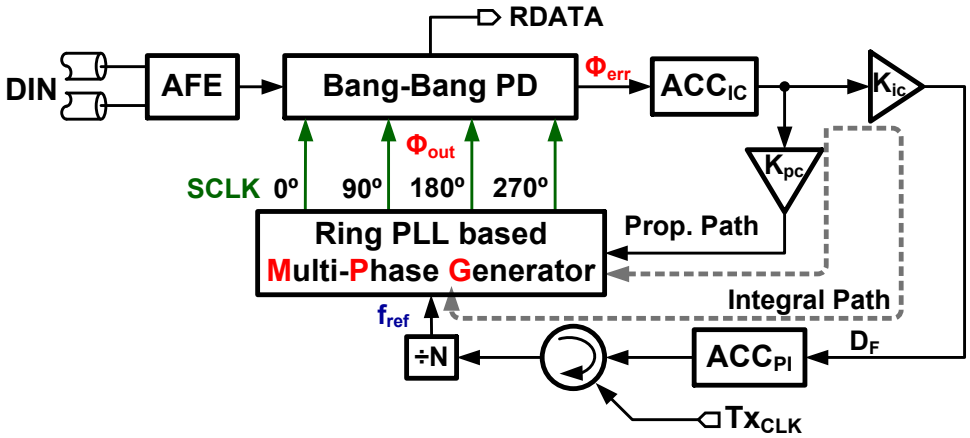


Figure 4.6: Block diagram of proposed clock and data recovery.

To alleviate the above problem, the loop filter control is split as shown in the proposed CDR block diagram in Fig. 4.6. The proportional path delay is minimized by directly adjusting the output phase of the RPLL similar to the proportional path in a conventional second-order CDR loop. Frequency error between received data and sampling clock is compensated for through integral path by tuning the reference frequency in response to frequency error. Proportional and integral path controls are discussed in detail below with loop gain analysis.

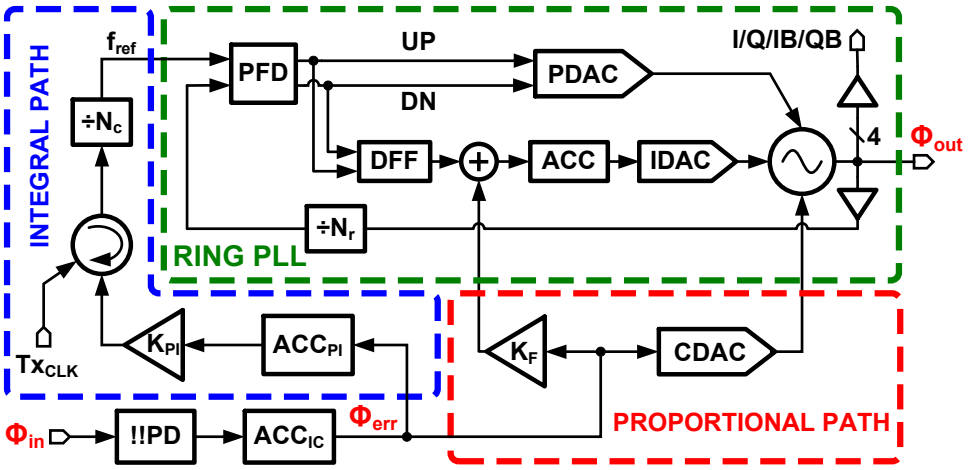


Figure 4.7: Block diagram for CDR proportional and integral path controls through ring PLL.

4.1.3 Loop Gain Analysis for Proposed CDR

The detailed block diagram for phase and frequency correction in the proposed CDR is shown in Fig. 4.7. It consists of a ring PLL which multiplies a low frequency reference f_{ref} to half-rate sampling clock (Φ_{out}) with in-phase and quadrature outputs. RPLL is implemented with analog proportional path and digital integral path similar to highly digital PLL in [20]. The CDR proportional path corrects for input phase error Φ_{in}. The proportional control is implemented by adding accumulated phase error (ACC_{IC} output) to current controlled ring oscillator (CCO) directly and to the integral path of the RPLL. The direct control of the oscillator through CDAC corrects for high frequency phase perturbations at the input. Phase addition with gain K_F through RPLL integral path compensates for low frequency sampling phase error. Open loop gain for input phase error only through proportional path of CDR is given by eq. 4.2.

$$LG_{prop,cdr}(s) = \frac{\Phi_{out}}{\Phi_{in}}\bigg|_{prop,cdr} = \frac{K_{bbpd} f_{ref}}{s} \times \frac{N_r LG_{rpll}(s)}{1 + LG_{rpll}(s)} \left[\frac{K_{pc,pr} + \frac{K_{pc,ir} f_{ref}}{s}}{K_{pr} + \frac{K_{ir} f_{ref}}{s}} \right] \tag{4.2}$$

where $LG_{rpll}(s)$ is open loop gain of ring PLL, N_r is frequency divide ratio in RPLL feedback path, and K_{pr} and K_{ir} are static gains for proportional and integral paths in an independent ring PLL, respectively. In CDR proportional path, K_{bbpd} is gain of !!PD, and $K_{pc,pr}$ and

$K_{pc,ir}$ are static proportional path gains (Φ_{out}/Φ_{err}) through CCO and RPLL integral path, respectively. Under the approximation that RPLL has much larger bandwidth compared to error signals in CDR proportional path, $LG_{rpll}/(1+LG_{rpll}) \approx 1$ holds true for all practical phase error inputs. With CDR proportional path gains scaled in proportion to RPLL gains, i.e., $K_{pc,pr} = \alpha K_{pr}$ and $K_{pc,ir} = \alpha K_{ir}$, eq. 4.2 is reduced to eq. 4.3 which is the same as proportional path control in conventional second-order CDR loop (eq. 4.1).

$$LG_{prop,cdr}(s) = \left.\frac{\Phi_{out}}{\Phi_{in}}\right|_{prop,cdr} = \frac{\alpha N_r K_{bbpd} f_{ref}}{s} \tag{4.3}$$

It is to be noted that all accumulators in the above expressions are clocked at f_{ref} and variation in clocking frequency and delays should be accounted for in loop gain which is ignored above for simplicity of expression. Figure 4.8 plots Φ_{out}/Φ_{in} for CDR proportional path where RPLL bandwidth is restricted to $f_{ref}/50$. Below RPLL bandwidth, proportionally scaled $K_{pc,pr}$ and $K_{pc,ir}$ give a 20 dB/dec frequency suppression similar to proportional path in conventional second-order CDR loop. In the absence of integral path control ($K_{pc,ir} = 0$), only a fraction of error Φ_{in} is corrected in steady state due to finite DC gain in the proportional path (eq. 4.2).

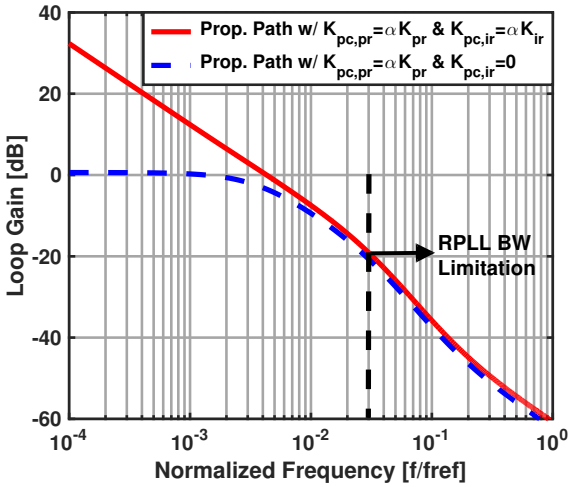


Figure 4.8: Open loop gains for CDR proportional path with and without integral path control in ring PLL.

Frequency error between input data and sampling clock appears as static offset at accumulator ACC_{PI} input (Fig. 4.7). PI applies a phase shift proportional to ACC_{PI} output to high frequency clock Tx_{CLK} which is often shared between Tx and Rx, and divided PI output is used as a reference for RPLL. Tx_{CLK} frequency $f_{TxCLK} = N_c f_{out}/N_r$ where f_{out} is sampling frequency for the receiver. Loop gain in the integral path of CDR is given by eq. 4.4 which can be approximated to the integral path in second order CDR loop under approximation $LG_{rpll}/(1 + LG_{rpll}) \approx 1$.

$$\begin{aligned} LG_{int,cdr}(s) &= \frac{K_{bbpd} K_{ic} f_{ref}^2}{s^2 N_c} \times \frac{N_r LG_{rpll}(s)}{1 + LG_{rpll}(s)} \\ &\approx \frac{K_{bbpd} K_{ic} f_{ref}^2 N_r}{s^2 N_c} \end{aligned} \tag{4.4}$$

where K_{ic} is static gain of the integral path from Φ_{err} to PI output. The integral path is the slowest path in the CDR loop correcting steady state frequency errors. Since !!PD has limited frequency error detection capability, it can be assisted by separate frequency error detection blocks like input divider based frequency detection in [29] or counting-based referenceless frequency detection in [30].

During clock and data recovery, the Rx front-end requires operation with maximum bandwidth and *DATA/EDGE* samplers at maximum clocking speed. A power-versus-performance tradeoff for these blocks is common in conventional architectures. The next section highlights problems with conventional designs and proposes a low power charge-based Rx front-end.

4.2 Rx Front-end

Figure 4.9 shows a simplified block diagram of a digital CDR with emphasis on the Rx front-end. It consists of a wide bandwidth amplifier to amplify the received signal before it is sampled by the following *DATA/EDGE* samplers. These samples are first synchronized and later deserialized to a rate low enough to be processed by the digital clock recovery loop. Power consumption in samplers and deserializer increases with the frequency of operation. The front-end samplers can be clocked at a lower frequency in sub-rate CDR architec-

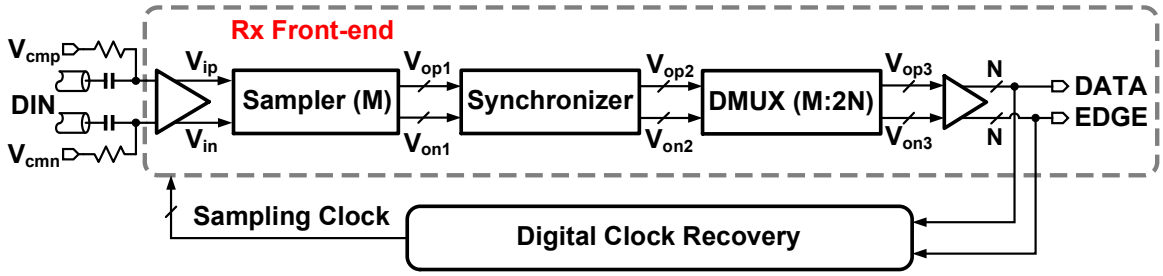


Figure 4.9: Simplified block diagram of a digital clock and data recovery.

tures [31] and save power compared to full-rate sampling in the CDR front-end [32]. But the low frequency operation is not necessarily energy-efficient for the receiver as a whole. This is because of the increased power consumption in multi-phase clock generation and distribution for sub-rate sampling in the front-end. In this work, CDR is implemented in half-rate topology while optimizing the overall power consumption in multiphase clock generation/distribution and Rx front-end together.

Given a sub-rate sampling in the front-end, a low deserialization ratio reduces the total deserialization power, but it increases power dissipation in digital clock recovery. Power dissipation in front-end samplers and deserializer amounts to a significant portion of the overall Rx power. To a large extent, front-end samplers and deserializer are based on regenerative amplifier. A regenerative amplifier amplifies the sampled input non-linearly to fixed output swings. Such amplifiers can be implemented with a current mode logic (CML) based latch or sense-amplifier-based flip-flop (SAFF) [13]. Unlike in a CML-based latch, power consumption scales with frequency in a SAFF, but parasitic capacitance at various circuit nodes experiencing rail-to-rail voltage swings increases inherent power dissipation. While the voltage swing is limited in a CML-based flip-flop, constant current itself leads to increased power. In this work, we seek to achieve the positives of both the prior flip-flops in a single flip-flop with low dynamic power dissipation and limited voltage swings at output and intermediate circuit nodes. In this bid, charge steered logic (CSL) provides a basic platform and is reviewed next.

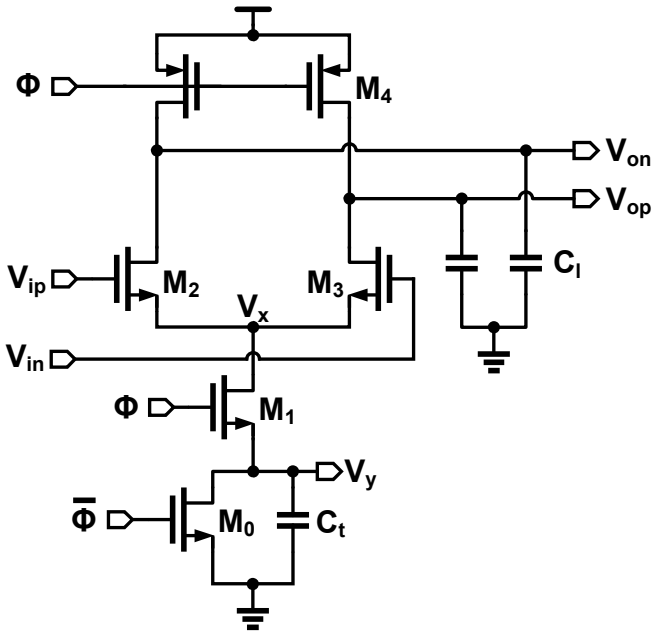


Figure 4.10: Schematic diagram of a charge-based amplifier (CbA).

4.2.1 Charge Steered Logic: Prior Art

Figure 4.10 shows a schematic of a charge-based amplifier (CbA) introduced in [3]. It operates in two phases, namely reset phase and active phase. Figure 4.11 shows transient signal waveforms at different nodes in CbA during the two phases. In reset phase (Fig. 4.11) clock signal Φ is low, outputs V_{op}/V_{on} are pulled to VDD, input pair M_2/M_3 are turned OFF by switch M_1, and tail capacitor C_t is discharged to GND through M_0. Source node of the input pair (V_x) is charged to $V_{cm} - V_{th} + |\Delta V|$ where V_{cm} is input common mode voltage, V_{th} is threshold voltage of the input pair, and ΔV is single-ended input swing. During active phase (Fig. 4.11), M_2/M_3 discharge V_{on}/V_{op} in proportion to input swings V_{ip}/V_{in}. As load capacitors (C_l) are discharged, charge is transferred from C_l to C_t and node potentials V_x and V_y rise. For a long active phase, V_x rises high enough to shut down M_2/M_3. In the end, CbA achieves limited swing operation at output and intermediate nodes with reduced power consumption. It is to be noted that there is no direct current sinking path between VDD and GND during the whole operation.

The CbA design has certain drawbacks if it is used as such while replacing regenerative amplifiers in CML-based latches or sense-amplifiers in SAFF. First, the amplifier gain is low

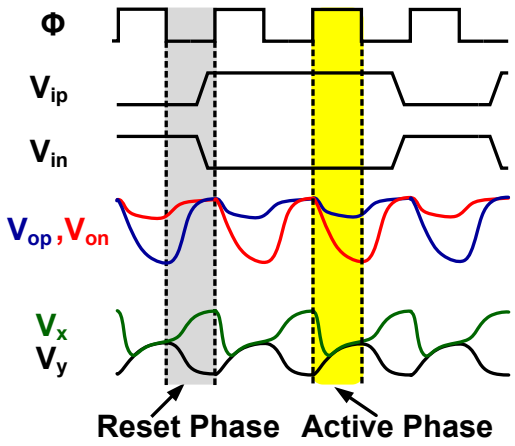


Figure 4.11: Transient signal waveforms in CbA.

and sensitive to input amplitude rather than polarity of the sampled input. This particular sensitivity to varying amplitude of the received signal makes it misfit for *DATA/EDGE* detection in digital CDRs. Second, the output of the amplifier is reset to differential zero value for a half clock period every clock cycle. Due to this zero reset value it requires multiple skewed clock phases to cascade CbAs, which becomes power intensive with multi-phase clock generation and distribution throughout sampling, synchronization, and deserialization.

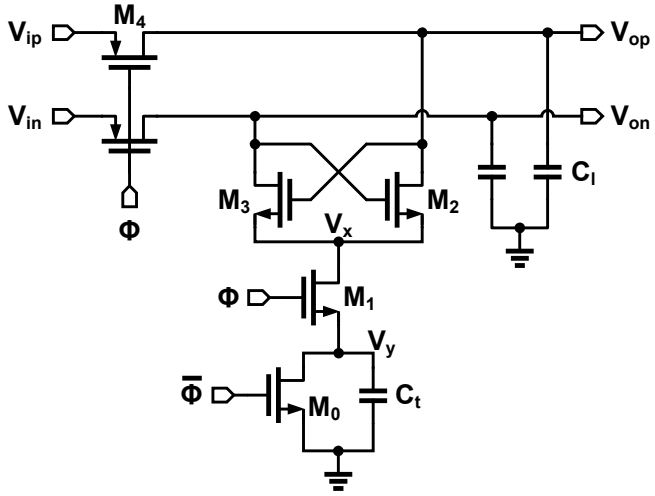


Figure 4.12: Schematic diagram of a charge-based NRZ latch in [3].

Figure 4.12 shows a non-return-to-zero (NRZ) charge steering latch used only in front-end sampling to overcome timing issues while mitigating time issues with high speed multi-phase clocks otherwise [3, 22]. It samples inputs on the cross-coupled NMOS pair when the clock is low and regenerates to a large swing when the clock goes high. In this manner

the latch maintains its output for one full clock period. But the latch suffers heavily from data-dependent ISI and requires a large current source to drive these latches. Therefore, this charge-based NRZ latch is not preferable for synchronization and deserialization stages. As such the charge-based amplifier and NRZ latch fail to replace a desired flip-flop with dynamic power consumption, limited input/output swings, and easy concatenation. Next, we propose a charge-based flip-flop which operates with limited input/output swings, delivers NRZ outputs, and uses only alternate clock phases for sampling and cascading.

4.2.2 Proposed Limited-swing Charge-based Flip-flop (LS-CFF)

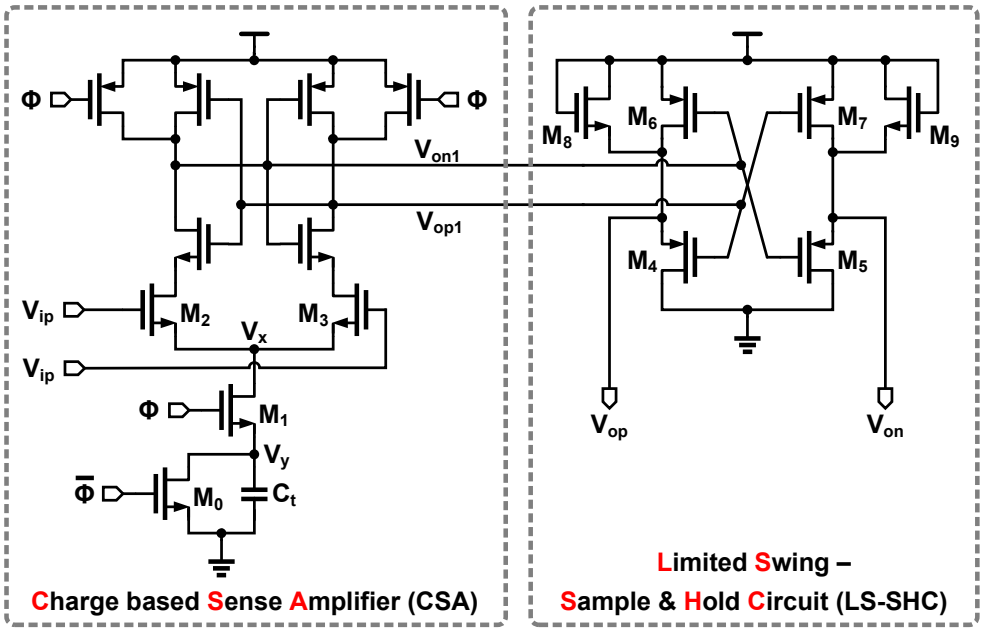


Figure 4.13: Schematic diagram of a proposed limited-swing charge-based flip-flop (LS-CFF).

Figure 4.13 shows a schematic of the proposed limited-swing charge-based flip-flop (LS-CFF) [33] which overcomes aforementioned problems. It consists of two parts, namely charge-based sense-amplifier (CSA) and a limited-swing sample-and-hold circuit (LS-SHC). CSA operates in two phases, specifically a reset phase and an active phase. During the reset phase, clock Φ is low and outputs V_{op1}/V_{on1} are reset to VDD as shown by the transient signal waveform in Fig. 4.14. Also, the input pair M_2/M_3 is switched OFF with source

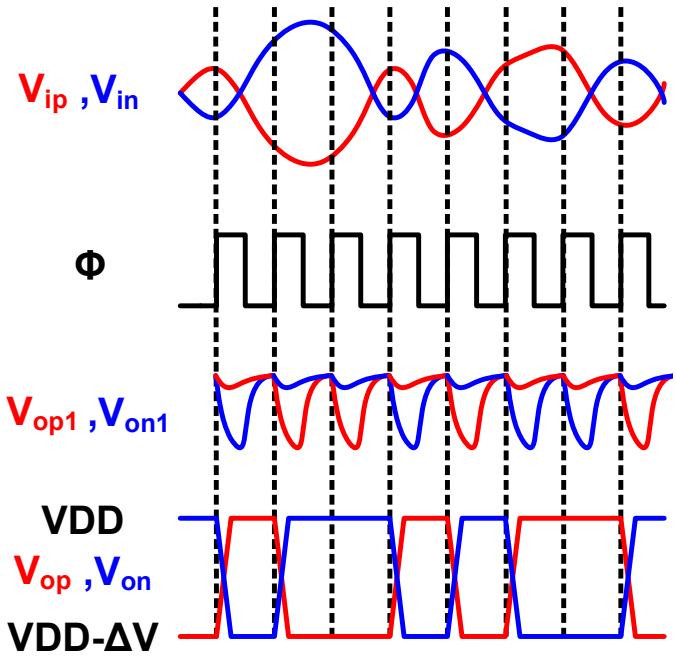
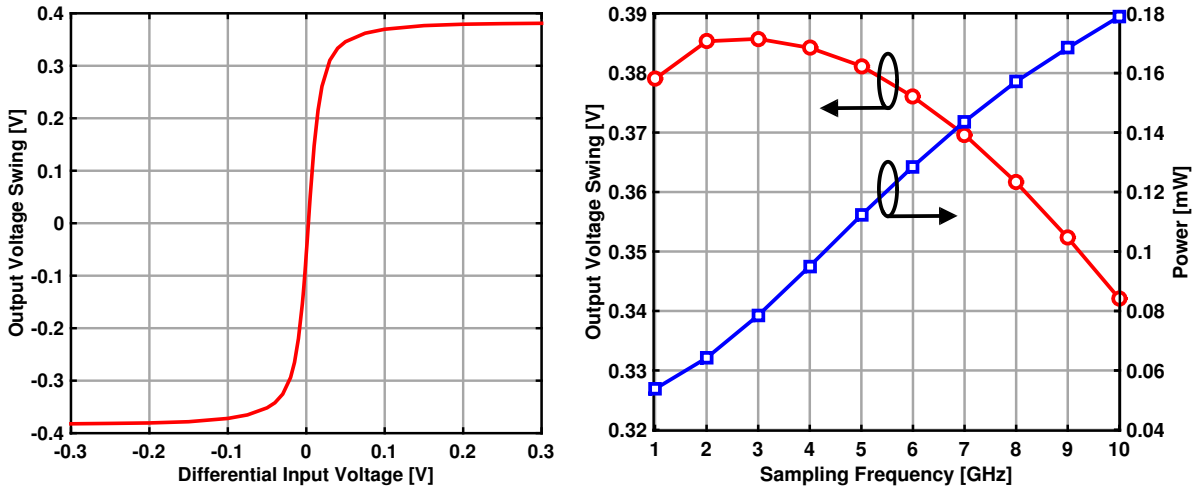


Figure 4.14: Transient signal waveforms in LS-CFF.

node V_x getting pulled to $V_{cm} - V_{th} + |\Delta V|$ and tail capacitor C_t getting discharged to *GND*. In active phase, inputs V_{ip}/V_{in} are sampled by M_2/M_3 and difference $V_{ip} - V_{in}$ is regenerated using cross-coupled inverters at CSA output. The regenerated output swing $|\Delta V|_{op,csa} = |V_{op1} - V_{on1}|$ is limited by charge transferred from V_{op1}/V_{on1} to C_t. In the extreme as $C_t \to \infty$, CSA behaves more like a regular sense-amplifier and $|\Delta V_{op,csa}| \to VDD$. For a finite C_t, CSA has a limited swing return-to-zero (RZ) output sensitive mostly to the polarity of the differential input sampled.

RZ outputs of CSA are sampled by PMOS transistors in the following sample-and-hold circuit (Fig. 4.13: LS-SHC). During active phase, PMOS transistor $M_6(M_7)$ charges output node $V_{op}(V_{on})$ to VDD whereas discharge at counter node $V_{on}(V_{op})$ is limited by threshold voltage of $M_5(M_4)$ and CSA output swings. Hence, the final output swing $(V_{op} - V_{on}) \propto (V_{op1} - V_{on1}) \propto \text{SIGN}(V_{ip} - V_{in})$. In reset phase, PMOS transistors $M_4 - M_7$ are switched OFF with their gates pulled to VDD. And NMOS transistors $M_8 - M_9$ help retain the sampled output during the reset phase. In the end, the CFF achieves a limited input/output swing operation and dynamic power consumption.

To verify CFF operation, LS-CFF is designed in TSMC 65 nm CMOS technology. The

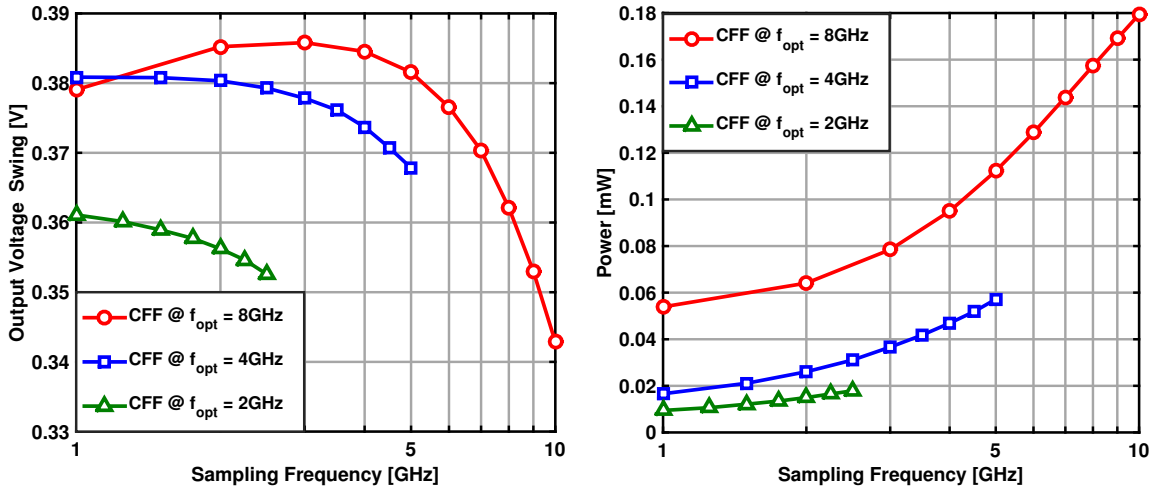


(a) (b)

Figure 4.15: (a) Differential output voltage swing vs. input voltage for LS-CFF clocked at 7 GHz. (b) Output voltage swing and power consumption vs. sampling frequency for LS-CFF.

flip-flop is simulated with rc-extracted parasitics, 10 fF external load, sinusoidal sampling clock at 7 GHz, 1 V supply voltage, and variable input voltage swing across simulations. Figure 4.15(a) shows simulated differential output voltage swing for LS-CFF. The output swing is limited and it saturates for differential inputs roughly $\geq$ 50 mV. When the same flip-flop is clocked at different frequencies with 100 mV differential input, Fig. 4.15(b) plots output average voltage swing and power consumption w.r.t. sampling frequency. Though the variation in output voltage swing is minimal, it is inversely proportional to sampling frequency (f_{clk}) over a large frequency range. This is because the gain for a sample-and-hold circuit is proportional to available discharge period ($\propto 1/f_{clk}$). The output swing drops at much lower frequencies due to increased leakage during the reset phase. This problem can be tackled by optimizing flip-flop design over a range of frequencies. LS-CFF can operate as a standard flip-flop with limited input/output swings as new digital levels. The output swings are maintained across PVT variations by controlling tail capacitor C_t value. Power consumption in LS-CFF varies linearly with frequency as shown along the y-axis on the right in Fig. 4.15(b). Here, clocking power is not included for the sampler, but it is comparable to clocking power in standard SAFF.

It is to be noted that charge steered form one point to another is a function of capacitance



(a) (b)

Figure 4.16: (a) Output voltage swing vs. frequency, and (b) power consumption vs. frequency for optimized LS-CFF.

at circuit nodes and time allotted for charge sharing ($\propto 1/f_{clk}$). Hence, power consumption in CFF can be minimized based on load capacitance and frequency of operation for the same output swing. Figure 4.16(a) shows variation in output voltage swing with sampling frequency when the design of the sampler is optimized at three independent sampling frequencies, namely 8 GHz, 4 GHz, and 2 GHz, respectively. Here, each sampler is simulated with rc-extracted parasitics, 10 fF external load capacitance, 1 V supply voltage, and 100 mV differential input. The plot confirms roughly the same output swing for the three samplers. An actual benefit from such optimization can be estimated by comparing the power consumption in these samplers at the same sampling frequency. Figure 4.16(b) reaffirms the reduction in power consumption at a given frequency for optimized samplers with roughly the same output swing. Similar to standard SAFF, LS-CFF is a building block for sampler, synchronizer, and deserializer (DMUX) in the Rx front-end. Frequency optimized design for LS-CFF leads to significant energy savings in the Rx front-end. At the very end of the deserialization stage, outputs need to be scaled to full-swing signals for digital processing with standards CMOS logic. This is possible with full-swing charge-based flip-flop as proposed next.

4.2.3 Full-swing Charge-based Flip-flop (FS-CFF)

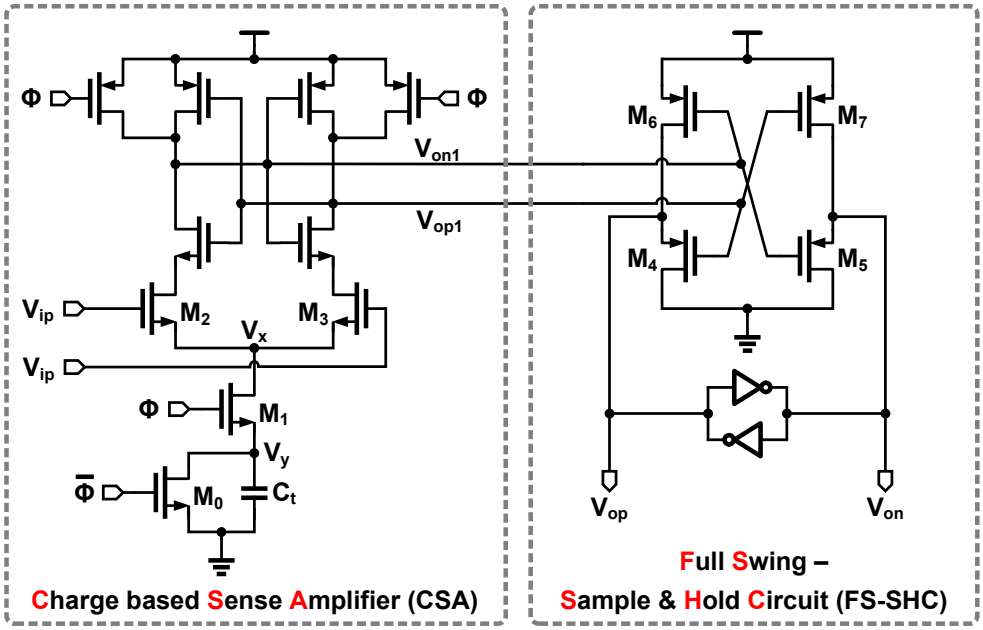


Figure 4.17: Schematic diagram of a full-swing charge-based flip-flop (FS-CFF).

Figure 4.17 shows a circuit diagram of a full-swing charge-based flip-flop (FS-CFF). It consists of two parts. The first part is a charge-based sense-amplifier (CSA) which is the same as that used in LS-CFF. A full-swing sample-and-hold circuit (FS-SHC) follows the CSA. In contrast to LS-CFF, SHC in FS-CFF employs an inverter-based cross-coupled latch at its output. PMOS transistors in SHC sample CSA outputs during active phase and regenerate it to rail-to-rail swings. Unlike the LS-SHC, a cross-coupled latch at FS-SHC output slows down the output transition. But, with larger time margins at the end of the deserialization stage, outputs are reliable for further use over a wide sampling period. Figure 4.18 shows variation in average output voltage swing and power consumption w.r.t. sampling frequency. Output voltage swing is constant at *VDD* as shown along the y-axis on the left, and power consumption scales linearly with frequency as depicted along the y-axis on the right. LS-CFF and FS-CFF are energy-efficient building blocks for the Rx front-end as discussed next.

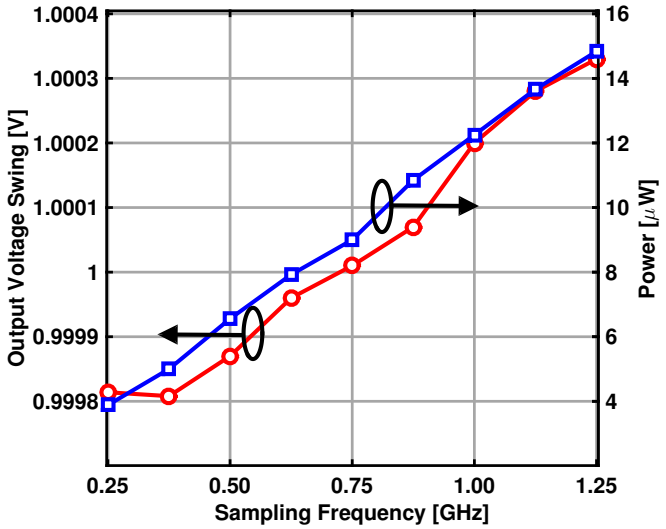


Figure 4.18: Output voltage swing and power consumption vs. sampling frequency for FS-CFF.

4.2.4 Charge-based Rx Front-end

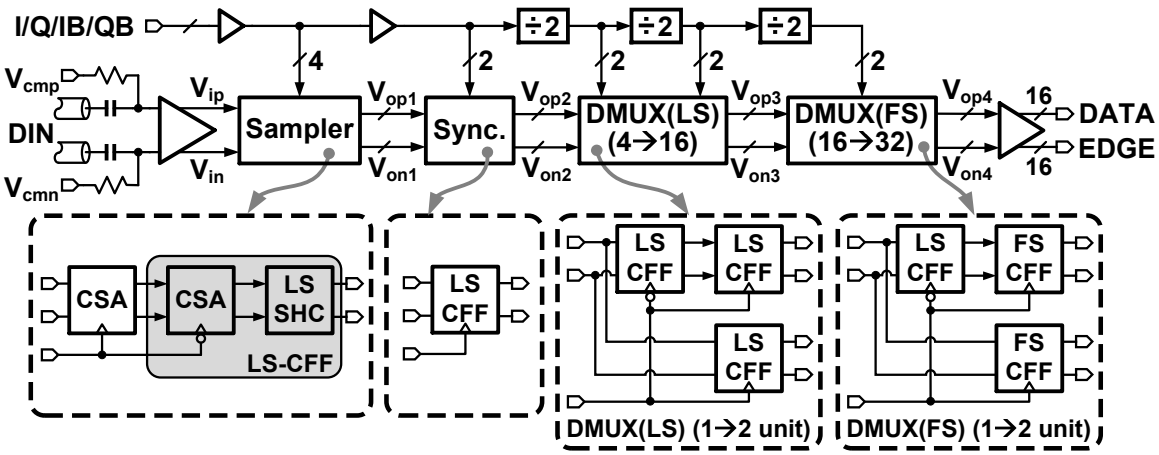
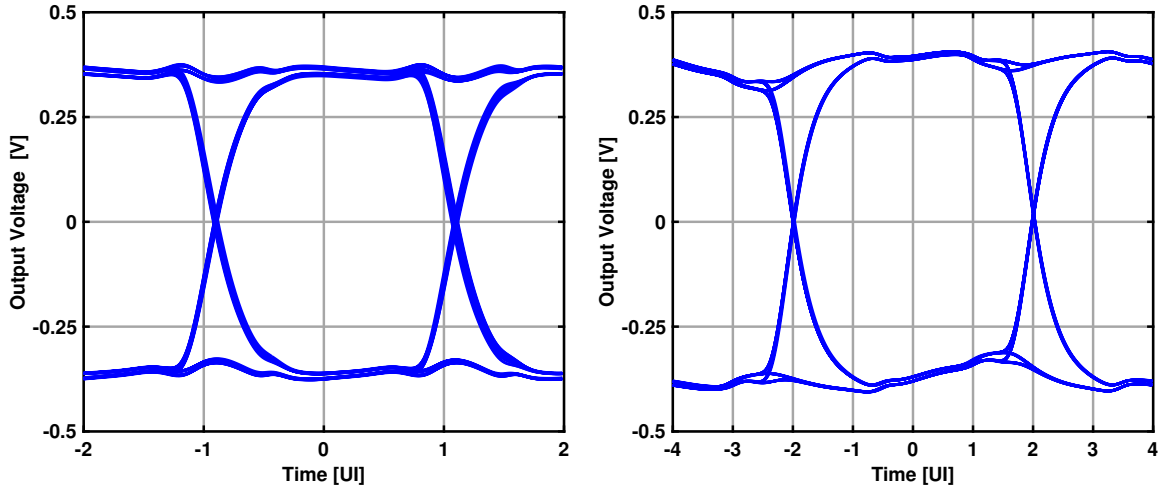


Figure 4.19: Detailed block diagram of the Rx front-end.

Figure 4.19 shows a detailed block diagram of the receiver front-end. It consists of a wide bandwidth amplifier driving *DATA/EDGE* samplers at half-rate with I/Q/IB/QB clock phases. The front-end samplers make use of series-connected CSAs similar to series-connected sense-amplifiers in [34]. The series-connected CSAs help in improving the sensitivity of samplers which operate on low amplitude received signals. At half-rate clocking, the four sampled values even(odd) *DATA/EDGE* are separated by a quarter clock cycle. These samples are synchronized by LS-CFF. The synchronized samples are first deserialized by a

factor of 4:8 and then 8:16 while using LS-CFF based 1:2 DMUX units (Fig. 4.19). Each stage of deserialization like 4:8(8:16) operates at a single clock frequency. Hence, each stage makes use of optimized LS-CFF for a given sampling frequency. The last deserialization stage uses FS-CFF as shown in Fig. 4.19 to restore full-swing levels for standard CMOS logic in the clock recovery loop. It is to be noted that sampling, synchronization, and deserialization are all accomplished with charge-based limited-swing dynamic logic without any intermediate buffers till the very end of the front-end.



(a) (b)

Figure 4.20: (a) Eye diagram at the output of front-end sampler. (b) Eye diagram at the output of first deserialization stage.

Figure 4.20(a) shows a simulated eye diagram at the output of front-end samplers. The differential output amplitude is roughly 400 mV and valid for 2 bit unit intervals (UIs) as shown. At the end of the first deserialization stage, outputs extend over 4 bit periods as shown in Fig. 4.20(b).

An important point to note is that clock signals have rail-to-rail swings unlike limited swing input/output signals for CFFs. Sampling clocks at different deserialization stages are generated using full-swing standard TSPC logic based clock divider. Now, the CLK-to-Q delay through TSPC-based divider and charge-based deserialization stages are different and it can be a problem for sampling during deserialization. To overcome timing issues with divided clocks at different deserialization stages, a divider with tunable delay is used

as shown in Fig. 4.21. Based on the *Sel* signal, the divider selects between rising edges of Φ_{in} and $\overline{\Phi_{in}}$ input to the divider and the output gets delayed by half of the input clock period. This tunable option helps to sample inputs in the charge-based DMUX with sufficient time margins across PVT variations. The *Sel* signal is evaluated to be 1/0 during pattern synchronization at the output of the deserializer.

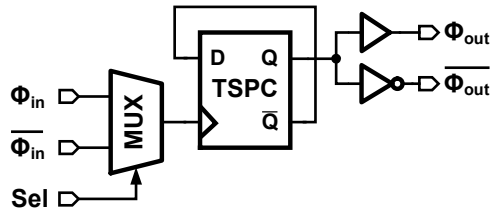


Figure 4.21: Block diagram of clock divider with tunable delay.

4.3 Voltage-mode Transmitter

A voltage-mode (VM) transmitter is an energy efficient method of signaling on the Tx side compared to current mode signaling. Current mode logic (CML) based output drivers consume four times more current than VM output drivers for the same output swing. However, much-needed equalization on the Tx side increases power consumption in VM Tx. This is specifically due to the way equalization is implemented in the VM O/P driver.

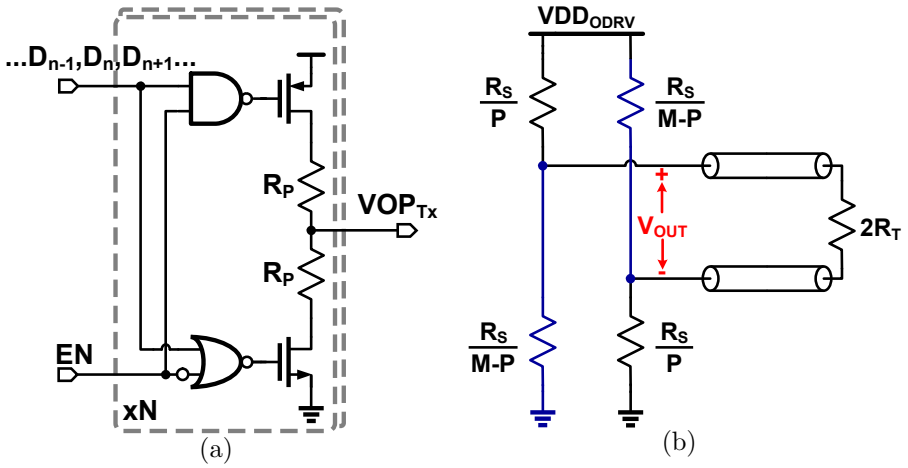


Figure 4.22: (a) Schematic diagram of single side of a uniformly segmented differential VM O/P driver. (b) Simplified schematic of VM O/P driver with embedded equalization.

Figure 4.22(a) shows a typical block diagram of a source-series terminated single side of a differential voltage-mode output driver [9]. The output driver is uniformly segmented into N segments, thereby incorporating channel termination and equalization. Each segment consists of PMOS/NMOS switches and poly-resistor $\mathrm{R_P}$. Total number of segments M ($\leq N$) active at any given time depends on the output impedance of each segment and impedance required for channel termination, $\mathrm{R_T}$. M selected segments are further distributed among bit controls $.., \mathrm{D_{n-1}}, \mathrm{D_n}, \mathrm{D_{n+1}}, ..$ for feedforward equalization. By design, output impedance of each segment is $\mathrm{R_S} = \mathrm{MR_T}$, so that the channel is terminated with impedance $\mathrm{R_S}/M = \mathrm{R_T}$ which is preferably its characteristic impedance. Consider a simplified circuit diagram of the above differential voltage-mode output driver with embedded equalization as shown in Fig. 4.22(b). The output driver is differentially terminated with impedance $2\mathrm{R_T}$. It implements equalization by transmitting variable output swing depending on bit sequence. During one of the equalization settings, P and M-P segments are switched to $\mathrm{VDD_{ODRV}}$, and M-P and P segments are switched GND on two sides of the differential output driver, respectively. Differential output voltage swing for Tx is given by eq. 4.5.

$$\mathrm{V_{OUT}} = \mathrm{VDD_{ODRV}} \left(\frac{\mathrm{P}}{\mathrm{M}} - \frac{1}{2} \right) \tag{4.5}$$

Now, resolution of equalization is $\Delta\mathrm{V_{OUT}}$ when one segment is redistributed among different equalization taps. For $\Delta\mathrm{P} = 1$, change in output voltage swing is given by eq. 4.6.

$$\Delta\mathrm{V_{OUT}} = \mathrm{VDD_{ODRV}}/\mathrm{M} \tag{4.6}$$

The resolution of equalization is inversely proportional to the total number of segments, M, and the total number of segments for a given resolution, $\Delta\mathrm{V_{OUT}}$, is upper-bounded by channel characteristic impedance. So, if M is increased to enhance the resolution of equalization, the pre-driver driving each O/P driver segment does not scale down proportionally beyond the technology-limited dimensions. This results in increased power consumption in pre-drivers [2]. Hence, there exists a power-versus-resolution tradeoff in voltage-mode equalization.

It has been observed that in ISI dominated signal transmission, easily implementable 2-tap FIR equalization in VM Tx has peculiar characteristics. Peak distortion analysis while equalizing channel loss with 2-tap FIR equalizer reveals that the main cursor coefficient is always greater than pre-/post cursor taps. If the sum of all equalizer coefficients is 1 for FIR-equalization on the Tx side, then the main cursor coefficient is > 0.5. This helps in non-uniform segmentation of equalizer taps during FIR equalization. It further leads to non-uniform segmentation of pre-driver and saving pre-driver power without compromising on resolution of equalization. Using this fact, a partially segmented voltage-mode transmitter is proposed next.

4.3.1 Proposed Partially Segmented VM Transmitter

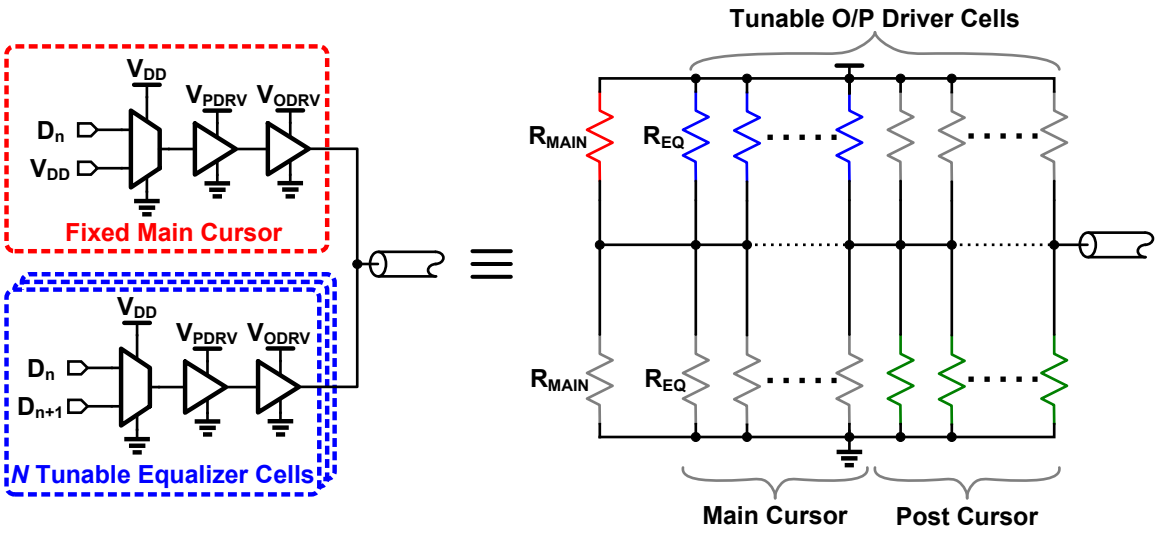


Figure 4.23: Block diagram of proposed partially segmented VM Tx w/ embedded de-emphasis.

Figure 4.23 shows a block diagram of a partially segmented voltage-mode Tx implementing 2-tap equalization. For simplicity it shows only a single-ended version of a differential output driver. The VM Tx consists of two blocks, namely fixed main cursor and a block containing N uniformly segmented tunable equalizer cells. O/P driver and pre-driver cells in each block operate at supply voltages V_{ODRV} and V_{PDRV}, respectively. Each O/P driver segment is a source-terminated resistive segment similar to segments in Fig. 4.22(a). The fixed main

cursor is controlled by bit D[n] which has an output impedance R_{MAIN} ($\leq 2R_T$) shown in the simplified resistive equivalent of the O/P driver on right. Tunable cells are shared between bits D[n] and D[n+1] to control de-emphasis for a given channel loss. Each tunable cell has an output impedance R_{EQ} ($\geq N \times 2R_T$). Hence, the total number of tunable segments is roughly halved compared to uniform segmentation for the same resolution (V_{ODRV}/N). The voltage generated by active resistors in the voltage divider is transmitted as a part of equalization. It is to be noted that total output impedance of the driver is $R_{MAIN}||R_{EQ}/N$ and matched to channel characteristic impedance R_T.

Power consumption in the pre-driver cell for each tunable equalizer cell is minimized with optimal sizing to drive the O/P driver cell. The fixed main cursor O/P driver is a scaled version of the tunable O/P driver cell. When a number of uniform equalizer segments are merged together to form a fixed main cursor, pre-driver sizing is not limited by technology, and parasitics for the unified pre-driver are minimized thereby reducing power dissipation in the pre-driver. Input to output delays for fixed main cursor path and tunable cell path should be matched for minimal circuit-induced ISI. Transmit output swing is governed by V_{ODRV} and output impedance of VM O/P driver is controlled by V_{PDRV}.

4.4 Proposed 14 Gb/s Serial Link

Figure 4.24 shows a block diagram of a 14 Gb/s transceiver (XCVR) incorporating the above proposed system and circuit-level innovations. The XCVR consists of an LC-DPLL, partially segmented VM transmitter, half-rate clock and data recovery, and ring PLL based multi-phase generator on the Rx side. Digital PLL employing LC-oscillator generates low jitter 7 GHz clock which is shared between Tx and Rx on the same chip. A divided 7 GHz clock is used to generate 16 parallel streams of random data at 0.875 Gb/s by an on-chip PRBS generator. A 16:1 serializer serializes these low-rate parallel streams to 14 Gb/s full-rate data. On the Tx side, an N-over-N based partially segmented VM O/P driver is realized with embedded 2-tap de-emphasis as discussed at block level in Section 4.3. Tx output swing is controlled by regulating supply voltage V_{odrv} for the O/P driver, and output impedance of the O/P driver is matched to channel characteristic impedance by regulating supply voltage

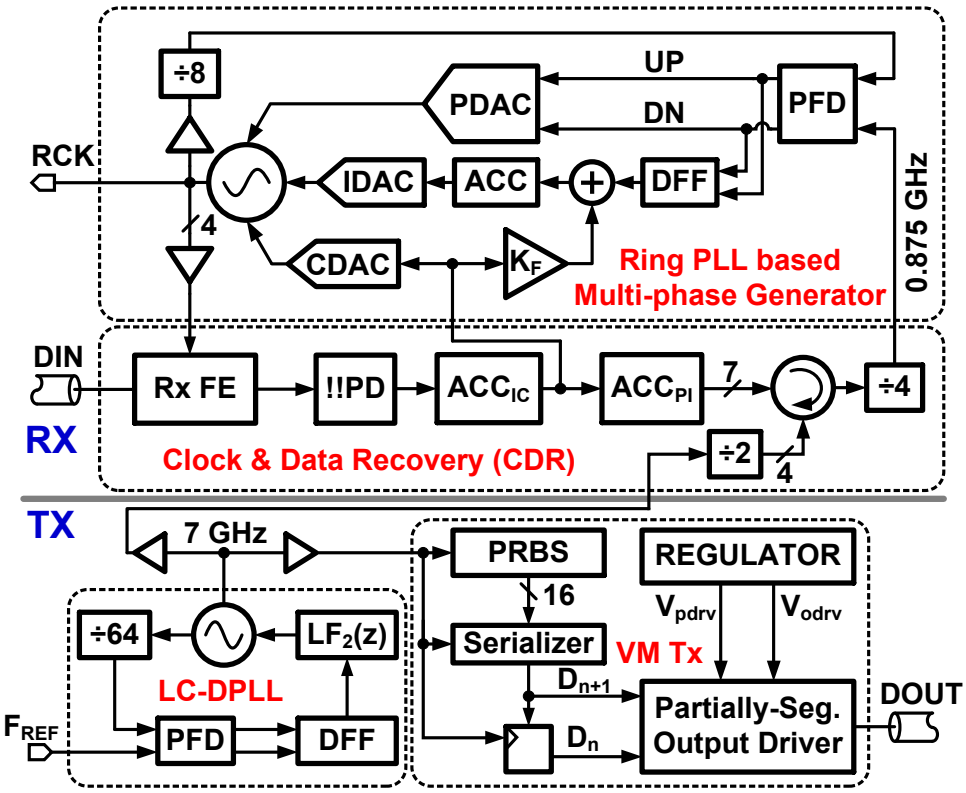


Figure 4.24: Block diagram of proposed 14 Gb/s transceiver.

of the pre-driver (V_{pdrv}) for a given peak-to-peak differential output swing (V_{odrv}).

On the Rx side, clock and data recovery is implemented in half-rate architecture for 14 Gb/s data input. Here, front-end samplers are clocked with I/Q/IB/QB clock phases which are generated by a ring PLL (RPLL) based multi-phase generator. The Rx front-end includes a wide bandwidth front-end amplifier, charge-based *DATA/EDGE* samplers, and 4:32 charge-based deserializer (Section 4.2). The deserialized information is processed by a synthesized !!PD which generates *Early/Late* signals signifying sampling phase error. !!PD output is filtered with an accumulator ACC_{IC}. The filtered output corrects for sampling phase error by adjusting output phase of RPLL through two different paths. In the first path, it controls the ring oscillator in RPLL through a current DAC (CDAC). In the second path, it adds the accumulated CDR phase error to integral path in RPLL with gain K_F. Also, the filtered output corrects for sampling frequency error by changing the reference frequency for RPLL through a phase interpolator (PI). PI derives its four phases for interpolation by dividing the shared 7 GHz clock by a factor of two. ACC_{PI} provides 7-bit interpolation code

for PI. PI interpolates clock phases at 3.5 GHz with 2.2 ps average resolution. PI output is divided by four providing a reference clock to RPLL. RPLL reference frequency is low enough for a feasible low power operation of RPLL building blocks including PFD, DFF, PDAC, IDAC, and accumulator. At the same time, the reference frequency is large enough to suppress ring oscillator phase noise with larger PLL bandwidth. Implementation details of key building blocks of the transceiver are discussed below.

4.4.1 LC-based Digital Phase Locked Loop (LC-DPLL)

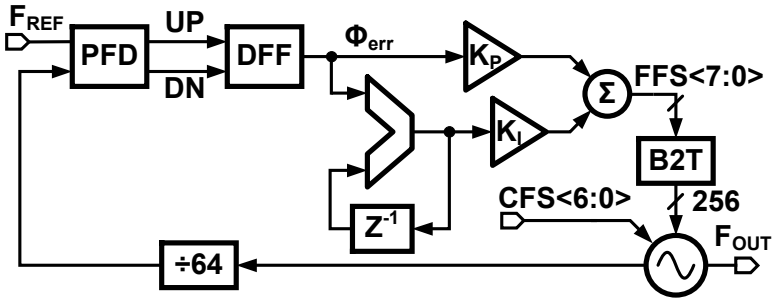


Figure 4.25: Block diagram of LC oscillator based digital PLL.

Figure 4.25 shows a block diagram of an LC oscillator based digital phase locked loop. DPLL generates a low jitter clock F_{OUT} at 7 GHz using an external reference F_{REF} at 109.375 MHz. Phase/frequency error between reference F_{REF} and divided output clock ($F_{OUT}/64$) is measured by an analog phase and frequency detector (PFD). PFD output UP-DN is digitized using DFF to give one bit error information Φ_{err} [20]. Φ_{err} is processed by a digital loop filter with gains K_P and K_I in proportional and integral paths, respectively. The 8-bit binary loop filter output $FFS\langle 7:0\rangle$ is converted to thermometer coded 256 fine frequency select signals to vary LC oscillator output frequency with 20 ppm resolution at 7 GHz. A digitally controlled LC oscillator with cross-coupled load on either side of the LC-tank provides near to full swing output amplitude and is designed similarly to the LC oscillator in [35]. An additional 7-bit binary coarse frequency select (CFS) signal tunes output frequency with resolution 300 ppm around 7 GHz. LC-DPLL output is later shared between Tx and Rx on the same chip.

4.4.2 Voltage Mode Transmitter

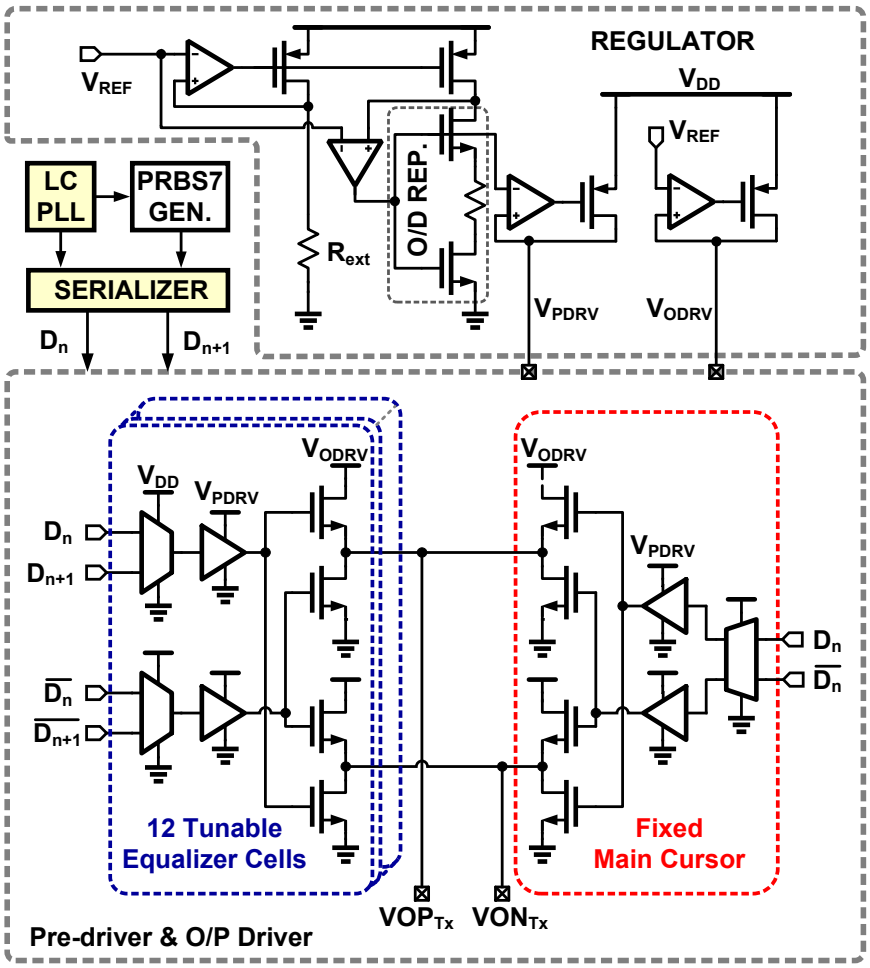


Figure 4.26: Block diagram of proposed voltage mode transmitter.

Figure 4.26 shows a detailed block diagram of a low swing VM transmitter implementing a N-over-N based partially segmented VM O/P driver. It consists of LC-DPLL, on-chip PRBS generator, 16:1 serializer, partially segmented O/P driver, and a voltage regulator. LC-DPLL output is divided to provide clocks for PRBS generator at 0.875 Gb/s and different stages of 16:1 serialization. 16:2 serialization is achieved using standard TSPC latches and transmission gate 2:1 MUX, whereas the final 2:1 serialization stage is realized using split-load 2:1 MUX in [36]. The split-load 2:1 MUX reduces data-dependent ISI with the same clock-to-Q delay for bits 1/0. The serialized full-rate data D_n/D_{n+1} is fed to partially segmented O/P driver.

The O/P driver implements 2-tap equalization by realizing a resistor-based voltage divider at Tx output and controlling voltage division by bits D_n/D_{n+1}. The driver consists of two main parts, namely a fixed main cursor and twelve tunable equalizer cells. The total output impedance of the O/P driver is tuned to 50 Ω by controlling supply voltage of pre-driver (V_{pdrv}) [37]. Each tunable cell can be selected between current bit D_n (main cursor) and next bit D_{n+1} (pre-cursor) for canceling pre-cursor ISI in this particular implementation. Pre-cursor and/or post-cursor dominance and their suppression during equalization depend on the channel. Nevertheless, the number of cells available for selection between main cursor and pre-/post cursor taps is limited to half the number of cells needed otherwise for uniformly segmented O/P driver. In fixed main cursor, pre-driver is optimized for main cursor coefficient > 0.5

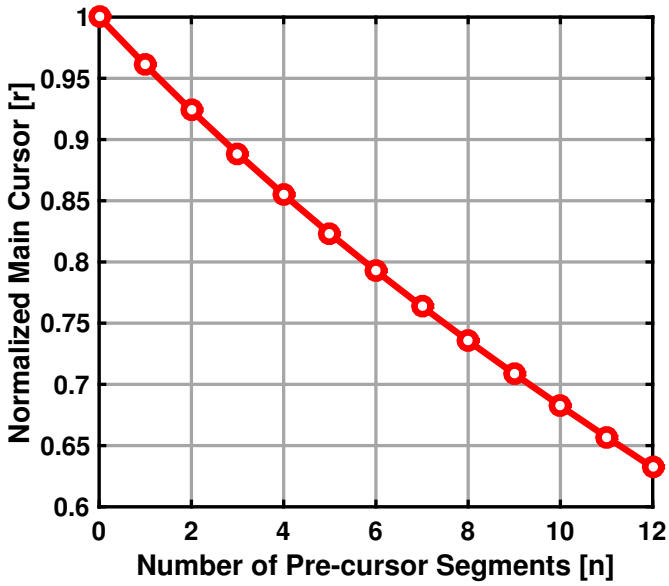


Figure 4.27: Variation in main cursor tap weight with varying equalizer tap weight.

Consider the above 2-tap equalization with coefficients r and -(*1*-r) for main cursor and pre-cursor, respectively. Figure 4.27 shows variation in total main cursor tap weight (r) while re-assigning tunable main cursor taps to pre-cursor taps (n). The variation in r is shown for the partially segmented O/P driver with rc-extracted parasitics. An average of 3 % resolution in equalizer coefficient r leads to 6 % ($= 2r$) resolution in output swing control. It would require a total of roughly 33 cells to implement uniformly segmented O/P driver with 3 % tap weight resolution. In partially segmented VM Tx, the total number of segments can be

minimized thereby reducing power consumption in the pre-driver.

4.5 Measurement Results

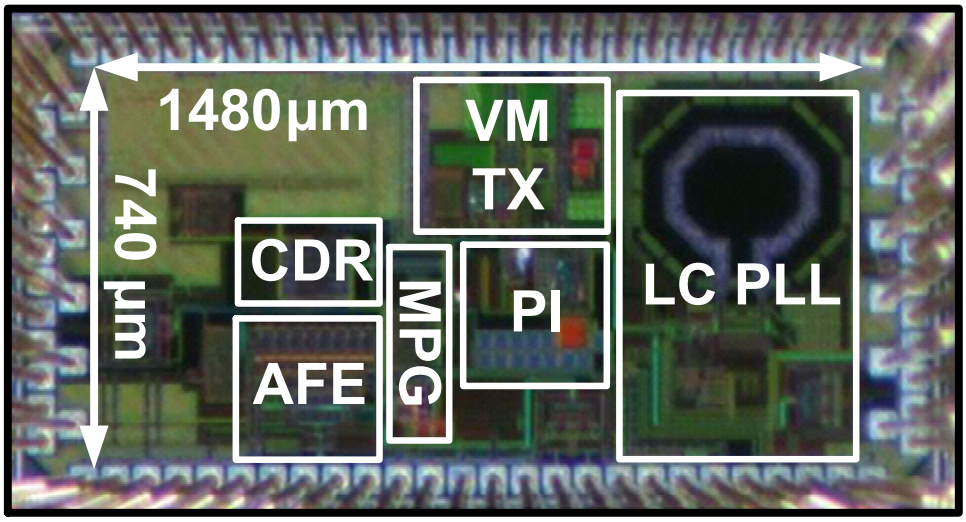


Figure 4.28: Die photo of the transceiver prototype.

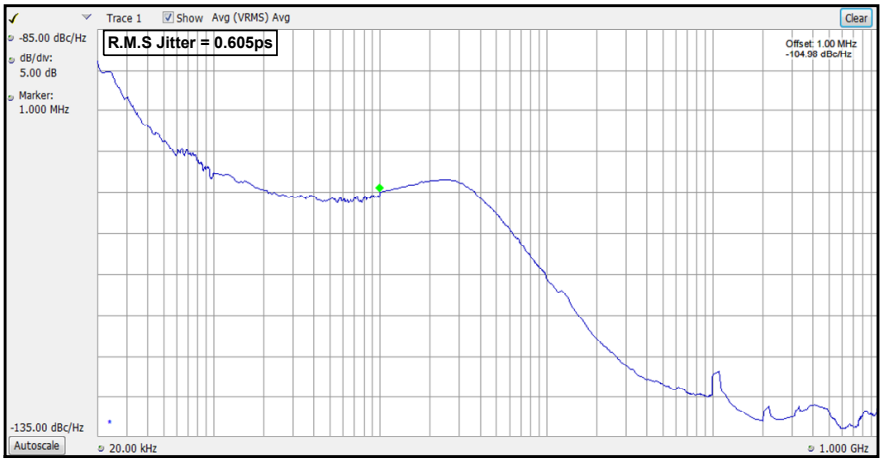


Figure 4.29: Phase noise spectrum of LC-DPLL locked at 7 GHz.

The proposed transceiver was implemented in TSMC 65 nm CMOS process. Figure 4.28 shows a chip micrograph of the prototype. It occupies an active area of 1.1 mm^2. The die was packaged in a standard 88 pin plastic QFN package and characterized using a four-layer FR-4 printed circuit board. The prototype was tested for peak data rate of 14 Gb/s. First, LC-DPLL is locked at 7 GHz using an external reference clock at 109.375 MHz from AWG7122C.

Figure 4.29 shows phase noise spectrum at the output of LC-DPLL and measured using Tektronix RSA5126A spectrum analyzer. When phase noise is integrated over frequency range 20 kHz-1 GHz, the DPLL has 0.605 ps_{rms} as integrated jitter. The peaking in phase noise spectrum and increased jitter is due to limited on-chip digital control for proportional and integral paths. The 7 GHz LS-DPLL output is shared between Tx and Rx on the same chip.

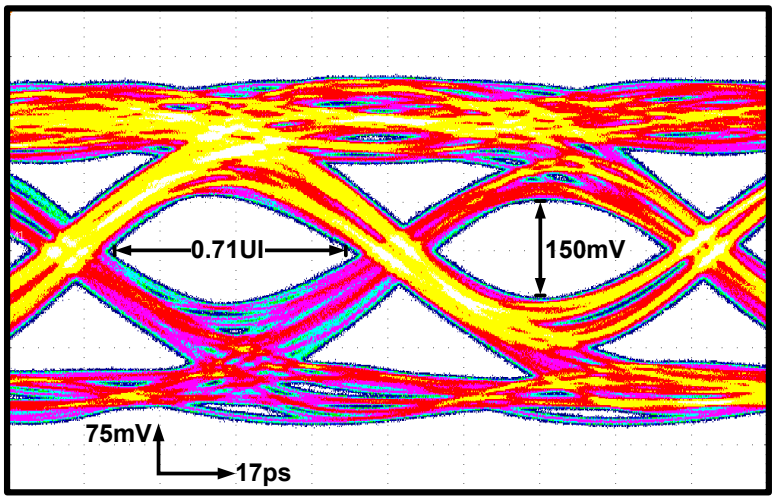


Figure 4.30: Differential transmit eye diagram at 14 Gb/s.

Using the above clock, on-chip 7-bit PRBS generator, and 16:1 serializer, a 14 Gb/s PRBS7 data pattern is realized and transmitted through a partially segmented voltage-mode output driver. Figure 4.30 shows a differential eye diagram measured at the output of Tx. The Tx is configured to transmit 0.4 V_{pp} differential amplitude. Tx signal path includes bond wire, package parasitics, μ-stripline on test board, SMA connectors, and SMA cables. Tx output has a vertical eye opening of 150 mV and horizontal eye opening of 0.71 UI. Figure 4.31 shows BER bathtub plots at the output of external FR-4 stripline channel driven by the VM Tx. The total channel loss in the signal path is 11 dB at 7 GHz. BER bathtub plots are measured using 80SJNB BER testing software available with Tektronix DSA8300. Without equalization the sampling time margin is < 0.1 UI for BER $< 10^{-12}$. When 2-tap FIR equalizer is enabled, sampling time margin improves to 0.36 UI at BER $< 10^{-12}$.

In an independent test setup, ring PLL is locked with a 0.875 GHz reference clock generated using an LC-DPLL and a phase interpolator. Using a fixed control code, PI interpolates

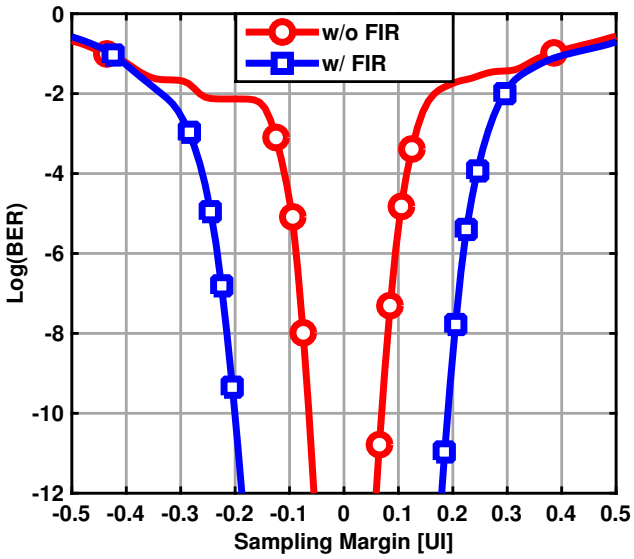


Figure 4.31: Bathtub plots at the output of channel w/ and w/o FIR equalization.

between 3.5 GHz I/Q clock phases obtained after dividing LC-DPLL output. PI output is further divided by four to provide 0.875 GHz reference clock. Figure 4.32 shows phase noise spectrum of the ring PLL output at 7 GHz. The ring PLL has an integrated r.m.s. jitter of 1.511 ps when phase noise is integrated over 20 kHz-1 GHz. From the phase noise spectrum, RPLL bandwidth is roughly greater than 30 MHz and surely greater than the desired CDR bandwidth.

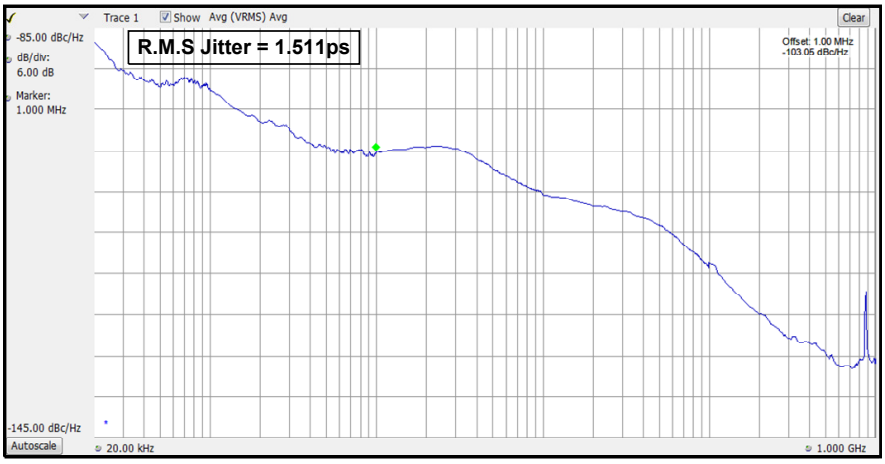


Figure 4.32: Phase noise spectrum of ring PLL output locked at 7 GHz.

Figure 4.33 shows clock waveform of the recovered clock when the receiver is locked to 14 Gb/s PRBS7 data fed externally from the bit error rate tester (BERT). The data is re-

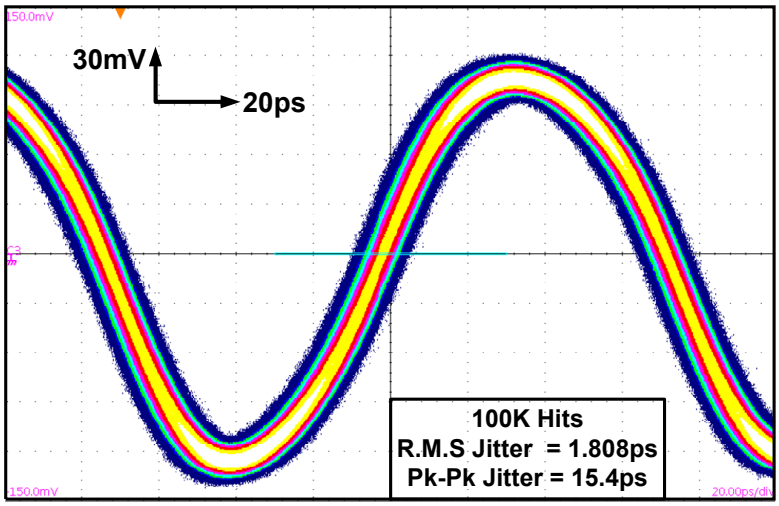


Figure 4.33: Recovered clock waveform.

covered with BER$< 10^{-12}$. Long term absolute jitter of the recovered clock is 1.808 ps_{rms} and 15.4 ps_{pp}, respectively. The digital clock and data recovery loop have unwanted significant clock delays during digital processing of phase error information, which results in increased jitter unless dampened by low loop bandwidth. In measurements, CDR loop bandwidth is reduced to counter loop delay [38]. CDR measurement results shown here are affected by loop delay, but the problem can easily be rectified in the next silicon revision. Figure 4.34 shows a jitter transfer (JTRAN) plot for CDR locked to data from external BERT. A low amplitude sinusoidal jitter is applied to measure JTRAN. A low CDR bandwidth, for the reason mentioned before, is observed here.

For jitter tolerance (JTOL) measurement, stressed 7-bit PRBS data at 7 Gb/s is fed to Rx from an external BERT. The data recovery is still at 14 Gb/s and a 7 GHz clock is recovered for the half-rate architecture. Half-rate stressed data restricts the data-dependent ISI in the Rx front-end. A minimum bit error rate of 10^{-12} is observed at 14 Gb/s for a certain peak-to-peak jitter applied at a given frequency. Figure 4.35 shows a jitter tolerance plot for the proposed CDR w.r.t. frequency of jitter applied. Y-axis shows peak-to-peak UI jitter applied when normalized to 14 Gb/s. The dip in JTOL curve is due to the large loop delay as mentioned before and analyzed in [38]. High frequency JTOL is limited to 1 UI as CDR loop response is limited by its bandwidth.

The taxonomy of power consumption in the proposed 14 Gb/s transceiver is shown in

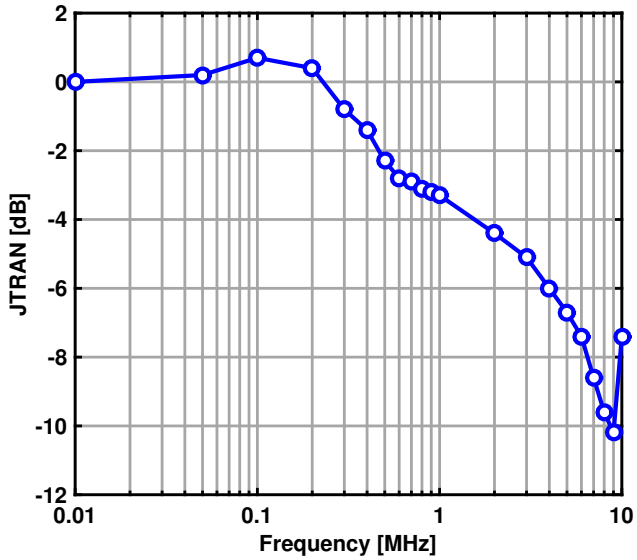


Figure 4.34: JTRAN curve for the proposed CDR locked at 14 Gb/s.

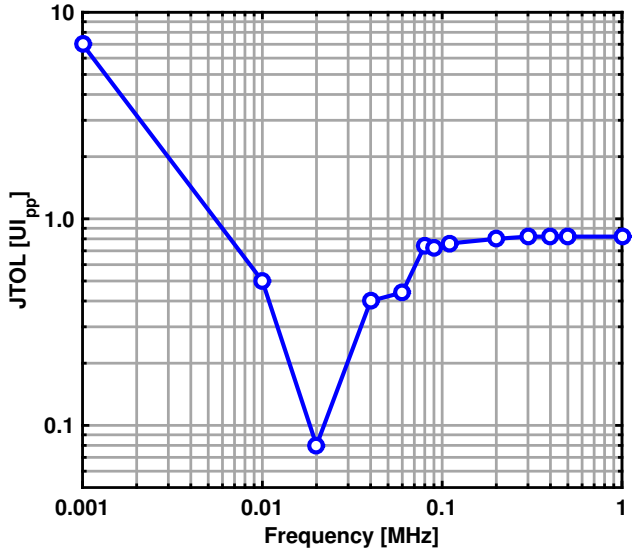


Figure 4.35: JTOL plot measured with stressed data input.

Fig. 4.36. It consumes a total power of 39.6 mW achieving energy efficiency of 2.8 mW/Gb/s. LC-DPLL generates on-chip 7 GHz clock while consuming only 3.6 mW at 1 V supply voltage. On the Tx side, the serializer consumes 5.9 mW from a 1.1 V supply. Supply regulated voltage-mode O/P driver, pre-driver, and 2-tap equalizer together with low dropout regulator consume 6.3 mW at 1 V supply. Phase interpolator, ring PLL, and digitally processing CDR block dissipate 2 mW, 3.2 mW, and 2 mW, respectively, at 1 V supply voltage. The Rx front-end consumes 16.6 mW from a 1.15 V supply. Table 4.1 summarizes the performance of the proposed 14 Gb/s transceiver and compares it with state-of-the-art transceivers operating at similar data rates.

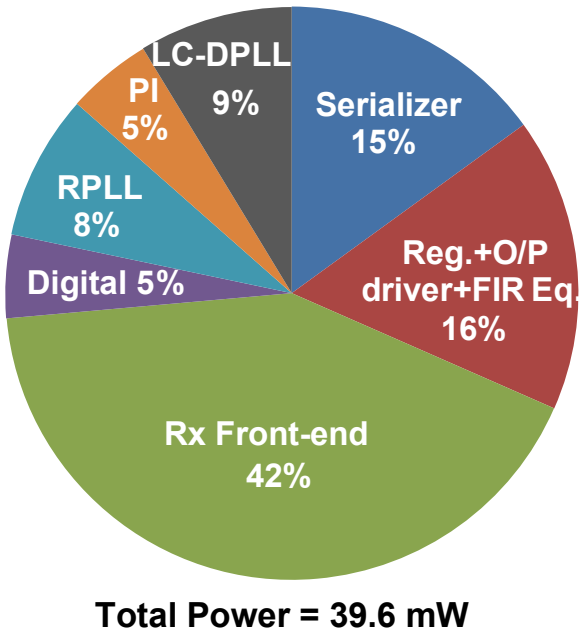


Figure 4.36: Power breakdown in 14 Gb/s serial link.

4.6 Summary

Serial links consume significant power in clock and data recovery while generating and distributing low jitter multi-phase sampling clock, sampling low swing received data, and de-serializing front-end samples to data rates low enough for further digital processing in CDR loop. On the Tx side, voltage-mode equalization gets power hungry for high resolution accuracy during equalization. In this chapter, issues with conventional serial link design

Table 4.1: Performance summary for 14 Gb/s XCVR and comparison to the state-of-the-art.

	This Work	ISSCC'14 [39]	JSSC'14 [40]
Technology [nm]	65	28	22
Data Rate [Gb/s]	14	20	8
Architecture	1/2-rate	1/2-rate	1/2-rate
Supply Voltage [V]	1.0/1.1/1.15	0.9/1.35	0.72
Transmit Swing (pk-pk) [V]	0.4	0.5	NA
Channel Loss [dB]	11	20	8
Equalizer	2-tap FIR	2-tap FIR/CTLE/2-tap DFE	3-tap CML Eq.
Rx CLK Jitter [ps]	1.8	NA	NA
Tx Power [mW]	12.5	42	6.36
Rx Power [mW]	23.6	77	6.43
PLL Power [mW]	3.5	11*	13.2*
FOM [pJ/bit]	2.8	6.5	3.25

* Power amortized over parallel lanes.

techniques are discussed in detail and new techniques are proposed to improve energy efficiency of the transceiver. Frequency and phase tracking as proposed on Rx side with single PI, a ring PLL, and shared Tx CLK is more energy-efficient compared to frequency and phase tracking with multiple PIs, multi-phase generator, and shared Tx CLK in conventional designs. The proposed charge-based flip-flop with limited i/p and o/p swings and dynamic power consumption replaces traditional full-swing flip-flops in front-end samplers and deserializers thereby reducing power consumption. A partially segmented voltage-mode output driver eases out the resolution-versus-range tradeoff in the 2-tap voltage-mode equalizer and reduces power consumption in pre-drivers. Incorporating all the proposed techniques, an energy efficiency of 2.8 mW/Gb/s is achieved for a 14 Gb/s transceiver. Inadvertent delays while processing !!PD output digitally forbid CDR to achieve wide JTOL bandwidth, but they can be easily corrected.

Chapter 5

Conclusion

Aggressive scaling of CMOS devices has enabled incredible amounts of functionality integration and enabled faster processing (speed) of large data (volume) with lower power consumption (power). Speed/volume/power vary across applications from a few kbps data-rate/kbytes data/sub-mW power in portable devices to multi-Gbps data-rate/peta-bytes data/MW power in data centers and supercomputers. While volume and data processing speed requirements are application specific, energy efficiency is always critical. It appears as low battery life in mobile devices while it dominates the cooling cost in large computer facilities. While addressing the increasing demand of speed and volume of data in various applications, the energy efficiency of serial links is one of the major bottlenecks. The emphasis of this research has been to design techniques at the architectural and circuit levels to improve energy efficiency of serial links. The next two sections summarize two specific projects during this research.

5.1 Voltage-Mode Tx with PWM-based De-emphasis

The attenuation, dispersion, and distortion of the transmitted signal by channel severely limits the data transfer rate. Overcoming channel-induced data rate penalty requires equalization, which incurs significant power penalty. More specifically, equalization complicates the design of both the transmitter and receiver. For instance, conventional voltage mode transmitters (VM-Tx) that are extremely energy-efficient compared to their current mode logic (CML) counterpart become grossly inefficient when equalization is embedded into them. To overcome this drawback, we proposed a time-based equalization scheme for a VM-Tx to achieve excellent energy efficiency while compensating for 28 dB channel loss at 5 Gb/s data

rate [37].

5.2 Low Power 14 Gb/s Transceiver

To meet the growing demand of data rates across backplane channels with a few-pJ/bit energy efficiency is a challenging task. On the Tx side, a VM output driver is commonly used, but suffers from range/resolution tradeoff when the output driver and pre-driver have to be segmented to implement the much-needed equalization and impedance matching on the Tx side. Secondly, the Rx side uses many high-speed samplers for data/edge (D/E) detection on the received signal and deserializer later to process D/E in digital clock and data recovery loop. The high-speed samplers and deserializers cost a significant portion of the total Rx power. In addition, the generation and routing of high-speed sampling clock phases add a major chunk to the Rx power. We proposed three different energy-efficient design techniques in a 14Gb/s transceiver [33]: (i) a partially segmented VM driver that improves power efficiency by more than 30% across all equalization settings, (ii) charge-based sampling flip-flop (CFF) with non-return-to-zero (NRZ) operation, and (iii) low power clock and data recovery architecture that reduces clock distribution power. The prototype transceiver employing these techniques achieves an energy efficiency of 2.8 pJ/bit while operating at 14 Gb/s data rate

References

[1] P. Kogge. (2008, Sept.) ExaScale computing study: Technology challenges in achieving exascale systems. [Online]. Available: http://www.cse.nd.edu/Reports/2008/TR-2008-13.pdf

[2] Y. Lu, K. Jung, Y. Hidaka, and E. Alon, "Design and analysis of energy-efficient reconfigurable pre-emphasis voltage-mode transmitters," *IEEE J. Solid-State Circuits*, vol. 48, no. 8, pp. 1898–1909, Aug. 2013.

[3] J. W. Jung and B. Razavi, "A 25-Gb/s 5-mW CMOS CDR/Deserializer," *IEEE J. Solid-State Circuits*, vol. 48, no. 3, pp. 684–697, Mar. 2013.

[4] T. G. 500. (2014, Nov) The Little Green500 List - November 2014. [Online]. Available: http://www.green500.org/lists/little201411

[5] P. Restle, T. McNamara, D. Webber, P. Camporese, K. Eng, K. Jenkins, D. Allen, M. Rohn, M. Quaranta, D. Boerstler, C. Alpert, C. Carter, R. Bailey, J. Petrovick, B. Krauter, and B. McCredie, "A clock distribution network for microprocessors," *IEEE J. Solid-State Circuits*, vol. 36, no. 5, pp. 792–799, May 2001.

[6] B. Casper, J. Jaussi, F. O'Mahony, M. Mansuri, K. Canagasaby, J. Kennedy, E. Yeung, and R. Mooney, "A 20Gb/s forwarded clock transceiver in 90nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2006, pp. 263–265.

[7] R. Fish. (2012, Jan) Future of computers - Part 2: The Power Wall. [Online]. Available: http://www.edn.com/design/systems-design/4368858/Future-of-computers--Part-2-The-Power-Wall

[8] G. E. Moore, "Cramming more components onto integrated circuits," *IEEE Solid-State Circuits Society Newsletter*, vol. 11, no. 5, pp. 33–35, Sept. 2006, reprinted from *Electronics*, vol. 38, no. 8, April 19, 1965, pp.114 ff.

[9] S. Fallahi, D. Cui, D. Pi, R. Zhu, G. Unruh, M. Lugthart, and A. Momtaz, "A 19 mW/lane SerDes transceiver for SFI-5.1 application," in *Proc. IEEE Custom Integrated Circuits Conf.*, Sept. 2011, pp. 1–4.

[10] H. Hatamkhani, K.-L. J. Wong, R. Drost, and C.-K. K. Yang, "A 10-mW 3.6-Gbps I/O transmitter," in *Symp. VLSI Circuits Dig. Tech. Papers*, Apr. 2003, pp. 97–98.

[11] W. Dettloff, J. Eble, L. Luo, P. Kumar, F. Heaton, T. Stone, and B. Daly, "A 32mW 7.4Gb/s protocol-agile source-series-terminated transmitter in 45nm CMOS SOI," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2010, pp. 370–371.

[12] R. Sredojevic and V. Stojanovic, "Digital link pre-emphasis with dynamic driver impedance modulation," in *Proc. IEEE Custom Integrated Circuits Conf.*, Sept. 2010, pp. 1–4.

[13] B. Nikolic, V. G. Oklobdzija, V. Stojanovic, W. Jia, J. K.-S. Chiu, and M. Ming-Tak Leung, "Improved sense-amplifier-based flip-flop: design and measurements," *IEEE J. Solid-State Circuits*, vol. 35, no. 6, pp. 876–884, June 2000.

[14] J.-R. Schrader, E. A. M. Klumperink, J. Visschers, and B. Nauta, "Pulse-width modulation pre-emphasis applied in a wireline transmitter, achieving 33 dB loss compensation at 5-Gb/s in 0.13-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 41, no. 4, pp. 990–999, Apr. 2006.

[15] B. Casper, M. Haycock, and R. Mooney, "An accurate and efficient analysis method for multi-Gb/s chip-to-chip signaling schemes," in *Symp. VLSI Circuits Dig. Tech. Papers*, June 2002, pp. 54–57.

[16] J. Bulzacchelli, C. Menolfi, T. Beukema, D. Storaska, J. Hertle, D. Hanson, P.-H. Hsieh, S. Rylov, D. Furrer, D. Gardellini, A. Prati, T. Morf, V. Sharma, R. Kelkar, H. Ainspan, W. Kelly, L. Chieco, G. Ritter, J. Sorice, J. Garlett, R. Callan, M. Brandli, P. Buchmann, M. Kossel, T. Toifl, and D. Friedman, "A 28-Gb/s 4-tap FFE/15-tap DFE serial link transceiver in 32-nm SOI CMOS technology," *IEEE J. Solid-State Circuits*, vol. 47, no. 12, pp. 3232–3248, Dec. 2012.

[17] P. K. Hanumolu, B. , R. Mooney, G.-Y. Wei, and U.-K. Moon, "Analysis of PLL clock jitter in high-speed serial links," *IEEE Trans. Circuits Syst. II*, vol. 50, no. 11, pp. 879–886, Nov. 2003.

[18] J. Poulton, R. Palmer, A. Fuller, T. Greer, J. Eyles, W. Dally, and M. Horowitz, "A 14-mW 6.25-Gb/s transceiver in 90-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2745–2757, Dec. 2007.

[19] K. Fukuda, H. Yamashita, F. Yuki, M. Yagyu, R. Nemoto, T. Takemoto, T. Saito, N. Chujo, K. Yamamoto, H. Kanai, and A. Hayashi, "An 8Gb/s transceiver with 3x-oversampling 2-threshold eye-tracking CDR circuit for -36.8dB-loss backplane," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2008, pp. 98–99.

[20] W. Yin, R. Inti, A. Elshazly, B. Young, and P. K. Hanumolu, "A 0.7-to-3.5 GHz 0.6-to-2.8 mW highly digital phase-locked loop with bandwidth tracking," *IEEE J. Solid-State Circuits*, vol. 46, no. 8, pp. 1870–1880, Aug. 2011.

[21] J. W. Jung and B. Razavi, "A 25Gb/s 5.8mW CMOS equalizer," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 44–45.

[22] J. W. Jung and B. Razavi, "A 25 Gb/s 5.8 mW CMOS equalizer," *IEEE J. Solid-State Circuits*, vol. 50, no. 2, pp. 515–526, Feb. 2015.

[23] A. Manian and B. Razavi, "A 40-Gb/s 9.2-mW CMOS equalizer," in *Symp. VLSI Circuits Dig. Tech. Papers*, June 2015, pp. C226–C227.

[24] M. Kossel, C. Menolfi, J. Weiss, P. Buchmann, G. von Bueren, L. Rodoni, T. Morf, T. Toifl, and M. Schmatz, "A T-coil-enhanced 8.5Gb/s high-swing source-series-terminated transmitter in 65nm bulk CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2008, pp. 110–111.

[25] J. Sonntag and J. Stonick, "A digital clock and data recovery architecture for multi-gigabit/s binary links," in *Proc. IEEE Custom Integrated Circuits Conf.*, Sept. 2005, pp. 537–544.

[26] B. Raghavan, D. Cui, U. Singh, H. Maarefi, D. Pi, A. Vasani, Z. Huang, A. Momtaz, and J. Cao, "A sub-2W 39.8-to-44.6Gb/s transmitter and receiver chipset with SFI-5.2 interface in 40nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2013, pp. 32–33.

[27] U. Singh, A. Garg, B. Raghavan, N. Huang, H. Zhang, Z. Huang, A. Momtaz, and J. Cao, "A 780mW 4×28Gb/s transceiver for 100GbE gearbox PHY in 40nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 40–41.

[28] P. Upadhyaya, J. Savoj, F.-T. An, A. Bekele, A. Jose, B. Xu, D. Wu, D. Furker, H. Aslanzadeh, H. Hedayati, J. Im, S.-W. Lim, S. Chen, T. Pham, Y. Frans, and K. Chang, "A 0.5-to-32.75Gb/s flexible-reach wireline transceiver in 20nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2015, pp. 56–57.

[29] R. Inti, A. Elshazly, B. Young, W. Yin, M. Kossel, T. Toifl, and P. Hanumolu, "A highly digital 0.5-to-4Gb/s 1.9mW/Gb/s serial-link transceiver using current-recycling in 90nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2011, pp. 152–153.

[30] G. Shu, W.-S. Choi, S. Saxena, T. Anand, A. Elshazly, and P. Hanumolu, "A 4-to-10.5Gb/s 2.2mW/Gb/s continuous-rate digital CDR with automatic frequency acquisition in 65nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 150–151.

[31] J. Kenney, D. Dalton, M. Eskiyerli, E. Evans, B. Hilton, D. Hitchcox, T. Kwok, D. Mulcahy, C. McQuilkin, V. Reddy, S. Selvanayagam, P. Shepherd, W. Titus, and L. DeVito, "A 9.95 to 11.1Gb/s XFP transceiver in 0.13μm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2006, pp. 864–873.

[32] H. Werker, S. Mechnig, C. Holuigue, C. Ebner, G. Mitteregger, E. Romani, F. Roger, T. Blon, M. Moyal, A. Melodia, J. Fisher, G. de Mercey, and H. Geib, "A 10 Gb/s

SONET-compliant CMOS transceiver with low cross-talk and intrinsic jitter," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2004, pp. 172–173.

[33] S. Saxena, G. Shu, R. K. Nandwana, M. Talegaonkar, A. Elkholy, T. Anand, S. J. Kim, W.-S. Choi, and P. K. Hanumolu, "A 2.8mW/Gb/s 14Gb/s serial link transceiver in 65nm CMOS," in *Symp. VLSI Circuits Dig. Tech. Papers*, June 2015, pp. C352–C353.

[34] B.-J. Lee, M.-S. Hwang, S.-H. Lee, and D.-K. Jeong, "A 2.5-10-Gb/s CMOS transceiver with alternating edge-sampling phase detection for loop characteristic stabilization," *IEEE J. Solid-State Circuits*, vol. 38, no. 11, pp. 1821–1829, Nov. 2003.

[35] T. Anand, M. Talegaonkar, A. Elkholy, S. Saxena, A. Elshazly, and P. K. Hanumolu, "A 7Gb/s rapid on/off embedded clock serial link transceiver with 20ns power-on time, 740μW off-state power for energy proportional links in 65nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2015, pp. 64–65.

[36] K. Fukuda, H. Yamashita, G. Ono, R. Nemoto, E. Suzuki, N. Masuda, T. Takemoto, F. Yuki, and T. Saito, "A 12.3-mW 12.5-Gb/s complete transceiver in 65-nm CMOS process," *IEEE J. Solid-State Circuits*, vol. 45, no. 12, pp. 2838–2849, Dec. 2010.

[37] S. Saxena, R. Nandwana, and P. Hanumolu, "A 5 Gb/s energy-efficient voltage-mode transmitter using time-based de-emphasis," *IEEE J. Solid-State Circuits*, vol. 49, no. 8, pp. 1827–1836, Aug. 2014.

[38] M.-J. Park and J. Kim, "Pseudo-linear analysis of bang-bang controlled timing circuits," *IEEE Transactions on Circuits and Systems—Part I: Regular Papers*, vol. 60, no. 6, pp. 1381–1394, June 2013.

[39] V. Balan, O. Oluwole, G. Kodani, C. Zhong, S. Maheswari, R. Dadi, A. Amin, G. Bhatia, P. Mills, A. Ragab, and E. Lee, "A 130mW 20Gb/s half-duplex serial link in 28nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 438–439.

[40] T. Musah, J. Jaussi, G. Balamurugan, S. Hyvonen, T.-C. Hsueh, G. Keskin, S. Shekhar, J. Kennedy, S. Sen, R. Inti, M. Mansuri, M. Leddige, B. Horine, C. Roberts, R. Mooney, and B. Casper, "A 4×32 Gb/s bidirectional link with 3-Tap FFE/6-Tap DFE and collaborative CDR in 22nm CMOS," *IEEE J. Solid-State Circuits*, vol. 49, no. 12, pp. 3079–3090, Dec. 2014.