

# Design Techniques for Energy-Efficient, Low Latency High Speed Wireline Links

*Nick Sutardja*



Electrical Engineering and Computer Sciences  
University of California, Berkeley

Technical Report No. UCB/EECS-2021-19

<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2021/EECS-2021-19.html>

May 1, 2021

Copyright © 2021, by the author(s).  
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

**Design Techniques for Energy-Efficient, Low Latency High-Speed Wireline Links**

by

Nicholas Sutardja

A dissertation submitted in partial satisfaction  
Of the requirements for the degree of  
Doctor of Philosophy

in

Engineering – Electrical Engineering and Computer Sciences

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Elad Alon, Chair  
Professor Borivoje Nikolić  
Professor Xin Guo

Spring 2019

Design Techniques for Energy-Efficient, Low Latency High-Speed Wireline Links

The dissertation of Nicholas Sutardja is approved.

---

Chair

Date

---

Date

---

Date

University of California, Berkeley  
Spring 2019

Design Techniques for Energy-Efficient, Low Latency High-Speed Wireline Links

Copyright 2019

by

Nicholas Sutardja

## Abstract

Design Techniques for Energy-Efficient, Low Latency High Speed Wireline Links

by

Nicholas Sutardja

Doctor of Philosophy in Engineering – Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Elad Alon, Chair

As data and computing systems get larger with more elements composing a single system, streamlined computation and data communication has put an ever-increasing demand on throughput of high speed SerDes. Industrial standards have responded to this trend by increasing the data-rate of chip I/Os, demanding doubling per-pin data-rate around every four years while the power budget remains the same. This implies that the energy efficiency of these links must improve all while being able to handle the harsher equalization environments seen at higher frequencies. To address the challenge of per-pin bandwidth, this thesis first presents various receive side equalization techniques used in a 60Gb/s non-return-to-zero (NRZ) link. In particular, a double data-rate (DDR) architecture uses current integration in several front-end equalization circuits, including the continuous-time linear equalizer (CTLE), feed-forward equalizer (FFE), and decision-feedback equalizer (DFE), demonstrated in a receiver frontend to achieve 60Gb/s operation with  $> 0.2$  UI-timing margin at  $1e-9$  BER, while consuming 173mW. The same architecture was utilized within a complete non-return-to-zero transceiver with adaptive equalization achieving 60Gb/s with  $>0.3$  UI opening at  $10^{-12}$  Bit Error Rate (BER), while consuming 288 mW and occupying 2.48 mm<sup>2</sup>.

Furthermore, supporting this throughput in distributed system with a ubiquitous communication standard calls for links, which are able to quickly turn on and off and operate efficiently in low utilization modes while supporting capability for maximum throughput. This thesis then goes into an analysis of the requirements motivating our architectural and circuit level decisions for a burst-mode, energy proportional wireline link. To achieve energy proportionality, a 2-tap switched-capacitor transmitter with FFE equalization is presented that allows for a fully dynamic architecture operating at a nominal data-rate of 20Gb/s while maintaining energy-efficiency during both high and low link utilization. Additionally, a rapid-on/off voltage controlled LC oscillator uses resonant clocking to save power by directly driving the data-path capacitive loads all while improving overall latency, and a phase interpolator with a phase

adjustable clock divider allows for the lowest achievable latency design for a 64:1 1-latch serializer implementation. The transmitter was taped out in TSMC's 28nm GP process, and achieves 1.2ns startup time and 0.72-0.62 pJ/bit at 1-20Gb/s while occupying 0.19mm<sup>2</sup>.

*Learn how to see.  
Realize that everything connects  
to everything else.*

Leonardo Da Vinci

To My Family



# Contents

<b>Contents .....</b>	<b>ii</b>
<b>List of Figures.....</b>	<b>iv</b>
<b>List of Tables .....</b>	<b>vii</b>
<b>1 Introduction.....</b>	<b>1</b>
1.1 Background.....	1
1.2 Wireline communication .....	2
1.2.1 Intersymbol interference .....	4
1.3 Equalization.....	4
1.3.1 Continuous time linear equalizer .....	5
1.3.2 Feed-forward equalizer .....	7
1.3.3 Decision feedback equalizer .....	8
1.4 Thesis organization .....	10
<b>2 Energy-efficient receive side design techniques for 60-Gb/s.....</b>	<b>11</b>
2.1 Current integration vs. resistive summation .....	11
2.2 Current integrating CTLE and DMUX .....	12
2.3 RX feed-forward equalization.....	13
2.3.1 Dynamic latches .....	14
2.3.2 FFE coefficient control .....	15
2.3.3 Integrating FFE+DFE .....	17
2.3.4 DDR receiver frontend .....	19
2.4 60Gb/s receiver frontend results .....	20
2.5 60Gb/s complete transceiver results .....	23
2.6 Conclusion.....	26
<b>3 Energy proportional communication.....</b>	<b>27</b>
3.1 Energy proportional efficiency .....	29
3.1.1 Comparison between burst-mode and data-rate back-off .....	30
3.1.2 Efficiency vs. effective data-rate and $P_{\text{standby}}$ .....	31
3.1.3 Multilane signaling.....	32

3.2 Transmitter design for energy efficiency .....	34
3.2.1 Comparison between voltage mode and current mode logic transmitters .....	35
3.2.2 VM and CML TX driver start-up considerations .....	36
3.2.3 Switched-capacitor TX driver .....	38
3.3 Switched-capacitor feed-forward equalization .....	42
3.3.1 Switched-capacitor feed-forward equalizer .....	42
3.3.2 Variable capacitance using capacitive DAC.....	44
3.3.3 Tap control with clock gating.....	45
<b>4 A 2-tap switched-capacitor FFE transmitter achieving 1-20 Gb/s at 0.72-0.62 pJ/bit .....</b>	<b>47</b>
4.1 Transmitter architecture and circuit design.....	47
4.1.1 Switched-capacitor feed forward architecture .....	48
4.2 Serializer design challenges .....	50
4.2.1 MUX serializer timing constraints.....	51
4.2.2 Divider with buffering for a 1-latch SER .....	53
4.2.3 1-latch serializer w/ phase adjustable divider .....	54
4.3 On/off clocking.....	59
4.3.1 Cross-coupled oscillator startup time .....	60
4.3.2 Rapid-on/off LC oscillator .....	61
4.3.3 Resonant clocking .....	62
4.4 High-speed phase alignment.....	64
4.4.1 Phase interpolator design.....	65
4.4.2 High-speed clock divider phase alignment .....	66
4.5 Measurement results.....	69
4.6 Conclusion.....	72
<b>5 Conclusion .....</b>	<b>74</b>
5.1 Thesis summary .....	74
5.2 Future directions.....	75
<b>Bibliography .....</b>	<b>76</b>
<b>Appendix A .....</b>	<b>82</b>

# List of Figures

Figure. 1. 1. Ubiquitous links in the wide area network.....	1
Figure. 1. 2. I/O per-pin bandwidth.....	2
Figure. 1. 3. I/O Data-rate vs. process node for recent (2015-2018) transceiver designs.....	3
Figure. 1. 4. Intersymbol interference.....	4
Figure. 1. 5. Wireline link architecture.....	5
Figure. 1. 6. CTLE equalization.....	6
Figure. 1. 7. Continuous time linear equalizers (a) Passive CTLE (b) Active CTLE.....	6
Figure. 1. 8. Feed-forward equalizer.....	8
Figure. 1. 9. Decision feedback equalizer.....	9
Figure. 2. 1. (a) Resistive summation (b) Current integration (c) Voltage vs. time for current integration vs. resistive summation.....	11
Figure. 2. 2. (a) Current integrating CTLE + DMUX (b) CTLE odd and even outputs over time.....	12
Figure. 2. 3. (a) TX FFE (b) RX FFE.....	13
Figure. 2. 4. CTLE + DMUX and following FFE latches . .....	14
Figure. 2. 5. Transconductance implementations for FFE coefficient (fi) control (a) Current-bias control (b) Differential pair unit cells (c) Source degeneration (d) Cascode with duty cycle control (e) Cascode with variable DC bias.....	15
Figure. 2. 6. (a) Cascode DC bias control integrator output voltage (b) Linearity comparison between current-bias control and cascode DC bias control.....	16
Figure. 2. 7. FFE + DFE integrator.....	17
Figure. 2. 8. Incomplete reset setting (post-cursor ISI).....	18
Figure. 2. 9. (a) Simplified DFE circuit diagram (b) Optimal reset accuracy.....	19
Figure. 2. 10. Receiver frontend architecture.....	20
Figure. 2. 11. (a) Die photo (b) Measurement setup and measured waveforms (All clock sources are synchronized.).....	21
Figure. 2. 12. (a) Eye Diagram at the channel output (b) Estimated pulse response.....	21
Figure. 2. 13. Bathtub curve after equalization.....	22
Figure. 2. 14. Transceiver architecture.....	23

Figure. 2. 15. Measurement setups for (a) channel frequency response, (b) pulse response, and (c) equalizer and CDR characterization.....	24
Figure. 2. 16. (a) Measured channel frequency response (b) TX + channel eye diagram (c) TX + channel pulse response.....	24
Figure. 2. 17. (a) Eye diagrams (b) Bathtub curve (c) Die photo. ....	25
Figure. 3. 1. Parallel processing.....	27
Figure. 3. 2. Unified communication.....	28
Figure. 3. 3. Burst-mode communication environment (interposer illustration example).....	28
Figure. 3. 4. Burst-mode data stream.....	29
Figure. 3. 5. Comparison between effective efficiency of burst-mode (blue) Data-rate back-off (red). Assumes a 1pJ/bit 40Gb/s nominal link is burst at 4KByte intervals with standby-power being 100x less than on-power. ....	31
Figure. 3. 6. Effective energy efficiency contour plot. (assumes $t_{data} / (t_{startup} + t_{shutoff}) = 1e3$ , 1pJ/bit nominal efficiency, 20Gb/s nominal data-rate).....	32
Figure. 3. 7. Comparison between data-rate back-off, throttling, and multilane control in a multilane signaling system.....	33
Figure. 3. 8. Multilane effective efficiency (vs. standby power).....	34
Figure. 3. 9. Comparison between voltage mode and current mode logic transmitters. (a) CML transmitter with differential termination at RX (b) VM transmitter with differential termination at RX. ( $R=Z_0=50\Omega$ , where $Z_0$ is the impedance of the transmission line).....	35
Figure. 3. 10. On/off CML (Current Source).....	37
Figure. 3. 11. On/off regulator desired for voltage mode driver designs.....	37
Figure. 3. 12. (a) NMOS LDO for TX supply regulation. (b) On/off instantaneous voltage drop. ....	38
Figure. 3. 13. Switched-capacitor driver with differential termination at RX.....	39
Figure. 3. 14. SC driver circuit model. ....	39
Figure. 3. 15. Switched-capacitor driver voltage over time.....	41
Figure. 3. 16. Switched-capacitor circuit model with 3-tap FFE.....	43
Figure. 3. 17. (a) B-bit binary capacitor DAC B binary weighted cells (b) B-bit thermometer capacitor DAC with 2B unit cells. ....	44
Figure. 3. 18. N-tap SC FFE circuit model with tap strength control.....	45
Figure. 3. 19. Pre-charge and charge of SC driver.....	46
Figure. 4. 1. Transmitter architecture.....	47

Figure. 4. 2. 2-tap SC FFE TX.....	48
Figure. 4. 3. Switched-capacitor TX cell operation. (a) pre-charge (b) “1” (c) “0”. .....	49
Figure. 4. 4. 2-tap DDR SC FFE layout.....	50
Figure. 4. 5. (a) Line rate serializer (b) DDR serializer.....	50
Figure. 4. 6. (a) 2-latch MUX & timing (b) 1-latch MUX & timing (c) Example 2-latch MUX, two-stage serializer timing paths (d) CMOS MUX. ....	51
Figure. 4. 7. CMOS latch.....	52
Figure. 4. 8. (a) 1-latch buffered divider chain (b) Timing waveforms.....	53
Figure. 4. 9. Latency path for 1-latch SER with buffering. ....	54
Figure. 4. 10. 1-latch adjustable divider chain.....	55
Figure. 4. 12. (a) Adjustable clock to serializer and divider timing paths (a) timing waveforms. 58	
Figure. 4. 13. Latency path for 1-latch serializer w/ adjustable divider. ....	59
Figure. 4. 14. Complementary cross-coupled LC oscillator. ....	59
Figure. 4. 15. Small signal model of a complementary cross-coupled LC oscillator. ....	60
Figure. 4. 16. On/off oscillator.....	61
Figure. 4. 17. Capacitor Array Unit Cell ( $C_{VAR}$ ). ....	62
Figure. 4. 18. (a) Clock buffering (Red) (b) Resonant clocking (Green). ....	63
Figure. 4. 19. On/off LC VCO resonant clocking.....	64
Figure. 4. 20. Divider and data phase alignment. ....	65
Figure. 4. 21. (a) 6-bit PI (b) Inverter PI ideal output (c) PI output waveform. ....	66
Figure. 4. 22. High-speed clock and divider phase alignment.....	67
Figure. 4. 23. Oscillator waveform timing.....	67
Figure. 4. 24. (a) 7-bit tunable delay circuit (b) Delay resolution (tbuffer).....	68
Figure. 4. 25. Die photo (TX). ....	69
Figure. 4. 26. (a) Testing setup image (b) Testing setup block diagram. ....	70
Figure. 4. 27. (a) 4Kbit Data. (b) 4Kbit Data (zoomed in). (c) Latency (zoomed in further).....	71
Figure. 4. 28. (a) UI spaced pulse response. (b) TX eye diagram at 20Gb/s for 4Kbits PRBS pattern. ....	71
Figure. 4. 29. (a) Effective efficiency from 1Gb/s to 20Gb/s. (b) OSC frequency vs. code. ....	72
Figure. A. 1. Complementary cross-coupled oscillator and its small signal model.....	82

# List of Tables

Table 2. 1: Comparison of high-speed receiver equalizers .....	22
Table 2. 2: Comparison of high-speed transceivers.....	25
Table 4. 1: Comparison of energy proportional and SC driver architectures .....	72

## Acknowledgements

First, I would like to wholeheartedly thank my professors: Professor Elad Alon and Professor Borivoje Nikolić for their continued support and belief in me over all these years.

Throughout my Ph.D., I have had the opportunity to work with many talented students here at Berkeley. I would like to thank Hanh-Phuc Le and Ruzica Jevtic for working with me on my first tape out ever, Jaeduk Han and Yue Lu for the many tapeouts together, Andrew Townley, Greg LaCaille, Bonjern Yang, and Nathan Narevsky on general tapeout and Cadence guidance, and Zhongkai Wang for BAG support. I would also like to thank the professors on my thesis committees: Professor Vladimir Stojanovic and Professor Xin Guo and the rest of my fellow BWRC colleagues and the BWRC staff.

Finally, I would like to thank my family: my mother for being the most positive person I know and imparting her mindset on me, my father for proving that passion is the most important thing in life, and my brother for his continued belief in me and camaraderie during graduate school, which helped during some of the hardest times of the Ph.D. journey.

# Chapter 1

## Introduction

### 1.1 Background

From our everyday personal devices to the massive, compute intensive clusters in the cloud, wireline links are ubiquitous. Whenever any data is transmitted over the Internet, countless wireline links operate in the background to send and receive data wherever it needs to go.

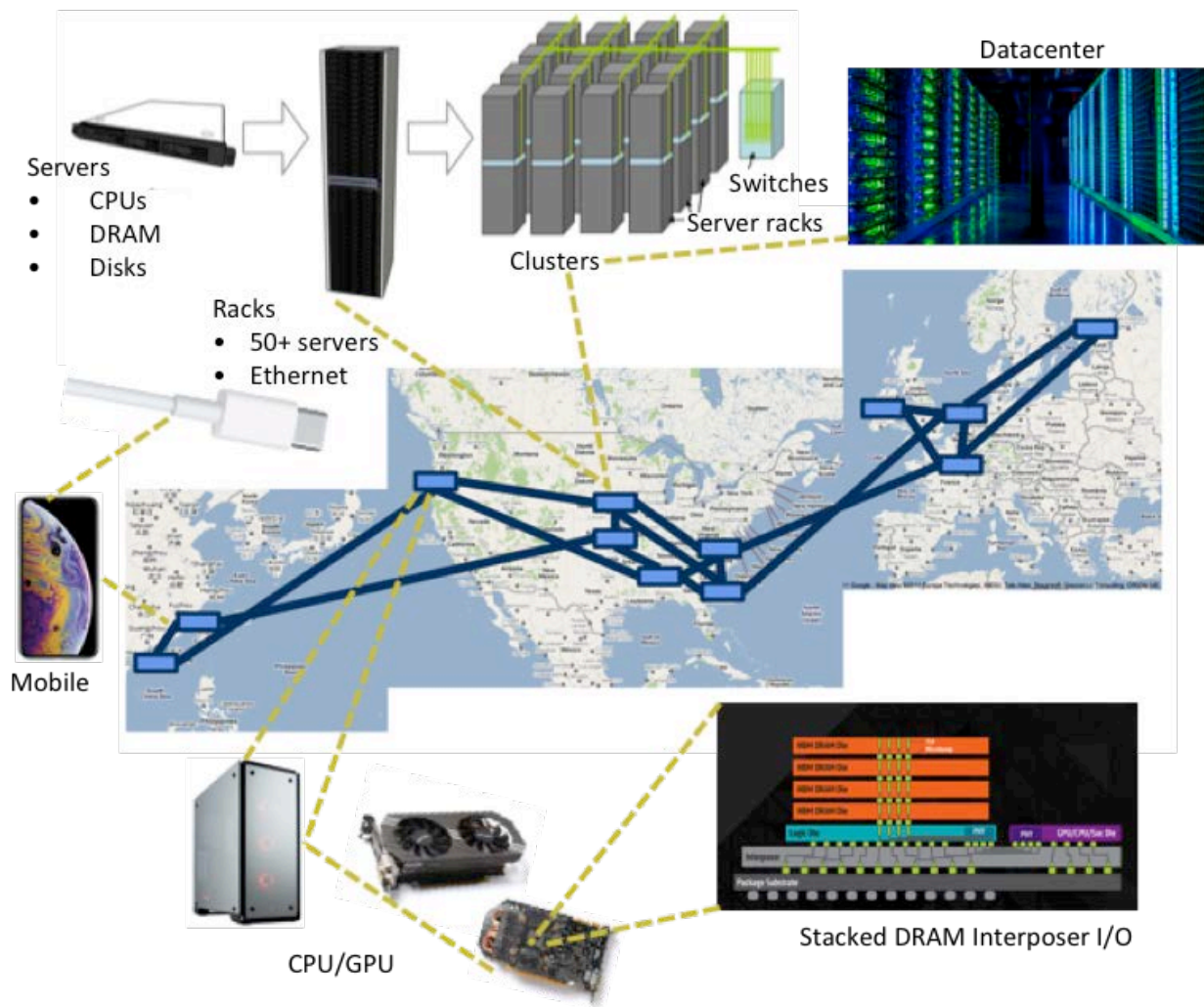


Figure. 1. 1. Ubiquitous links in the wide area network.

The rapid growth of Internet connectivity has required systems to become more distributed and data oriented. Specifically, computing systems have grown larger with more interconnections between various ASICs, with more of these ASICs being application specific [1].



Physical limitations on pins and a limited energy budget for this streamlined computation and data communication have put an ever-increasing demand on communication and throughput of high-speed links. Throughput must improve by increasing per-pin bandwidth [2] and clustering massive I/Os [3], all while improving per-lane efficiency. Industrial standards have responded to this trend by increasing the data-rate of chip I/Os, demanding doubling per-pin data-rate around every four years while the power budget remains the same.

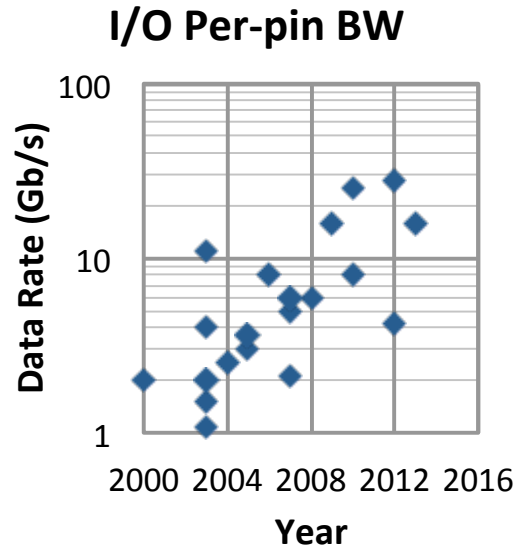


Figure. 1. 2. I/O per-pin bandwidth.

As Moore's Law comes to an end [25], the introduction of more modules (versus one monolithic IC) implies more off-chip communication, meaning that link latency (a nanosecond) becomes more significant. Supporting this throughput in multi-chip module, distributed systems as seen in stacked 3-D and 2.5-D GPU-memory machine-learning driven systems calls for links, which are able to quickly turn on and off and operate efficiently in low utilization modes while supporting capability for maximum throughput. The future of intelligent computing will evolve with this mentality in that highly specialized functional blocks will need to communicate not only with high throughput but also with extremely low latency to accomplish specific tasks [4].

## 1.2 Wireline communication

Off-chip communication without equalization is limited up to a certain frequency (~100 MHz over a 1 meter wire) [13]. Wireline communication systems aim to make use of modern VLSI techniques to allow for the capability of sending data at much higher throughputs over significant distances. Various channel non-idealities and the increasing attenuation at higher frequencies has lead to significant efforts in the design of wireline transceivers.

Over the years a large amount of effort has been put into designing circuits to transmit and receive data through communication links in large backplane environments at higher data-rates by serializing digital data to be transmitted and then deserializing it in the receiver (SerDes). While these environments often remain the same due to legacy considerations, over the years, links have been tasked with operating at higher and higher data-rates. Increasing demands on off-

chip throughput and limited energy budgets require transceivers to operate in harsher link environments as there is more attenuation at higher frequencies due to the skin-effect and more dielectric losses [5].

Faster technology nodes have allowed us push data-rate through higher frequencies allowed through transistor improvements in digital (and mixed-signal) circuits. However, while scaling has aided us on some fronts, various advancements in transceiver equalization techniques as well as simultaneous channel design as seen in CPU-CPU links [6] and capacitively coupled links [7] have helped to ease data-rate and energy limitations that legacy backplane conditions have presented. In fact, our CPU-CPU link uses a 0.7m Twinax cable to operate at 60Gb/s in 65nm. Today, data-rates of published transceivers have been designed to operate upwards of 60Gb/s/channel especially at the more recent technology nodes [2].

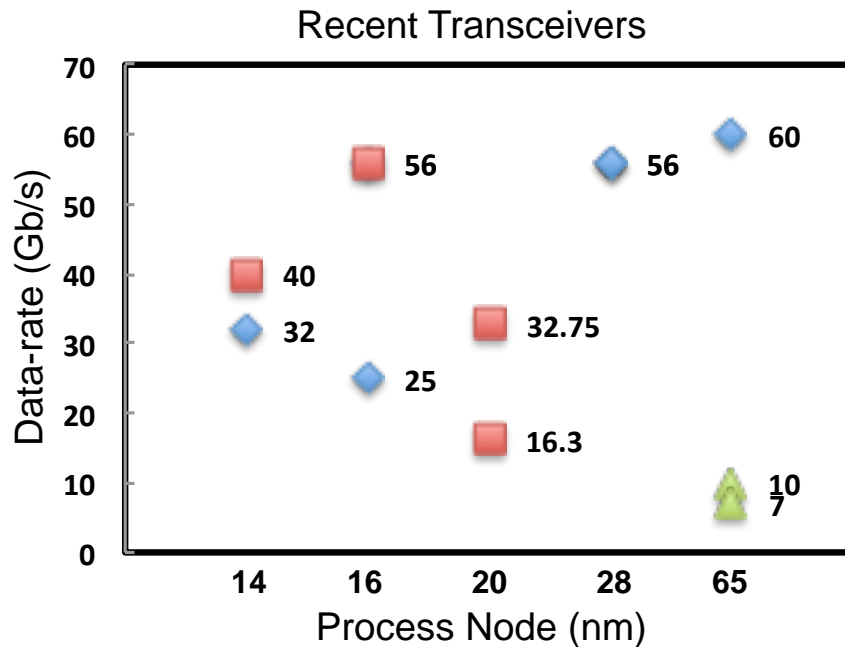


Figure. 1.3. I/O Data-rate vs. process node for recent (2015-2018) transceiver designs.

Fig. 1.3 shows the data-rate of various, selected transceiver designs published by 2018 and their associated technology nodes (always-on transceivers in blue, data-rate flexible transceivers in red, and energy proportional transceivers in green). From this plot, we can see that there have been several designs approaching 60Gb/s, with the one 65nm 60Gb/s design described in further detail in Chapter 2. Furthermore, while technology improvements have lead to some level of data-rate scaling, there is a wide design space, for varying applications. For all of these links, differences in architectural choices due to variations on data-rate and/or channel losses, have lead to this range of designs.

Specifically, more designs are moving toward data-rate flexibility or energy proportionality, where links are not always expected to be utilized at maximum data-rate. The data-rate flexible transceivers (in red), while having flexibility in functional data-rate, are often only energy efficient at their maximum data-rate, since this flexibility is achieved through changing nominal data-rate, while system power remains largely the same. The energy proportional transceivers (in green) demonstrate burst-mode links which operate comparatively more efficiently during low link utilization, but are limited in their speeds.

Ultimately, all high-speed off-chip I/O requires a wireline transceiver due to physical limitations electrical signaling presents. Described in the next section, intersymbol interference (ISI) is at the core of this discussion, and is the starting point to explaining why equalization is necessary, especially as higher demands on throughput require links to operate at higher data-rates.

## 1.2.1 Intersymbol interference

Examining how various bits are affected after going through a non-ideal channel, Fig. 1.4 depicts the received waveforms from an ideal pulse ( $w_{pulse}(t)$ ) and data waveform ( $w_{wf}(t)$ ).

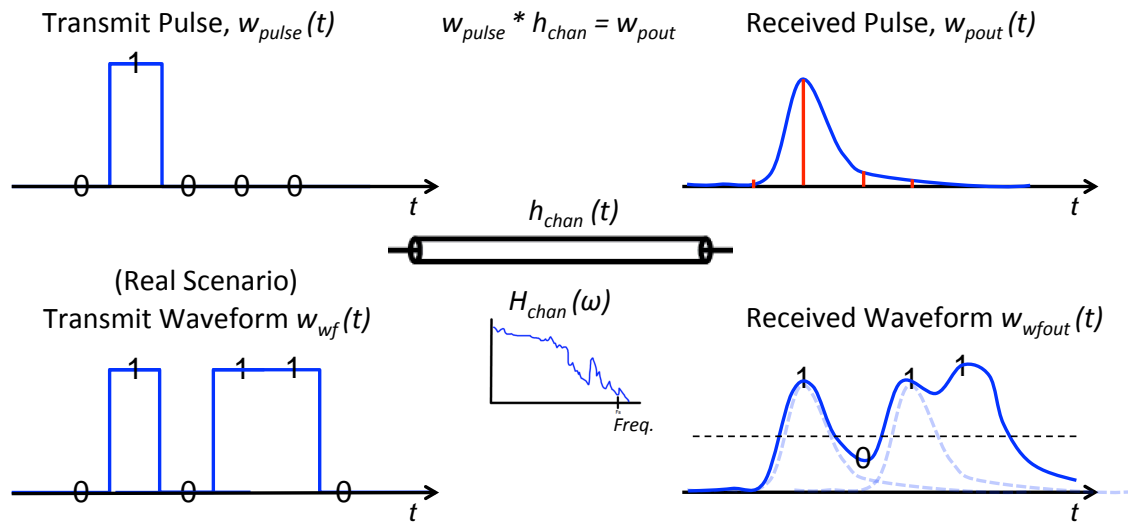


Figure. 1. 4. Intersymbol interference.

The received pulse response ( $w_{pout}(t)$ ) when sampled at the data-rate contains energy that is spread out in time after the channel. These UI spaced signal values are known as pre-cursors and post-cursors depending on if they are before or after the largest signal value, usually<sup>1</sup> chosen to be the cursor. The channel ( $h_{chan}(t)$ ) spreads out the energy, making the correct values of  $w_{wfout}(t)$  harder or impossible to resolve. Especially as our sampling rate increases and more energy leaks into neighboring bits, resolving the difference between a “1” and “0” data level (compared to a threshold) becomes ambiguous or data-dependent. This interference degrades noise margins, reducing the maximum frequency at which the link can operate. This phenomenon is known as intersymbol interference, and it only intensifies at higher frequencies. To counteract these bandwidth limitations caused from both pre-cursor and post-cursor ISI, various equalization techniques are implemented in the transmitter and receiver.

<sup>1</sup> At high data-rates, when the channel contains a comparable pre-cursor, it is often more efficient to choose the pre-cursor as the cursor, and subsequent taps (including the largest energy value) are cancelled by the DFE [9].

## 1.3 Equalization

Equalization is the effort that is performed to clean up signals before the slicer to make sure the received bits are the “same” as those that were transmitted. Typically, a wireline link seeks to have enough equalization to achieve error free operation over a significant amount of bits ( $> 10^{12}$ ) to avoid having to restart, causing interruption in communication. In order to ensure this, transceivers seek a bit error rate (BER) of  $< 10^{-12}$  through a large enough timing window so that they are tolerant to noise and clock jitter. A generic wireline link architecture is shown in Fig. 1.5 where data must go through the channel with its associated pulse response.

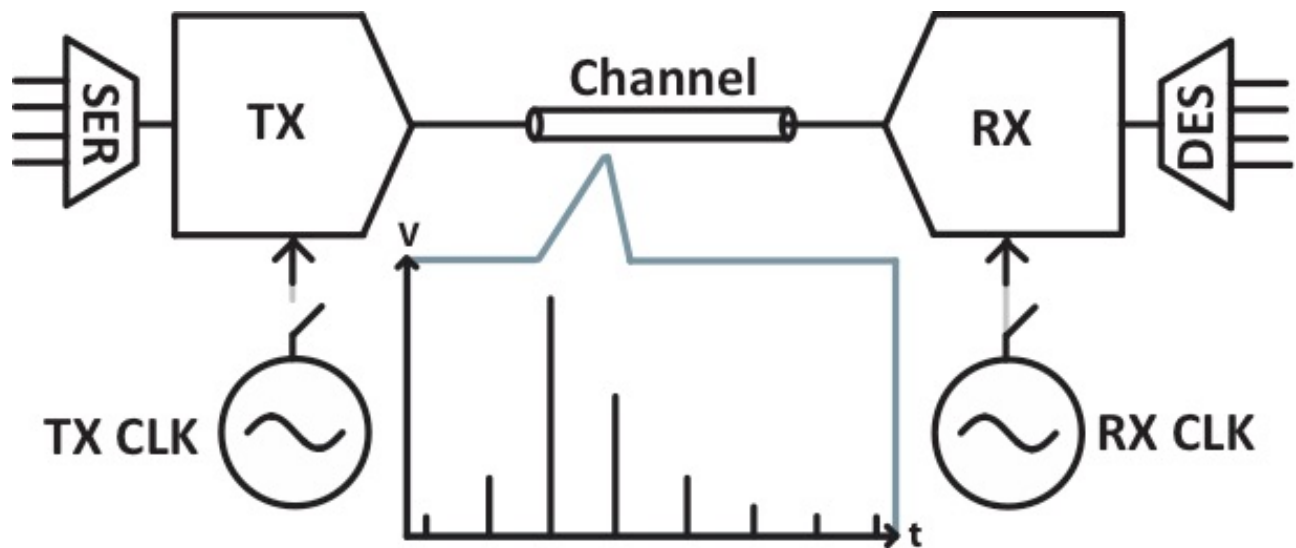


Figure. 1. 5. Wireline link architecture.

First, a serializer translates the digital (coming from CPU, GPU, memory, etc. ... ) data bits into a serial bitstream. Equalizers are employed before the RX slicer to cancel pre-cursor and post-cursor ISI seen from sending data through the non-ideal pulse response. These equalizer structures include the continuous time linear equalizer (CTLE), feed-forward equalizer (FFE), and decision feedback equalizer (DFE), all of which are instrumental in providing ISI cancellation by mitigating errors caused by pre-cursor and post-cursor taps introduced from the channel. Finally, data is deserialized and made available to the rest of the system for further processing. The following sections discuss the various uses for RX equalizers in a wireline link.

### 1.3.1 Continuous time linear equalizer

The continuous time linear equalizer is typically used as the first linear equalizer in the RX to boost high frequency content that has been attenuated in the channel and to attenuate low frequency content to flatten out the overall frequency response. This accomplishes equalization by placing a high pass filter following the (typically) low pass filter caused by the channel (Fig. 1.6).

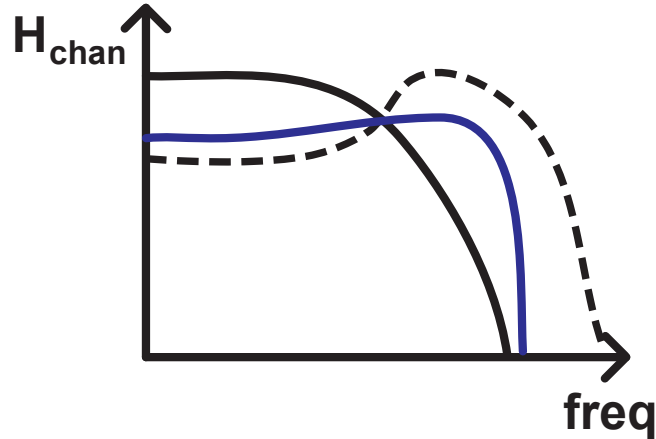
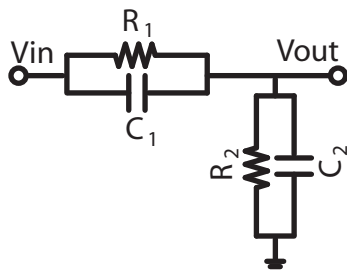


Figure. 1. 6. CTLE equalization.

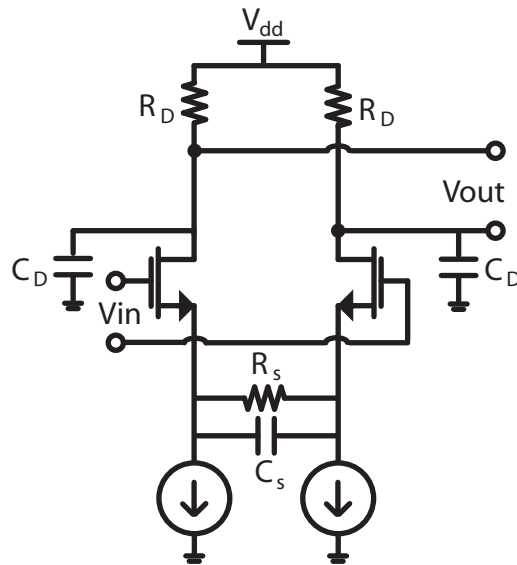
CTLEs are commonly realized as a passive CLTE or active CLTE shown in Fig. 1.7.

### Passive CLTE



(a)

### Active CTLE



(b)

Figure. 1. 7. Continuous time linear equalizers (a) Passive CTLE (b) Active CTLE.

The following analysis determines how we can size the passive CTLE (Fig. 1.7(a)) for the amount of peaking (equalization) desired. Simple KCL leads us to the frequency response  $H(s) = V_{out}(s)/V_{in}(s)$  as

$$H(s) = \frac{R_2}{R_1 + R_2} \frac{1 + R_1 C_1 s}{1 + \frac{R_1 R_2}{R_1 + R_2} (C_1 + C_2) s} \quad (1.1)$$

where  $\omega_z = \frac{1}{R_1 C_1}$  and  $\omega_p = \frac{1}{\frac{R_1 R_2}{R_1 + R_2} (C_1 + C_2)}$ , and the DC gain and high frequency gain are  $A_{DC} = \frac{R_2}{R_1 + R_2}$  and  $A_{HF} = \frac{C_1}{C_1 + C_2}$ .

The amount of peaking achieved is simply

$$Peaking_{passive\ CLTE} = \frac{A_{HF}}{A_{DC}} = \frac{R_1 + R_2}{R_2} \frac{C_1}{C_1 + C_2} \quad (1.2)$$

We can similarly find the peaking from the active CTLE (Fig. 1.7(b)) from calculating the frequency response as

$$H(s) = \frac{g_m}{C_D} \frac{s + \frac{1}{R_S C_S}}{\left(s + \frac{1 + g_m R_S / 2}{R_S C_S}\right) \left(s + \frac{1}{R_D C_D}\right)} \quad (1.3)$$

where the DC gain and high frequency gain are

$A_{DC} = \frac{g_m R_D}{1 + g_m R_S / 2}$  and  $A_{peak} = g_m R_D$  (assuming the  $\frac{1}{R_D C_D}$  pole is the second pole).

$$Peaking_{active\ CLTE} = \frac{A_{peak}}{A_{DC}} = 1 + g_m R_S / 2 \quad (1.4)$$

The addition of the pole and zero of the CTLE allows us to flatten out the overall frequency response seen after the channel by boosting high frequency content compared to low frequency content. This allows for a flatter frequency response that provides long-tail ISI cancellation to mitigate ISI induced from many post-cursors that are often too costly to equalize for directly. On the other hand, large amounts of ISI compared to the cursor value ( $V_{ISI}/V_{CURSOR}$ ) usually are not directly handled by the CLTE. This ISI usually appears, due to the low pass nature of a channel, at the pre-cursors and post-cursors closest to the cursor. While the CTLE presents a continuous time solution for pre-emphasis, channels at high-speeds often require more sophisticated methods of equalization. A feed-forward equalizer can be designed in the discrete domain for further equalization.

## 1.3.2 Feed-forward equalizer

The feed-forward equalizer is typically used to provide more gain at higher frequencies and to mitigate pre-cursor ISI. This is achieved through sending bits through a finite impulse response filter (FIR) filter. Fig. 1.8. Figure presents an N+1-tap FFE with 1 pre-cursor and N-1 post-cursors that achieves this function.

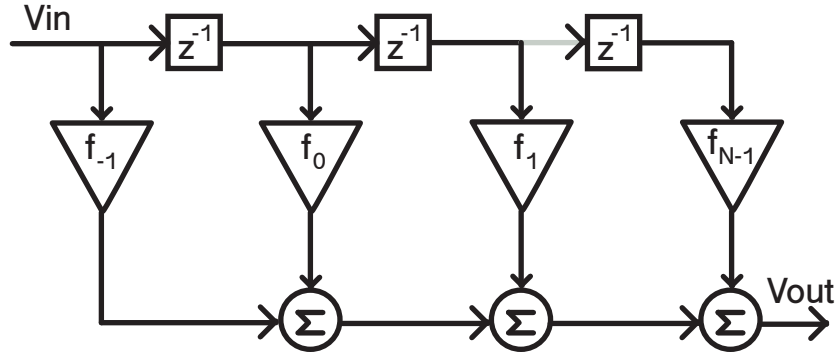


Figure. 1. 8. Feed-forward equalizer.

The following FFE equation demonstrates the functionality an  $N+1$ -tap FFE with 1 pre-cursor, cursor, and  $N-1$  post-cursors.

$$V_{OUT}[k] = \sum_{i=-1}^{N-1} f_i * V_{IN}[k - i] \quad (1.5)$$

The FFE allows us to transmit a weighted sum of shifted bits to cancel ISI. Since  $V_{OUT}[k]$  is a sum of linearly scaled delayed signals and we can choose which  $k$  is the desired cursor, pre-cursor ISI can be cancelled, making the FFE a worthwhile equalizer even with a limited number of taps.

For instance, a 2-tap FFE with a negative  $f_1$  (and positive  $f_0$ ) can provide useful equalization as the negative  $f_1$  value (and positive  $f_0$ ) can realize a high pass characteristic to provide gain to the channel's high-frequency content compared to the low-frequency content. Two opposite sign bits (as in a clock pattern) give us a maximum gain of  $(f_0 + f_1)$ . Conversely, two consecutive bits (as in a DC pattern) give us minimum gain of  $(f_0 - f_1)$ .

While the FFE is useful to equalize pre-cursor ISI and provide high frequency gain, it amplifies high frequency noise as well. Furthermore, additional equalization in the FFE implies the reduction of the cursor tap weight, since headroom limits the output swing of the design ( $\sum |f_i| = 1$ ).

Since the high frequency gain of the FFE amplifies more high-frequency noise, more equalization leads to more noise enhancement. This leads us to our next equalizer, the decision feedback equalizer, which takes advantage of a slicer to erase noise and memory effects introduced from the channel.

### 1.3.3 Decision feedback equalizer

A decision feedback equalizer (DFE) is a non-linear equalizer that allows us to make a symbol decision by quantizing the input with a slicer and subtract out ISI using a feedback FIR filter (Fig. 1.9). The  $N$ -tap DFE shown cancels out  $N$  post-cursors up to the resolution of the DFE tap ( $d_i$ ) coefficients. As the slicer allows us to remove all noise due to post-cursors from the channel, the  $d_i$  can be directly chosen to equal the equalized channel response post-cursor values right before the DFE (at  $V_{in}$ ). Enough ISI needs to be cancelled prior to the slicer, so that noise does not allow for wrong decisions to be made, for a certain amount of consecutive bits.

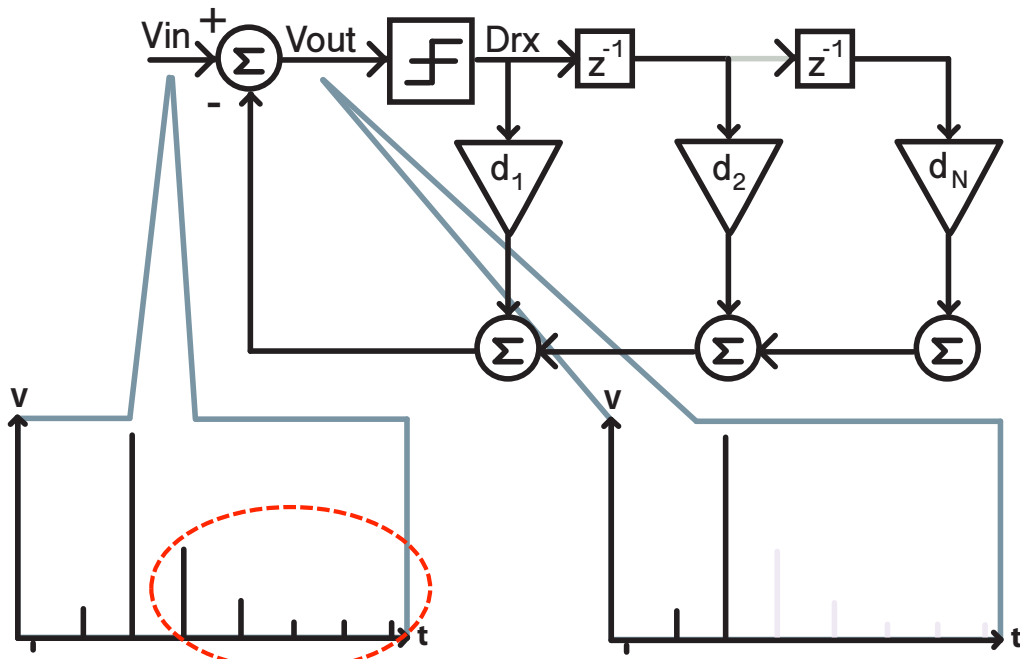


Figure. 1. 9. Decision feedback equalizer.

This non-linear nature (due to the slicer) of the DFE allows the  $k^{th}$  post-cursor to be directly cancelled out as

$$V_{out,k} = V_{in,k} - d_1 * D_{out,k-1} - d_2 * D_{out,k-2} \dots - d_N * D_{out,k-N} \quad (1.6)$$

Since only subsequent bits are fed back to the summing node, only post-cursor ISI can be cancelled. Even though it cannot cancel pre-cursor ISI, the DFE can directly cancel out post-cursor ISI without any noise amplification (unlike in the case of the FFE, where the additional linear noise-enhancing, amplifiers are necessary before the slicer). This circuit is extremely valuable since post-cursors are completely eliminated up to the resolution and dynamic range of the DFE tap coefficients. Typically, earlier  $d_i$ 's are designed to handle more current to be able to equalize larger post-cursors.

As DFEs approach higher-speeds (equalizing for higher data-rates), the limiting factor for high-speed DFE operation is closing the feedback timing path within 1 Unit Interval (UI). Specifically this is dependent on the technology's  $f_t$  (when current gain is unity or approximately  $\frac{1}{2\pi} \frac{g_m}{C_{gs}}$ ). Loop unrolling [8] has been used to get around this feedback loop by creating concurrency (performing look-ahead computation in order resolve the previous bit before the next arrives) but introduces additional delay into the critical paths of later (non-unrolled) DFE taps. Furthermore, loop enrolling does not scale well as it would involve  $2^N$  comparators for  $N$  taps. Various other techniques are introduced in the 66Gb/s DFE design in [9], where the summer is merged with the latch to reduce loading on the first tap, pushing the data-rate for high-speed DFEs. We take advantage of this technique, along with several other techniques for equalization at high speeds.



## 1.4 Thesis organization

As this chapter has motivated the need for equalization and high-speed links, the following chapter explains various techniques that push the boundaries of throughput and energy efficiency in the receiver through design of energy efficient CTLE, RX FFE and DFE. Various techniques including double data-rate (DDR) architecture, cascode tap control, integration, and incomplete reset settling are employed to enable efficient, receive-side equalization at 60Gb/s. These techniques are demonstrated in two separate designs. First a receiver frontend was taped out in a 65nm TSMC process achieving 60Gb/s, equalizing an ISI of  $1.54 V_{ISI}/V_{CURSOR}$  and error free operation over  $1e12$  bits. Then, a complete 60Gb/s transceiver taped out in a 65nm TSMC process achieves 4.8pJ/bit, equalizing 21dB of loss at 30GHz over a 0.7-m Twinax cable.

Chapter 3 introduces the concept of energy proportional communication, motivating the need for low-latency, burst-mode communication. Furthermore, various system and architectural level considerations and how they affect startup time and quiescent current are discussed for an energy proportional transmitter. With the design constraints for an efficient, burst-mode transmitter in mind, Chapter 4 explains various techniques for limiting quiescent current consumption and startup time and presents a 2-tap switched-capacitor FFE transmitter achieving 1-20 Gb/s at 0.72-0.62 pJ/bit achieving a latency of 1.2ns with a 64:2 1-latch serializer in a 28nm TSMC process. Finally, Chapter 5 summarizes the thesis and points out future areas of research.

## Chapter 2

### Energy-efficient Receiver Design Techniques for 60-Gb/s

While various designs have demonstrated high-speed receiver circuits including high-speed DFEs in [9][10][11], further power reductions are required to meet the stringent power constraints and make these more attractive for widespread implementation. In particular, current integration is implemented in various receiver circuits allowing for power savings compared to the continuous-time summation counterparts. Furthermore, receive side FFE is demonstrated for pre-cursor ISI cancellation in the receiver, saving power compared to doing the same in the transmitter. Finally, incorporating all these techniques together, a DDR, integrating CTLE, FFE and DFE is demonstrated in a 60Gb/s non-return-to-zero transceiver with adaptive equalization achieving 60Gb/s with  $>0.3$  UI opening at  $10^{-12}$  Bit Error Rate, while consuming 288 mW and occupying  $2.48 \text{ mm}^2$ .

#### 2.1 Current integration vs. resistive summation

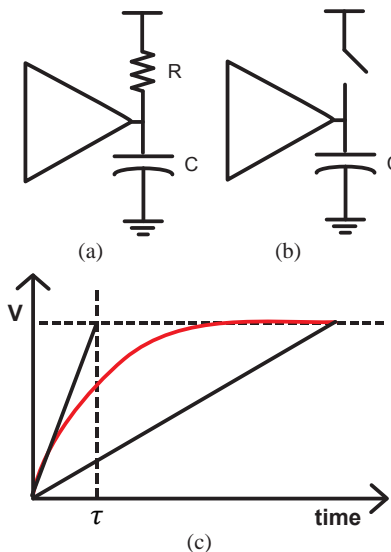


Figure. 2. 1. (a) Resistive summation (b) Current integration (c) Voltage vs. time for current integration vs. resistive summation.

The first technique that allows for energy-efficient, high-speed operation is known as current integration, which was first proposed in [12] for a low-power DFE. A current integration circuit (Fig. 2.1(b)) as opposed to resistive current summation circuit (Fig. 2.1(a)) makes use of the clock to pre-charge (or reset) the output voltage to  $V_{dd}$  (or a common mode voltage as in [13]) and then discharges the output voltage based on the output capacitance  $C$  and transconductance in the integration phase. This technique is well suited for DDR architectures since there is a natural reset phase for half the clock period. Fig. 2.1 compares current integration and resistive current summation.

While resistively loaded summation experiences exponential settling time to achieve a desired gain (in red), current integration allows for the same gain to be achieved in a shorter time or power to be saved given the same time to achieved the desired gain (Fig. 2.1(c)).

An extensive analysis in the current integrating DFE and FFE shows a  $\sim 3x$  power savings compared to resistively loaded stages [13]. Current integration, as opposed to continuous-time allows for  $N_\tau$  savings in power (1.7).

$$\frac{I_{\text{integrating summer}}}{I_{\text{continuous-time summer}}} \approx \frac{T_{\text{bit}}}{\tau} = N_\tau \quad (1.7)$$

We use this technique in our CTLE, DFE + FFE summer described in the following sections.

## 2.2 Current integrating CTLE and DMUX

The resistively loaded CTLE as in (Fig. 1.7) is not sufficient for high-speeds, as it requires a lot of power as described in the previous section. Current integration for the CTLE through replacing the resistors with PMOS resetting switches allows the CTLE to function as an integrator, and it also gives us the added benefit of a two-phase system (for DDR operation). With one modification (adding an additional NMOS cascode transistor similar to the cascode sampling proposed in [14]) to provide two distinct differential outputs ( $V_{oute}$  and  $V_{outo}$ )), we can demultiplex our full-rate data with this integrating stage to provide the odd and even data for the rest of the DDR receive chain. This first CTLE + DMUX stage provides an energy efficient way to have CTLE and provide a demultiplexer (DMUX) for odd and even data paths.

We employ current integration, with two cascode resetting stages and a shared NMOS input pair with resistive and capacitive degeneration (Fig. 2.2).

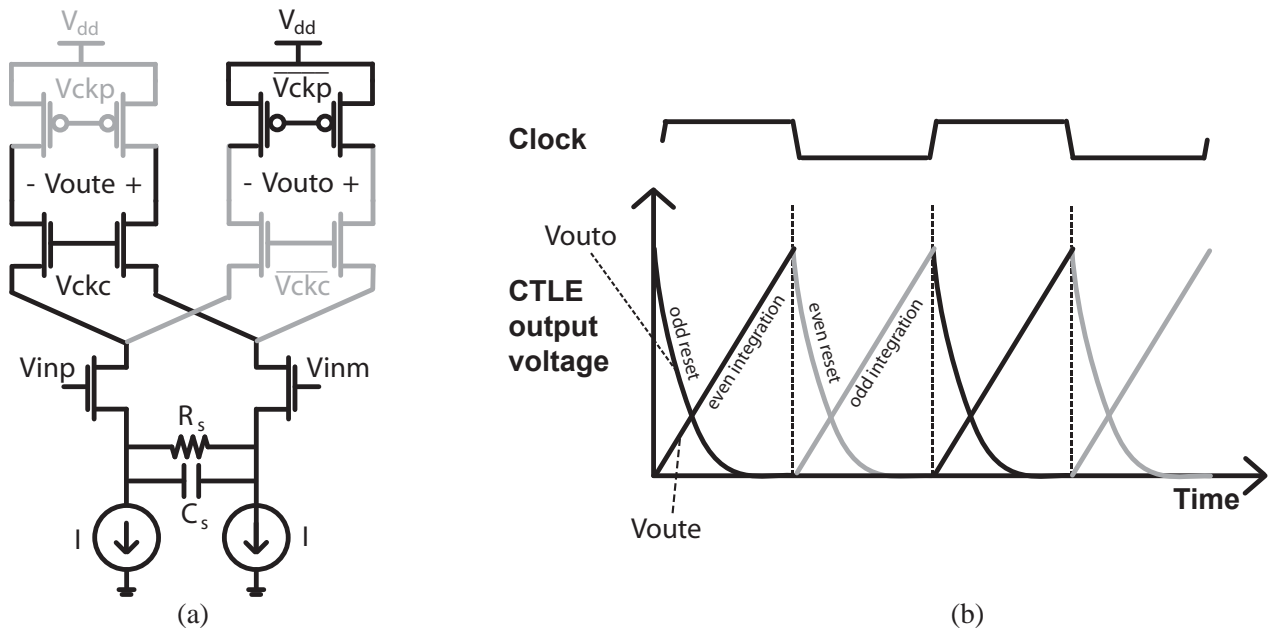


Figure. 2. 2. (a) Current integrating CTLE + DMUX (b) CTLE odd and even outputs over time.

The odd PMOS and even NMOS cascode switches in the upper part of the circuit are activated at the same time (same with the even PMOS and odd NMOS cascode), while the bottom NMOS differential inputs are always on. This alternating integration and reset is delineated by the black and grey shading in the circuit in Fig. 2.2(a). Fig. 2.2(b) shows the ideal CTLE odd and even output voltages, where each path resets and integrates in opposite phases. Reset in this diagram occurs fully, as the differential output voltages reach 0V differential every cycle. Furthermore, the DC level of the cascode clock can allow for additional control over overall gain.

The next technique that we use to save energy is receiver feed-forward equalization (RX FFE) explained in the following section.

## 2.3 RX feed-forward equalization

We compare TX feed-forward equalization with RX feed-forward equalization in Fig. 2.3.

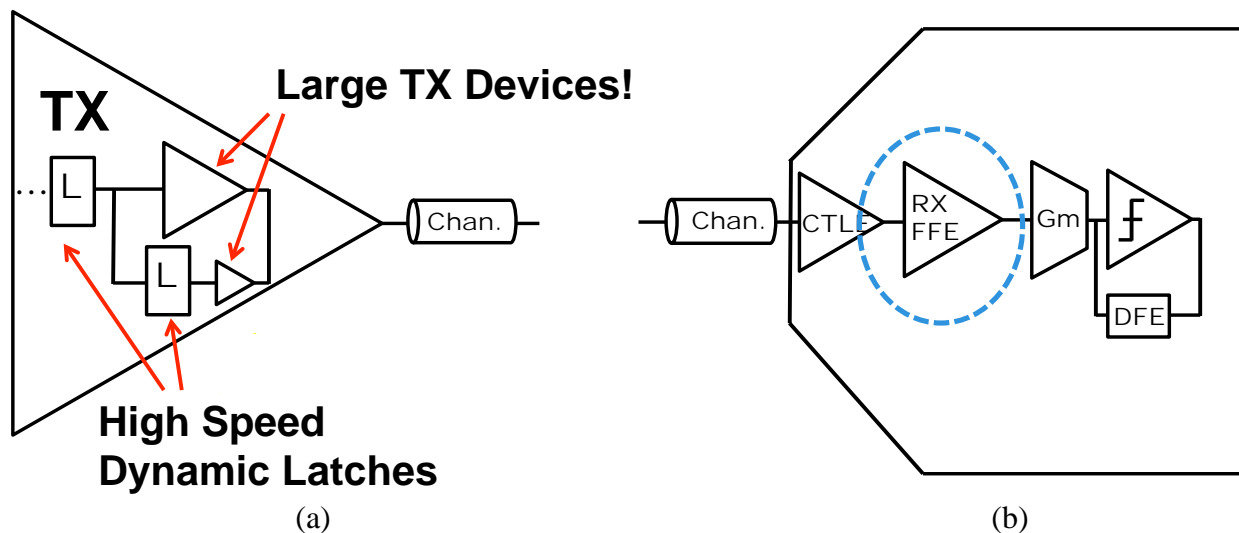


Figure. 2. 3. (a) TX FFE (b) RX FFE.

Receive-side equalization is commonly achieved using predominantly CTLE and DFE. Since FFE requires multiple stages of delay, it is typically done in the digital domain at the transmitter.

Implementing FFE in the TX might require more power compared to having the same function in the receiver. Some high-speed TX FFE's are shown with various implementation differences in [15] where 7-tap feed-forward equalizer is designed with current mode logic delay elements and output driver and in [16] where a 1-UI delay line consisting of inductors and capacitors is used to create delay for a 4-tap FFE. These delay elements and drivers consume considerable amounts of current to achieve FFE equalization in these designs.

However, the presence of pre-cursor ISI from our channel means that we still need feed-forward equalization. As in the TX-FFE, the RX FFE achieves the same FIR filter function to sum delayed and amplified versions of the signal. Especially for multi-tap designs as in [13], considerable power can be saved, since power is based on the analog delay elements needing to only drive  $g_m$  stages in the RX rather than large digital delay elements needing to drive  $50\Omega$  loads in the TX driver. However, one of the challenges lies in how to actually realize linear,

analog delay elements in the receiver. We make use of analog dynamic latches to create the UI spaced delay in our DDR RX FFE.

### 2.3.1 Dynamic latches

Since integrating CTLE's resetting (or return-to zero or RZ behavior) waveforms (Fig. 2.2(b)) mean that the output signal of one stage is not available to the next during integration, we need to first convert the RZ waveforms into non-return-to-zero (NRZ) before using the CTLE output for further receiver stages. An analog dynamic latch can be used for this purpose, shown in Fig. 2.4. In addition to creating the NRZ waveforms, the latches naturally create the UI-delay for the FFE + DFE integrator.

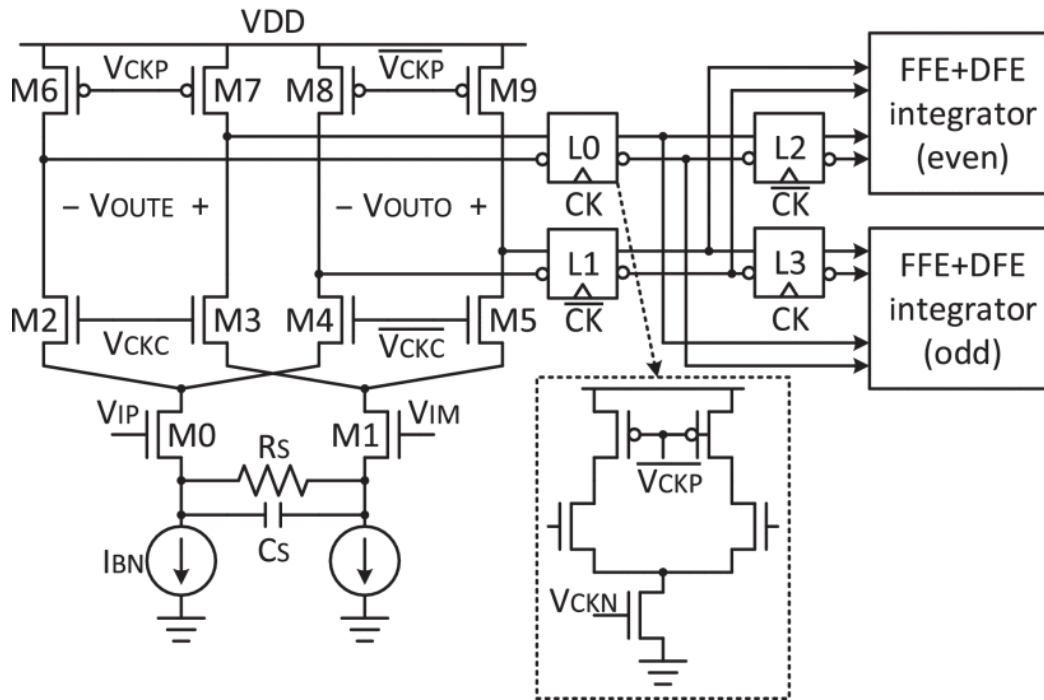


Figure. 2. 4. CTLE + DMUX and following FFE latches .

Since any gain before the slicer must be analog in nature, linearity must be preserved in the delay elements. The high-speed analog delay latch is similar to the digital dynamic latches in [9], but is designed to have higher input transistor overdrive voltage ( $V^*$ ) for linear signal-transfer characteristics. The Dynamic latch provides gain when both  $V_{ckn}$  and  $V_{ckp}$  are high (both on), and the parasitic capacitance at the output nodes hold the value while off. Input NMOS devices are sized to have enough overdrive voltage to support the expected swing seen at the inputs.

Careful consideration is placed into designing these latches with appropriate common mode voltages with gains around or slightly above 1 (to ensure linearity and not amplify noise unnecessarily). Hence, multiple DACs are used for the individual bias voltages of  $V_{ckn}$ ,  $V_{ckp}$ , and  $V_{ckc}$  to be able to control both latch gain and integration window. Furthermore, swing control of the gate voltage is also implemented for the current sources (to keep them in saturation) using explicit capacitive division, added to the clock network.

## 2.3.2 FFE coefficient control

To implement the coefficients ( $f_i$ 's) in (1.5), variable effective transconductance at the integrating node controls how much current is integrated per tap. Five different possible methods to achieve this are shown in Fig. 2.5.

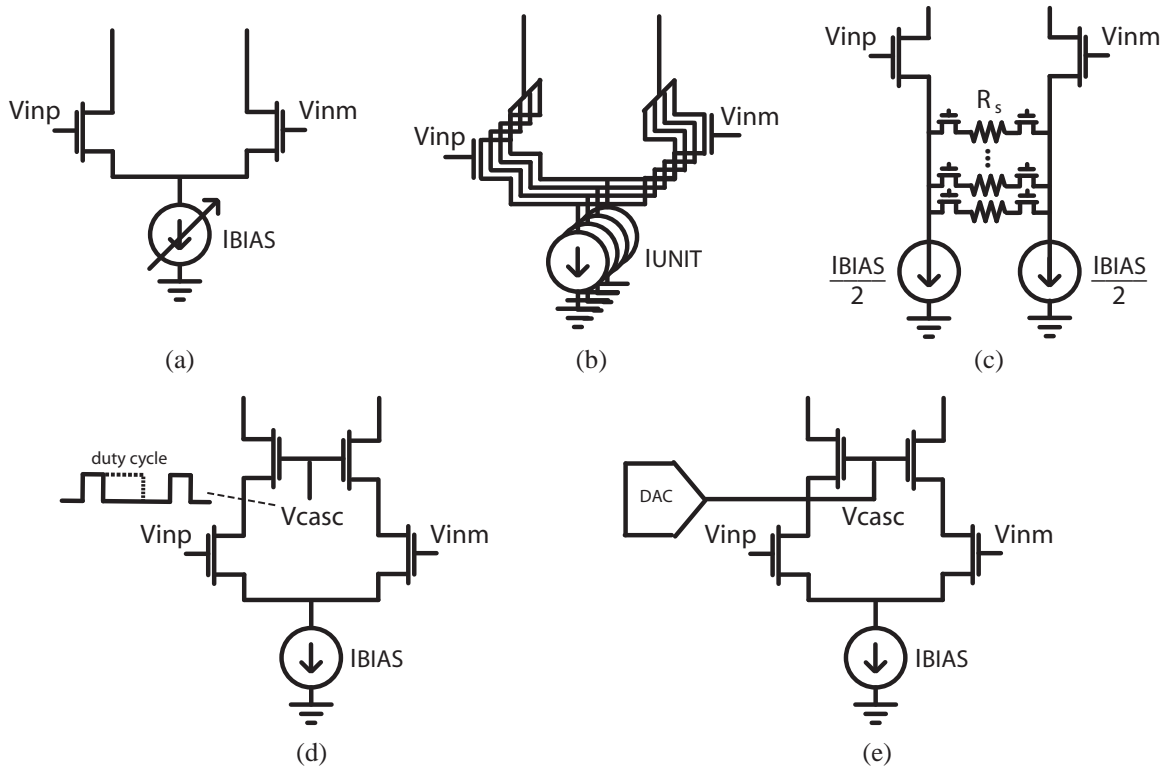


Figure. 2. 5. Transconductance implementations for FFE coefficient ( $f_i$ ) control (a) Current-bias control (b) Differential pair unit cells (c) Source degeneration (d) Cascode with duty cycle control (e) Cascode with variable DC bias.

The first, most common method is current-bias control (Fig. 2.5(a)), which is widely used in DFEs, allowing for a variable amount to be integrated in the differential pair. However, this approach is infeasible for feed-forward equalization because FFE requires linear signal-transfer characteristics. Specifically, the input signals to the FFE ( $V_{inp}$  and  $V_{inm}$ ) are both still analog and must be further equalized by later stages. Reducing tail current (for the same sized input pair) when smaller  $f_i$  is desired would reduce the  $V^*$  of the input pair, making the input pair start to clip as signals get large. This leads to nonlinear distortion that cannot be directly compensated through equalization. Alternatively, the DFE gets away with this implementation because the inputs to the differential pair are digital (and the input transistors operate as switches).

To get around this linearity issue, Fig. 2.5(b) employs unit cells of differential pairs to modulate the amount of current integrated. Multiple differential pair unit cells are configured in parallel, and a subset is turned on to achieve the desired  $f_i$ . Since  $V^*$  remains constant with gain control, linear signal-transfer characteristics are preserved. Fig. 2.5(c) presents another solution to linearity by adding a variable degeneration resistance ( $R_s$ ). While they both preserve linearity,

these solutions suffer from limited minimum transistor-size constraints, limiting tap resolution or power efficiency. In order to achieve a high resolution (high number of bits  $b$ ) using the unit cell method, minimum size constraints would soon limit how small each differential pair could be sized. Due to many minimum-sized switches introducing a lot of capacitance on the tail, gain at high frequency will not be as expected, but instead much higher. Additionally, capacitive loading on the integration node would be determined by the  $2^b$  times the minimum size and not the actually desired current, making this architecture's per-tap power scale exponentially with the number of bits. Resistive degeneration suffers from a similar issue, as there is a limit as to how much degeneration can be achieved with minimum-sized switches.

To support a large dynamic range with high resolution without sacrificing linearity or directly increasing power consumption, we propose a cascode current control scheme in Fig. 2.5(d) and Fig. 2.5(e) where an additional cascode NMOS switch is added in between the input pair and output node.  $V_{casc}$  is controlled by modifying the duty cycle of the input into the cascode in Fig. 2.5(d), where integration time is modulated to vary the gain. A digital to time converter [17] can be used for this purpose.

For our design, we use a DAC to set the DC bias of  $V_{casc}$ , which sets the drain voltage and input impedance of the input differential pair devices changing the overall gain. Since the overdrive voltage of the input pair is preserved, the signal linearity is largely unaffected by the gain setting. Both the duty cycle control and variable DC bias for the cascode achieve the linearity we desire; however, we go with the DC bias, since a DAC costs less power (compared to power introduced from digital switching in a digital to time converter). It is important to note that, if no DACs (and no quiescent current) are desired for energy proportional communication as we will introduce in Chapter 3, duty cycle control can be a more attractive choice.

Fig. 2.6 shows the integrator output voltage time domain characteristics for different gain settings and compares the linearity of the cascode control method to conventional current-bias control.

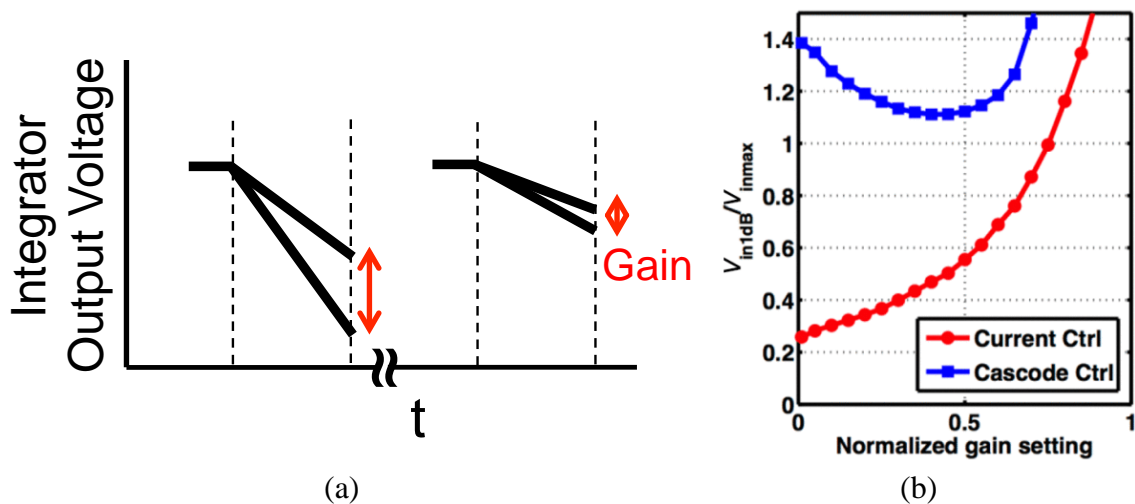


Figure. 2. 6. (a) Cascode DC bias control integrator output voltage (b) Linearity comparison between current-bias control and cascode DC bias control.

From Fig. 2.6(a), gain variation is achieved from varying differential output strength as transconductance impedance is modulated by the DC bias of the cascode switch. In Fig. 2.6(b),

$V_{in1dB}$  is the input voltage at which the gain drops by 1 dB from its target value, and  $V_{inmax}$  is the maximum input swing expected to be fed into the FFE stage. In the conventional, current-bias control (in red),  $V_{in1dB}/V_{inmax}$  varies from 0.2 to above 1.4 across gain settings. This variation in gain is not acceptable for a linear amplifier; hence, it cannot be used in the FFE, since it would add ISI that cannot be equalized. Instead, the cascode DC bias control (in blue) shows much better linearity with  $V_{in1dB}/V_{inmax}$  being relatively flat across gain settings (1.4 to 1.1).

It is important to note the while linear signal-transfer characteristics must be preserved as to not introduce signal-dependent ISI, tap strength vs. code does not need to be linear. Instead, it only has to be monotonic so that the adaptation loops settle appropriately.

### 2.3.3 Integrating FFE+DFE

Since DFE taps do not need to have linear signal-transfer characteristics as in the FFE taps, the DFE employs current-bias control for DFE taps. Rather than summing the FFE and DFE with separate summers, an FFE+DFE can be combined into an integrating summer stage (Fig. 2.7).

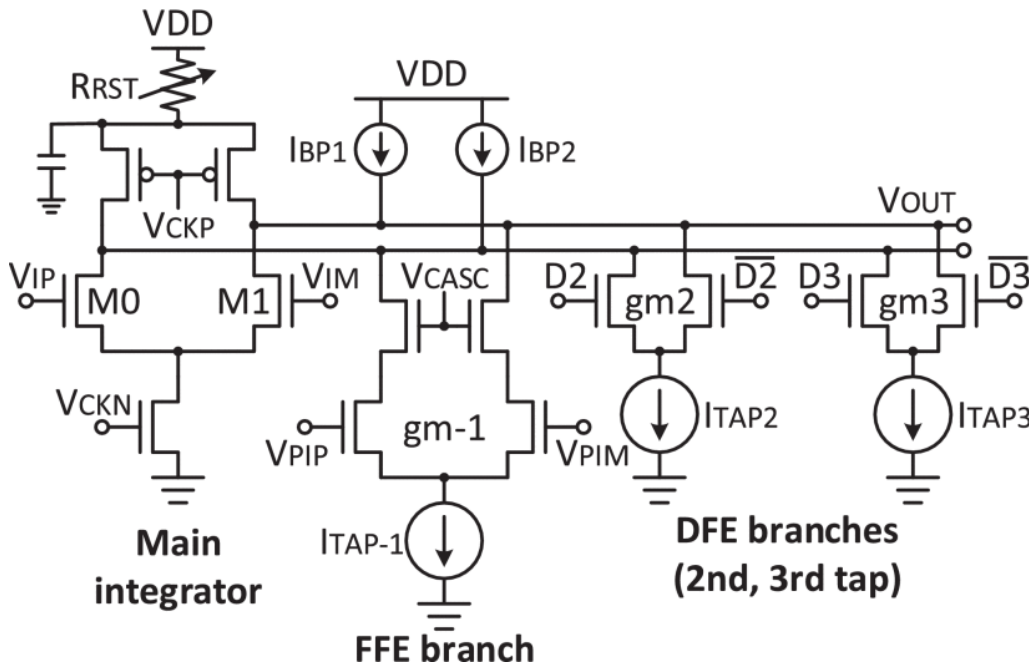


Figure. 2. 7. FFE + DFE integrator.

Variable  $R_{RST}$  at the supply of the integrating summer allows for variable common-mode-reset voltage. Additional decoupling capacitance is added to this supply for robustness. The FFE pre-cursor and DFE's tap-2 and tap-3 post-cursors are summed at this node, while the DFE's tap-1 is summed at a following stage shown in the following section.

Instead of trying to settle the output node completely in the short reset window ( $\sim 16ps$  at our data-rate) to avoid introducing any ISI, the resetting switch was sized to only settle to  $\sim 85\%$  of the final value ( $2\tau$  of settling) since the PMOS switches in 65nm are substantially worse than NMOS switches, and hence introduce considerably more capacitive loading. Not fully settling introduces extra ISI, which is demonstrated in Fig. 2.8 below.



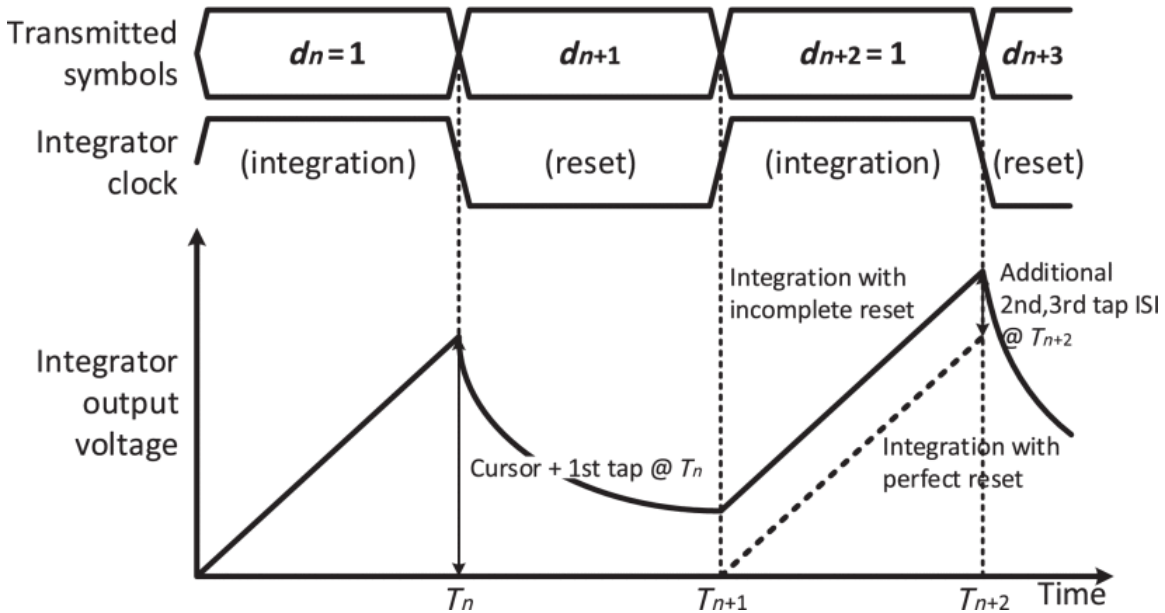


Figure. 2. 8. Incomplete reset setting (post-cursor ISI).

Fig. 2.8 shows the integrator output voltage during integration and reset for sequential cycles assuming  $2\tau$  of settling. As seen from the incomplete settling at  $T_{n+1}$ , the integrator is not given enough time to completely settle to  $0V$  differential (compared to the perfect reset from the dotted line and also shown in the ideal integrator output voltages in Fig. 2.2(b)). This incomplete settling time translates to second and third tap ISI (and also in later taps, but these are assumed to be small due to the suppression from multiple resets). This extra ISI is handled by overdriving the DFE (using increased  $d_i$  coefficients), costing additional power in the form of increased loading from increased maximum current in the post-tap differential pairs.

This tradeoff of incomplete resetting (due to the increased time constant of PMOS devices) and the additional tap strength required to cancel the introduced ISI (Fig. 2.9(a)) can be quantified through the following equations and the optimal reset error desired for minimal power is simulated to be 85% from Fig. 2.9(b).

From [13], the power consumption of a current-integrating DFE ( $P_{int}$ ) at data-rate  $f_s$ , gain  $G$ , and load capacitance  $C_L$  was shown to be

$$P_{int} \propto \frac{I_{nom} \cdot 2(1 + k_{reset})}{1 - \frac{G \cdot f_s}{\omega_{T,in}} \cdot \gamma \left(1 + k_{tap0} \frac{v_c}{V_i^*} \beta\right)} \cdot 2(1 + k_{reset}) \quad (1.8)$$

where  $I_{nom} = C_L V_i^* G f_s$  is the current-consumption of a class-A amplifier without self loading,  $\omega_{T,in}$  is the transit frequency of the input transistor,  $V_i^*$  is the overdrive voltage of the input transistor,  $v_c$  is the input cursor amplitude,  $k_{tap0}$  is the ratio between the ISI and cursor,  $\gamma$  is the drain-to-gate-capacitance ratio,  $\beta = \omega_{T,tap}/\omega_{T,in}$  is the ratio between the transit frequencies of the tap transistors and the input transistor, and  $k_{reset}$  is the factor by which the capacitance of the summing node needs to be increased to include reset capability. The effects of incomplete reset

are captured by an equivalent RC network composed of a PMOS reset device and load capacitances at the output where the time constant of the reset network can be expressed as

$$\tau_{reset} = \tau_p \left( \frac{1 + k_{reset}}{k_{reset}} \right) = \frac{\alpha_r}{f_s N_{\tau,reset}} \quad (1.9)$$

where  $\tau_p = r_{op} C_{Dp}$  is the PMOS time constant without additional loads,  $\alpha_r$  is the fraction of each UI spent for resetting, and  $N_{\tau,reset}$  is the number of reset time constants desired.

$$k_{reset} = \frac{\tau_p}{\frac{\alpha_r}{f_s N_{\tau,reset}} - \tau_p} \quad (1.10)$$

Tap strength is increased by  $e^{-N_{\tau,reset}}$  to compensate for the increased ISI induced from incomplete settling.

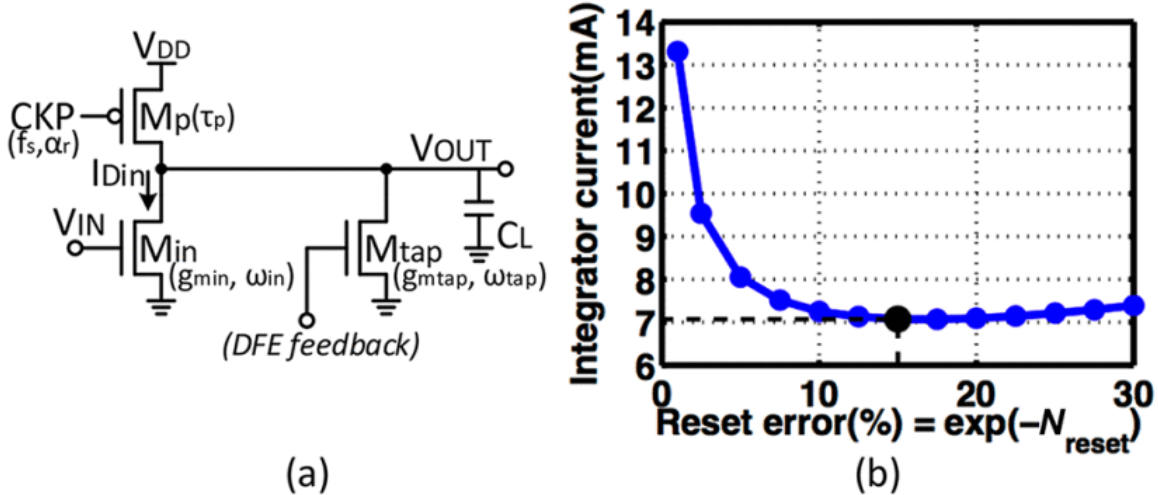


Figure. 2. 9. (a) Simplified DFE circuit diagram (b) Optimal reset accuracy.

The optimal reset accuracy is relatively flat around 15% reset error and saves almost half the power compared to almost completely resetting (Fig. 2.9(b)).

### 2.3.4 DDR receiver frontend

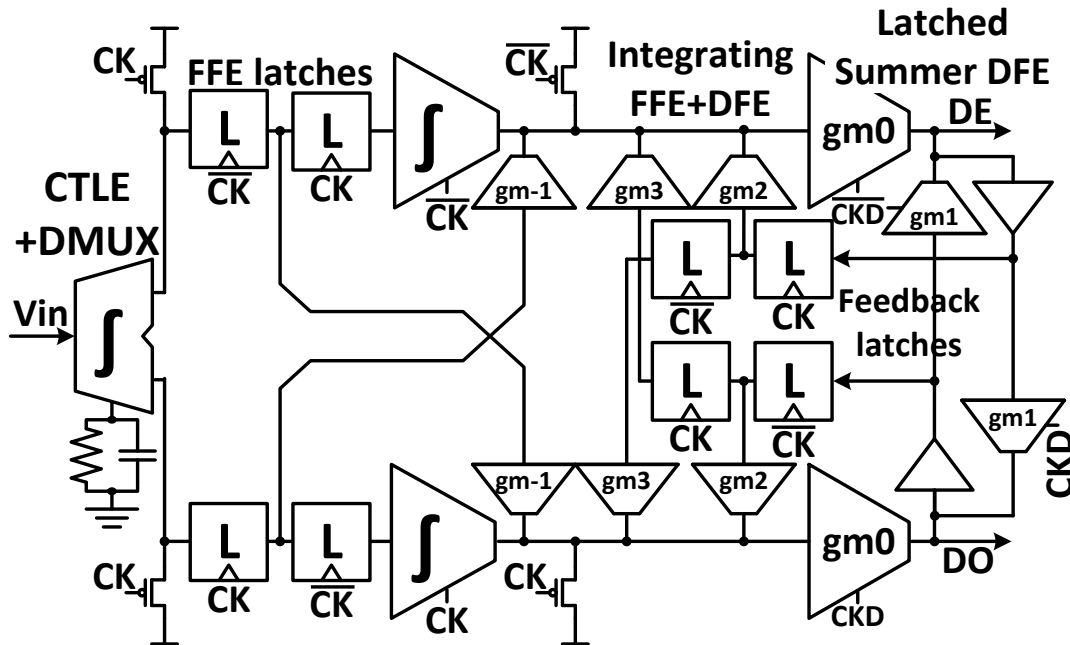


Figure. 2. 10. Receiver frontend architecture.

The receiver frontend architecture is shown in Fig. 2.10, where full-rate data is equalized and demultiplexed by an integrating CTLE + DMUX. Latches turn the CTLE output into NRZ signals, which are further delayed and taken as the UI-delayed signals for the DDR FFE. The integrating FFE+DFE summer combines the pre-cursor FFE tap and the DFE's tap-2 and tap-3 summing. A dedicated latch is used as the summer for the first DFE tap as in the DFE in [9] to meet the stringent-timing requirements of closing the first DFE tap.

## 2.4 60Gb/s receiver frontend results

This architecture was first demonstrated in a receiver frontend in a 65nm process (Fig. 2.11(a)). The equalizer-core circuit occupies  $0.012 \text{ mm}^2$ , and the fabricated chip occupies  $0.16 \text{ mm}^2$  (except for the pad, ESD, and t-coil area). The chip was as directly soldered onto a Nelco 4000-13 PCB via flip-chip bumps to minimize parasitic loading from package structures.

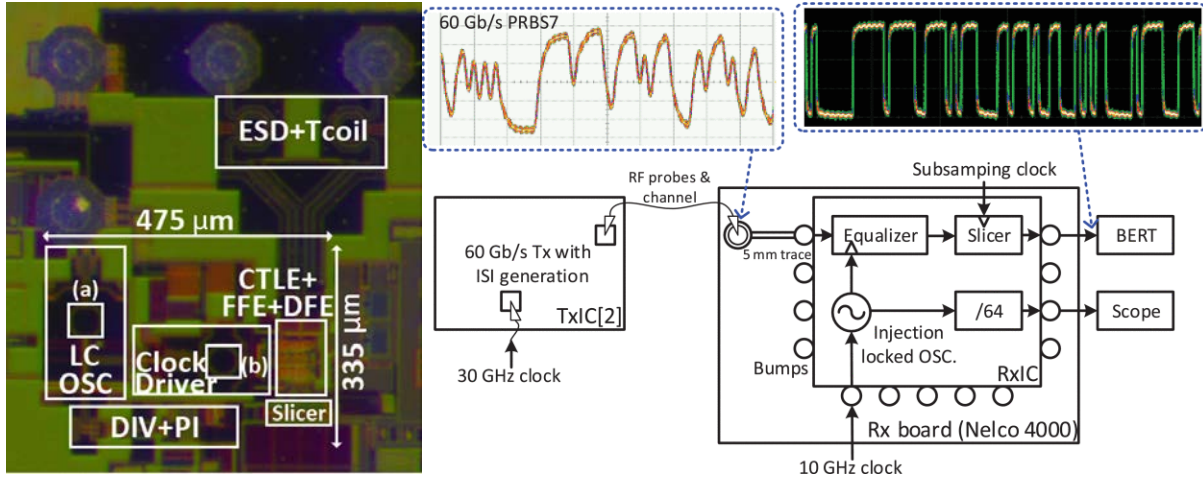


Figure. 2. 11. (a) Die photo (b) Measurement setup and measured waveforms (All clock sources are synchronized.).

Since no 60Gb/s signal source was available for the measurement, the pattern-generator/channel-emulator circuit from the chip described in [9] was reused for testing the receiver (shown in the measurement setup in Fig. 2.11(b)). The band-limited transmitter provided differential PRBS7 signals with emulated channel profiles for our receiver to equalize.

Eye diagram measured at the input of the receiver frontend evaluation board shows a closed eye before equalization (Fig. 2.12(a)) with an estimated pulse response in Fig. 2.12(b).

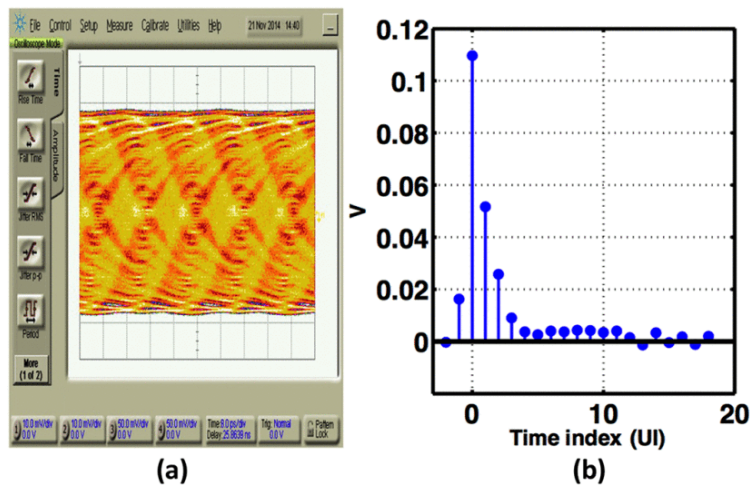


Figure. 2. 12. (a) Eye Diagram at the channel output (b) Estimated pulse response.

The chip was tested with a 10 GHz clock generator (Keysight E8267D) synchronized with a 30 GHz transmitter clock source (Keysight E8257D) to provide the injection clock (with the oscillator injection locking enabled) with different phases for bathtub characterization. Furthermore, a Keysight 86130A BERT measures the BER of the reconstructed PRBS7 pattern from 1/128x sub-samplers, clocked by an external source (Keysight E8267D). Under these conditions, the receiver front-end recovers the transmitted PRBS pattern and operates at 60Gb/s error-free over  $10^{12}$  bits for the phase offset shown in the bathtub curve in Fig. 2.13 (and with  $> 0.2$  UI-timing margin at  $1e-9$  BER).

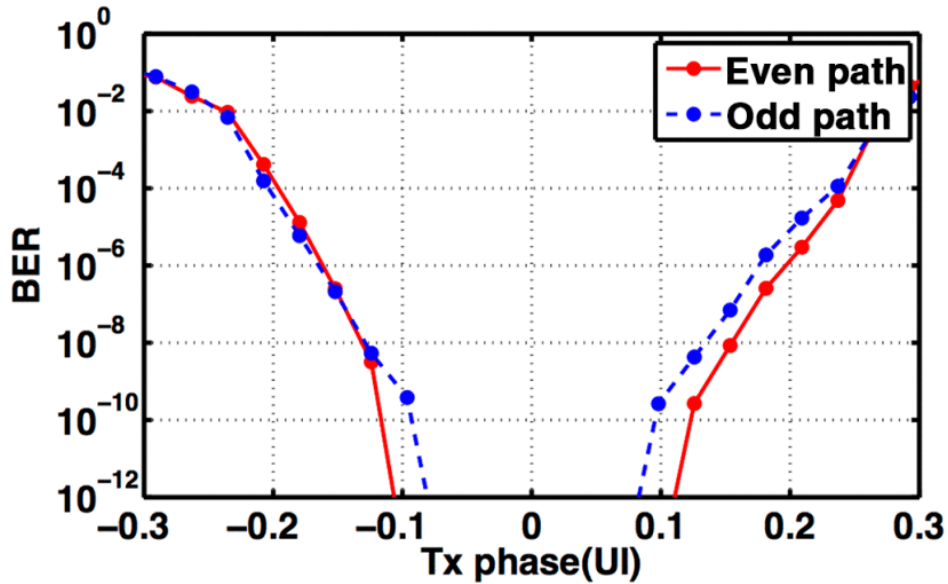


Figure. 2. 13. Bathtub curve after equalization.

The receiver achieves 60 Gb/s, consuming 173mW (138mW from a 1.2V and 35mW from 1.0V supply) in a 65nm process, demonstrated to equalize 1.54 times the cursor amplitude of ISI. Table I compares the design with prior 56-80 Gb/s equalizer designs.

Table 2. 1: Comparison of high-speed receiver equalizers

Reference	[9] JSSC'2013	[10] JSSC'2014	[11] VLSI'2014	This work
Process	65nm CMOS	130nm SiGE	20nm CMOS	65nm CMOS
Data-rate (Gb/s)	66	80	56	60
Equalizer	3-tap DFE	1-tap DFE	External 2-tap FFE (6 dB) CTLE 1-tap DFE	CTLE 2-tap FFE 3-tap DFE
$V_{ISI}/V_{CURSOR}$ or channel loss (dB)	1.65	12 dB	23 dB	1.54
Power (mW)	46	4000	177*	173
Equalizer	46	1772		48
Deserializer	N/A	N/A		28
Clock generation	N/A	N/A	N/A	52 <sup>o</sup>
Clock distribution	N/A	2228		45
Efficiency (pJ/bit)	0.7	50	3.16	2.88

\*Includes equalizer, 4:16 DES, clock distribution

Includes output buffer

<sup>o</sup>LC oscillator + divider + PI

Integration (in the CTLE, FFE, and DFE) is shown to be an energy-efficient technique for equalization at high speeds, as the complete receiver frontend consumes 2.88pJ/bit, equalizing ISI of 1.54 times the cursor.

The receiver frontend was demonstrated again in a complete transceiver design, and the results shown next.

## 2.5 60Gb/s complete transceiver results

The data-path circuitry adapted from the receiver was implemented in a complete transceiver architecture operating at 60Gb/s with adaptive equalization and a baud-rate CDR. A current mode logic (described in further detail in chapter 3) transmitter was used to implement the transmitter at higher speeds. The transceiver architecture with 1:128 SERDES ratio, 3-tap TX FFE, 2-tap RX FFE, CTLE, and 3-tap DFE, is shown in Fig. 2.14, and the clocking, adaptation and CDR is described in detail in [18].

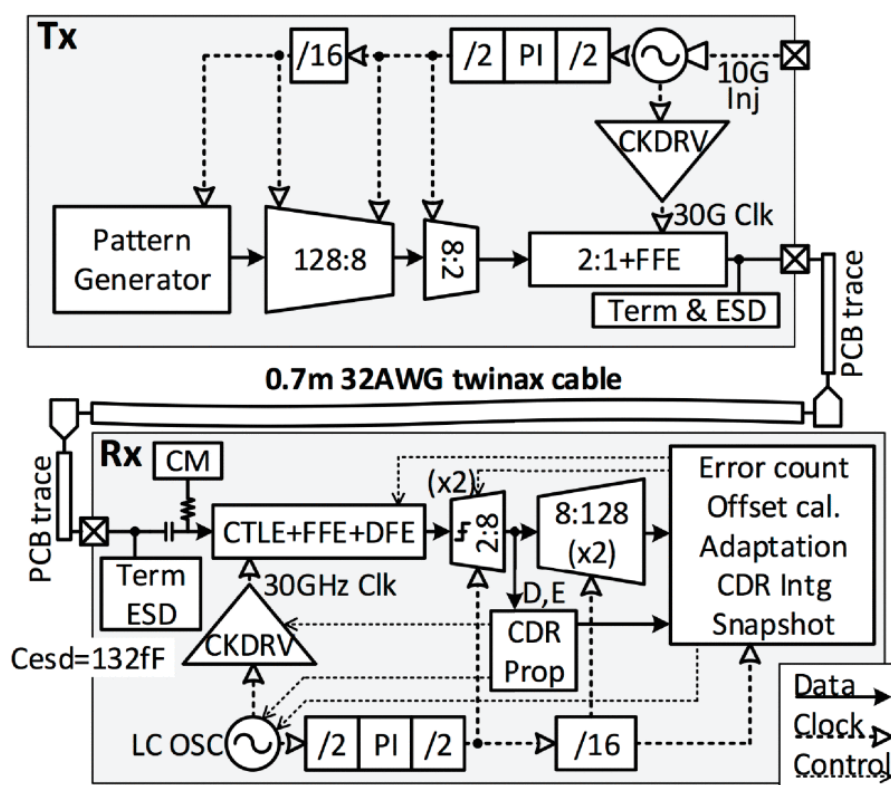


Figure. 2. 14. Transceiver architecture.

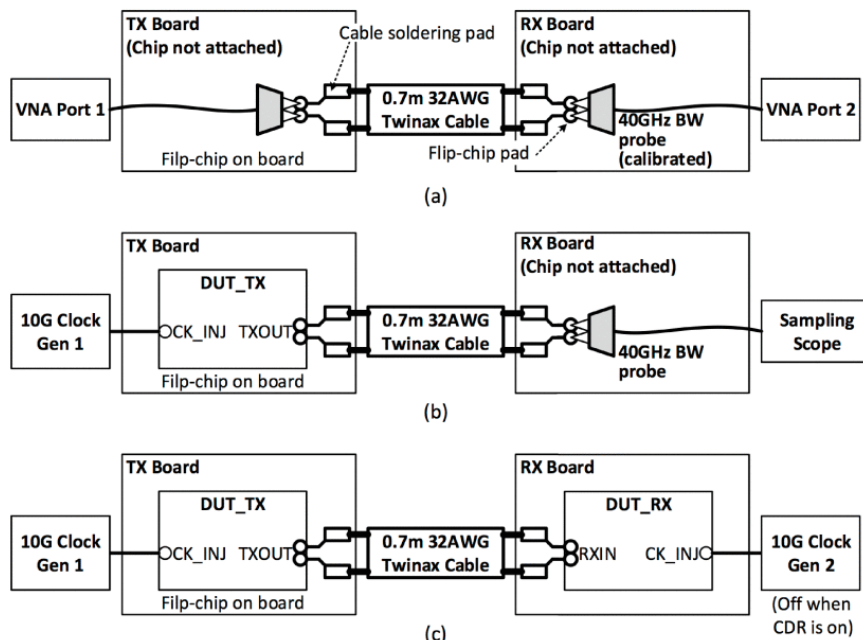


Figure. 2. 15. Measurement setups for (a) channel frequency response, (b) pulse response, and (c) equalizer and CDR characterization.

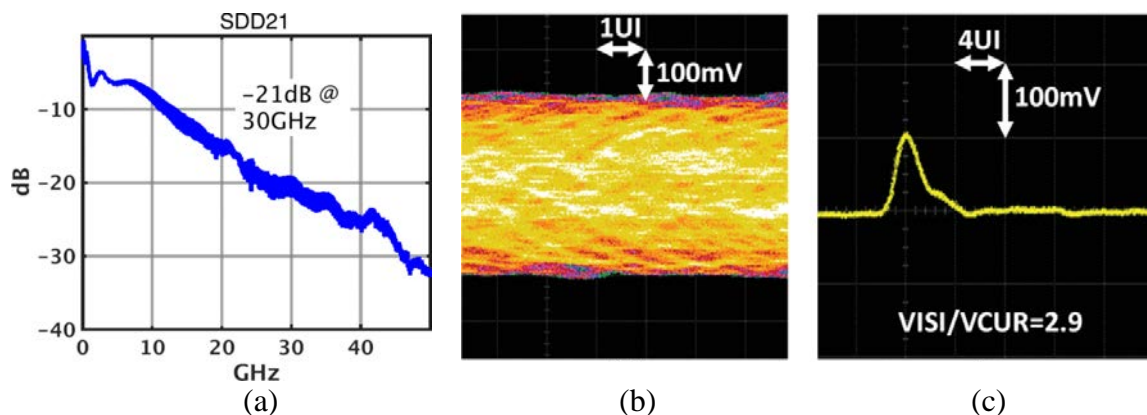


Figure. 2. 16. (a) Measured channel frequency response (b) TX + channel eye diagram (c) TX + channel pulse response.

Figure 2.16(a) shows the measured  $S_{21}$  of the channel from the configuration in Fig. 2.15(a), showing -21dB of insertion loss at 30GHz. The measurement setup in Fig. 2.15(b,c) where the a Keysight E8267D is used to generate a 10-GHz reference clock to injection lock the transmitter's 30-GHz LC oscillator is used to characterize the TX + channel eye diagram before equalization, showing a closed eye (Fig. 2.16(b)) and  $V_{ISI}/V_{cursor}$  of 2.9. In particular, pre-cursor ISI, significant post cursor ISI, and long-tail ISI are observed in the measured pulse response (Fig. 2.16 (c)), necessitating the use of the complete CTLE+FFE+DFE equalizer chain.

With equalization turned on, on-chip eye diagrams and a bit error rate bathtub curve were measured with the setup in Fig. 2.15(c).

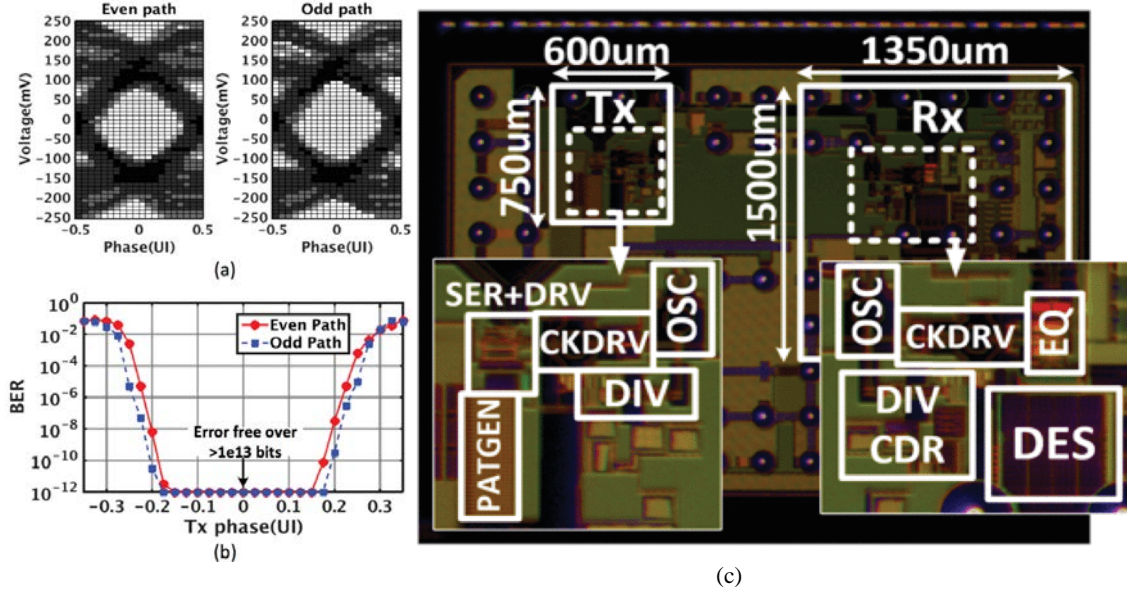


Figure. 2. 17. (a) Eye diagrams (b) Bathtub curve (c) Die photo.

The design achieves  $>0.3$  UI eye opening at  $10^{-12}$  BER for both the odd and even paths (Fig. 2.17(a,b)). The die photo is shown in (Fig. 2.17(c)) with the transceiver occupying  $2.48 \text{ mm}^2$  (TX:  $0.45 \text{ mm}^2$  and RX:  $2.03 \text{ mm}^2$ ), and the performance is compared against various prior PAM4 and NRZ 54+ Gb/s transceivers in Table II below.

Table 2. 2: Comparison of high-speed transceivers

Reference	[19] VLSI' 2016	[20] ISSC' 2017	[21] VSL' 2015		[22] ISSCC' 2016	This work
<b>Modulation</b>	PAM4	PAM4	PAM4	NRZ	NRZ	<b>NRZ</b>
<b>Process</b>	16nm	40nm	40nm	40nm	28nm	<b>65nm</b>
<b>Data-rate (Gb/s)</b>	56	56	54.1~56.8	55.5~56.5	65	<b>60</b>
<b>Channel loss (db)</b>	25	25	N/A	N/A	18.4	<b>21</b>
$V_{ISI}/V_{CURSOR}$	-	-	N/A	N/A	-	<b>2.9*</b>
<b>Equalizer</b>	3-tap TX FFE CTLE DSP	3-tap TX FFE CTLE 3-tap DFE	CTLE	CTLE	2-tap TX FFE CTLE 1-tap DFE	<b>3-tap TX FFE 2-tap RX FFE CTLE 3-tap DFE</b>
<b>SERDES ratio</b>	1:32	1:64	1:16 (TX) 1:2 (RX)	1:8	1:32	<b>1:128</b>
<b>Adaptation</b>	Y	N	N	N	Y	<b>Y(per-path)</b>
<b>Eye opening</b>	N/A	25% @ $1e-9$	N/A	N/A	28% @ $1e-9$	
<b>Tx Power (mW)</b>	140	200	290	450	104.7	<b>152</b>
<b>Rx Power (mw)</b>	370	382	420	220	141.7	<b>136</b>
<b>Tot. Power (mW)</b>	550 <sup>o</sup>	602 <sup>‡</sup>	710	670	246.4 <sup>†</sup>	<b>288</b>
<b>Tx Efficiency (pJ/bit)</b>	2.5	3.57	5.17	8	1.87	<b>2.53</b>
<b>Rx Efficiency (pJ/bit)</b>	6.61	6.82	7.5	3.93	2.53	<b>2.26</b>
<b>Efficiency (pJ/bit)</b>	9.82	10.75	12.67	11.96	4.4	<b>4.8</b>

\* $V_{ISI}/V_{CURSOR}$  is measured from a probing setup with additional  $<2\text{dB}$  loss @ 30GHz



<sup>◊</sup>DSP power not included, 40mW clocking power

<sup>‡</sup>20mW clocking power

<sup>†</sup>Clocking power is amortized over 2 lanes

The NRZ transceiver equalizes  $V_{ISI}/V_{CURSOR}$  of 2.9 with a transmitter power of 152mW and receiver power of 136mW, achieving 4.8 pJ/bit at 60 Gb/s.

## 2.6 Conclusion

Various high-speed receiver equalization techniques are described to achieve 60Gb/s, demonstrated in two chips fabricated in 65nm. The integration techniques, requiring dynamic data-path circuitry strongly contribute to the efficiency achieved in the RX (2.26 pJ/bit in the complete transceiver). While much focus was placed on the receiver architecture (DDR, integration, and incomplete reset) for minimal power consumption, our transmitter still employed traditional, less energy-efficient methods (current mode logic driver) to achieve 60Gb/s operation. Furthermore, ultra-high-speed (56Gb/s+) links are still sparingly used because they consume too much power (even at the energy-efficiencies demonstrated) to be adopted for all off-chip communication channels (requiring varying levels of speed).

In order to provide more ubiquitous, higher throughput capabilities to the rest of the I/O, an energy proportional link, requiring less overall energy for any given data-rate is desired. In particular, data-rate flexibility is desired, requiring data-path circuitry to be dynamic (as in the case of the dynamic latches and integrators in our receiver front-end equalizers, and not the case in our transmitter architecture). The next chapter introduces the concept of energy proportional communication, and goes further into energy efficient transmitter techniques that can be used for energy-efficient, data-rate-flexible communication.

## Chapter 3

### Energy Proportional Communication

As we become less reliant on improvements manifested through pure transistor scaling (exhibited by the slowdown of Moore's Law), there has been a shift toward more designs incorporating parallelism and moving further away from Von Neumann architectures, since it has become too costly to be limited by a single CPU's memory access latency (Figure. 3.1).

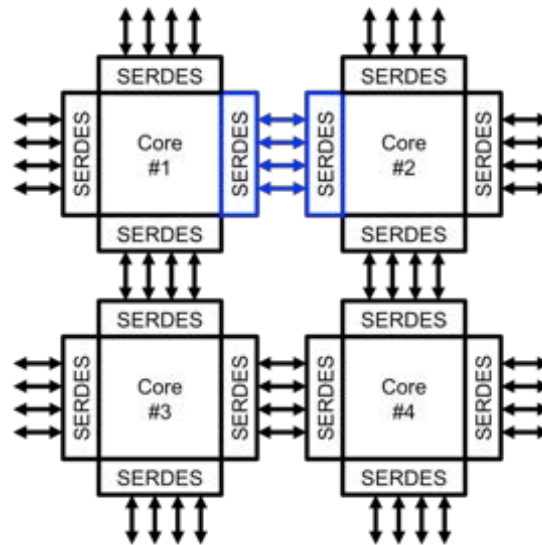


Figure. 3. 1. Parallel processing.

Although Von Neumann computing makes it undesirable to go off chip (to memory or other computing elements), systems more than ever need to communicate with multiple application specific ASICs (not just due to memory). Off-chip latency is costly but is needed in the case of memory due to limited on-chip memory (cache) size constraints, and this problem presents itself again as we add more CPUs.

Multiple CPUs/GPUs and separate ASICs for accelerators mean systems are beginning to rely on more hops between chips to accomplish a task, demanding even more distributed communication in the following years [25]. There has been an increasing demand for unified communication standards [24][25] for various reasons, whether it be flexibility, efficiency, and/or fundamental design requirements. A low latency, unified standard for chip-to-chip communication is needed for distributed computing (Fig. 3.2).

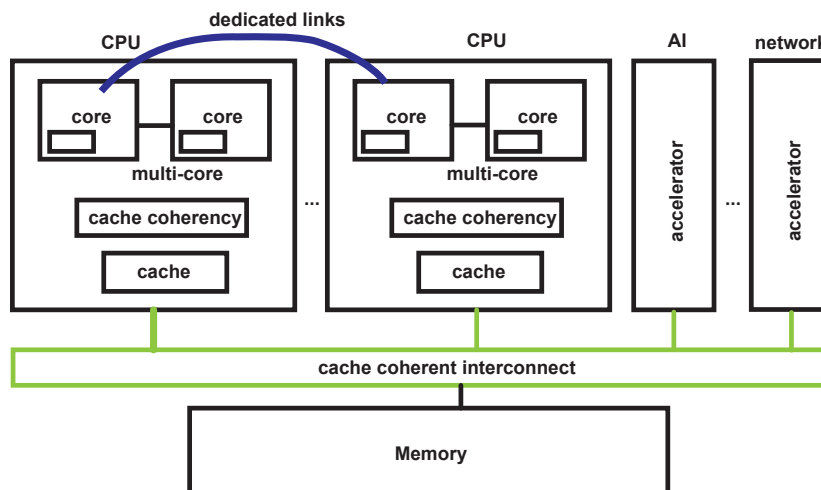


Figure. 3. 2. Unified communication.

Figure. 3.2 shows both high-speed, dedicated CPU-CPU links as described in Chapter 2 and a unified communication channel that is necessary between various ASICs where dedicated links are not feasible. When deciding the requirements that a unified link like this should have, energy efficiency and latency are important to consider. Ideally, it should support the maximum data-rate for high-speed connections while also maintaining energy proportional efficiency for lower data-rate communication. Furthermore, the ability to save power by operating in standby can be an additional benefit. Finally, burst-mode functionality allows for data to be transmitted always at the maximum data-rate, inherently being the lowest latency solution.

Due to advancements in packaging technology (and interposers typically used to spread connections off chip to a wider pitch), various systems have begun to take advantage of packaging technology to make connections between chips within the package. One example can be seen through the interconnect-technology advancements (and die stacking) in High Bandwidth Memory (HBM), where dies are meant specifically for I/O redistribution [26]. For systems like this, the large number of signals available mean I/O energy costs must be reduced. Various designs seek to operate in these short reach conditions mentioned including the 25Gb/s/pin ground-referenced single-ended serial link presented by Poulton [27] and 56Gb/s/lane transceiver for common electrical I/O short reach standards by Shibasaki in [22].

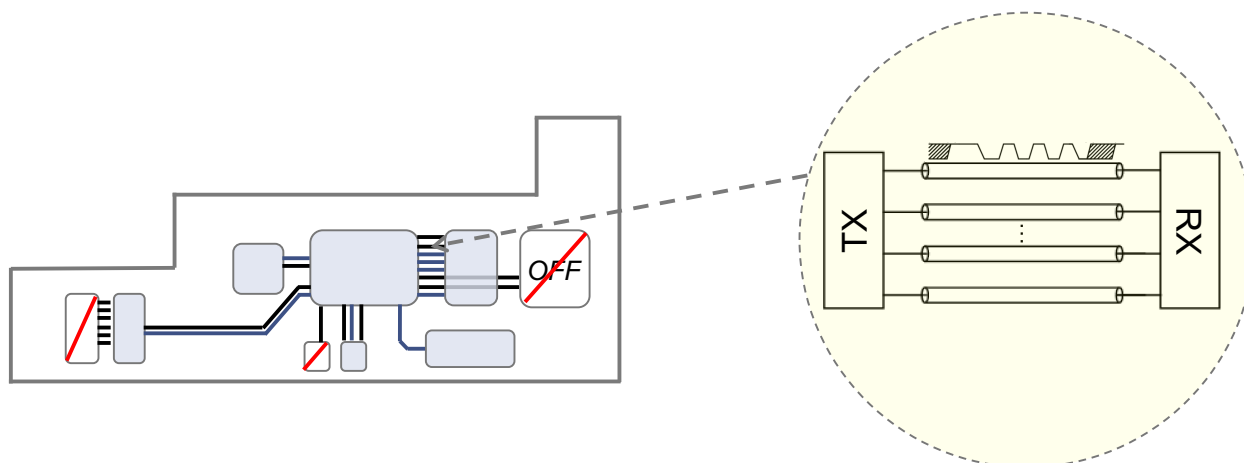


Figure. 3. 3. Burst-mode communication environment (interposer illustration example).

Fig. 3.3 illustrates a generic example of an environment where one unified standard could support ASIC-ASIC communication on an interposer. In this example, various links could be on standby when certain links are not communicating, or when certain blocks are off. Furthermore, burst-mode communication can offer flexibility by supporting lower data-rate standards by providing throughput in bursts to achieve a desired effective data-rate, since various communication standards require different link bandwidths (USB, HDMI, DDR, CPU-CPU, etc. ...). Fig. 3.4 shows the relevant characteristics of a data stream in a burst-mode signaling system.



Figure. 3. 4. Burst-mode data stream.

$$D_{effective} = \frac{D_{packet}}{t_{data} + t_{startup} + t_{shutoff} + t_{standby}} \quad (3.1)$$

Effective data-rate ( $D_{effective}$ ) in (3.1) can easily be controlled by sending valid data bursts of length ( $t_{data}$ ) at the nominal link data-rate. Limiting standby power, start-up and shut-off time are important considerations that affect energy efficiency and will be examined in the following section.

### 3.1 Energy proportional efficiency

Various flexible data-rate links have been presented by using the same circuitry for all data-rates and/or bypassing (or disabling) limited portions of the equalization architecture when communicating at lower data-rates [28][29][30], or by operating in NRZ vs. PAM-4 [31]. These flexible transceivers, while performing efficiently (around <10pJ/bit) at max data-rate meet standard requirements to support lower data-rates of legacy backplanes but consume effectively the same power when backed-off to lower data-rates.

Energy efficiency (Power (J) /Data-rate (bit)) would therefore degrade proportionally as data-rate decreases during back off since power remains constant while data-rate scales. It is important to note this is not the case in all flexible data-rate links, as some achieve varying amounts of efficiency scaling by disabling some circuitry for selected data-rates. For simplicity, we assume power remains constant for data-rate flexible links in the following analysis.

Energy proportional communication instead aims to achieve close to constant energy efficiency regardless of data-rate. Instead of varying the nominal data-rate to operate over a range of data-rates, these links achieve energy proportionality by operating at a nominal high-speed data-rate when on and by being placed into standby otherwise. Hence, efficiency will always be based on the nominal power and data-rate plus the startup overhead. Effective efficiency can be determined by considering how much overhead is spent while turning the link on and off and how long the data is valid compared to the standby period.

Effective efficiency ( $\epsilon_{effective}$ ) for these systems can be calculated as

$$\varepsilon_{effective} = \frac{E_{data} + E_{startup} + E_{shutoff} + E_{standby}}{D_{effective}} t_{total} \quad (3.2)$$

The following section demonstrates how effective efficiency in a burst-mode link compares to a data-rate back-off link.

### 3.1.1 Comparison between burst-mode and data-rate back-off

Communication links with the capability to back off in data-rate allow for throughput flexibility, but lack in energy efficiency during back-off compared to an energy proportional link since they consume effectively the same power or similar power, especially when operated at low data-rates.

On the other hand, burst-mode communication always operates at the nominal link data-rate, so efficiency scales based on (3.2) instead of simply linearly scaling in the data-rate back-off case where energy is constant. Assuming a 40Gb/s nominal data-rate link with 1pJ/bit efficiency, burst-mode and back-off efficiencies are compared and plotted vs. data-rate in Fig. 3.5.

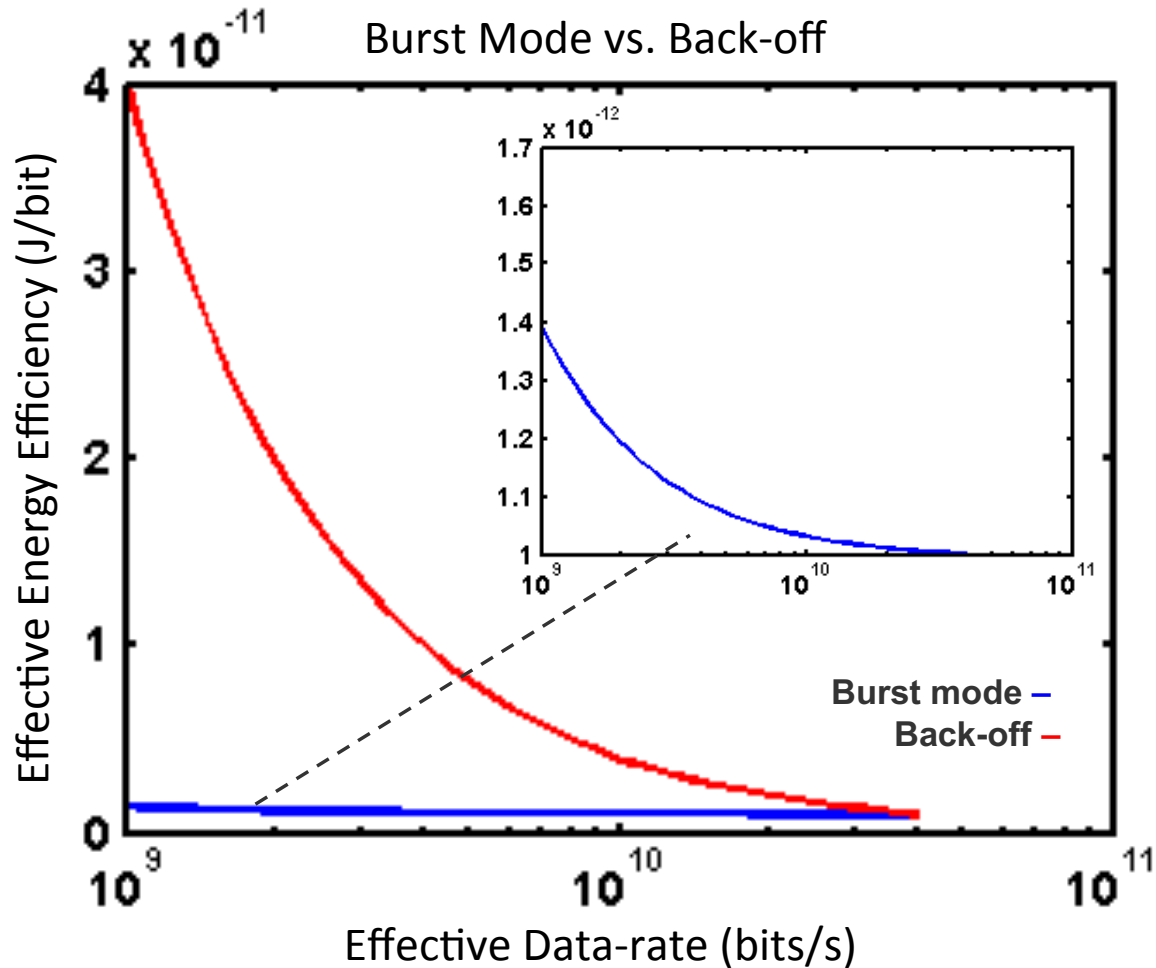


Figure. 3. 5. Comparison between effective efficiency of burst-mode (blue) Data-rate back-off (red). Assumes a 1pJ/bit 40Gb/s nominal link is burst at 4KByte intervals with standby-power being 100x less than on-power.

In the data-rate back-off scenario (red), the  $\epsilon_{effective,back-off}$  degraded by 4x when operating at 1Gb/s instead of the nominal 40Gb/s data-rate. Energy proportional communication is exemplified for burst-mode scenario (blue), since  $\epsilon_{effective,burst\ mode}$  only degraded by 40% at 1Gb/s. The burst-mode scenario therefore exhibits energy proportionality since it is relatively constant (especially compared to the back-off scenario). The next section discusses exactly how various burst-mode overheads affect energy efficiency.

### 3.1.2 Efficiency vs. effective data-rate and $P_{standby}$

How important is standby power and how does energy efficiency scale differently with link activity? Links where low activity is more frequent would benefit the most from minimizing standby power, but by how much?

The following contour plot shows how energy efficiency scales as standby power and link activity changes (Fig. 3.6).

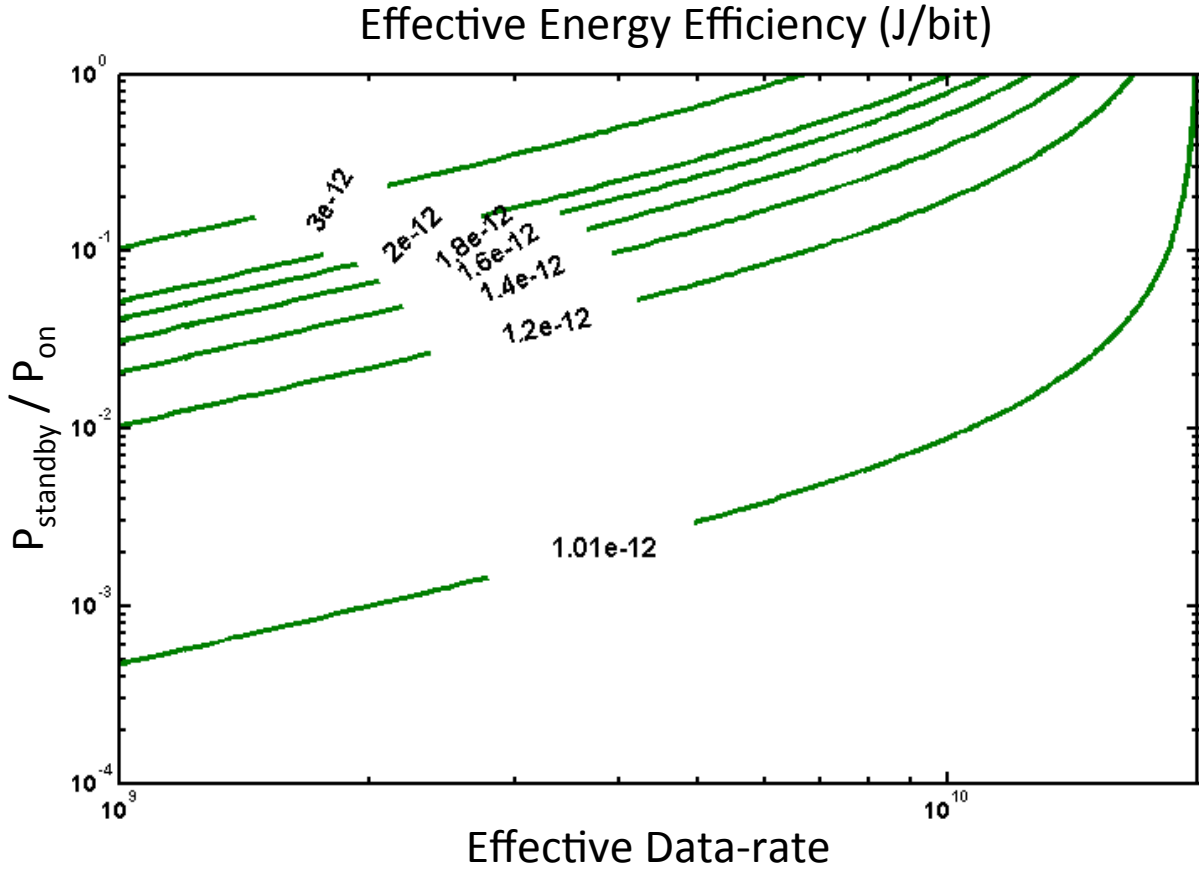


Figure. 3. 6. Effective energy efficiency contour plot. (assumes  $t_{data}/(t_{startup} + t_{shutoff}) = 1e3$ , 1pJ/bit nominal efficiency, 20Gb/s nominal data-rate)

$P_{standby}/P_{on} = 10^{-1}$  gives us around 3x degradation in efficiency when we get to 1Gb/s operation. Whereas,  $P_{standby}/P_{on} = 10^{-2}$  gives us around 20% degradation in efficiency when we get to 1Gb/s operation. If the universal burst-mode link expects to be used for a ~1Gb/s data-rate channel, it should also be designed to have energy proportional efficiency down to at least 1Gb/s. Hence, aiming for  $P_{standby}/P_{on} = 10^{-2}$  makes for a reasonable design target. While  $P_{standby}/P_{on} = 10^{-3}$  or less standby power consumption would allow for better energy efficient scaling during back-off, architectural design and technology constraints often make it difficult to achieve tens of  $\mu W$  of power. Especially when looking at the marginal efficiency benefits the compared to the to the  $P_{standby}/P_{on} = 10^{-2}$  countour line,  $P_{standby}/P_{on} = 10^{-2}$  gives us added flexibility in architectural design choices without hurting us too much on this trade-off.

### 3.1.3 Multilane signaling

It is important to consider the energy efficiency for the overall systems that will ultimately benefit from burst-mode communication. Multilane systems demand for burst-mode signaling in that they must support a large amount throughput extremely quickly. Throughput and latency go

hand in hand when data needs to be queued up due to bandwidth limitations introduced from going off-chip. Here, throughput becomes a limiting factor on latency.

Aggregated energy efficiency in a multilane signaling environment will ultimately guide us in our architectural choices. A similar optimization can be taken further for a system with multiple variable data-rate lanes. A comparison between three options: data-rate back-off, throttling, or multilane control illustrated in Fig. 3.7 below.

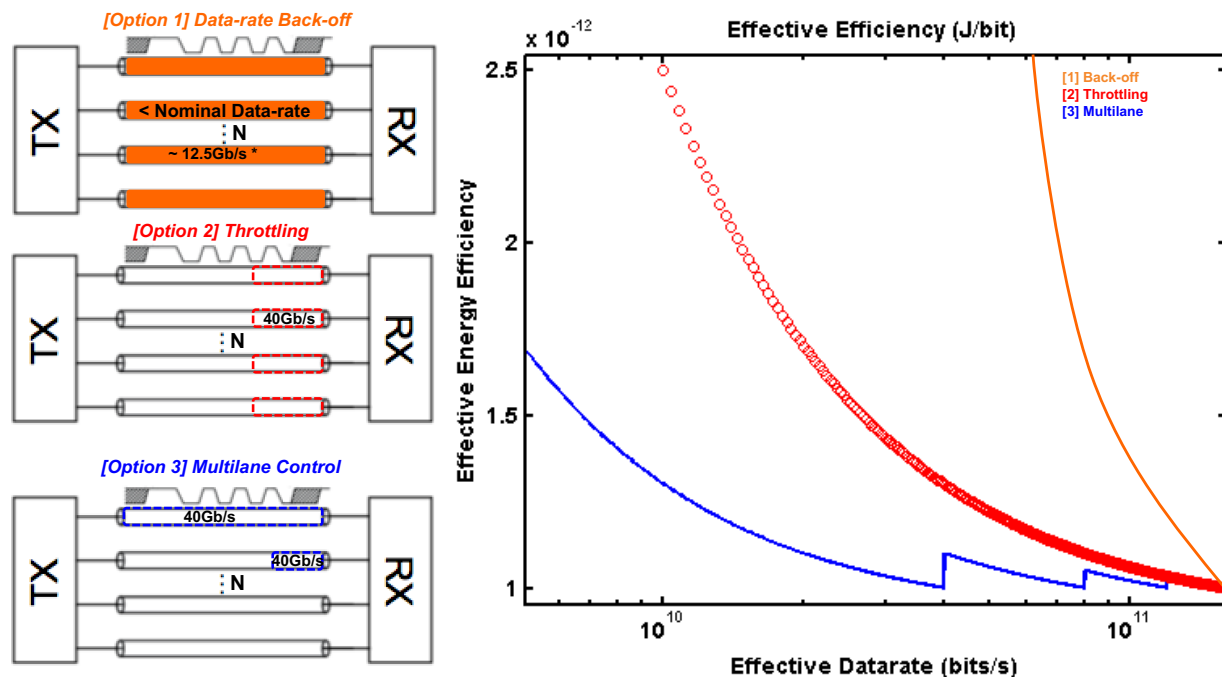


Figure. 3. 7. Comparison between data-rate back-off, throttling, and multilane control in a multilane signaling system.

This example demonstrates a multiple lane system where an effective data-rate of 50Gb/s (out of a possible of 200Gb/s) for a nominal link data-rate of 40Gb/s is desired. Effective energy efficiency is plotted vs. effective data-rate. Option 1 is the continuous link with data-rate back-off, Option 2 shares the load equally among burst-mode links, and Option 3 allows for individual links to be in standby when throughput is not required.

Assuming that transceivers are designed to operate at a larger range of sampling frequencies, Option 1's scaling is such that the power remains a constant for all regimes. Similar to the one link case, this does not scale well since power is fixed regardless of data-rate. Throttling in Option 2 provides a much more suitable roll-off than in 1, allowing for much more energy efficient systems. While Option 3 provides much better energy efficiency at reduced data-rates, it suffers from extra latency compared to Option 2 and would require more complicated digital circuitry for actual implementation. Option 3 calls for per-lane clocking (a per-lane oscillator), motivating our proposed burst-mode link presented in Chapter 4. Various reasons including limited loading on the VCO and clock distribution challenges at high speed also motivate per-lane clocking.

The effect of varying standby power on effective Efficiency under Option 3 is examined further in Fig. 3.8.



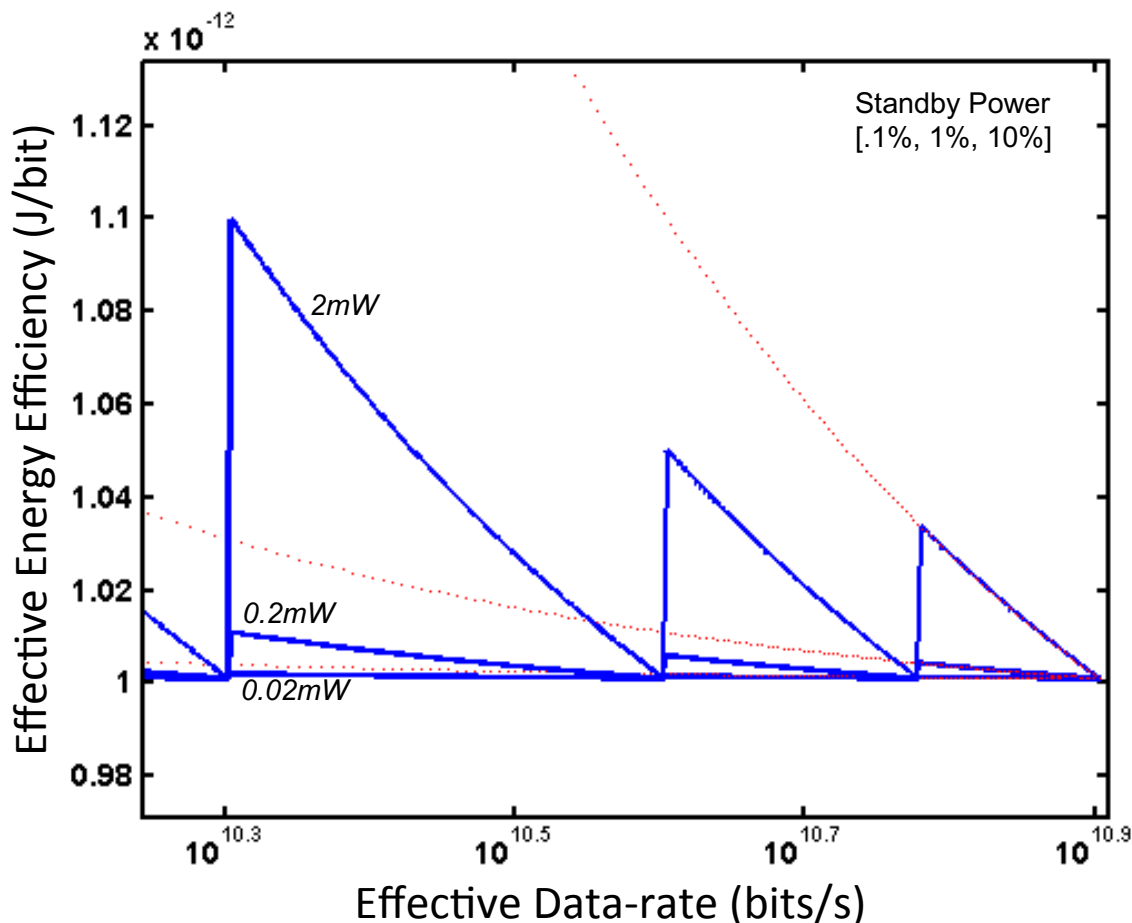


Figure. 3. 8. Multilane effective efficiency (vs. standby power).

In Fig. 3.8, the blue curves represent multilane control for standby power consumption of 0.1%, 1%, and 10% of on power consumption. The red dotted line represents multilane throttling which follows efficiency for the single lane burst-mode case.

These system level burst-mode constraints lead us to how we should approach our design. The next section discusses the architectural decisions that are made when designing for energy efficiency.

## 3.2 Transmitter design for energy efficiency

The circuit that naturally consumes the most power in the transmitter is the output driver. The output driver typically consists of either a voltage mode (VM) or current mode logic (CML) driving stage, which must be terminated to the impedance of the channel. Their efficiencies are compared analytically in the following section.

### 3.2.1 Comparison between voltage mode and current mode logic transmitters

Typically for always-on, high-speed wireline transmitters, this design choice is made based on power efficiency and technology limitations under a differentially terminated condition. The differential versions of these drivers are shown in Fig. 3.9. The CML driver operates off of the nominal supply voltage ( $V_{dd}$ ) and requires a constant current source to deliver power. Early versions of such a transmitter were demonstrated in [32][33]. The VM or source-series-terminated (SST) driver has been proposed as an alternative to the CML driver due to its benefits in energy efficiency with various versions demonstrated in [34][35][36][37]. The CML and VM drivers under TX and RX differentially matched (to line impedance) conditions are shown in Fig. 3.9 below.

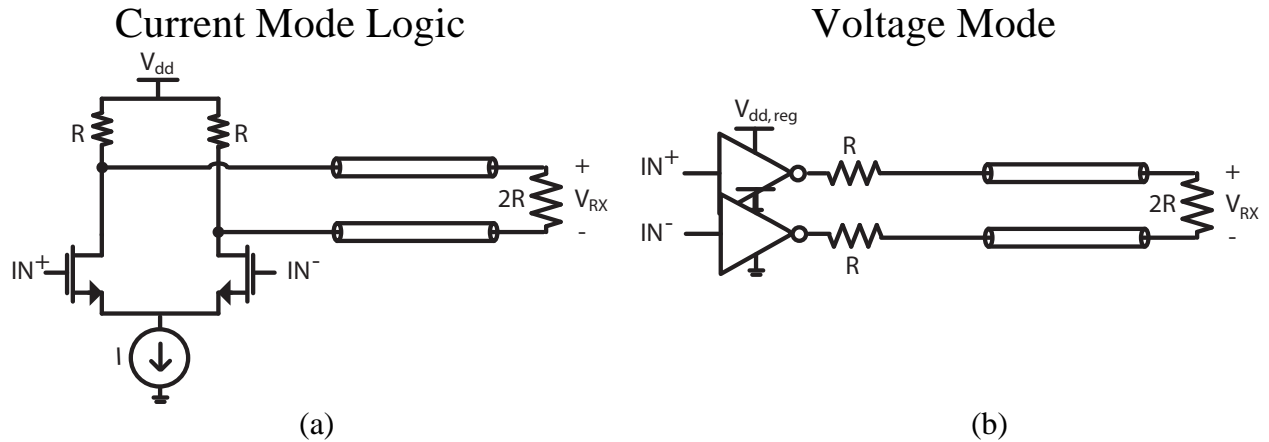


Figure. 3. 9. Comparison between voltage mode and current mode logic transmitters. (a) CML transmitter with differential termination at RX (b) VM transmitter with differential termination at RX. ( $R=Z_0=50\Omega$ , where  $Z_0$  is the impedance of the transmission line).

Power efficiency limits driver in the matched case for the VM and CML driver are compared in the following analysis. In Fig. 3.9 (a) only  $I/4$  (single-ended) gets to the RX termination. As follows, the voltage across the RX termination ( $V_{RX}$ ), can be found as

$$V_{RX} = \left(\frac{I}{4}\right)(2R) - \left(-\frac{I}{4}\right)(2R) = IR \quad (3.3)$$

$$I = V_{RX}/R \quad (3.4)$$

In Fig. 3.9 (b), the VM driver operates off of a regulated supply ( $V_{dd,reg}$ ), determining its output power found as

$$V_{RX} = V_{dd,reg}/2 - (-V_{dd,reg}/2) = V_{dd,reg} \quad (3.5)$$

Since current  $I = V_{dd,reg}/4R$ ,

$$I = V_{RX}/4R \quad (3.6)$$

Comparing (3.4) and (3.6) shows that there is a 4x reduction in driver current (or power) in the ideal VM driver compared to the CML driver. Hence, when optimizing for power in high-speed designs, the voltage mode driver is usually the choice as opposed to CML. However, the CML does have the benefit over the VM in that it can operate at higher frequencies due to technology limitations of the VM driver. Various CML drivers are used in high-speed transceiver designs as in the 66Gb/s PRBS generator in [29] and the high-speed NRZ and PAM-4 drivers in [38][39], but consume considerable amounts of power to achieve the high data-rates.

Even though the VM driver seems to be the optimal when considering instantaneous power, start-up time considerations and quiescent current consumption must be considered in deciding which architecture to choose for a burst-mode transmitter. The following section describes the challenges both architectures face.

### 3.2.2 VM and CML TX driver start-up considerations

Taking into account the circuitry that needs to be turned on before sending data and off to save power during standby ultimately leads us to our decision for our TX driver. From our efficiency analysis, aiming for *standby power/on power* =  $10^{-2}$  (vs.  $10^{-1}$  or worse) gives us considerable gains in energy proportional communication through a larger range of effective data-rates. The architecture we choose should be able to efficiently support both high-speed and slower-speed standards. Hence, major power consuming circuits should be able to turn on and off instantaneously and or consume close to no power.

Ultimately, both the CML driver and VM driver have various challenges that lead us to choose a different architecture.

In order to allow for on/off behavior in the CML driver, one solution involves adding start-up circuitry to turn off the current source during standby in order to achieve a nominal “off” mode. This would involve charging up the current source NMOS  $C_{gs}$  completely before being operational. Charging this current source gate capacitance is typically nontrivial since the TX driver drives a  $50\Omega$  load (Fig. 3.10).

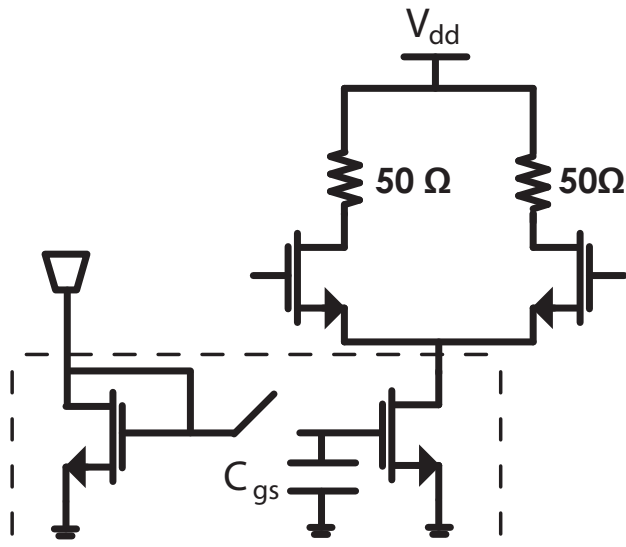


Figure. 3. 10. On/off CML (Current Source).

The on/off CML in Fig. 3.10 can also be separated (with a switch) at the drain of the current source as in the receive side data-amplifier in the source synchronous link in [39], resulting in a startup time of over five digital clock cycles. As we desire minimal quiescent current and a startup time around a nanosecond, we consider other options for the transmitter driver.

In the VM case, various regulators should be able to turn on and off quickly and consume little to no quiescent current during standby. Fig. 3.11 shows a typical VM driver incorporating an on/off regulator. Several challenges include biasing  $Vg$  and  $Vg'$ , and making sure that there is little supply drop, leading to interference dependent on data length.

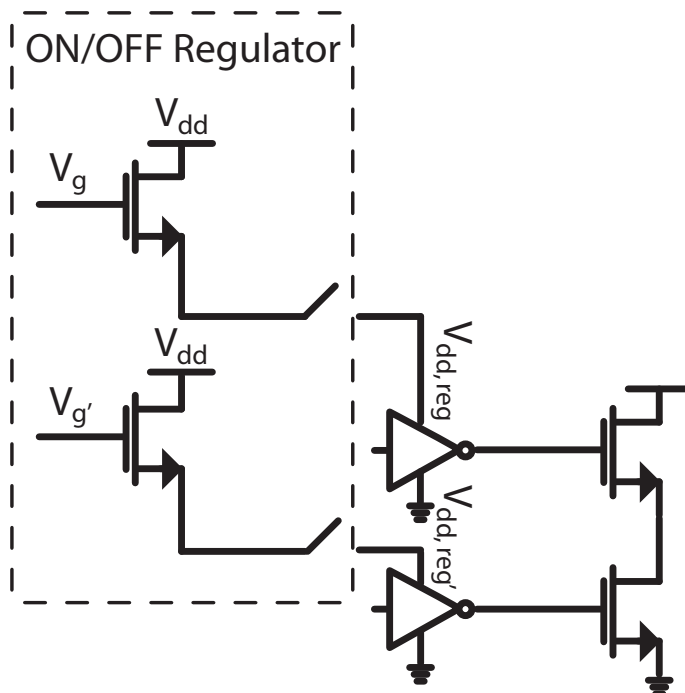


Figure. 3. 11. On/off regulator desired for voltage mode driver designs.

On/off behavior in the VM driver can be achieved through turning off the supply (effectively the regulator that supplies the VM driver). A low dropout regulator (LDO) consisting of a high gain differential amplifier (to set  $V_g$ ) and active NMOS can be used as the active supply for this circuit (Fig. 66 (a)). In this scenario, start-up can be achieved through either charging the various capacitances in the LDO, or directly disconnecting the LDO from the circuit. This option provides various concerns, and ultimately either adds capacitance that needs time to be charged or causes an instantaneous voltage drop on the supply domain as seen in Fig. 3.12 (b).

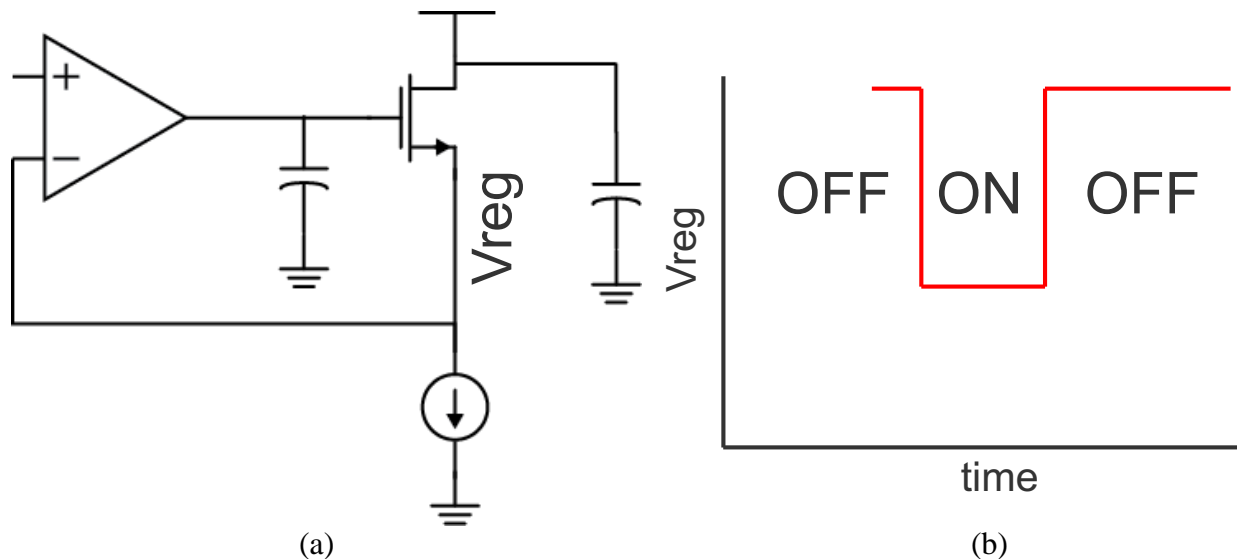


Figure. 3. 12. (a) NMOS LDO for TX supply regulation. (b) On/off instantaneous voltage drop.

Various ways around this issue could be considered. Gating the regulator from the TX (and operating the TX at 50% duty-cycle) could allow for measurable voltage drop while on but ultimately places limits on how we can operate our burst-mode link. A switched-capacitor DC-DC converter [40] (instead of the amplifier loop) could also be used to control the gate voltage on the regulator. A switched-capacitor DC-DC converter would require some additional oscillators on chip, adding to the standby power budget, but could be an attractive solution to burst-mode regulator designs. Since the VM driver is quite involved in its design for optimal termination with equalization [37] and requires complicated switched-capacitor regulator circuitry for on/off operation, the switched-capacitor driver explained in the following sections is chosen for our design.

### 3.2.3 Switched-capacitor TX driver

Switched-capacitor driver was first used as a single-ended short reach serial link by Poulton in [41], achieving 20Gb/s. The switched-capacitor (or charge pump) architecture is desirable since it draws the same current from the supply regardless of data polarity compared to other single-ended architectures which suffer from noise generated from data dependent supply current variation.

Requiring one switched-capacitor driver to pre-charge and another to be operational, this architecture seems to be perfectly suitable for DDR operation. A DDR switched-capacitor driver with differential termination is shown in Fig. 3.13.

## Switched-capacitor TX Driver

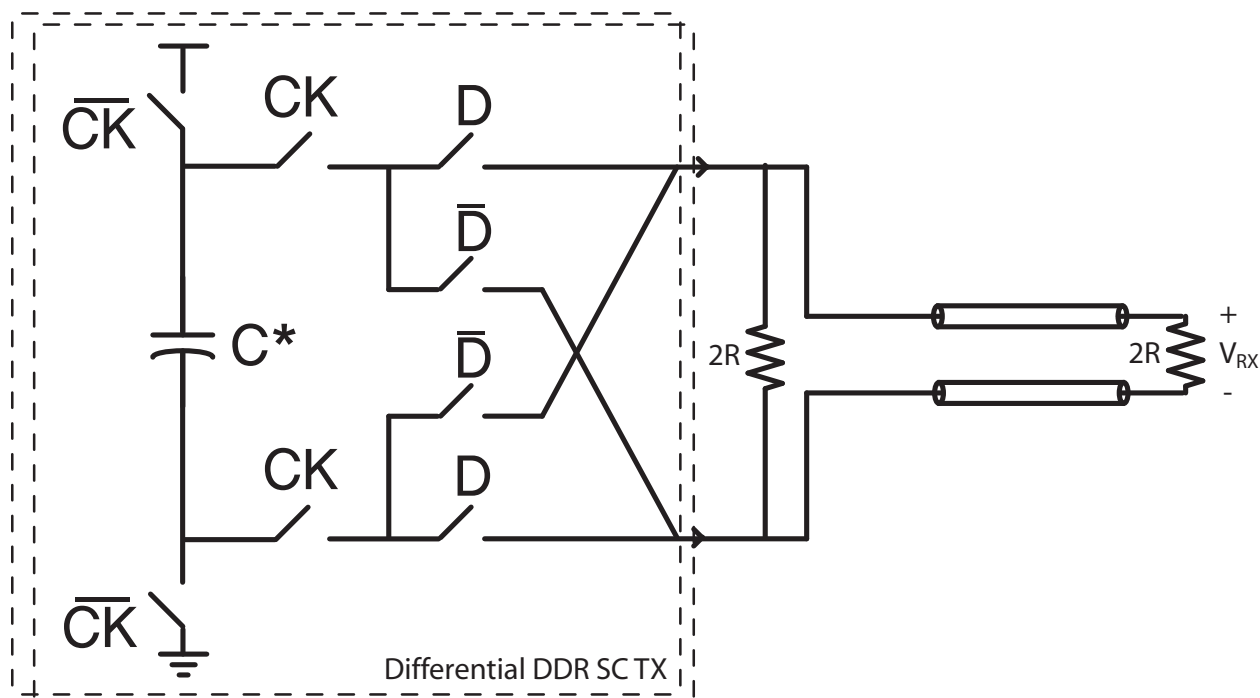


Figure. 3. 13. Switched-capacitor driver with differential termination at RX.

Two differential SC TX drivers (one for pre-charge and the other to send data) driving a differentially terminated transmission line are shown in Fig. 14.  $C^*$  is charged when  $CK$  is low, and discharged through the  $D$  or  $\bar{D}$  paths (depending on data polarity) when  $CK$  is high.

The SC driver becomes an attractive option once we consider a scenario where the TX has to operate in standby or turn on and off quickly. From Fig. 14, we can see that simply disconnecting  $CK$  can stop the driver from switching. Instead of waiting for large current sources to charge to achieve the “on” state, simply disconnecting and connecting the  $CK$  to the driver could allow for a much shorter (faster) latency than a VM or CML driver. Hence, the SC driver allows us on-mode to simply be when the  $CK$  is oscillating and off-mode to be when the  $CK$  is not. The following circuit model is used to calculate the voltage over time (and maximum voltage) of the SC driver.

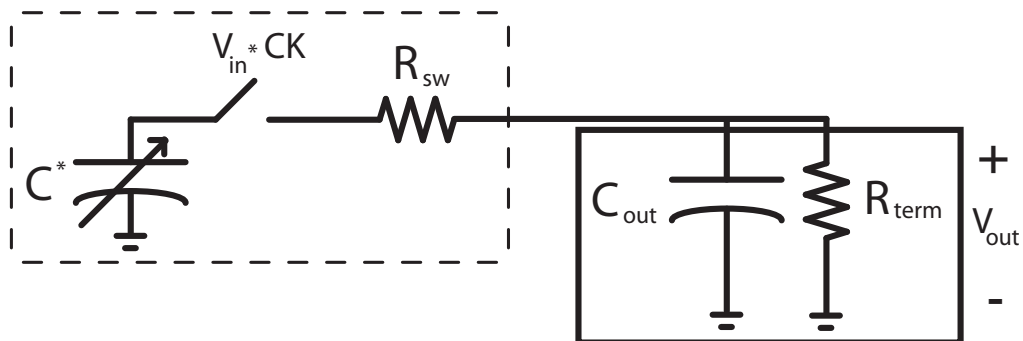


Figure. 3. 14. SC driver circuit model.

The switched-capacitor driver can be analyzed through the simplified, single-ended circuit model from Fig. 3.14 as follows similarly to [27], where the flying capacitor ( $C^*$ ) is pre-charged to an initial voltage ( $v_{in,0}$ ) in series with the switch with resistance ( $R_{sw}$ ) and the transmitter output impedance is modeled as a shunt capacitance ( $C_{out}$ ) in parallel with  $R_{term}$ . For this analysis,  $R_{sw}$  accounts for the total (sum) of the two resistances from the series switches for CK and D.

We can write out the time domain equation from simple KCL analysis as

$$\frac{dV_{in}}{dt} C^* + \frac{dV_{out}}{dt} C_{out} + \frac{V_{out}}{R_{term}} = 0 \quad (3.7)$$

Taking the Laplace transform with initial condition  $V_{in}(0) = v_{in,0}$ ,  $V_{out}(0) = 0$

$$(sV_{in}(s) - v_{in,0})C^* = -\left(sV_{out}(s)C_{out} + \frac{V_{out}(s)}{R_{term}}\right) \quad (3.8)$$

where

$$V_{in}(s) = V_{out}(s) \left( sC_{out} + \frac{1}{R_{term}} \right) R_{sw} + V_{out}(s) \quad (3.9)$$

Substituting (3.9) into (3.8), we get

$$sV_{out}(s) \left( sC_{out}R_{sw} + \frac{R_{sw}}{R_{term}} + 1 \right) C^* - v_{in,0}C^* = -V_{out}(s) \left( sC_{out} + \frac{1}{R_{term}} \right) \quad (3.10)$$

$$V_{out}(s) \left( s^2 C_{out} R_{sw} C^* + s \left( \frac{R_{sw}}{R_{term}} C^* + C^* + C_{out} \right) + \frac{1}{R_{term}} \right) = v_{in,0} C^* \quad (3.11)$$

Dividing by  $R_{sw}C_{out}C^*$  on both sides,

$$V_{out}(s) \left( s^2 + s \left( \frac{1}{R_{term}C_{out}} + \frac{1}{R_{sw}C_{out}} + \frac{1}{R_{sw}C^*} \right) + \frac{1}{R_{sw}C^*} \frac{1}{R_{term}C_{out}} \right) = \frac{1}{R_{sw}C_{out}} v_{in,0} \quad (3.12)$$

Rearranging gives us  $V_{out}(s)$  as

$$V_{out}(s) = \frac{\frac{1}{R_{sw}C_{out}} v_{in,0}}{s^2 + \left( \frac{1}{R_{sw}C^*} + \frac{1}{R_{sw}C_{out}} + \frac{1}{R_{term}C_{out}} \right) s + \frac{1}{R_{sw}C^*} \frac{1}{R_{term}C_{out}}} \quad (3.13)$$

We can calculate the system's two poles  $p_{1,2}$

$$= \frac{-\left(\frac{1}{R_{sw}C^*} + \frac{1}{R_{sw}C_{out}} + \frac{1}{R_{term}C_{out}}\right) \pm \sqrt{\left(\frac{1}{R_{sw}C^*} + \frac{1}{R_{sw}C_{out}} + \frac{1}{R_{term}C_{out}}\right)^2 - 4\frac{1}{R_{sw}C^*R_{term}C_{out}}}}{2} \quad (3.14)$$

Simplifying, we get

$$p_{1,2} = \frac{\frac{1}{R_{sw}C^*} + \frac{1}{R_{sw}C_{out}} + \frac{1}{R_{term}C_{out}}}{2} \left( 1 \pm \sqrt{\frac{4\frac{1}{R_{sw}C^*}\frac{1}{R_{term}C_{out}}}{\left(\frac{1}{R_{sw}C^*} + \frac{1}{R_{sw}C_{out}} + \frac{1}{R_{term}C_{out}}\right)^2}} \right) \quad (3.15)$$

The two-pole system (3.13) has the following time domain voltage characteristics. This can be characterized through two exponentials that govern the dynamic, time domain behavior of this circuit (Fig. 3.15).

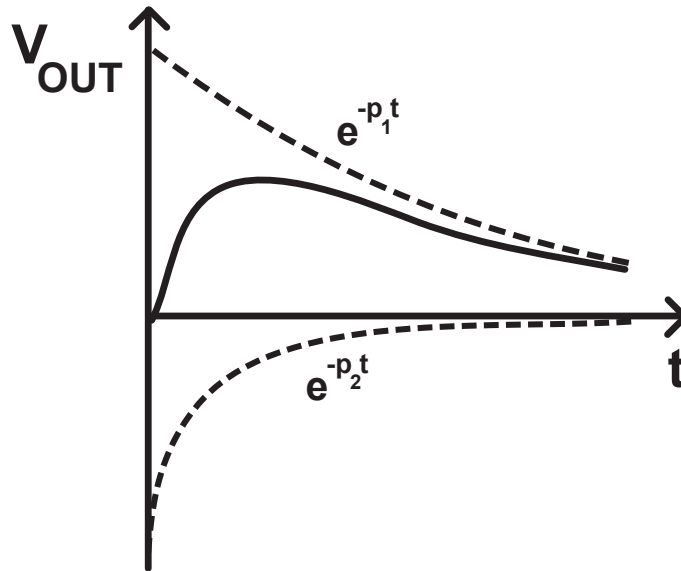


Figure. 3. 15. Switched-capacitor driver voltage over time.

Assuming  $p_2 > p_1$ , we end up with  $V_{out}(t)$  governed by the following equation.

$$V_{out}(t) = \frac{1}{R_{sw}C_{out}} v_{in,0} (e^{-p_1 t} - e^{-p_2 t}) \quad (3.16)$$

We can therefore calculate the time when  $t$  is max, with  $\frac{dV_{out}(t)}{dt} = 0$

$$\frac{dV_{out}(t)}{dt} = \frac{1}{R_{sw}C_{out}} v_{in,0} (p_2 e^{-p_2 t} - p_1 e^{-p_1 t}) = 0 \quad (3.17)$$

Simplifying and taking the natural log of both sides we get



$$\ln\left(\frac{p_2}{p_1}\right) - p_2 t = -p_1 t \quad (3.18)$$

$$t_{max} = \frac{1}{p_2 - p_1} \ln\left(\frac{p_2}{p_1}\right), \quad (3.19)$$

and the maximum voltage (or swing) we should expect to achieve is found as

$$V_{out}(t_{max}) = \frac{1}{R_{sw}C_{out}} v_{in,0} \left( e^{-p_1 \frac{1}{p_2 - p_1} \ln\left(\frac{p_2}{p_1}\right)} - e^{-p_2 \frac{1}{p_2 - p_1} \ln\left(\frac{p_2}{p_1}\right)} \right) \quad (3.20)$$

$$= \frac{1}{R_{sw}C_{out}} v_{in,0} \left( \left(\frac{p_2}{p_1}\right)^{\frac{-p_1}{p_2 - p_1}} - \left(\frac{p_2}{p_1}\right)^{\frac{-p_2}{p_2 - p_1}} \right) \quad (3.21)$$

$$V_{out}(t_{max}) = \frac{1}{R_{sw}C_{out}} v_{in,0} \left(\frac{p_2}{p_1}\right)^{\frac{-p_2}{p_2 - p_1}}. \quad (3.22)$$

This analysis is only an approximation, since there are non-idealities from non-trivial parasitics involved in the system including  $C^*$ 's top and bottom plate capacitances, and parasitic capacitances of the two series switches in our topology from Fig. 3.13.

### 3.3 Switched-capacitor feed-forward equalization

Equalization in the transmitter can often lead to more relaxed constraints on the receiver, due to the increased cursor comparatively to the pre-cursors and post-cursors. Furthermore, higher-order equalization can be implemented at the transmitter, since it is straightforward to add digital delays. Although we have mentioned earlier in Chapter 2 that equalization is preferred in the receiver (due to analog delays having to drive small  $g_m$  stages), some TX FFE for pre-emphasis is often desirable depending on overall link constraints. Particularly, [6] uses a 3-tap TX FFE as well as a 2-tap RX FFE for pre-cursor cancellation. Hence, it is not uncommon to see links that budget for TX FFE as well as RX FFE.

Ultimately, system level, architecture level, channel, and technology constraints all contribute to where (and how much) equalization is needed. While a decision feedback equalizer is not achievable on the transmit side, feed-forward equalization can be since it is a linear option. In particular, we discuss various methods to vary the coefficients for a switched-capacitor FFE.

#### 3.3.1 Switched-capacitor feed-forward equalizer

Just as delay, gain, and tap control strength can be achieved in various ways for an RX FFE (and any equalizer needing coefficient control), a switched-capacitor FFE has various

architectural choices that can affect the overall performance of the equalizer. Fig. 3.16 shows the switched-capacitor circuit model with 3-tap feed forward equalization.

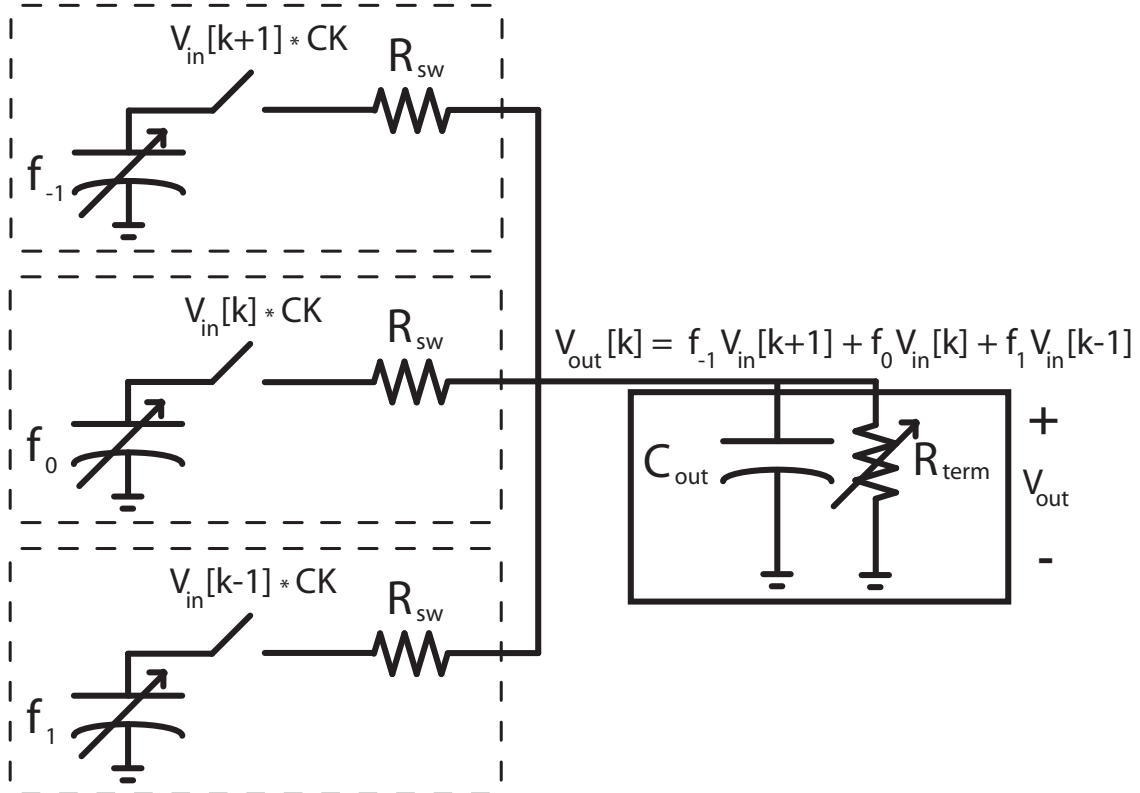


Figure. 3. 16. Switched-capacitor circuit model with 3-tap FFE.

$V_{out}[k]$  is represented as

$$V_{out}[k] = f_{-1}V_{in}[k + 1] + f_0V_{in}[k] + f_1V_{in}[k - 1] \quad (3.23)$$

for the 3-tap FFE with 1 pre-cursor and 1 post-cursor in this example, but can be extended for arbitrary number of taps. To implement the FFE algorithm, the architecture should be able to effectively delay data ( $V_{in}[k - i]$ ) and control tap strength ( $f_i$ ).

Again, we want to avoid CML latches or any circuit elements that require a constant current source. This further motivates the DDR design, since latches need to only operate at half-rate. Thus, achieving  $V_{in}[k - i]$ 's are straightforward as long as technology permits CMOS latch design at the desired operating frequency. CMOS latches allow for fully dynamic circuits to achieve the  $V_{in}[k - i]$ 's, but they are limited in operating frequency. Technology node limitations can drive this architectural choice, as it is often the case the high-speed designs require, especially at older nodes (65nm or above), dynamic latches for this purpose [9].

On the other hand, there are several possible ways we can vary  $f_i$ . At first glance, it may seem nice to control  $f_i$  by controlling the duty cycle of the switch. However, completely charging  $C_i$  is important in order to not introduce any data-dependent ISI into the system, which cannot be equalized. The switches should therefore be sized to almost fully settle (at least  $3\tau$ 's ( $R_{sw}C^*$ )) within half a clock period.

### 3.3.2 Variable capacitance using capacitive DAC

Adding a capacitor in series with the capacitor adds an additional switch both the recharge and charge phase. We can consider varying the  $C_i^*$ 's by making the capacitors into a multi-bit DACs as in Fig. 17 below.

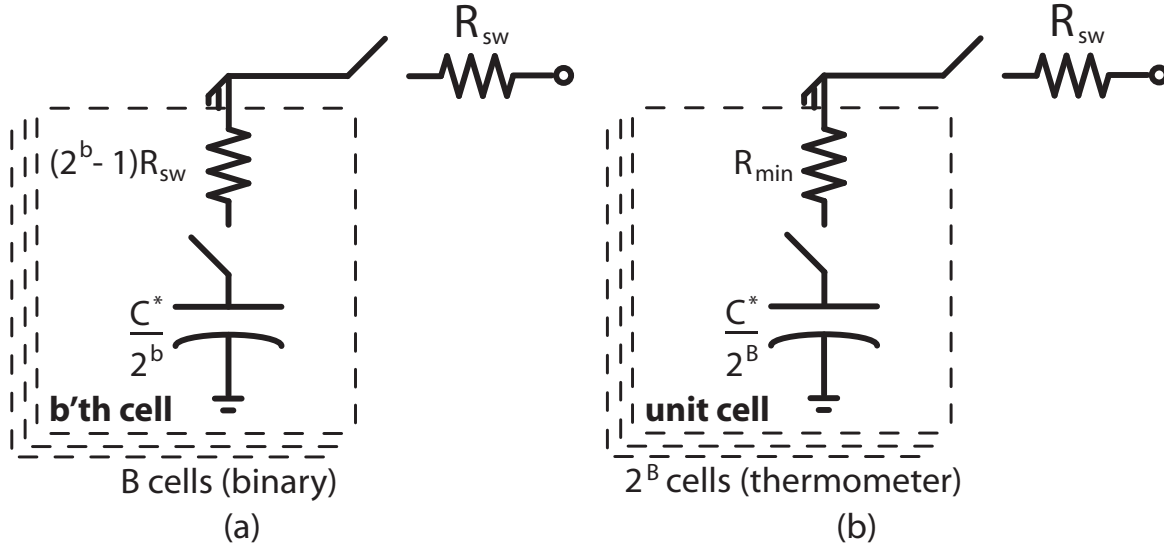


Figure. 3. 17. (a)  $B$ -bit binary capacitor DAC  $B$  binary weighted cells (b)  $B$ -bit thermometer capacitor DAC with  $2^B$  unit cells.

Fig. 3.17 shows the ideal model of  $B$ -bit binary and thermometer CAP DACs. The binary DAC (in Fig. 3.17(a)) shows the necessary switch sizing for varying capacitance values for equivalent settling time. Since  $\frac{C^*}{2^b}$  would require  $\frac{1}{2^b}$  switch size for  $\tau = R_{sw}C^*$  (switch resistance is inversely proportional to switch size), two switches in series with a capacitor  $\frac{C^*}{2^b}$  would allow for the DAC switches to be sized down by at most  $\frac{1}{2^b - 1}$ .

$$\frac{C^*}{2^b}(R_{sw} + (2^b - 1)R_{sw}) = R_{sw}C^* \quad (3.24)$$

With increasing  $b$ , switch size eventually becomes limited by minimum transistor sizing constraints.  $R_{min}$  is the minimum switch resistance based on the minimum of the  $b^{th}$  switch resistance and the resistance of a minimum size switch as

$$R_{min} = \min(R_{minsize}, (2^b - 1)R_{sw}) \quad (3.25)$$

A similar calculation shows that the  $B$ -bit thermometer CAP DAC (Fig. 3.17(b)) with  $2^B$  cells each with switches size to  $R_{min}$  has a similar constraint to (3.25).

Although these aforementioned solutions present sizing methods for settling equivalence to the single flying capacitor driver, using a capacitive DAC becomes less attractive once we consider that parasitic loading of the switches (and passive wiring) adds additional loading and

that driving multiple switches requires a  $v_{th}$  drop per series switch. Increased voltage levels or other additional circuitry would be needed to overcome the  $v_{th}$  drop.

We would like to take the DAC out of the signal path, as many equalizer aim to do; however, several tap control options involving supply regulators/DACs consume both area and quiescent current are unsuitable for our design target.

### 3.3.3 Tap control with clock gating

Instead, we have chosen to implement  $f_i$ 's by having multiple unit elements of switched-capacitors and adding the switch in the clock path. This solution takes the additional switch out of the data-path but provides less resolution to the equalizer coefficients. Fig. 3.18 shows the circuit model for our tap control strategy.

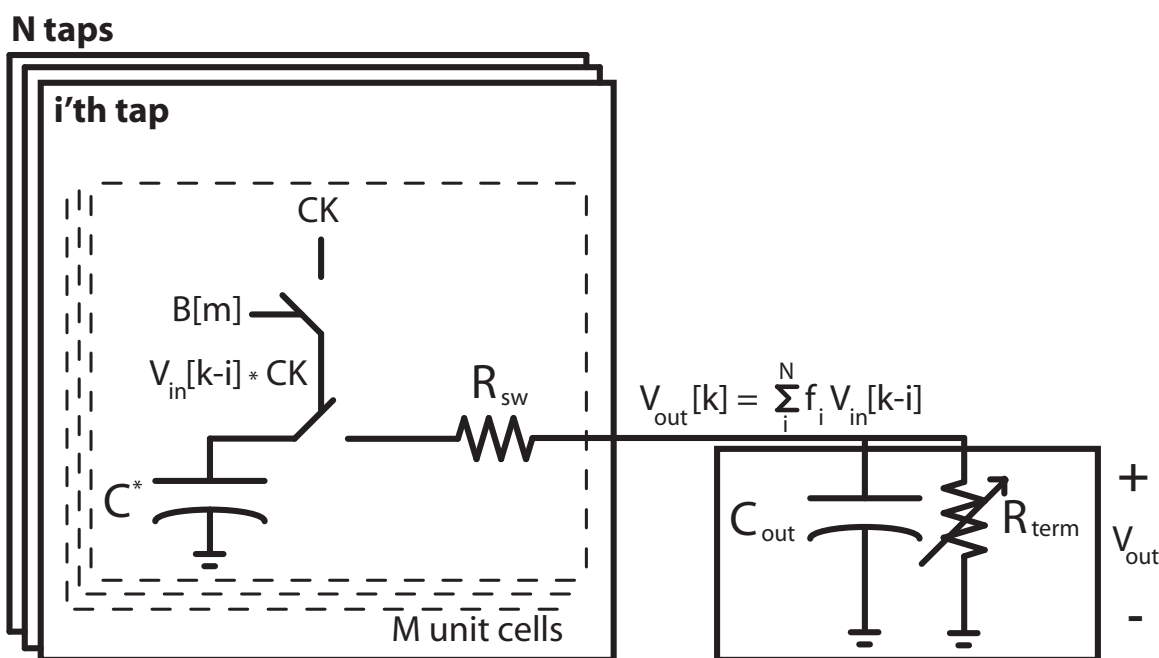


Figure. 3. 18. N-tap SC FFE circuit model with tap strength control.

The N-tap SC FFE circuit model shows N taps of switched cap drivers that are made up of M unit cells ( $B[m] = 1 \dots M$  to control tap strength ( $f_i$ )) to achieve the FFE function ( $V_{out}[k] = \sum_{i=1}^N f_i V_{in}[k-i]$ ). We use a fixed fly capacitance  $C^*$  and short the outputs of the unit cells (and taps) to achieve the summation. Adjustable  $R_{term}$  allows the configurable driver to have matching termination to the line impedance.

The switch on the clock simply adds delay that should be consistent for all circuits clocked by the high-speed, half-rate clock in a DDR system. It is important to distribute a differential clock, since 50% duty cycle is important for matching odd and even paths. Additionally, timing of pre-charge and charge should not overlap, as it can induce data-dependent ISI (Fig. 3.19).

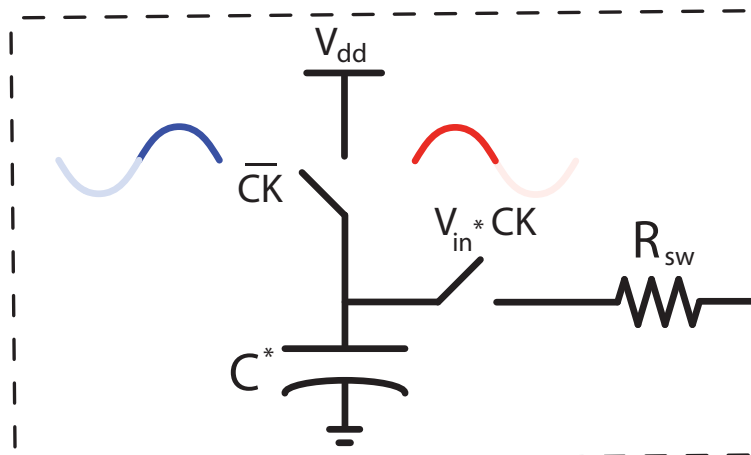


Figure. 3. 19. Pre-charge and charge of SC driver.

A sinusoidal clock is beneficial, as the non-infinite slope of the sinusoid essentially creates non-overlapping to aid in this condition. Of course, less swing (or similarly duty cycle) implies increased switch sizing in order to meet settling time constraints.

Now that we have described the system and architectural constraints for a burst-mode transmitter, the next chapter presents the design of a 2-tap SC FFE TX realized in a 28nm TSMC process.

## Chapter 4

### A 2-tap Switched-Capacitor FFE Transmitter Achieving 1-20 Gb/s at 0.72-0.62 pJ/bit

As chip-to-chip systems realize more large-scale, massive aggregated I/O (Tb/s), they demand rapid-on/off I/O to achieve maximum energy efficiency during both high and low link utilization. Furthermore, fast startup time or latency becomes more important as we approach constant energy efficiency vs. data-rate (energy proportional) operation. Various energy proportional links have been presented in [39][42][43] to address these needs. However, their architectures do not provide truly dynamic operation, do not operate with close to zero quiescent current, or have longer startup times that limit their efficiencies during low utilization.

High-speed receiver architectures [44][45] have been shown with efficient, continuous operation at high speeds and could be suitable for on/off operation due to their dynamic operation; however, traditional voltage mode or current mode logic solutions for the transmitter either require complicated circuitry for rapid on/off or require additional standby power. This work provides a solution for the transmitter incorporating a low latency, 1-latch serializer architecture and a switched-capacitor driving stage [39][41] that can be operated fully dynamically through resonant clocking with an on/off LC OSC to achieve 1.2ns startup time, 0.1mW standby power and 12.4mW power during operation.

#### 4.1 Transmitter architecture and circuit design

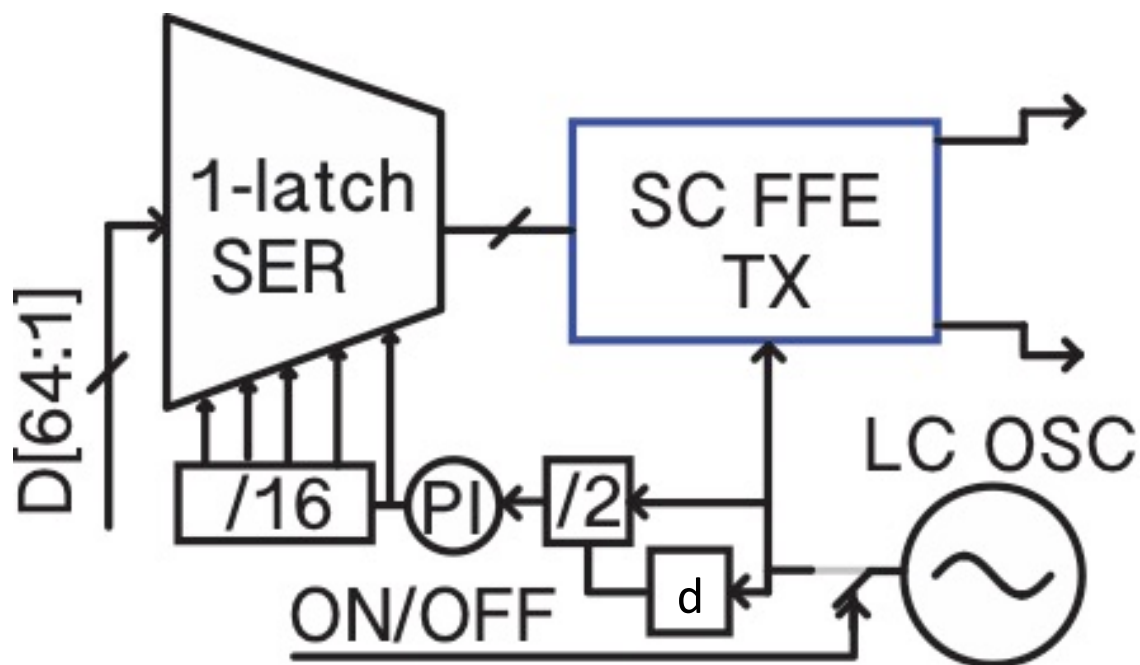


Figure. 4. 1. Transmitter architecture.

The transmitter architecture is chosen to minimize latency and standby power, while maintaining energy efficiency. The transmitter consists of a 64:2 1-latch CMOS MUX based SER, and 2-tap DDR SC FFE that is driven directly by a rapid on/off LC OSC through an adjustable clock divider chain (Fig. 4.1).

### 4.1.1 Switched-capacitor feed forward architecture

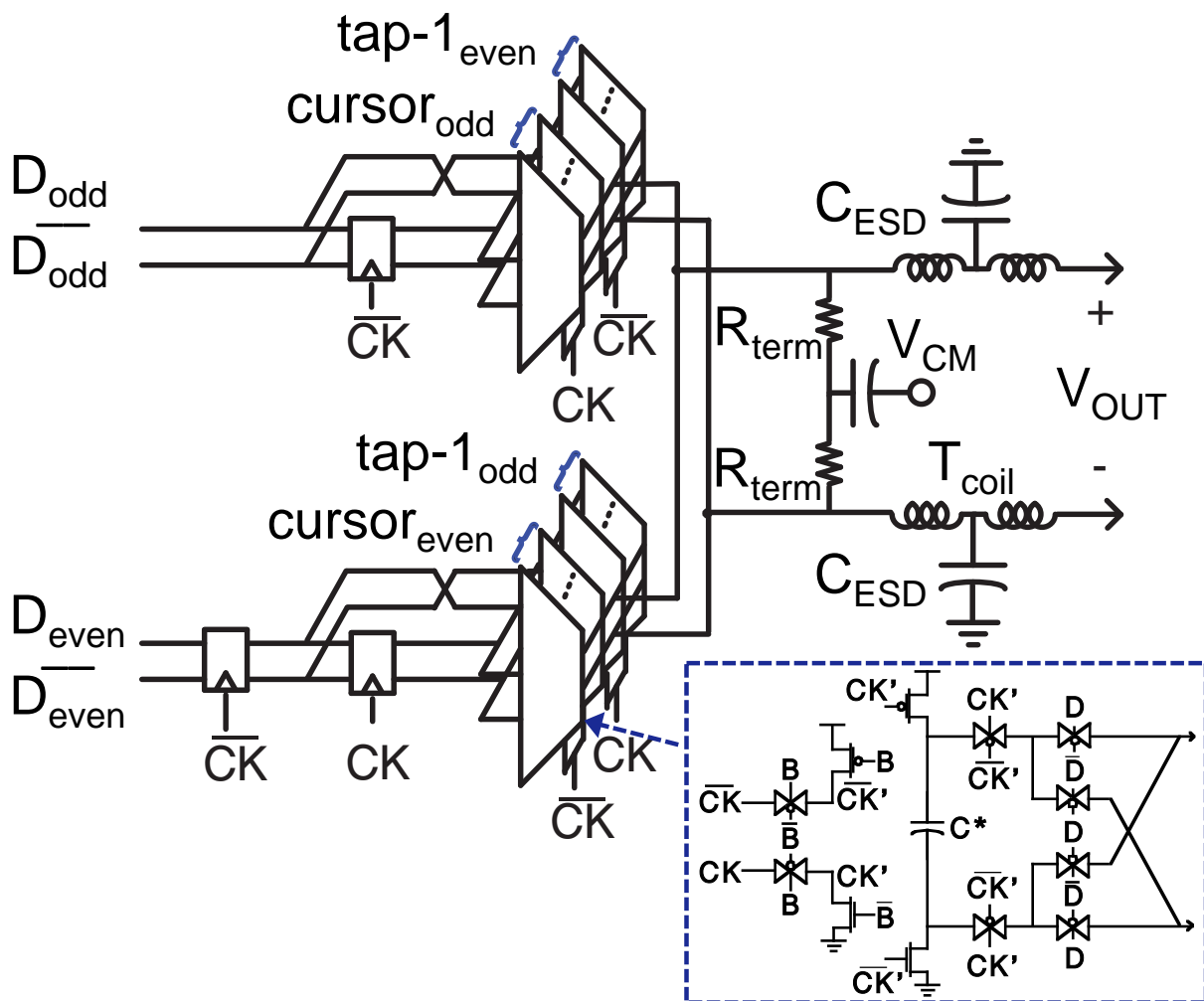


Figure. 4. 2. 2-tap SC FFE TX.

Shown in Fig. 4.2, the SC Feed Forward Equalizer (FFE) transmitter consists of seven differential unit cells of switched-capacitor transmitters each with its own capacitor  $C^*$ , with one of the unit elements (tap-1) designated for canceling pre-cursor intersymbol interference (ISI) and the remaining six unit elements dedicated for the cursor. Tap-1 is fixed to be  $-f_1$  since a high pass characteristic is almost always more beneficial for the types of channels we are trying to equalize.

Operating in dual data-rate (DDR), the switched-capacitor circuits charge  $C^*$  during the first phase of  $CK'$  and discharge  $C^*$  through the bottom or top direction depending on the value of the

data (D) during the second phase of CK'. The operation of the SC architecture (consisting of two phases for DDR operation) is demonstrated in Fig. 4.3 below.

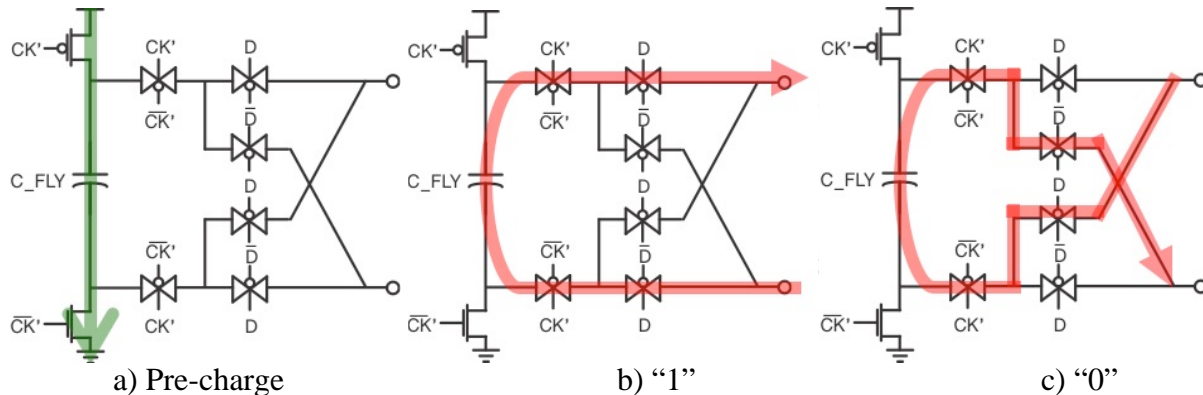


Figure 4.3. Switched-capacitor TX cell operation. (a) pre-charge (b) "1" (c) "0".

In phase 1 (Fig. 4.3(a)), C\_FLY is fully charged. Then in phase 2, C\_FLY is discharged from V+ to V- depending if D is high or low (Fig. 4(b or c)). Full transmission gates allow for more consistent on resistance over voltage across the switch. Main tap strength is controlled through B, which determines how many unit SC TX's are active, allowing for variable equalization strengths by varying the overall swing. The SC TX's design gives us a way to toggle operation by having the high-speed clock control the switches instead of having complicated regulators and/or DACs. Due to this fact, the SC TX's improved standby power and startup time compared to its more power efficient VM TX counterpart makes it the natural choice when optimizing for latency and energy efficiency rather than instantaneous power. Finally, the output is terminated and DC coupled to set  $V_{CM}$ , and drives an ESD capacitor and  $T_{coil}$ . Figure. 4.4 shows the layout of the 2-tap DDR SC FFE.

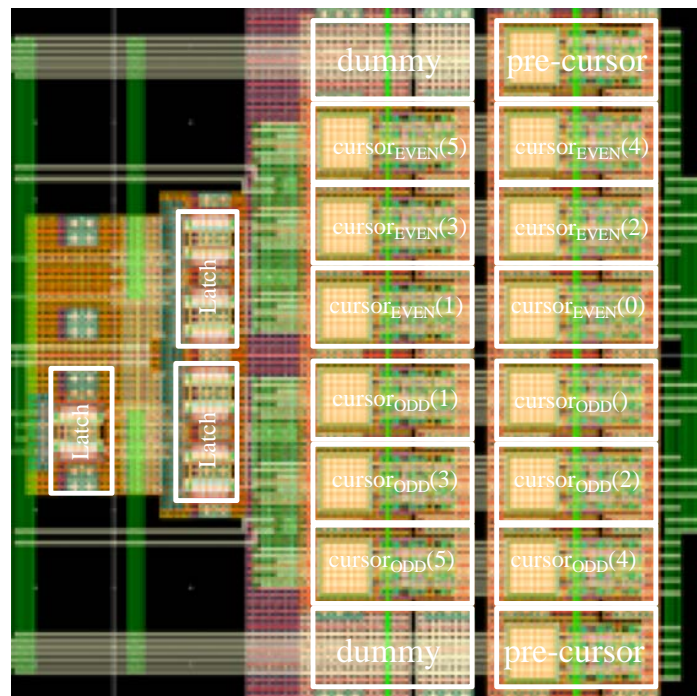




Figure. 4. 4. 2-tap DDR SC FFE layout.

Layout symmetry across the x-axis is conserved for odd and even paths. The even switched-capacitor FFE is above, while the odd is below. The differential 10GHz clock is distributed from the left-center and is shielded when distributed with ground lines between individual signals. Latches on the left are shown which provide the UI-spaced delay for the pre-cursor and post-cursor.

## 4.2 Serializer design challenges

Serialization serves an important function in every communication link to serialize digital bits to be transmitted at the high-speed line data-rate. Fig. 4.5 depicts a 64:1 line rate serializer and a 64:2 DDR serializer.

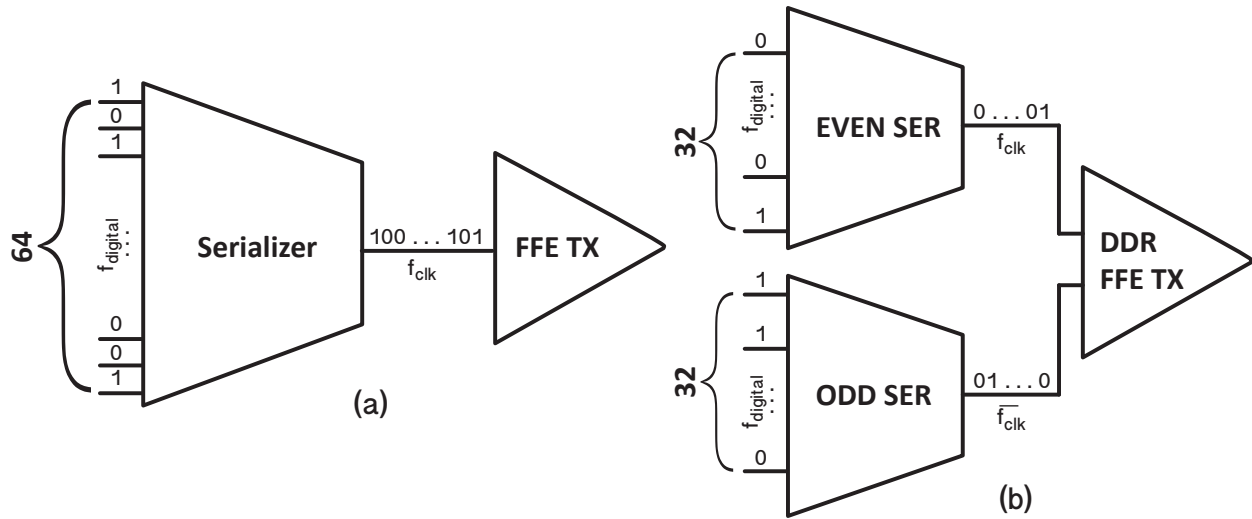


Figure. 4. 5. (a) Line rate serializer (b) DDR serializer.

A line rate transmitter requires serializes the digital bits to the line data (Fig. 4.5(a)). The line rate serializer provides data at the data-rate for a data-rate TX. This requires high-speed serialization, which is especially power consuming for the final stage. Serialization is typically achieved through CMOS shift registers for low frequency bits and MUX's for higher frequency stages [46]. Furthermore, CML based MUXs are typically the only viable serialization options at high speeds and have the same quiescent current issue as in the TX drivers [47].

Alternatively, for a DDR transmitter, serialization only up to half-rate is required. Since the serializer does not operate at the data-rate, it can be designed without resorting to the demanding CML MUX SERs as in [38][48]. In our design (Fig. 4.5(b)), we incorporate two 32:1 serializers (one odd and one even) to serialize each 32 bits to half the data-rate. The odd and even serializer outputs are then sent to the odd and even DDR SC data paths and combined at the output (providing the final stage of serialization to the full, nominal data-rate of 20Gb/s).

The MUX based serializer is traditionally made up of a structure similar to the 2-latch MUX shown in Fig. 4.6(a). Since this transmitter is intended to operate over short, multi-package environments with limiting latency constraints, this design incorporates a fully MUX based serializer composed of 1-latch CMOS MUXs (Fig. 4.6(b, d)) to save latency in the SER.

## 4.2.1 MUX serializer timing constraints

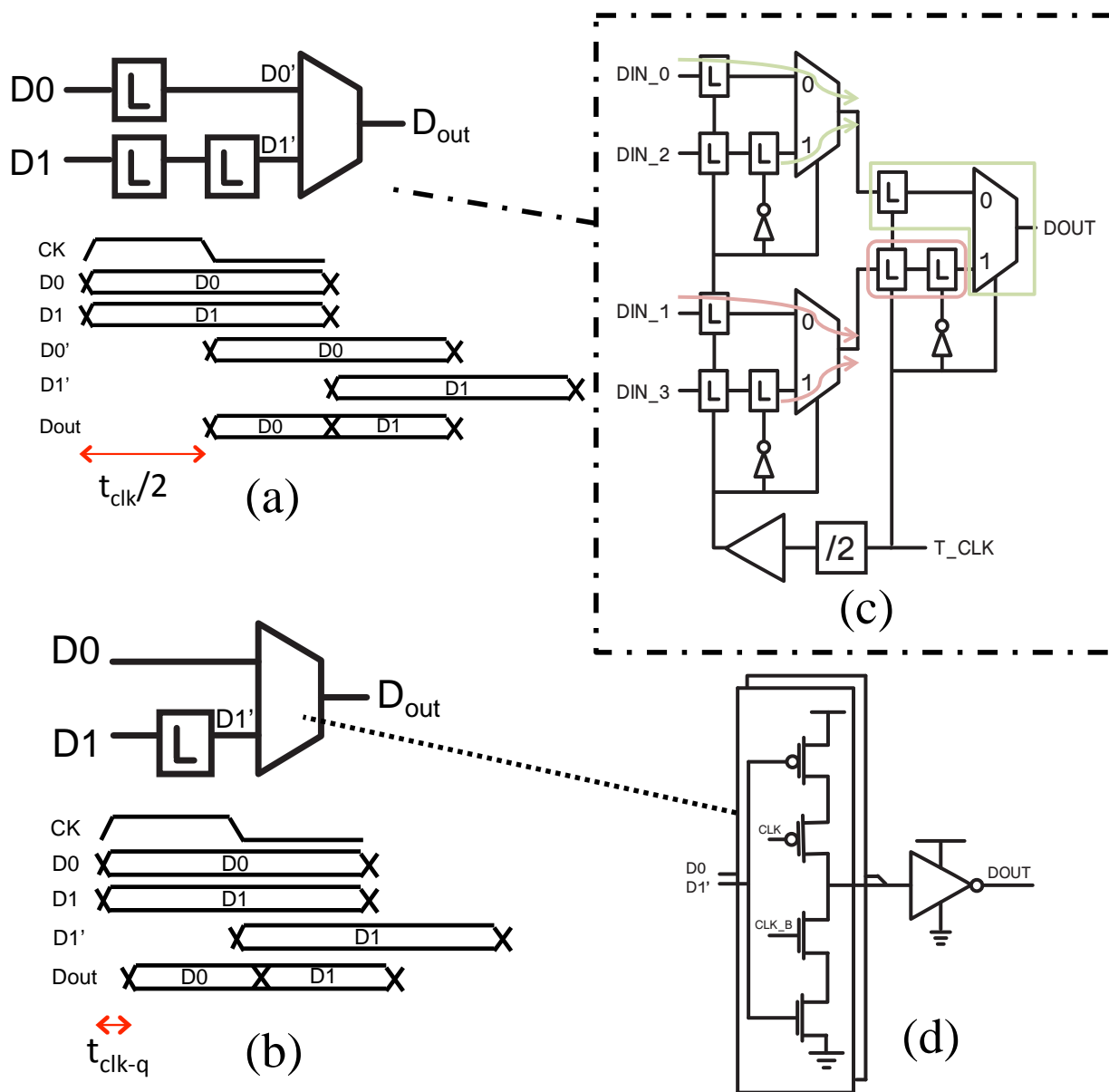


Figure 4. 6. (a) 2-latch MUX & timing (b) 1-latch MUX & timing (c) Example 2-latch MUX, two-stage serializer timing paths (d) CMOS MUX.

For the 2-latch architecture, it is fairly straightforward to ensure proper functionality by making sure setup and hold time constraints are met. These can be analyzed through looking at the timing paths for a two-stage example serializer as seen in Fig. 4.6(c). By combining the setup and hold time constraints for the various timing paths, proper operation of the serializer can be achieved as long as the buffer delay is bounded by (4.1), where  $\alpha$  accounts for the clock skew, which is typically around 10% mainly due to wire length variations, capacitive coupling, and varying capacitance on clock inputs.

$$\begin{aligned}
& \frac{T_{CLK} - \max(t_{clkb-q,MUX}, t_{clk-q,MUX}) - \max(t_{su}, t_{su})}{1 - \alpha} - t_{DIV2} \\
& > t_{BUF} > \\
& \frac{\max(t_{hold}, t_{hold}) - \min(t_{clkb-q,MUX}, t_{clk-q,MUX})}{1 - \alpha} - t_{DIV2}
\end{aligned} \tag{4.1}$$

Equation (4.1) shows that the MUX based serializer requires a specific amount of delay on the divider chain to ensure proper functionality. It is important to notice that not only does the 2-latch SER go through 1 flip-flop every stage (Fig. 4.6(a, c)), but it also necessitates per stage buffer latency in the divider chain. Therefore, having 2-latches per MUX adds both a latency and power penalty for the serializer, latency being more important for earlier stages and power being more important for later ones.

Alternatively, the 1-latch MUX architecture (Fig. 4.6(b)) inherently has less delay and does not require such buffering to meet timing constraints. We analyze a 1-latch MUX serializer, which saves half a CPU-clock cycle in latency from the first serialization stage and allows for only one latch to be necessary for each stage of MUX serialization.

$$t_{BUF} < \frac{0.5T_{CLK} - t_{clk-q,MUX} - t_{su}}{1 - \alpha} - t_{DIV2} \tag{4.2}$$

The buffer delay is no longer bounded by the flip-flop's hold time constraint (4.2), since it is sufficient that the input data to the MUX is ready during the latch's transparent phase. The one-latch serializer functions appropriately as long as the amount of delay in the divider chain is equivalent to the delay in the serializer stage that it is clocking.

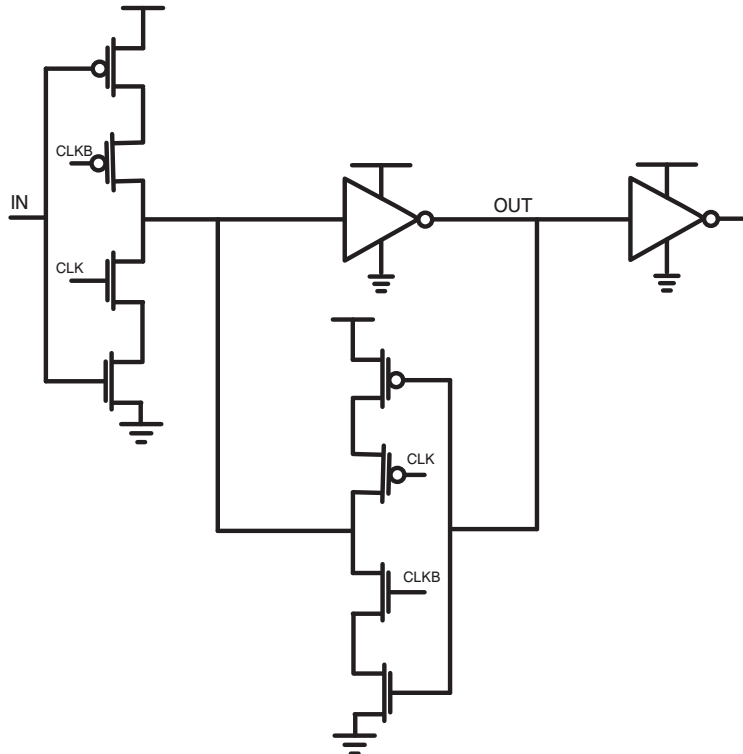


Figure. 4. 7. CMOS latch.

The CMOS latch consists of a clocked inverter followed by an inverter with a regenerative, clocked inverter to hold the intermediate node with positive feedback shown in Fig. 4.7. When CLK is high,  $\overline{IN}$  propagates to the internal node and then subsequently to  $OUT$ . In the next phase when CLK is low, the feedback inverter turns on and holds the state of  $\overline{IN}$  at the internal node. This is different from a simple clocked buffer, as the latch's positive feedback provides memory to store state information of two stable states (0 or 1). The regenerative clocked inverter was sized to be  $\frac{1}{4}$  of the forward path inverter.

## 4.2.2 Divider with buffering for a 1-latch SER

A divider can be designed to support the 1-latch MUX SER that includes buffering to ensure proper timing paths (Fig. 4.8(a)).

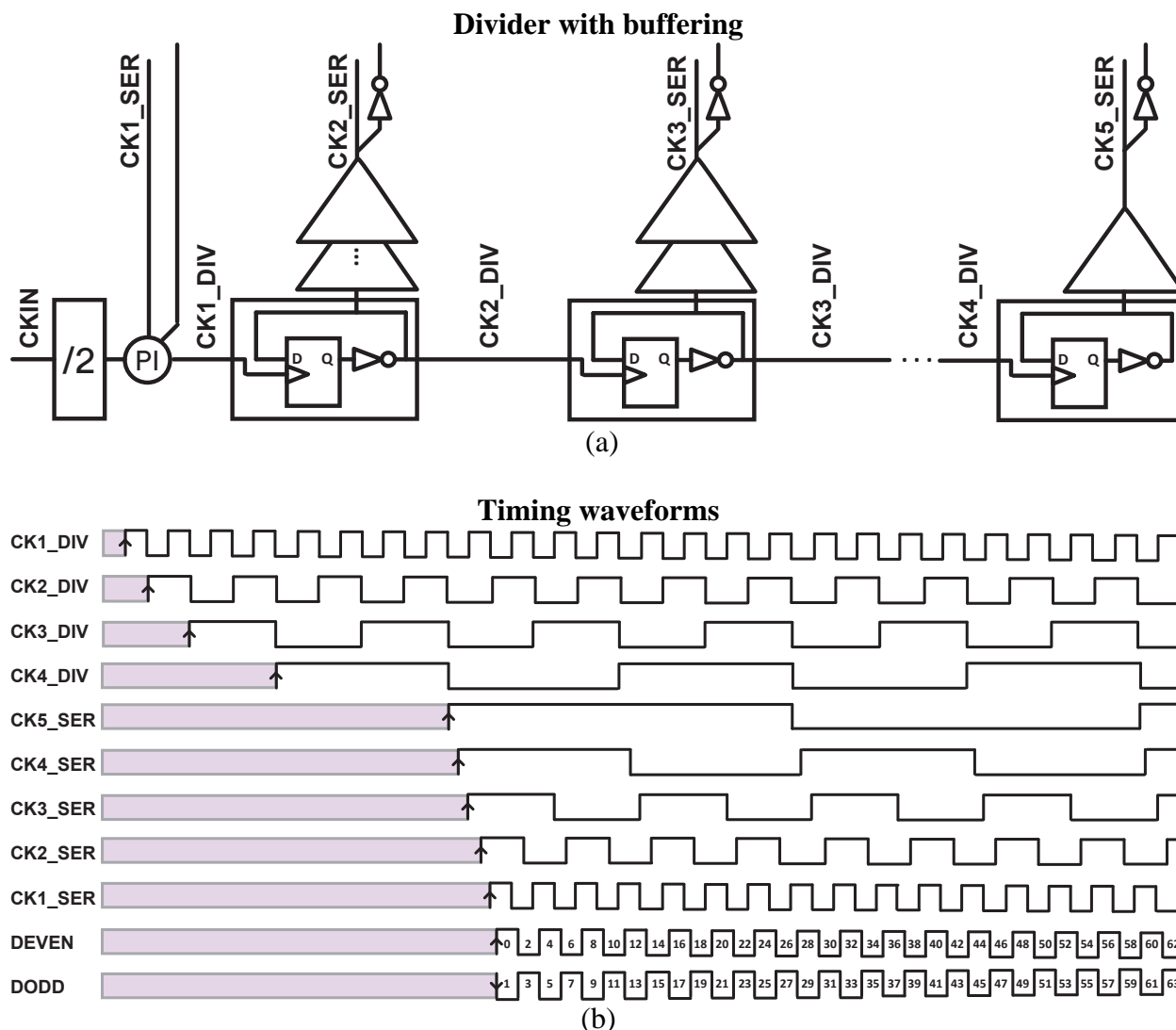


Figure. 4. 8. (a) 1-latch buffered divider chain (b) Timing waveforms.

The divider consists of CMOS, divide-by-2 unit elements with buffering to the serializer. The buffering can be designed to meet timing constraints as long as the delay from the divider chain is equivalent to the delay in the data path. Specifically, the clock should rise (or fall) accounting for  $t_{MUX,d-q}$ , since the following MUX should be clocked when the output is ready. From the timing diagram above in Fig. 4.8(b), buffering is necessary for the latches to be transparent when data is ready. Specifically  $CK1_{SER}$ ,  $CK2_{SER}$ ,  $CK3_{SER}$ , and  $CK4_{SER}$  would require a lot of buffering (delay lines) or similarly power. Furthermore, absolute process and temperature variation of these delay lines get worse as the delay increases. It is important to note that extraneous bits before and after the valid data do not affect functionality since they are eventually combined with the on/off clock in the final driving stage (SC TX), so unwanted bits do not propagate to the transmitter output.

We now consider the latency of the complete 1-latch serializer architecture (Fig. 4.9). Since the clock starts from the on/off signal, latency is determined by the path shown in red, as the divided clocks need to be ready in order for us to serialize the bits in each stage of the serializer. Since we are consecutively adding delay in each stage of the serializer to satisfy this constraint, we are adding up to one first-stage clock cycle (typically  $> 1$  ns) of delay to the minimum time it takes to start up the equalizer.

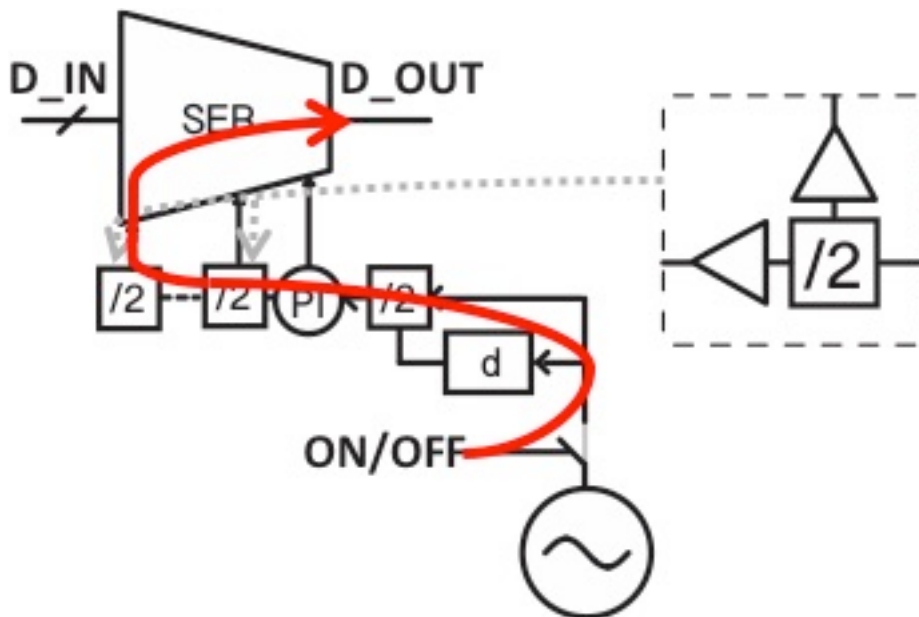


Figure. 4. 9. Latency path for 1-latch SER with buffering.

This path gets longer depending on how many stages are in the serializer but is always dominated by the first stage delay, which is approximately equivalent to the combined delay of the rest of the following serializer stages.

### 4.2.3 1-latch serializer w/ phase adjustable divider

Adjustable phases in the divider chain allow appropriate phases to clock each stage in our 1-latch MUX SER architecture, as it is not obvious to exactly line up the phases of the clock without intentionally adding delay on the clock divider path to match the delay of the serializer.

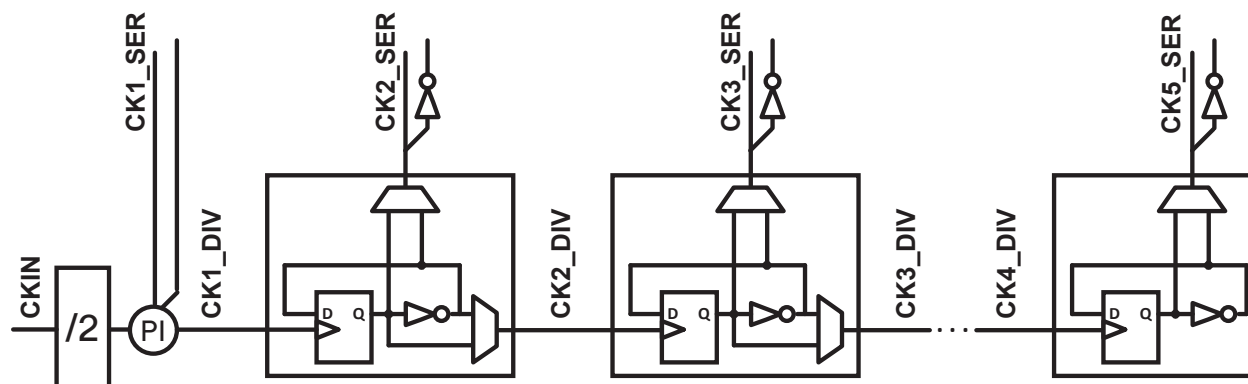


Figure. 4. 10. 1-latch adjustable divider chain.

From Fig. 4.10, each divider consists of a CMOS divide-by-2 and MUXs for divide and serializer paths. Adjustability in the clocks supplied to the serializer path allows for a significant decrease in the required buffering, as opposite polarities can be chosen to give us effective clock delay. Adjusting the divider path clock polarities allows for the slowest clock ( $CK5_{SER}$ ) to be delayed a variable amount, enabling deviations in overall SER latency. Fig. 4.11 shows the complete 64:2 1-latch MUX serializer with an adjustable phase divider.

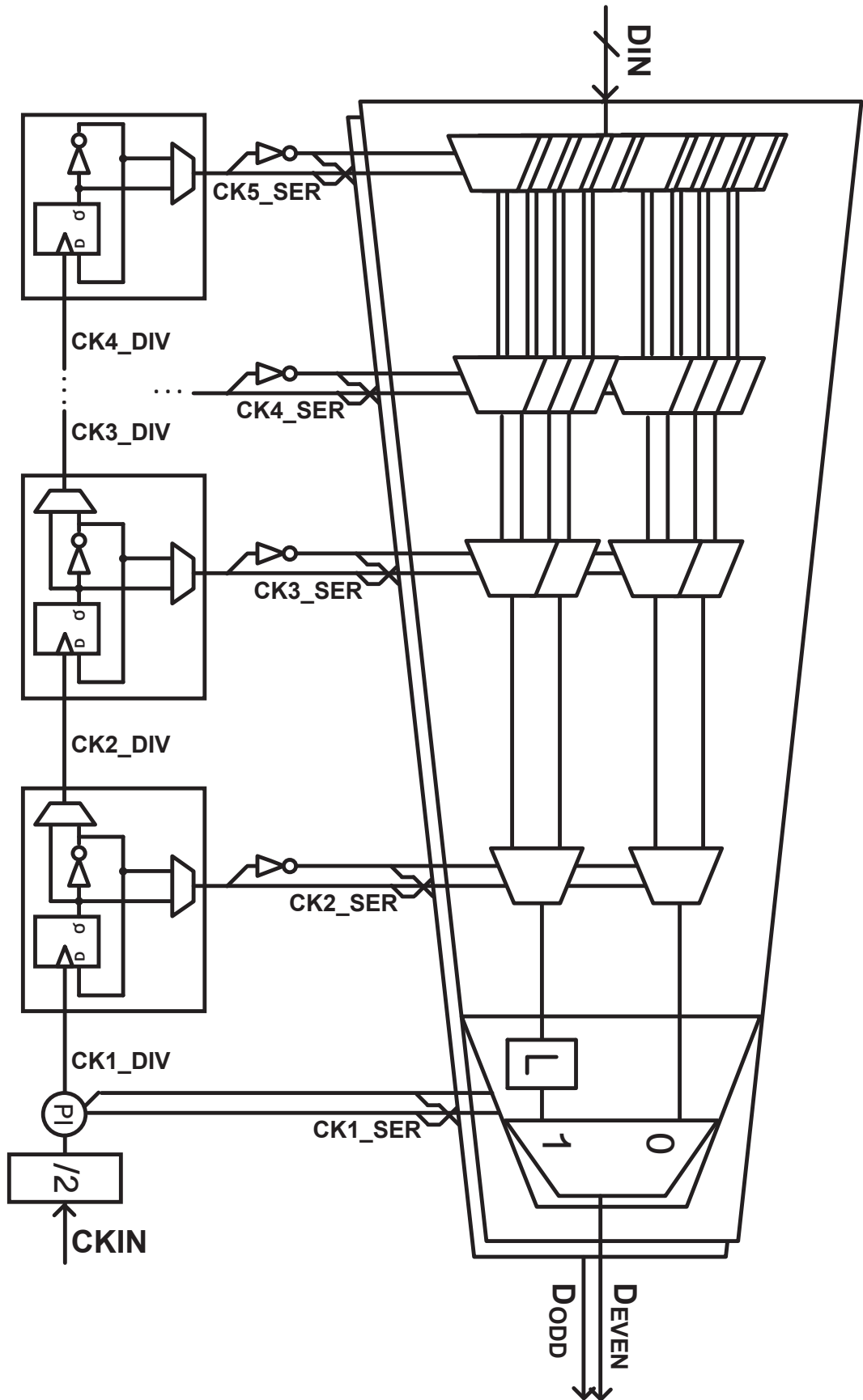
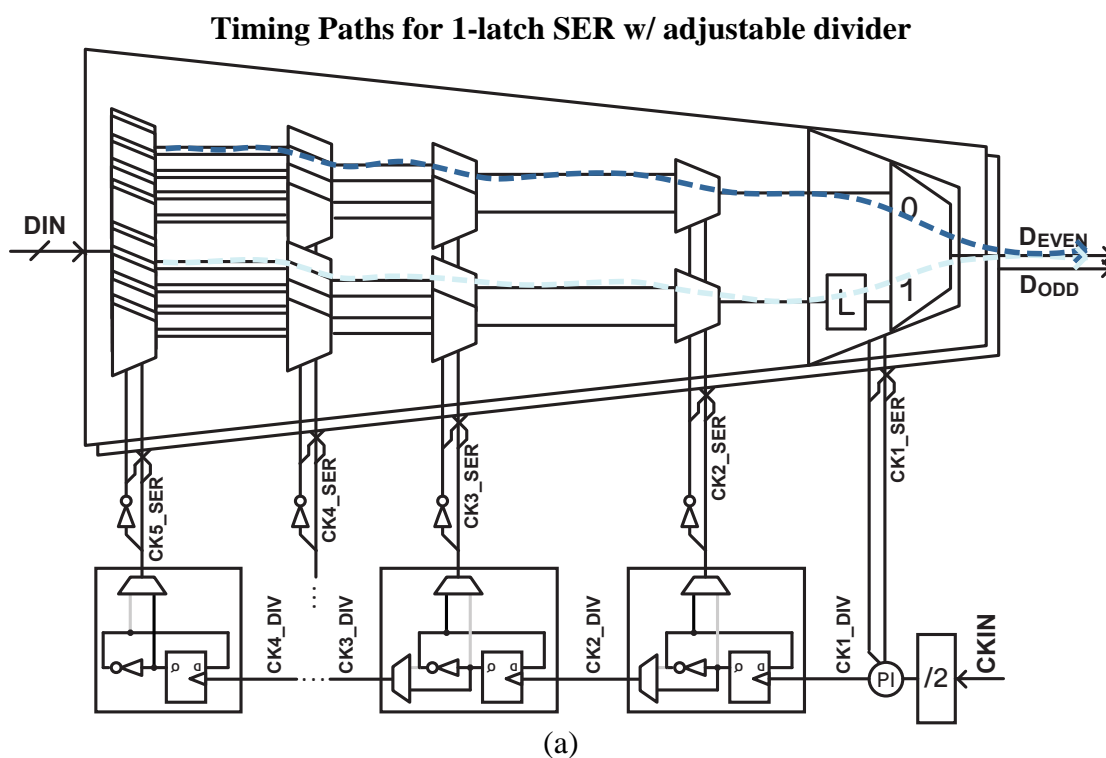


Figure. 4. 11. 1-latch MUX SER architecture.

This serializer consists of five stages of 1-latch MUX's to provide 32:1 serialization for each serializer (odd and even). Even and odd values of  $D_{IN}$  are fed as inputs to the even and odd serializers respectively, and  $D_{EVEN}$  and  $D_{ODD}$  are then provided as inputs to the DDR switched-capacitor driver. Each serializer stage is supplied with separate differential clocks (with the odd SER supplied the opposite polarity for the differential clocks).

It is important to note that the design is dependent on the exact d-q and clk-q delays present in the system in determining which select bits to be chosen for proper functionality. Since latches will propagate inputs due to their transparent (asynchronous) nature, timing is critical in this design. Fig. 4.12 shows the even serializer's timing paths for  $D[0]$  and  $D[2]$  data and associated timing diagrams for an example setting of the adjustable divider chain. The rest of the bits (remaining timing paths) follow similarly.





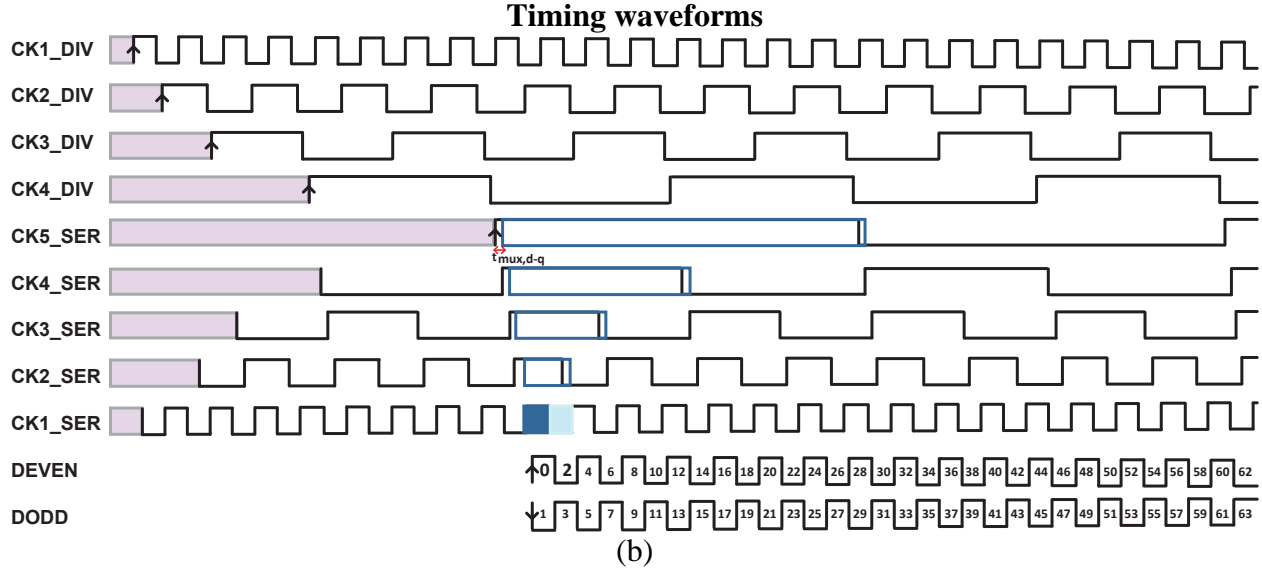


Figure. 4. 12. (a) Adjustable clock to serializer and divider timing paths (a) timing waveforms.

Unused divider paths are in grey, and the divider settings in this example are as follows:

$$\begin{aligned}
 CK1_{DIV} &= CK1_{IN} + \Phi_{PI} \\
 CK1_{SER} &= \overline{CK}_{DIV} \\
 CK2_{DIV} &= \overline{CK1}_{DIV}/2 \\
 CK2_{SER} &= \overline{CK2}_{DIV} \\
 CK3_{DIV} &= \overline{CK2}_{DIV}/2 \\
 CK3_{SER} &= \overline{CK3}_{DIV} \\
 CK4_{DIV} &= \overline{CK3}_{DIV}/2 \\
 CK4_{SER} &= \overline{CK4}_{DIV} \\
 CK5_{SER} &= \overline{CK4}_{DIV}/2
 \end{aligned}
 \tag{4.3}$$

Since each SER clock and divider clock can be chosen with opposite phase, we can appropriately choose divider settings without having large amounts of buffering as in the non-adjustable divider. (Small amounts of buffering is added for some serializer clocks for optimal timing) Adjustable 180° phases of the clock to both the SER and the divider is sufficient to ensure that data is ready while the appropriate latch is transparent.

The latency of this path is similar for this case with the adjustable divider, but has a variable latency path that is determined by the divider settings (Fig. 4.13). This path is shown in pink and has minimal latency for settings chosen in the above example.

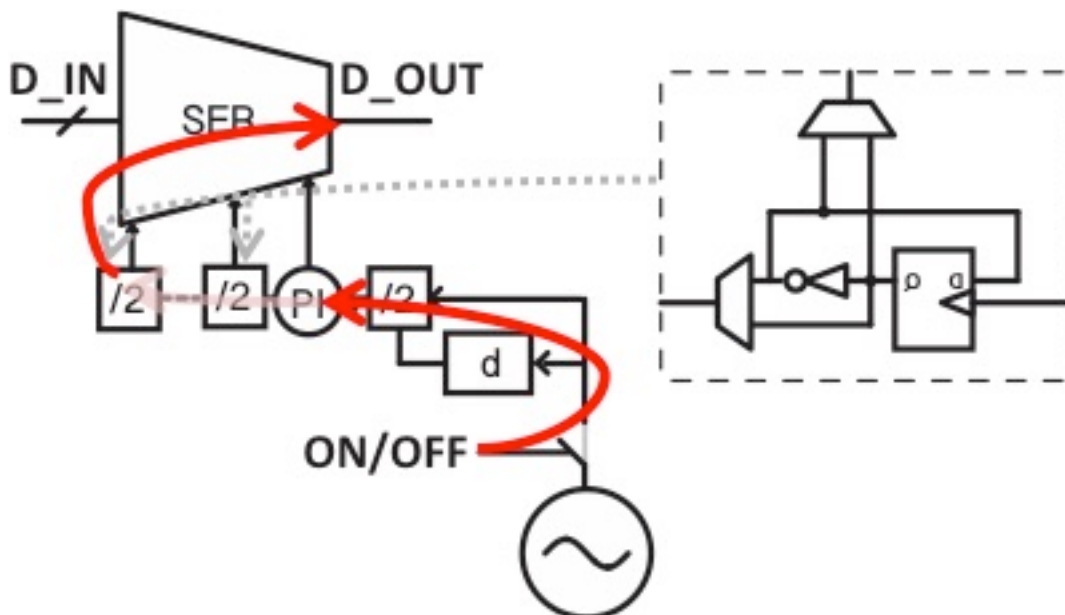


Figure. 4. 13. Latency path for 1-latch serializer w/ adjustable divider.

Clock phase can be reliably chosen depending on exact delay of the chain to save power from extensive buffering in the alternative buffered divider.

### 4.3 On/off clocking

Now that we have carefully designed our data-path circuitry to be functional while the clock is functional, this section explains the rapid-on/off clock design that drives those dynamic circuits.

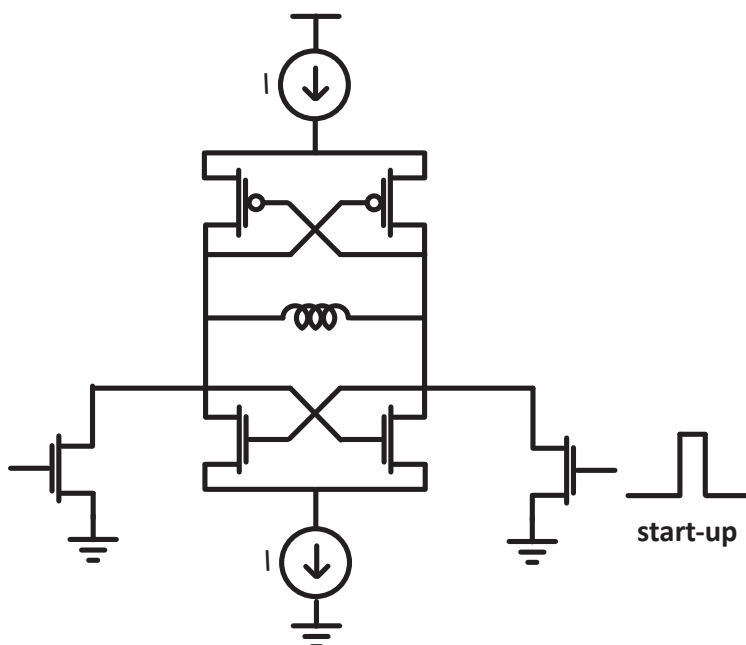


Figure. 4. 14. Complementary cross-coupled LC oscillator.

Since clocking circuits are designed to operate with a full swing clock DC biased around  $V_{dd}/2$ , it would also be desirable to have an oscillator that is biased around mid supply. A traditional NMOS cross-coupled LC oscillator would provide an output DC biased around  $V_{dd}$ ; however, a complementary cross-coupled LC oscillator [49] can conveniently provide the desired DC output condition and has twice the  $-1/g_m$  due to both the PMOS and NMOS cross-coupled pairs. Furthermore, as we are in 28nm, extra PMOS loading (for comparable NMOS width) no longer presents an issue as it did in previous technology nodes. A complementary cross-coupled LC oscillator containing NMOS startup switches is shown in Fig. 4.14 above.

This design (with the single NMOS startup switch) requires a start-up current pulse to turn on. This means that the current is dependent on the size of this switch, which affects the overall loading at the output. The next section discusses the startup time of this complementary cross-coupled LC oscillator.

### 4.3.1 Cross-coupled oscillator startup time

Oscillator startup time has been analyzed in [50][51] where small signal models have been used to determine the oscillator startup behavior (i.e. growth of the amplitude of oscillation). Similarly, we can take the small signal model shown in Fig. 4.15 to determine the startup time.

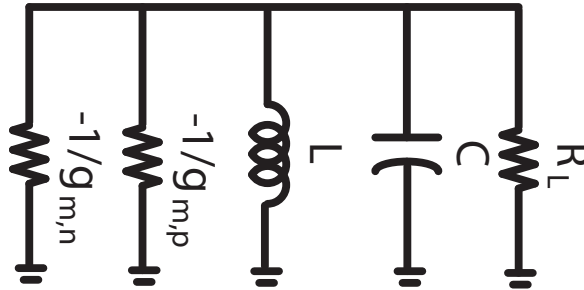


Figure. 4. 15. Small signal model of a complementary cross-coupled LC oscillator.

The small signal model circuit can be analyzed analogously to the NMOS cross-coupled LC OSC in [52], except  $-1/g_{m,p}$  is in parallel with  $-1/g_{m,n}$ .

Gong through the analysis of the parallel second-order circuit (derived in Appendix A), start-up time follows from (A.8) since  $Q_0$  cannot arbitrarily be lowered, especially for resonant-clocked architectures where the driver loads are presented directly at the oscillator. Hence,

$$T_{startup} = \frac{2Q_0}{\omega_0} \ln \frac{V_{max} \omega_0 C \sqrt{1 - 1/4Q_0^2}}{I_0} \quad (4.4)$$

where  $Q_0 = \omega_0 RC$ ,  $R = \frac{R_L}{g_{m,p} || g_{m,n} R_L - 1}$ , and there is an initial current  $I_0$  and 0 initial voltage applied to the tank as in 4.14.

From this equation, lower  $Q_0$  actually results in faster startup time. In order to speed up the startup time of the oscillator, lower  $Q_0$  is desired, meaning that startup is faster when no loading is present in the system.

Also,  $I_0$  can be increased to reduce startup time, but increasing  $I_0$  increases the size of the transistors that will eventually be used for injecting current into the tank, resulting in increased the parasitic tank capacitance. Since we would like to be able to decouple startup time from  $Q_0$ , we desire another way to startup our oscillator.

### 4.3.2 Rapid-on/off LC oscillator

Instead of directly adding startup switches at the outputs of the oscillator as in Fig. 4.14, we modify the startup circuitry by having two stacked switches with the top switch configured in positive feedback [52]. The rapid-on/off LC oscillator employs a modified version of the complementary cross-coupled oscillator to employ both NMOS and PMOS cross-coupled pairs, providing positive feedback drive strength from both rails and bias the clock at around half the supply voltage (Fig. 4.16).

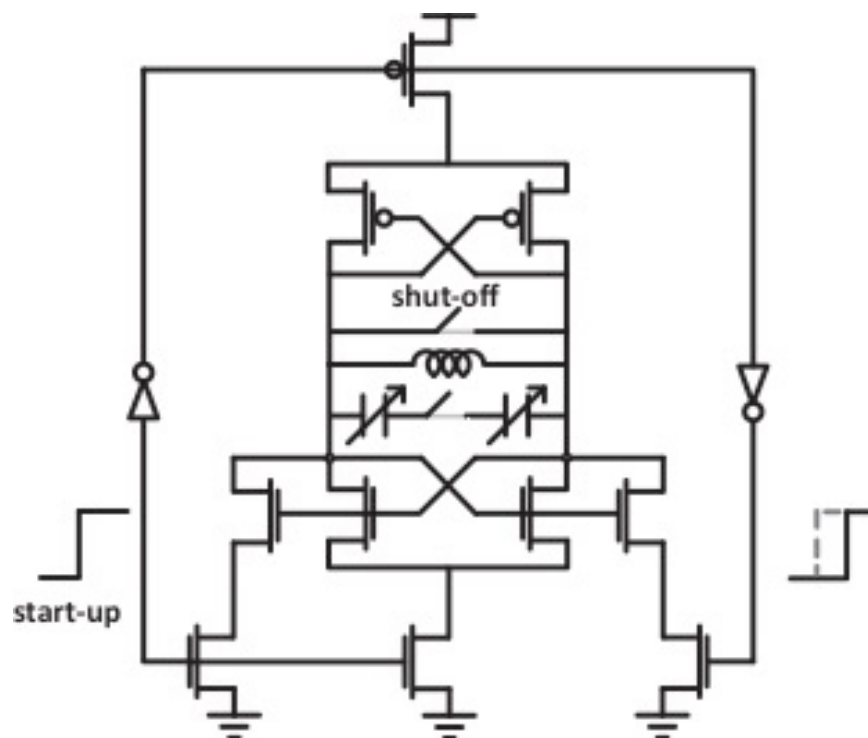


Figure 4. 16. On/off oscillator.

When the startup switches are on in this configuration, the top transistors in the stack are configured in positive feedback, so they can (and should) be left on for additional  $-1/g_m$  during operation. Hence, we can startup the oscillator by pulling down one side of the tank to start the oscillation followed shortly by the other side. Afterward, the startup transistor can simply remain on during the “on” state, removing the need for a current pulse.

Furthermore, since the startup transistor is repurposed (in positive feedback) for some  $-1/g_m$ , this design achieves much faster startup speed compared to its counterpart with containing

only a single pull-down switch. Turn on is achieved through directly pulling down one side of the oscillator with the on/off signal turning on one pair of NMOS (and then the other pair shortly after through a small amount of buffering) as in [52].

During the “on” state, current sources are turned on, and the startup switch stacks are turned on sequentially with the left side first and then the right side after a small amount of buffering to increase in negative impedance. In “off” state, current sources are turned off, and the outputs are shorted together.

Due to various sources of variability (corners or temperature variation), the oscillator has a 4-bit digitally controlled capacitor array ( $C_{VAR}$ ) to precisely tune the oscillation frequency (Fig. 4.17).

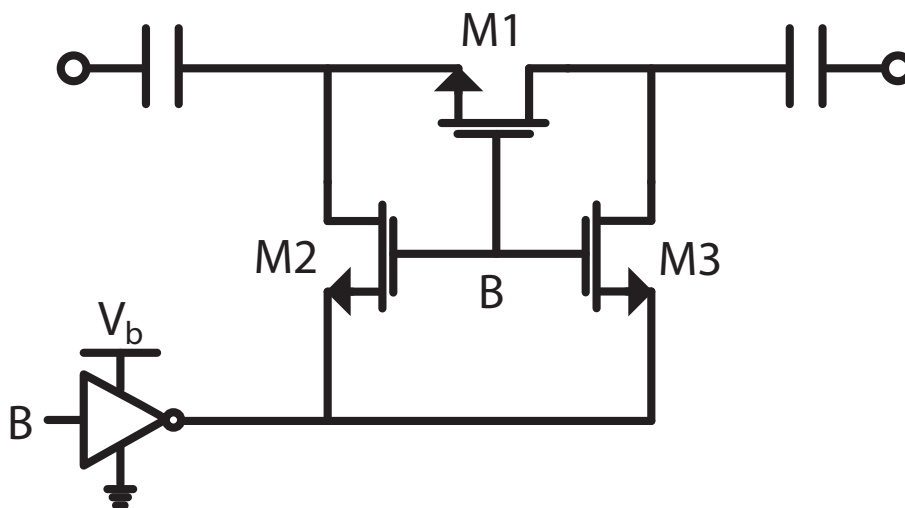


Figure. 4. 17. Capacitor Array Unit Cell ( $C_{VAR}$ ).

The center switch reduces parasitic capacitance compared to having the switches connected to the oscillator outputs. Bias voltage needs to be considered for the internal nodes around the switch during on/off instantaneous switching.  $V_b$  ensures the switches (M2, M3) are on and off during on/off conditions.

The oscillator operates at 10GHz to provide the half-rate clock for the even and odd SC data-paths (and is also the input to the divider chain). Discussed in the following section, the OSC’s inductor  $L_{OSC}$  directly resonates out the capacitance of the switches seen in the TX driving stage, saving power compared to conventional clock buffering.

### 4.3.3 Resonant clocking

Traditional clock buffering makes use of buffers and a clock tree to distribute the clock, especially over large distances [53]. Depending on clock frequency, the clock is distributed either at the nominal frequency or at a lower frequency and then multiplied up to the nominal frequency locally [54]. Resonant clocking, on the other hand, can save buffer power since the clock directly resonates out the loading capacitance to determine the oscillation frequency [55]. A comparison of clock buffering vs. resonant clocking is shown in Fig. 4.18 (a, b).

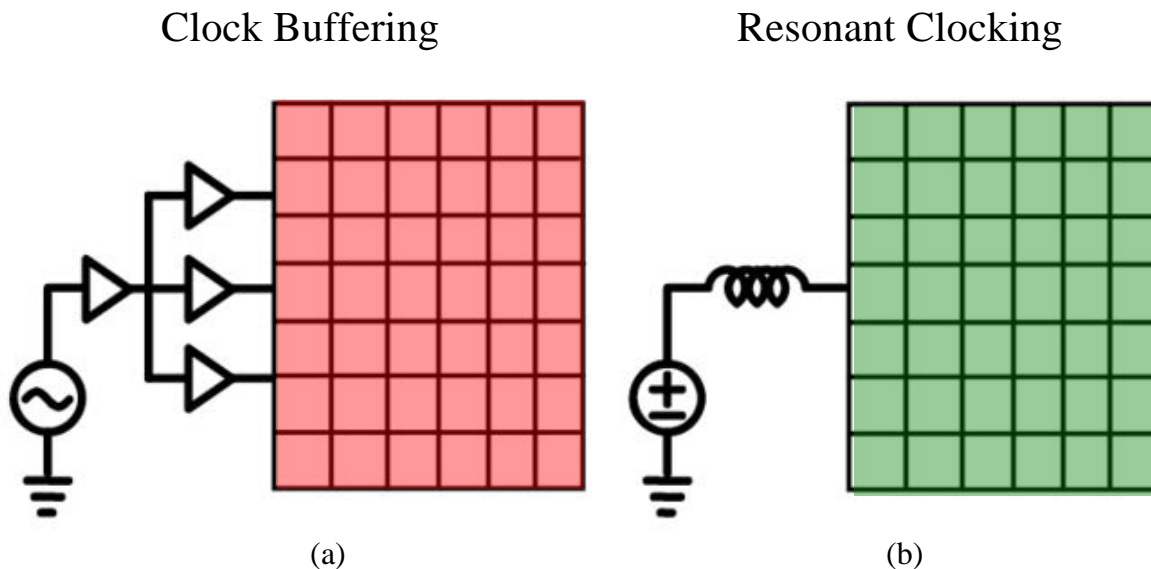


Figure. 4. 18. (a) Clock buffering (Red) (b) Resonant clocking (Green).

Even though most microprocessors distribute clocks using clock buffering, this becomes extremely costly at high speeds, since the energy required is directly proportional to switching frequency and it is difficult to distribute a matched, differential clock. Instead, our design makes use of resonant clocking to directly resonate out the load capacitance of the TX with the oscillator's inductor.

While some designs save clocking overhead by having multiple transmitters clocked through one oscillator, we desire to have per-lane, on/off control, making resonant clock a more attractive option since it would only require one load network per clock. Overloading the oscillator runs into minimum size issues for the oscillator inductance and also limits the frequency of operation. While the increased capacitive load will increase  $Q_0$  and increase the startup time of the oscillator, the energy savings from not having an always-on clock buffer leads us to our architecture in which the TX load directly loads the oscillator.

The intrinsic loading and loading network as shown in Fig. 4.19 determine the resonance frequency of this structure.

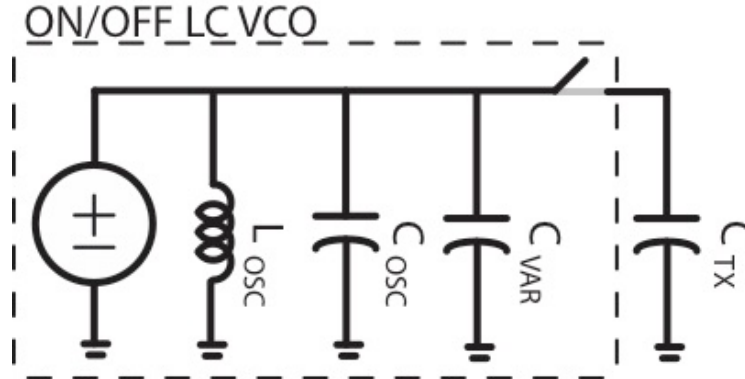


Figure. 4. 19. On/off LC VCO resonant clocking.

In the figure above,  $C_{TX}$  is seen directly in parallel at the oscillator's output network. Therefore, its resonant frequency is determined by the parasitic capacitance of the oscillator ( $C_{OSC}$ ), added variable capacitance for additional frequency tuning range ( $C_{VAR}$ ), load cap of the TX ( $C_{TX}$ ), and the oscillator inductance ( $L_{OSC}$ ) according to (4.5).

$$f_{osc} = \frac{1}{2\pi\sqrt{L_{OSC}(C_{OSC} + C_{VAR} + C_{TX})}} \quad (4.5)$$

## 4.4 High-speed phase alignment

Now that we have discussed the clock strategy and phase alignment of the SER's lower bits, all that is left is to make sure the last stage of serializer is appropriately aligned with the divided high speed clock. As shown in Fig. 4.20, even and odd serializer outputs should be timed such that they are valid for the entire time the high-speed clock ( $CK_{IN}$ ) is high or low (depending on odd or even).

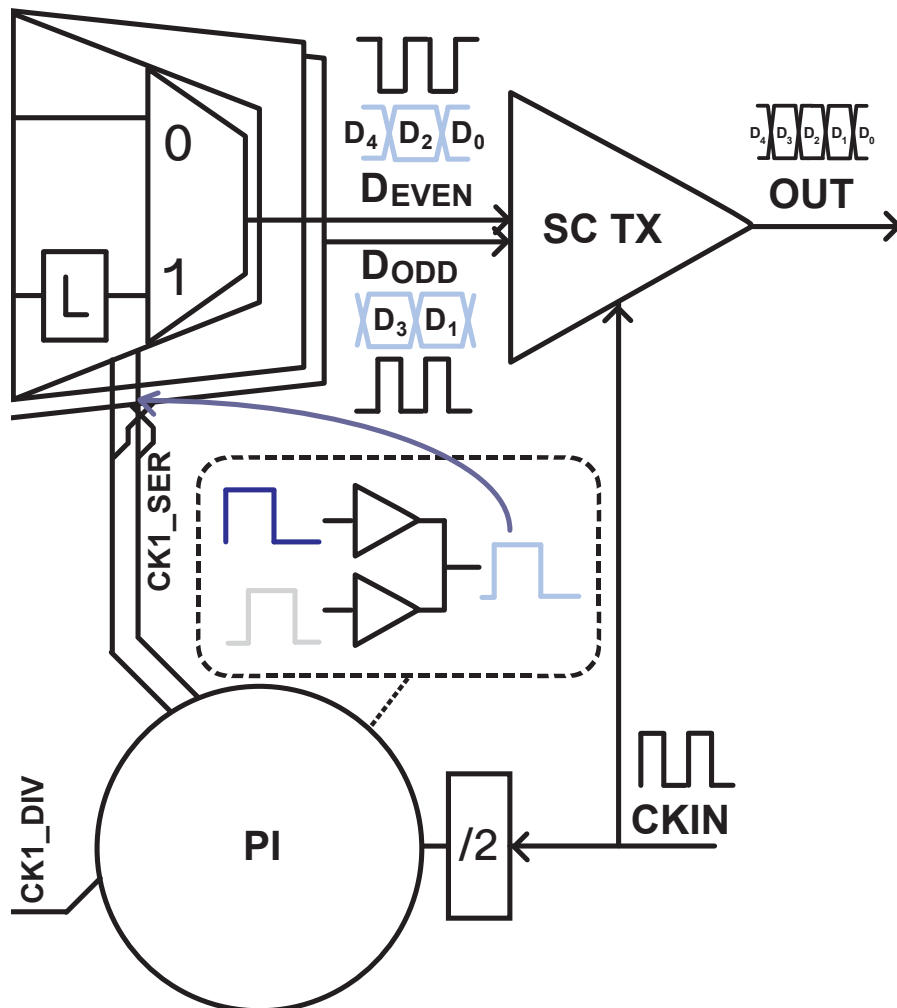


Figure. 4. 20. Divider and data phase alignment.

Instead of trying to time the SER output to be in phase with  $CK_{IN}$ , we instead align the phase of the clock for the last stage of the serializer [13]. Since the high-speed clock goes through the divider (which has intrinsic delay), it must be retimed to remain in-phase. A phase interpolator is used for this purpose and is described in the next section. Having full ( $360^\circ$ ) phase control over the last stage clocking guarantees that we can align the phases.

#### 4.4.1 Phase interpolator design

Digital to phase converters [56] are important building blocks in serial communication systems, as they often require fine resolution to line up various digital signals. Digital to phase converters (or phase interpolators) are often employed for clock and data recovery as the data needs to be well-matched with the clock in order to eliminate timing related errors [57]. Our design incorporates an inverter-based digital to phase converter [58] in our divider chain to align the last stage serializer with the high-speed clock, since an inverter-based PI conveniently does not need any level converters nor does it consume static current as in its or current combining (or



Gilbert cell [59]) phase interpolator counterpart. Furthermore, a current combining PI can be designed with high resolution, but we don't need as many bits for our purposes.

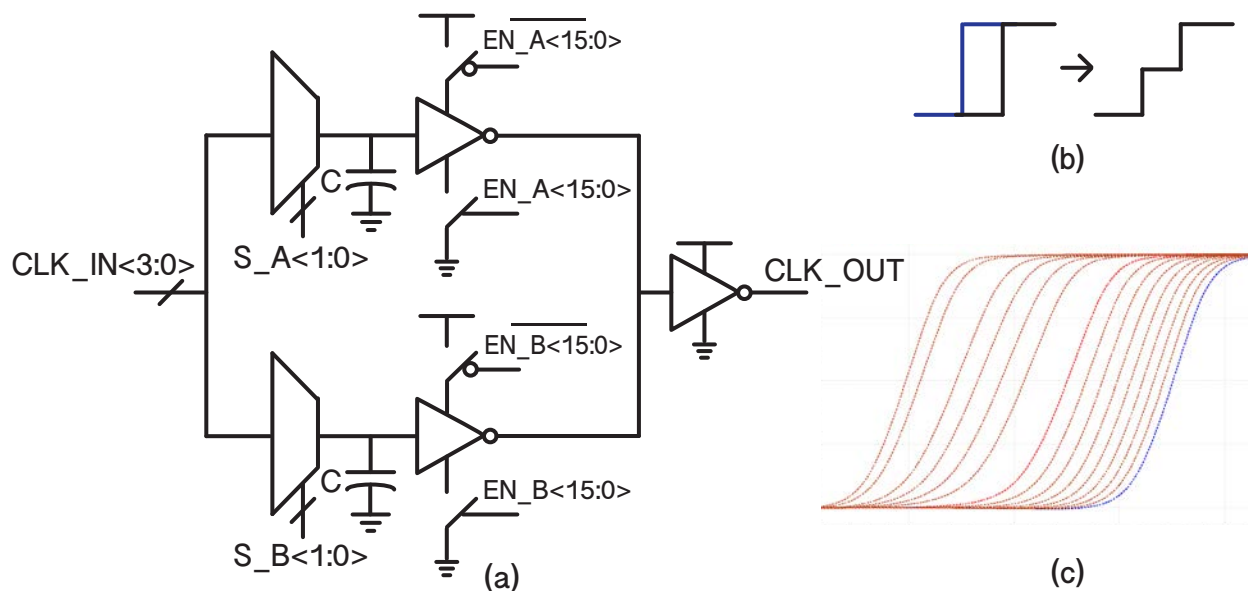


Figure. 4. 21. (a) 6-bit PI (b) Inverter PI ideal output (c) PI output waveform.

The 6-bit inverter-based PI (Fig. 4.21(a)) clocks the final stage of the serializer to ensure the right phase for the even and odd data. The inverter based phase interpolator operates by blending together two adjacent  $90^\circ$  clocks (i.e.  $CLK\_IN < 3 >$  and  $CLK\_IN < 0 >$  or  $CLK\_IN < 1 >$  and  $CLK\_IN < 2 >$ , etc. ...). The ideal output of an inverter-based PI can be seen in Fig. 4.21(b) where the staircase function results from the average of the two transition periods. To avoid this metastable condition, not only should we add rise time to the signals before combining, but also the signals should be close enough in phase. Typically, explicit capacitance is added to slow down the rise and fall time of the clock signals to be combined. The first 2 MSB's come naturally from the choice of  $CLK\_IN$ , and the 4 LSB's are determined by the number of inverters chosen per path. Various PI codes are simulated in Fig. 4.21(c) to show the operation of the inverter based PI (gap due to not all codes shown here). Here, the red signals show the various achievable phases by combining  $CLK\_IN < 0 >$  and  $CLK\_IN < 1 >$ . The following blue signal is the first code for the following inputs of  $CLK\_IN < 1 >$  and  $CLK\_IN < 2 >$ . While linearity is not the biggest concern, it is important that this PI control is monotonic, and that can be ensured by checking this condition across fast corners.

#### 4.4.2 High-speed clock divider phase alignment

While the clock may be almost immediately supplied to the driving stage, data will only be valid once the serializer turns on and is ready. Hence, we need a way to control the clock phase of the serializer independently from the switched-capacitor driver. Fig. 4.22 shows a block diagram of how we resolve the aforementioned alignment issue.

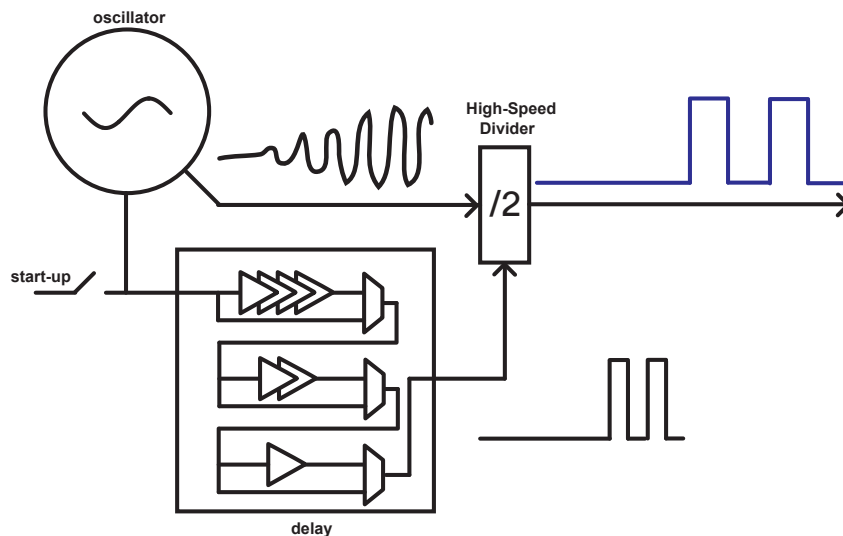


Figure. 4. 22. High-speed clock and divider phase alignment.

Ensuring the serializer's stability requires that the divider chain is activated with consistent start-up conditions. If we directly supply the serializer with the clock of the oscillator, as we do with the switched-capacitor driving stage, the first high speed latch of the divider will not be guaranteed to deterministically start up with the right conditions. The serializer should wait until the clock has started and that it is in a rising or falling edge condition with full swing. An example rapid-on/off waveform labeled with the relevant conditions is shown in Fig. 4.23.

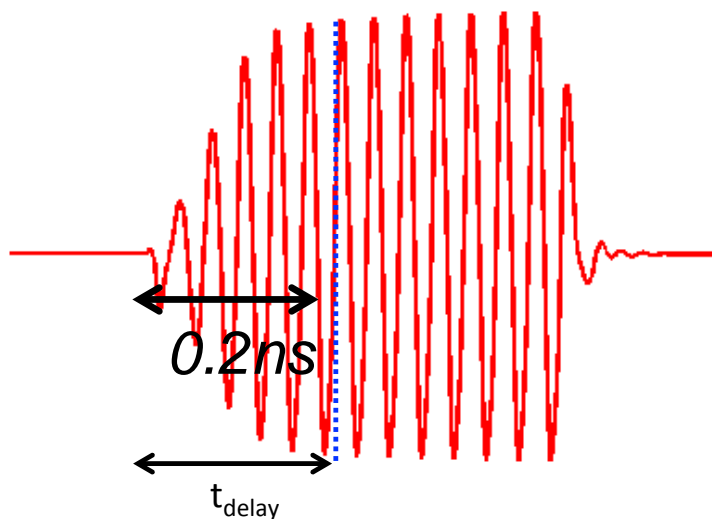


Figure. 4. 23. Oscillator waveform timing.

Simulation shows that several (5+) cycles are required for the oscillator to reach full swing and a stable oscillation frequency. The tunable delay ( $t_{delay}$ ) allows us to ignore the period of time shown in the example oscillator waveform where the clock has not yet reached full swing. To ensure that the enough time has passed before the oscillator is fully on, a tunable delay shown in Fig. 4.24 is added in between the oscillator and divider chain to ensure both correct phase and sufficient amplitude of the clock is provided as an input to the serializer.

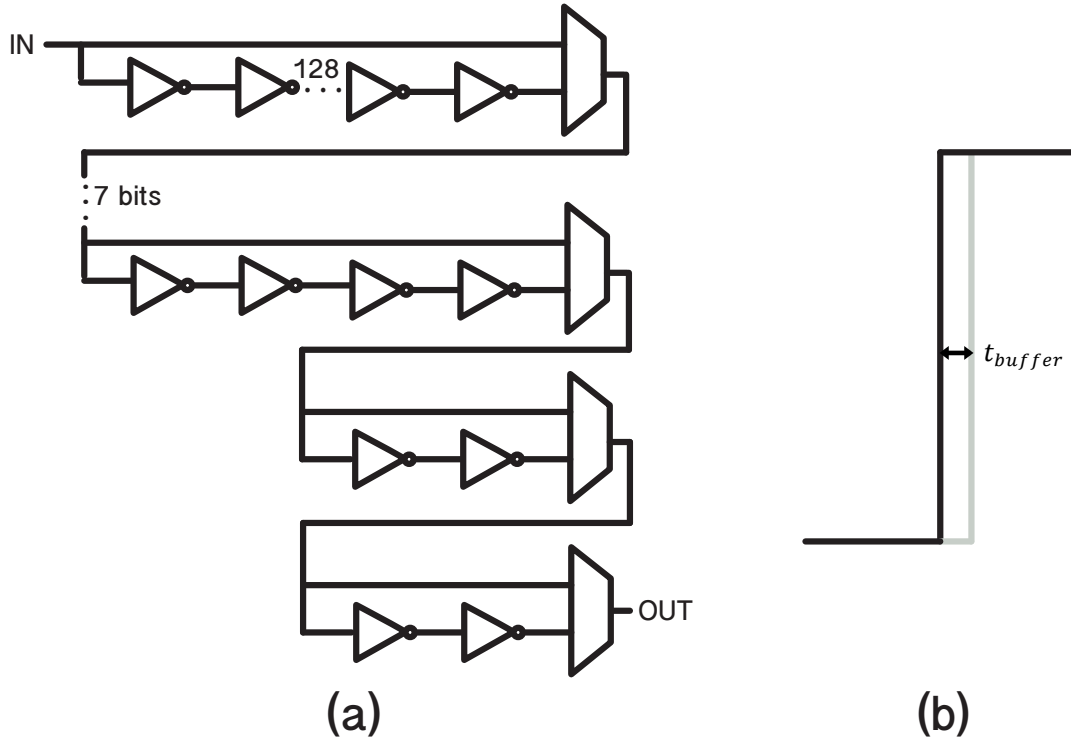


Figure. 4. 24. (a) 7-bit tunable delay circuit (b) Delay resolution ( $t_{buffer}$ ).

Delay is precisely tuned from with an  $N = 7$  bit tunable delay circuit consisting of inverters shown in Fig. 4.24(a). Binary weighed delay is determined by which paths of inverters are enabled.

There are many ways to build a tunable delay line. Since we are optimizing for minimal quiescent current, we would like to stay away from DACs that consume constant current. This approach consists entirely of dynamic elements (MUXs and inverters).

The delay of this N-bit tunable delay circuit can be determined by the following equations

$$t_{delay}(code) = code * t_{buffer} + (N + 1) * t_{mux} \text{ where } 0 < code < 2^{N+1} \quad (4.6)$$

where  $N = 7$  in our scenario.

The drawback of this approach is that it consumes dynamic power based on the number of inverters involved and that there is a minimum delay determined by the number of MUX's in the delay line as shown in (4.6). Again, we are not as concerned about minimizing dynamic power as we are with static power.

The minimum delay  $t_{delay}(0)$  is

$$t_{delay}(0) = (N + 1) * t_{mux} \quad (4.7)$$

or equivalently 8 MUX delays.

However, the minimum delay is not a problem as well, since we intentionally need to delay the clock by far more than 8 MUX delays due to the minimum start-up time of the LC oscillator,

which is much larger than  $t_{delay}(0)$ . The tunable delay circuit gives us 100ps to 1.5ns of delay with 5.7ps resolution (buffer delay) in Fig. 4.24(b).

## 4.5 Measurement results

The transmitter was fabricated in a 28nm TSMC process and occupies a rectangular region of  $0.19mm^2$  in Fig. 4.25 below.

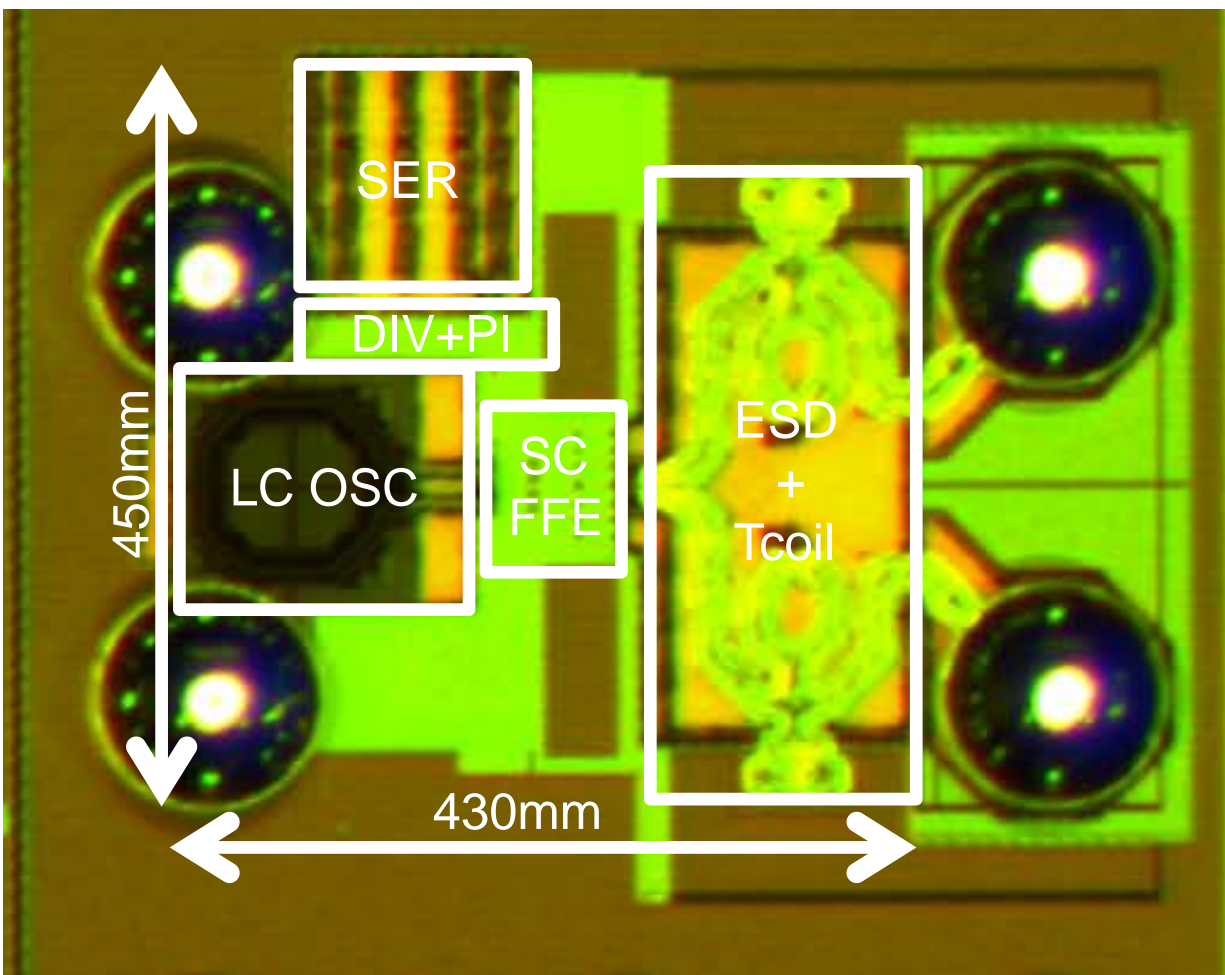


Figure. 4. 25. Die photo (TX).

The chip is directly attached via flip-chip bumps to minimize parasitic loading on a testing board made out of Megtron6-5670 material. To characterize the design, a DSA90804A scope was used to measure the equalized pulse response and burst-mode 4Kbit PRBS data pattern (loaded through scan) after 2.92mm connectors and 5inch coax cables shown in Fig. 4.26.

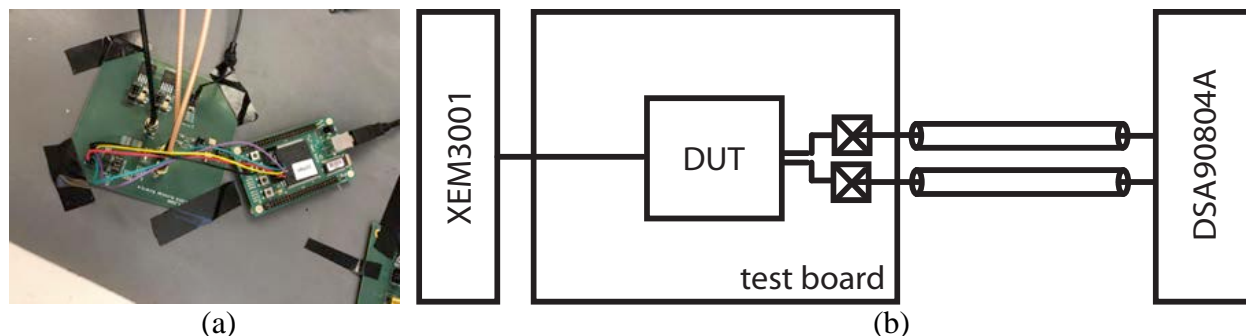
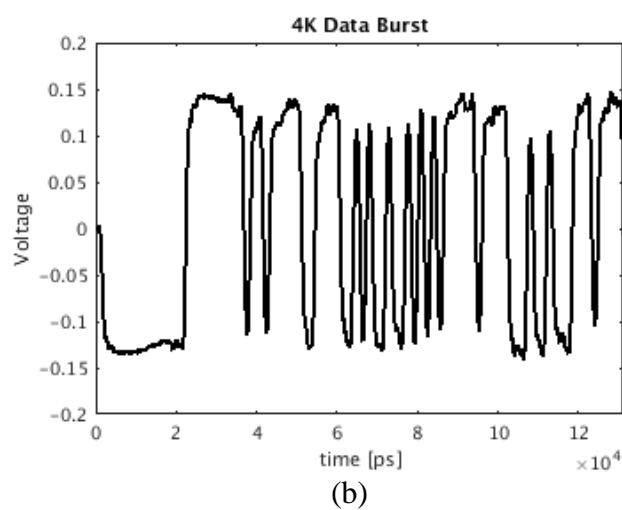
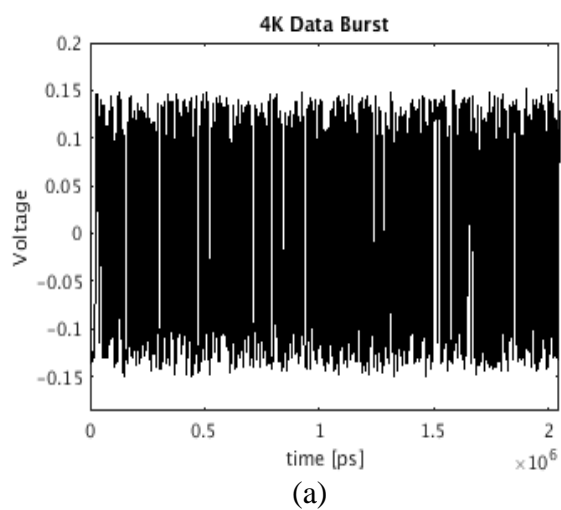
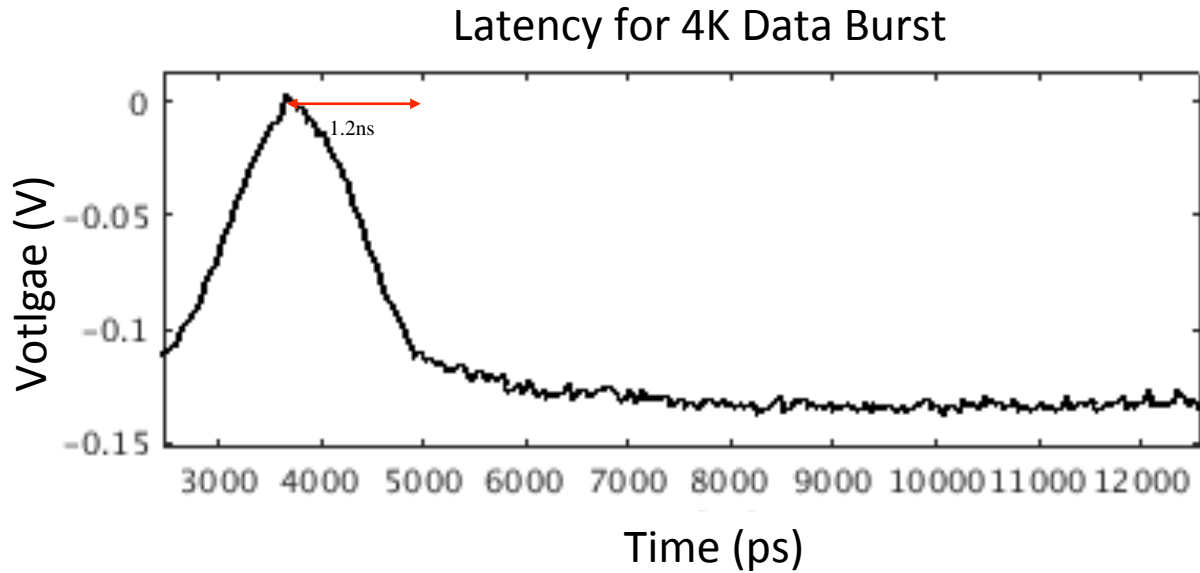


Figure 4. 26. (a) Testing setup image (b) Testing setup block diagram.

According to the loss of the cable, the output swing was measured to be 260mV. With the testing setup in Fig. 4.26(b) (using an Opal Kelly XEM3001 to load a 4Kbit pattern on chip), the burst-mode functionality of the transmitter was verified through 4K matching input and output bits, where the differential output of a 4Kbit pattern is seen in Fig. 4.27(a). Fig. 4.27(b) shows the same data zoomed in from the beginning of a valid data stream. Latency of 1.2ns was measured from 0V differential, found by sending repeating data blocks with the same duty cycle and measuring the time between the last and first bits between adjacent blocks (increasing duty cycle until data until it is no longer valid) (Fig. 4.27(c)).





(c)

Figure. 4. 27. (a) 4Kbit Data. (b) 4Kbit Data (zoomed in). (c) Latency (zoomed in further).

The measured, equalized pulse response and the TX eye diagram for 4Kbits of a PRBS data burst are shown in Fig. 4.28.

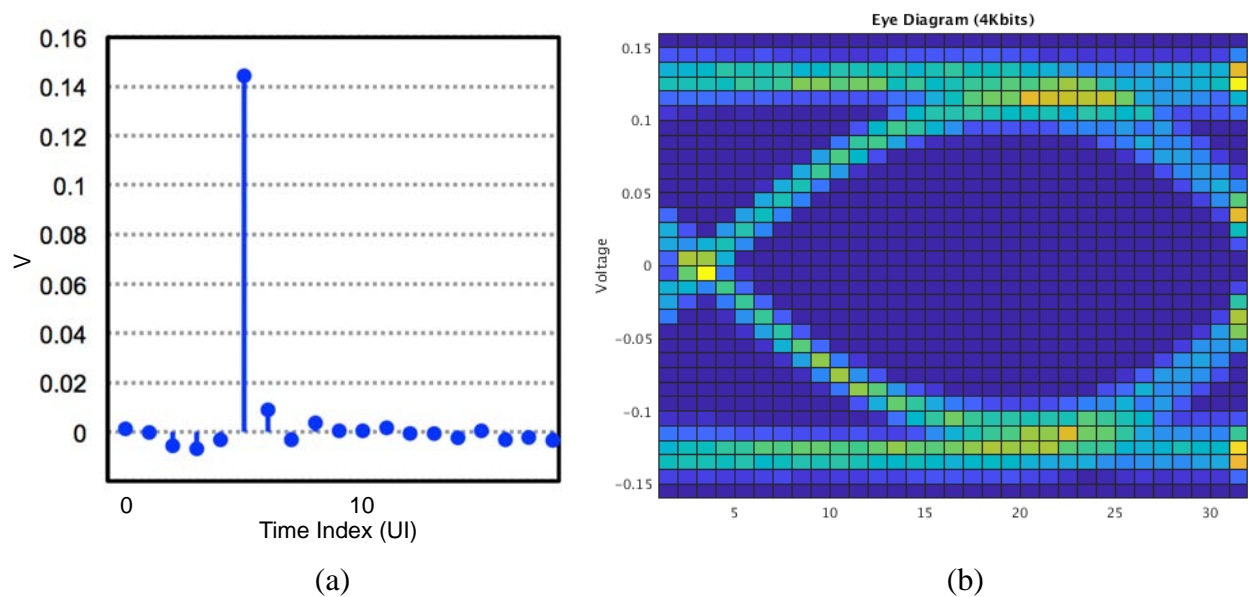


Figure. 4. 28. (a) UI spaced pulse response. (b) TX eye diagram at 20Gb/s for 4Kbits PRBS pattern.

Energy efficiency at 20 Gb/s was measured to be 0.62 pJ/bit and 0.72 pJ/bit at 1 Gb/s and is plotted for various effective data-rates in Fig. 4.29(a). The LC OSC operates from 9.7 to 10.1 GHz as seen in Fig. 4.29(b).

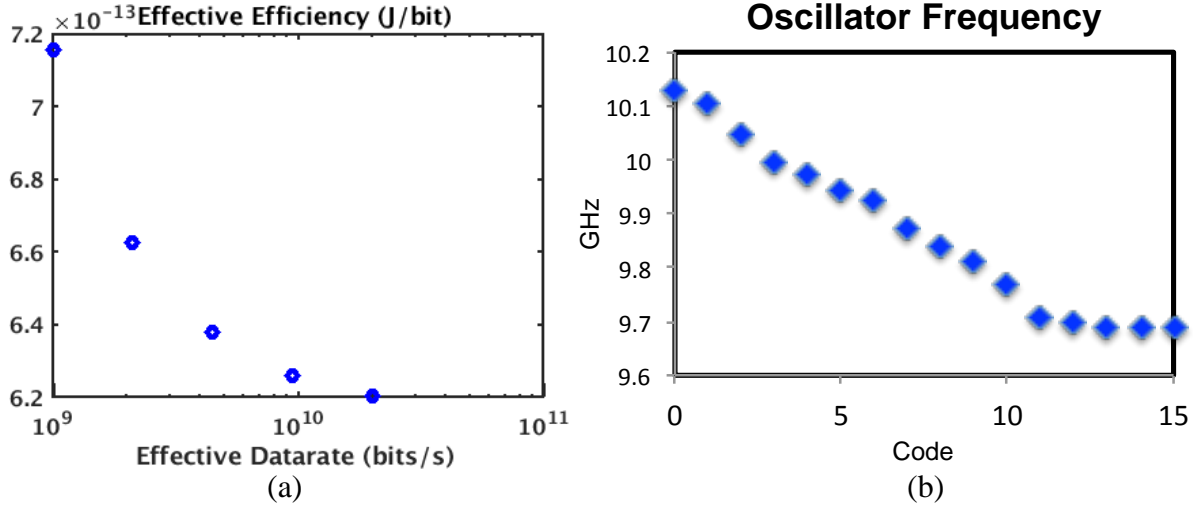


Figure. 4. 29. (a) Effective efficiency from 1Gb/s to 20Gb/s. (b) OSC frequency vs. code.

Table 4. 1: Comparison of energy proportional and SC driver architectures

Reference	[42] JSSC'15	[39] JSSC'18	[43] ISSCC'18	<b>This work</b>
Process	65nm	65nm	16nm	<b>28nm</b>
Supply (V)	1.1/1	1.0	.95/.85	<b>1.0</b>
Max Data-rate (Gb/s)	7	10	25	<b>20</b>
TX Equalizer	3-tap FFE	3-tap FFE	Pre- emphasis	<b>2-tap FFE</b>
TX On-Power (mW)	28.7	16.1	11.2	<b>12.4</b>
TX Architecture	CML	SC	SC	<b>SC</b>
Rapid-on/off	Yes	Yes	No	<b>Yes</b>
Standby Power (mW)	0.74	0.155	N/A	<b>0.1</b>
TX Latency (ns)	0.5	N/A	5	<b>1.2</b>
Efficiency* (pJ/bit)	9.8-9.1	3.8-3.6	N/A-1.17	<b>0.72- 0.62</b>
Serialization ratio	16:1	16:1	16:1	<b>64:1</b>
Output (mV)	500	450	300	<b>260</b>
Area	0.39	0.12	0.08	<b>0.19</b>

\*1Gb/s-Max Data-rate

Table III shows the summary of the burst-mode transmitter. This work provides a 2-tap FFE SC TX at 20Gb/s with comparable on-power, standby power, and TX latency to [39][43]. Furthermore, our serialization ratio of 64:1 (4x more than 16:1 in [39][42][43]) provides inherent added latency challenges that we address in the design of the low latency, 1-latch, SER.

## 4.6 Conclusion

High throughput, energy proportional links allow for the possibility of unified on-board communication standards. Burst mode enables massive aggregated bandwidth and minimizes

latency especially at reduced data-rates, and energy and pin constraints demand that massive I/O applications employ burst-mode communication. Furthermore, startup time/latency becomes more important when tasks require more trips between numerous ASIC modules. This paper describes an energy efficient, energy proportional solution (0.72-0.62 pJ/bit) for the transmit side that incorporates various techniques on the serializer for low latency (1.2ns). The design includes a 64:2 1-latch MUX SER with phase adjustable clocking, fast on/off LC OSC, and a DDR 2-tap SC FFE for dynamic equalization at 20Gb/s operation in  $0.19mm^2$ .



## Chapter 5

### Conclusion

#### 5.1 Thesis summary

The increasing demands on data and the end of Moore's Law means systems can no longer increase capacity (computational or communication) by pure transistor scaling alone. As systems become more complicated with more chips composing a single module, limited space for heat dissipation and a limited number of pins per chip mean I/O data-rate must scale in terms of efficiency and throughput accordingly. These considerations lead to the design of high-speed link architectures, able to send data at high-speed per-pin bandwidths with the lowest possible energy. Energy efficient, high-speed equalization techniques are demonstrated in the high-speed receiver and transceiver designs.

- An integrating CTLE + DMUX allows for DDR operation in the receiver, all while equalizing long-tail ISI in the channel. An RX FFE with variable cascode gate bias integration summing and delay through analog dynamic latches allows for 2-tap FFE to reduce pre-cursor ISI.
- Furthermore, FFE + DFE integrating summer with incomplete settling and a dedicated latch summer for the first tap allows for optimal energy efficiency and 3-taps of DFE equalization.
- These techniques are first demonstrated in a 65nm process allowing for a receiver frontend to achieve 60 Gb/s and equalize a  $V_{ISI}/V_{CURSOR}$  of 1.54, while consuming 173mW. It is employed again in a 65nm, complete NRZ transceiver to achieve 60 Gb/s and equalize a 21-dB loss channel, while consuming 288mW.

While energy-efficient data-rate scaling is important as it allows for the realization of higher-speed I/O, research into energy proportional communication is important as well if we want these high-speed I/O to be ubiquitous across more pins on a die. Especially as system-level, computational complexity increases due to more distributed tasks, latency becomes important, motivating the need for high-speed, burst-mode SerDes standards. For a burst-mode design, dynamic architectures that may not be the most power efficient during operation are shown to be valuable, despite the fact that high-speed SerDes has gone the direction of lowest achievable power despite the costs on area or standby power (especially with multi-bit DACs).

- In particular, our design demonstrates a fully dynamic transmitter, incorporating techniques to avoid DACs and limit quiescent current and latency, through the design of the 2-tap SC FFE TX, 1-latch MUX serializer, control methods for coefficient or delay control.
- The design is demonstrated to achieve 1-20Gb/s with 1.2ns latency, while consuming 100mW of standby power and 12.4mW power during operation. Able to transmit data with serialization of 64:1 with minimal latency, the proposed architecture provides a transmit side solution to ubiquitous, energy proportional communication.

## 5.2 Future directions

While this thesis covered receive-side techniques at high speed and a separate, burst-mode transmitter architecture for energy proportional communication, a similarly-low latency burst-mode receiver could be designed as well. Since most of the receiver architecture in the 60Gb/s design employs dynamic techniques for equalization, they could be repurposed for burst-mode operation, as it has been shown in the transmitter architecture that a fast-on/off oscillator works well for burst-mode functionality. Specifically, challenges remain in designing a low latency, burst-mode CDR and removing the many DACs that bias various switches of the equalizers. Furthermore, voltage regulation techniques for burst-mode design like switched-capacitor DC-DC converters [40][60] could be an attractive solution for supply regulation under burst-mode regimes. Finally, integrating the low latency SerDes with a microprocessor (or similar computational unit) with simultaneous latency optimizations could further uncover developments in energy proportional communication.

## Bibliography

- [1] C. M. Jha, L. Choobineh, A. Jain, (2015) *Microelectronics Thermal Sensing: Future Trends*. In: Jha C. (eds) *Thermal Sensors*. Springer, New York, NY
- [2] D. C. Daly, L. C. Fujino and K. C. Smith, "Through the Looking Glass - The 2018 Edition: Trends in Solid-State Circuits from the 65th ISSCC," in *IEEE Solid-State Circuits Magazine*, vol. 10, no. 1, pp. 30-46, winter 2018.
- [3] M.-S. Lin, C.-C. Tsai, C.-H. Chang, W.-H. Huang, Y.-Y. Hsu, S.-C. Yang, C.-M. Fu, M.-H. Chou, T.-C. Huang, C.-F. Chen, T.-C. Huang, S. Adham, M.-J. Wang, W. W. Shen, A. Mehta "A 1 Tbit/s Bandwidth 1024 b PLL/DLL-Less eDRAM PHY Using 0.3 V 0.105 mW/Gbps Low-Swing IO for CoWoS Application," in *Solid-State Circuits, IEEE Journal of*, vol.49, no.4, pp.1063-1074, Apr. 2014.
- [4] B. Collins, J. Deng, K. Li, and L. Fei-Fei. Towards scalable dataset construction: An active learning approach. In *ECCV08*, pages I: 86– 98, 2008.
- [5] V. M. Stojanovic, "Channel-Limited High-Speed Links: Modeling, Analysis and Design," Stanford University, 2004.
- [6] J. Han, N. Sutardja, Y. Lu and E. Alon, "Design Techniques for a 60-Gb/s 288-mW NRZ Transceiver With Adaptive Equalization and Baud-Rate Clock and Data Recovery in 65-nm CMOS Technology," in *IEEE Journal of Solid-State Circuits*, vol. 52, no. 12, pp. 3474-3485, Dec. 2017.
- [7] C. Thakkar, S. Sen, J. Jaussi and B. Casper, "A 32 Gb/s Bidirectional 4-channel 4 pJ/b Capacitively Coupled Link in 14 nm CMOS for Proximity Communication," in *IEEE Journal of Solid-State Circuits*, vol. 51, no. 12, pp. 3231-3245, Dec. 2016.
- [8] K. K. Parhi, "High-speed architectures for algorithms with quantizer loops," *IEEE International Symposium on Circuits and Systems*, New Orleans, LA, USA, 1990, pp. 2357-2360 vol.3.
- [9] Y. Lu and E. Alon, "Design techniques for a 66 Gb/s 46 mW 3-tap decision feedback equalizer in 65 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 48, no. 12, pp. 3243–3257, Dec. 2013.
- [10] A. Awmy, L. Moeller, J. Junio, J. C. Scheytt, and A. Thiede, "Design and measurement techniques for an 80 Gb/s 1-tap decision feedback equalizer," *IEEE J. Solid-State Circuits*, vol. 49, no. 2, pp. 452–470, Feb. 2014.
- [11] T. Shibusaki, W. Chaivipas, Y. Chen, Y. Doi, T. Hamada, H. Takauchi, T. Mori, Y. Koyanagi, H. Tamura, "A 56-Gb/s receiver front-end with a CTLE and 1-tap DFE in 20-nm CMOS," in *Symp. VLSI Circuits Dig.*, 2014, pp. 1–2.

- [12] M. Park, J. Bulzacchelli, M. Beakes, and D. Friedman, "A 7 Gb/s 9.3mW 2-tap current-integrating DFE receiver," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, 2007. ISSCC'07*, pp. 230–599.
- [13] C. Thakkar, N. Narevsky, C. D. Hull, and E. Alon, "Design techniques for a mixed-signal I/Q 32-coefficient Rx-feedforward equalizer, 100-coefficient decision feedback equalizer in an 8 Gb/s 60 GHz 65 nm LP CMOS receiver," *IEEE J. Solid-State Circuits*, vol. 49, no. 11, pp. 2588–2607, Nov. 2014.
- [14] Y. Duan and E. Alon, "A 12.8 GS/s time-interleaved ADC with 25 GHz effective resolution bandwidth and 4.6 ENOB," *IEEE J. Solid-State Circuits*, vol. 49, no. 8, pp. 1725–1738, Sep. 2014.
- [15] A. Momtaz and M. M. Green, "An 80 mW 40 Gb/s 7-Tap T/2-Spaced Feed-Forward Equalizer in 65 nm CMOS," in *IEEE Journal of Solid-State Circuits*, vol. 45, no. 3, pp. 629–639, March 2010.
- [16] M. Chen and C. K. Yang, "A 50–64 Gb/s serializing transmitter with a 4-tap, LC-ladder-filter-based FFE in 65-nm CMOS," *Proceedings of the IEEE 2014 Custom Integrated Circuits Conference*, San Jose, CA, 2014, pp. 1–4.
- [17] G. W. Roberts and M. Ali-Bakhshian, "A Brief Introduction to Time-to-Digital and Digital-to-Time Converters," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 3, pp. 153–157, March 2010.
- [18] J. Han, "Design and Automatic Generation of 60Gb/s Wireline Transceivers," University of California, Berkeley, 2017.
- [19] Y. Frans et al., "A 56 Gb/s PAM4 wireline transceiver using a 32-way time-interleaved SAR ADC in 16nm FinFET", *Proc. IEEE Symp. VLSI Circuits*, pp. 1–2, Jun. 2016.
- [20] P. J. Peng, J. F. Li, L. Y. Chen, and J. Lee, "A 56 Gb/s PAM-4/NRZ transceiver in 40nm CMOS," in *IEEE Int. Solid-State Circuits Conf. ISSCC'2017, Dig. Tech. Papers*, Feb. 2017, pp. 110–111.
- [21] J. Lee, P.-C. Chiang, and C.-C. Weng, "56 Gb/s PAM4 and NRZ SerDes transceivers in 40nm CMOS," in *Proc. IEEE Symp. VLSI Circuits*, Jun. 2015, pp. 118–119.
- [22] T. Shibusaki, T. Danjo, Y. Ogata, Y. Sakai, H. Miyaoka, F. Terasawa, M. Kudo, H. Kano, A. Matsuda, S. Kawai, T. Arai, H. Higashi, N. Naka, H. Yamaguchi, T. Mori, Y. Koyanagi, H. Tamura, "A 56Gb/s NRZ-electrical 247mW/lane serial-link transceiver in 28nm CMOS," *2016 IEEE International Solid-State Circuits Conference (ISSCC)*, San Francisco, CA, 2016, pp. 64–65.
- [23] M. Slota. (Oct. 3–4, 2018). OpenCAPI and its Roadmap. OpenPOWER Summit Europe [Online]. Available: <https://openpowerfoundation.org/wp-content/uploads/2018/10/Myron-Slota.OPF-Summit-Oct-2018-OC-and-RM-v3.pdf>

- [24] S. Sutardja, "1.2 The future of IC design innovation," *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, San Francisco, CA, 2015, pp. 1-6.
- [25] G. E. Moore, "No exponential is forever: but "Forever" can be delayed!" *IEEE International Solid-State Circuits Conference, Digest of Technical Papers. ISSCC'2003*, San Francisco, CA, Feb. 2003, pp. 20-23.
- [26] M. O'Connor. (Jun. 14, 2014). Highlights of the High Bandwidth Memory (HBM) Standard. The Memory Forum. [Online]. Available: <http://www.cs.utah.edu/thememoryforum/mike.pdf>
- [27] J. W. Poulton, J. M. Wilson, W. J. Turner, B. M. Zimmer, X. Chen, S. S. Kudva, S. Song, S. G. Tell, N. Nedovic, W. Zhao, S. R. Sudhakaran, C. T. Gray, W. J. Dally, "A 1.17-pJ/b, 25-Gb/s/pin Ground-Referenced Single-Ended Serial Link for Off- and On-Package Communication Using a Process- and Temperature-Adaptive Voltage Regulator," in *IEEE Journal of Solid-State Circuits*, vol. 54, no. 1, pp. 43-54, Jan. 2019.
- [28] Y. Frans, D. Carey, M. Erett, H. Amir-Aslanzadeh, W. Fang, D. Turker, A. Jose, A. Bekele, J. Im, P. Upadhyaya, Z. Wu, K. Hsieh, J. Savoj, K. Chang, "A 0.5–16.3 Gb/s Fully Adaptive Flexible-Reach Transceiver for FPGA in 20 nm CMOS," in *IEEE Journal of Solid-State Circuits*, vol. 50, no. 8, pp. 1932-1944, Aug. 2015.
- [29] P. Upadhyaya, J. Savoj, F.-T. An, A. Bekele, A. Jose, B. Xu, D. Wu, D. Turker, H. Aslanzadeh, H. Hedayati, J. Im, S.-W. Lim, S. Chen, T. Pham, Y. Frans, K. Chang, "3.3 A 0.5-to-32.75Gb/s flexible-reach wireline transceiver in 20nm CMOS," *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, San Francisco, CA, 2015, pp. 1-3.
- [30] P. Upadhyaya, C. F. Poon, S. W. Lim, J. Cho, A. Roldan, W. Zhang, J. Namkoong, T. Pham, B. Xu, W. Lin, H. Zhang, N. Narang, K. H. Tan, G. Zhang, Y. Frans, K. Chang, "A fully adaptive 19-to-56Gb/s PAM-4 wireline transceiver with a configurable ADC in 16nm FinFET," *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, San Francisco, CA, 2018, pp. 108-110.
- [31] J. Kim, A. Balankutty, A. Elshazly, Y.-Y. Huang, H. Song, K. Yu, F. O'Mahony, "3.5 A 16-to-40Gb/s quarter-rate NRZ/PAM4 dual-mode transmitter in 14nm CMOS," *IEEE International Solid-State Circuits Conference, Digest of Technical Papers. ISSCC'2015*, San Francisco, CA, 2015, pp. 1-3.
- [32] C.-K. K. Yang and M. A. Horowitz, "A 0.8-/spl mu/m CMOS 2.5 Gb/s oversampling receiver and transmitter for serial links," in *IEEE Journal of Solid-State Circuits*, vol. 31, no. 12, pp. 2015-2023, Dec. 1996.
- [33] W. J. Dally and J. Poulton, "Transmitter equalization for 4-Gbps signaling," in *IEEE Micro*, vol. 17, no. 1, pp. 48-56, Jan.-Feb. 1997.

- [34] H. Wilson and M. Haycock, "A six-port 30-GB/s nonblocking router component using point-to-point simultaneous bidirectional signaling for high-bandwidth interconnects," in *IEEE Journal of Solid-State Circuits*, vol. 36, no. 12, pp. 1954-1963, Dec. 2001.
- [35] M. Kossel, C. Menolfi, J. Weiss, P. Buchmann, G. v. Bueren, L. Rodoni, T. Morf, T. Toifl, M. Schmatz "A T-Coil-Enhanced 8.5 Gb/s High-Swing SST Transmitter in 65 nm Bulk CMOS With  $\ll$  -16 dB Return Loss Over 10 GHz Bandwidth," in *IEEE Journal of Solid-State Circuits*, vol. 43, no. 12, pp. 2905-2920, Dec. 2008.
- [36] K. Fukuda *et al.*, "An 8Gb/s Transceiver with 3 $\times$ -Oversampling 2-Threshold Eye-Tracking CDR Circuit for -36.8dB-loss Backplane," *IEEE International Solid-State Circuits Conference - Digest of Technical Papers. ISSC'2008*, San Francisco, CA, 2008, pp. 98-598.
- [37] R. Sredojevic and V. Stojanovic, "Digital link pre-emphasis with dynamic driver impedance modulation," in *IEEE Custom Integrated Circuits Conference 2010*, 2010, pp. 1-4.
- [38] P. Chiang, H. Hung, H. Chu, G. Chen and J. Lee, "2.3 60Gb/s NRZ and PAM4 transmitters for 400GbE in 65nm CMOS," *IEEE International Solid-State Circuits Conference Digest of Technical Papers. ISSCC'2014*, San Francisco, CA, 2014, pp. 42-43.
- [39] D. Wei, T. Anand, G. Shu, J. E. Schutt-Ainé and P. K. Hanumolu, "A 10-Gb/s/ch, 0.6-pJ/bit/mm Power Scalable Rapid-on/off Transceiver for On-Chip Energy Proportional Interconnects," in *IEEE Journal of Solid-State Circuits*, vol. 53, no. 3, pp. 873-883, March 2018.
- [40] H. P. Le, "Design Techniques for Fully Integrated Switched-Capacitor Voltage Regulators," University of California, Berkeley, 2013.
- [41] J.W. Poulton, W.J. Dally, X. Chen, J.G. Eyles, T.H. Greer, S.G. Tell, C.T. Gray, "A 0.54pJ/b 20Gb/s ground-referenced single-ended short-haul serial link in 28nm CMOS for advanced packaging applications," in *IEEE International Solid-State Circuits Conference Digest of Technical Papers. ISSCC'2003*, vol., no., pp.404-405, 17-21, Feb. 2013.
- [42] T. Anand, M. Talegaonkar, A. Elkholy, S. Saxena, A. Elshazly and P. K. Hanumolu, "A 7 Gb/s embedded clock transceiver for energy proportional links," in *IEEE Journal of Solid-State Circuits*, vol. 50, no. 12, pp. 3101-3119, Dec. 2015
- [43] J. M. Wilson et al., "A 1.17pJ/b 25Gb/s/pin ground-referenced single-ended serial link for off- and on-package communication in 16nm CMOS using a process- and temperature-adaptive voltage regulator," in *IEEE International Solid-State Circuits Conf. ISSCC'2018*, San Francisco, CA, 2018, pp. 276-278.
- [44] J. Han, Y. Lu, N. Sutardja, K. Jung, and E. Alon, "Design techniques for a 60 Gb/s 173 mW wireline receiver frontend in 65 nm CMOS technology," *IEEE J. Solid-State Circuits*, vol. 51, no. 4, pp. 871-880, Apr. 2016.

- [45] E. Chang, N. Narevsky, J. Han and E. Alon, "An Automated SerDes Frontend Generator Verified with a 16NM Instance Achieving 15 GB/S at 1.96 PJ/Bit," *2018 IEEE Symposium on VLSI Circuits*, Honolulu, HI, 2018, pp. 153-154.
- [46] J. Kim, J.-K. Kim, B.-J. Lee, M.-S. Hwang, H.-R. Lee, S.-H. Lee, N. Kim, D.-K. Jeong, W. Kim, "Circuit techniques for a 40Gb/s transmitter in 0.13/ $\mu$ m CMOS," *ISSCC. IEEE International Digest of Technical Papers. Solid-State Circuits Conference. ISSC'2005*, San Francisco, CA, 2005, pp. 150-589 Vol. 1.
- [47] S. Kaeriyama, Y. Amamiya, H. Noguchi, Z. Yamazaki, T. Yamase, K. Hosoya, M. Okamoto, S. Tomari, H. Yamaguchi, H. Shoda, H. Ikeda, S. Tanaka, T. Takahashi, R. Ohhira, A. Noda, K. Hijoka, A. Tanabe, S. Fujita, N. Kawahara, "A 40 Gb/s Multi-Data-Rate CMOS Transmitter and Receiver Chipset With SFI-5 Interface for Optical Transmission Systems," in *IEEE Journal of Solid-State Circuits*, vol. 44, no. 12, pp. 3568-3579, Dec. 2009.
- [48] A. A. Hafez, M. Chen and C. K. Yang, "A 32-to-48Gb/s serializing transmitter using multiphase sampling in 65nm CMOS," *IEEE International Solid-State Circuits Conference Digest of Technical Papers. ISSC'2013*, San Francisco, CA, 2013, pp. 38-39.
- [49] J. Craninckx, M. Steyaert and H. Miyakawa, "A fully integrated spiral-LC CMOS VCO set with prescaler for GSM and DCS-1800 systems," *Proceedings of CICC 97 - Custom Integrated Circuits Conference*, Santa Clara, CA, USA, 1997, pp. 403-406.
- [50] A. Rusznyak, "Start-up time of CMOS oscillators," in *IEEE Transactions on Circuits and Systems*, vol. 34, no. 3, pp. 259-268, March 1987.
- [51] Y. Tsuzuki, T. Adachi and Ji Wen Zhang, "Fast start-up crystal oscillator circuits," *Proceedings of the 1995 IEEE International Frequency Control Symposium (49th Annual Symposium)*, San Francisco, CA, USA, 1995, pp. 565-568.
- [52] L. Kong and E. Alon, "A 21.5mW 10+Gb/s mm-wave phased-array transmitter in 65nm CMOS," *2012 Symposium on VLSI Circuits (VLSIC)*, Honolulu, HI, 2012, pp. 52-53.
- [53] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolić, *Digital Integrated Circuits*, 2/e. Pearson Education, 2003, p. 761.
- [54] I. A. Young, M. F. Mar and B. Bhushan, "A 0.35  $\mu$ m CMOS 3-880 MHz PLL N/2 clock multiplier and distribution network with low jitter for microprocessors," *IEEE International Solids-State Circuits Conference. Digest of Technical Papers. ISSC'1997*, San Francisco, CA, USA, 1997, pp. 330-331.
- [55] A. J. Drake, K. J. Nowka, T. Y. Nguyen, J. L. Burns and R. B. Brown, "Resonant clocking using distributed parasitic capacitance," in *IEEE Journal of Solid-State Circuits*, vol. 39, no. 9, pp. 1520-1528, Sept. 2004.
- [56] Ju-Ming Chou, Yu-Tang Hsieh and Jieh-Tsorng Wu, "A 125MHz 8b digital-to-phase converter," *IEEE International Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC'2003*, San Francisco, CA, USA, 2003, pp. 436-505 vol.1.

- [57] P. Hanumolu, V. Kratyuk, G. -Y. Wei and U. -K. Moon, "A Sub-Picosecond Resolution 0.5-1.5GHz Digital-to-Phase Converter," *Symposium on VLSI Circuits, 2006. Digest of Technical Papers. VLSI'2006*, Honolulu, HI, 2006, pp. 75-76.
- [58] M. Chen, A. A. Hafez and C. Ken, "A 0.1-1.5 GHz 8-bit inverter-based digital-to-phase converter using harmonic rejection," *2012 IEEE Asian Solid State Circuits Conference (A-SSCC)*, Kobe, 2012, pp. 145-148.
- [59] B. Gilbert, "A precise four-quadrant multiplier with subnanosecond response," in *IEEE Journal of Solid-State Circuits*, vol. 3, no. 4, pp. 365-373, Dec. 1968.
- [60] B. Zimmer, Y. Lee, A. Puggelli, J. Kwak, R. Jevtic, B. Keller, S. Bailey, M. Blagojevic, P. Chiu, H.-P. Le, P. Chen, N. Sutardja, R. Avizienis, A. Waterman, B. Richards, P. Flatresse, E. Alon, K. Asanovic, B. Nikolic, "A RISC-V vector processor with tightly-integrated switched-capacitor DC-DC converters in 28nm FDSOI," *2015 Symposium on VLSI Circuits (VLSI Circuits)*, Kyoto, 2015, pp. C316-C317.
- [61] L. Kong, "Energy-Efficient 60GHz Phased-Array Design for Multi-Gb/s Communication Systems," University of California, Berkeley, 2014.



## Appendix A

### Start-up time of complementary LC oscillator

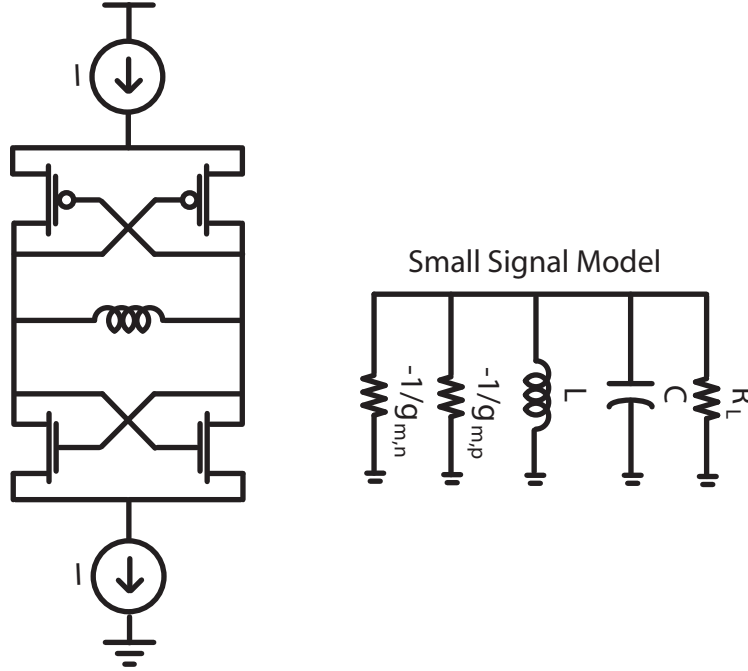


Figure. A. 1. Complementary cross-coupled oscillator and its small signal model.

The small signal model can be used to calculate the oscillator dynamics, based on the negative impedance of the oscillator's cross-coupled NMOS ( $-1/g_{m,n}$ ) and negative impedance of the oscillator's cross-coupled PMOS ( $-1/g_{m,p}$ ) along with the combined load capacitance and inductor and its load resistance.

With a similar analysis to [61], the oscillator dynamics are set by the following equation where the tank voltage ( $V_{out}$ ) can be expressed as

$$V_{out} - \frac{L}{R} \frac{dV_{out}}{dt} + LC \frac{d^2V_{out}}{dt^2} = 0 \quad (\text{A.1})$$

where

$$R = \frac{R_L}{g_{m,p} || g_{m,n} R_L - 1}. \quad (\text{A.2})$$

Assuming the tank voltage is of the form  $V_0 e^{kt}$ , (A.1) can be written as

$$V_0 e^{kt} - \frac{L}{R} k V_0 e^{kt} + LC k^2 V_0 e^{kt} = 0 \quad (\text{A.3})$$

$$1 - \frac{L}{R}k + LCk^2 = 0 \quad (\text{A.4})$$

We can replace R, L, and C by  $\omega_0 = \frac{1}{\sqrt{LC}}$  and  $Q_0 = \omega_0 RC = \frac{R}{\omega_0 L}$  and simplify the equation to the second order system

$$\frac{k^2}{\omega_0^2} - \frac{k}{\omega_0 Q_0} + 1 = 0. \quad (\text{A.5})$$

The roots of (A.5) can be found as

$$k_{1,2} = \frac{\omega_0}{2Q_0} \pm \frac{\omega_0}{2} \sqrt{\frac{1}{Q_0^2} - 4} \quad (\text{A.6})$$

$$k_{1,2} = \omega_0 \left( \frac{1}{2Q_0} \pm \sqrt{\frac{1}{4Q_0^2} - 1} \right) \quad (\text{A.7})$$

Depending on the sign on  $\frac{1}{4Q_0^2} - 1$ , the start-up behavior of the oscillator will be different.

For  $Q_0 > 1/2$ , the roots are complex (with  $k_{1,2}$  being complex conjugates). Hence,  $V_{out}(t)$  can be written as,

$$V_{out}(t) = \frac{I_0}{\omega_0 C \sqrt{1 - \frac{1}{4Q_0^2}}} e^{\frac{\omega_0 t}{2Q_0}} \sin \left( \sqrt{1 - \frac{1}{4Q_0^2}} \omega_0 t \right) \quad (\text{A.8})$$

assuming 0 initial voltage is applied to the tank, with initial current  $I_0$ . Waveform behavior can be characterized as a sinusoidal waveform that increases exponentially with amplitude. Amplitude increases until the  $g_{m,p}$  and  $g_{m,n}$  decrease to the point where  $g_{m,p} || g_{m,n} R_L$  decreases to 1 and  $Q_0$  is infinite (as  $R = \frac{R_L}{g_{m,p} || g_{m,n} R_L - 1}$ ), making the exponential constant ( $e^0 = 1$ ).

The time it takes to arrive at peak amplitude assuming maximum amplitude ( $V_{max}$ ) can be written as

$$T_{start} = \frac{2Q_0}{\omega_0} \ln \frac{V_{max} \omega_0 C \sqrt{1 - 1/4Q_0^2}}{I_0}. \quad (\text{A.9})$$

For  $Q_0 < 1/2$ , both poles are real, so the voltage can be written as

$$V_{out}(t) = \frac{I_0}{2\omega_0 C \sqrt{\frac{1}{4Q_0^2} - 1}} e^{\frac{\omega_0 t}{2Q_0}} \left( e^{\omega_0 t \sqrt{\frac{1}{4Q_0^2} - 1}} - e^{-\omega_0 t \sqrt{\frac{1}{4Q_0^2} - 1}} \right) \quad (\text{A.10})$$

Since the second term decays exponentially over time, the resulting behavior is set by the first term that exponentially increases without any sinusoidal waveform. The waveform increases in amplitude until  $Q_0$  reaches  $1/2$ . Oscillation amplitude increases in amplitude increase much faster, as amplitude increases (as seen from the product of two exponentials in (A.10) vs. one exponential in (A.8)). From these derivations, low  $Q_0$  is desired to speed up the oscillation process. However, this case requires  $Q_0 < 1/2$ , which is hard to achieve for resonant-clocked architectures.