University of California
Santa Barbara

Analysis and Design of Precision Timing Circuits using Pulse Mode Event Signaling

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Electrical and Computer Engineering

by

Prashansa Mukim

Committee in charge:

Professor Forrest Brewer, Chair
Professor James Buckwalter
Professor Luke Theogarajan
Professor Yuan Xie

March 2021

The Dissertation of Prashansa Mukim is approved.

Professor James Buckwalter

Professor Luke Theogarajan

Professor Yuan Xie

Professor Forrest Brewer, Committee Chair

December 2020

Analysis and Design of Precision Timing Circuits using Pulse Mode Event Signaling

by

Prashansa Mukim

Acknowledgements

First and foremost, I would like to express my deepest gratitude towards my advisor Professor Forrest Brewer for the innumerable ways in which he has supported me over the last five years. He decided to take me on as his student even when I was unsure about my goals and took the time to teach me everything ranging from the subtleties of designing chips to analysis strategies to writing better code to deconstructing large problems, and most importantly how to be creative in my thinking. Not only has Forrest been a wealth of knowledge, he has also been patient, supported me when I have struggled and given me ample opportunity to learn from my mistakes. Discussing crazy research ideas with Forrest was one of the most enjoyable parts of grad school, and I am extremely grateful for how generous he has been with his time.

I would also like to thank my committee members, all of whom I have been fortunate to work with and learn from during different stages of my PhD. Professor Luke Theogarajan, whose inspiring classes I have taken more than once and learnt how important it is to get the fundamentals right in order to succeed in any endeavor. I also am thankful for his encouragement, trust in me when I was his TA and his valuable feedback on my research. I could not have done without Professor Yuan Xie's support when I first started my PhD. I thank him for the opportunity to work on the CHIPS project and helping me identify important research directions. Working with the SEAL team taught me one of my first lessons on doing collaborative research and inspired me to have a strong work ethic. I also could not have completed my research without Professor Jim Buckwalter's support, who very generously provided me with access to his measurement lab and an opportunity to tapeout in 22nm. The time that he has taken to review my research and provide feedback has also been greatly valuable. I would also like to thank Professor Peng Li for all the stimulating discussions related to Spiking Neural Networks that have

helped me discover future research interests.

I am extremely thankful for the support of all my lab-mates during the course of my PhD. It was a pleasure to work with David, who was my lab neighbor and co-TA on several occasions and always inspired me with his thoroughness. I thank him for all his work on finishing chip tapeouts, help with resolving IT issues (that had a way of appearing at the most inopportune moments) and being a friend. I also would like to thank Aditya, Merritt, Carrie, Joseph, Di and Matt for guiding me when I first joined the lab and making HFH 2164 a fun place to be. I would like to express my gratitude towards Val, for all her guidance, helping make paperwork less daunting and giving the best advice. Thanks also to Avery, Bear and Paul for all their support and always being prompt with renewing key accesses.

Thanks to all my friends in Santa Barbara (Kruthika, Ravi, Akshar, Leah, Bharat, Bharath, Anna, Sadia) for their support and camaraderie over all these years and all the potlucks and game nights, Padma, Divya and Shubhra, who were my companions while taking classes when we spent all our time being at CSIL and my friends in India (Sameeksha, Richa, Meenal, Vasu, Shireen, Nidhi, Rajni) for always being there for me. I will be forever grateful to my parents Archana and Anil Mukim for all their love, for always encouraging me to make my own choices, being proud of the littlest of my achievements and for being my biggest pillars of strength. I also would like to thank my sister Priyanka for being my constant companion and my brother-in-law Ayush for all the good times while visiting India.

My sincere gratitude to Professor Ramesh Gaonkar who encouraged me to come to the US to pursue a PhD and has always looked out for me. Finally, I would like to thank the almighty for the events that led me to be at a place as beautiful as Santa Barbara, pursue my dreams, acquire knowledge, form personal connections with so many wonderful people and have the opportunities to make a difference.

Curriculum Vitæ

Prashansa Mukim

Education

2020	Ph.D. in Electrical and Computer Engineering, University of California, Santa Barbara.
2017	M.S. in Electrical and Computer Engineering, University of California, Santa Barbara.
2013	B.E. in Electronics and Communication, Netaji Subhas Institute of Technology, New Delhi, India

Publications

- **P. Mukim** and F. Brewer, "Multi-Wire Phase Encoding: A Signaling Strategy for High Bandwidth, Low Power Data Movement", IEEE Transactions on Very Large Scale Integration (VLSI) Systems (under review)
- **P. Mukim**, A. Dalakoti, D. McCarthy, C. Segal, M. Miller, J. F. Buckwalter and F. Brewer, "Design and Analysis of Collective Pulse Oscillatots", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2020
- **P. Mukim**, A. Dalakoti, D. McCarthy, B. Pon, C. Segal, M. Miller, J. F. Buckwalter and F. Brewer, "Distributed Pulse Rotary Traveling Wave VCO: Architecture and Design", IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2019
- M. Miller, A. Dalakoti, **P. Mukim** and F. Brewer, "Low Phase Noise Pulse Rotary Wave Voltage Controlled Oscillator", International SoC Design Conference (ISOCC), 2017
- M. Miller, C. Segal, D. McCarthy, A. Dalakoti, **P. Mukim** and F. Brewer, "Impolite High Speed Interfaces with Asynchronous Pulse Logic", Great-Lakes Symposium on VLSI (GLSVLSI), 2018
- P. Gu, D. Stow, **P. Mukim**, S. Li and Y. Xie, "Cost-efficient 3D Integration to Hinder Reverse Engineering During and After Manufacturing", Asian Hardware Oriented Security and Trust Symposium (AsianHOST), 2018
- J. Mekie, **P. Mukim** and K. Kale, "Impact of Process Variations on Synchronizer Performance: An Experimental Study", IEEE International Conference on VLSI Design and Embedded Systems (VLSID), 2018

List of Figures

List of Tables

Abstract

Analysis and Design of Precision Timing Circuits using Pulse Mode Event Signaling

by

Prashansa Mukim

This thesis presents an analytical framework and circuit solutions to a host of timing problems that are applicable to systems of varying scales. These non-standard solutions provide higher performance alternatives to the issues of multi-phase clock distribution and larger-scale on-chip or on-interposer communication and coherence. The underlying circuits for all these solutions are pulse mode asynchronous and built using a logic family of gates that encode both data and the time of arrival as atomic pulses. These gates incorporate local negative feedback loops leading to useful dynamic behavior that is exploited in the design of low-noise closed loop timing circuits.

First, Collective Pulse Oscillators (or CPOs), the simplest closed loop pulse gate circuits are presented. CPOs achieve FoMs better than conventional ring oscillators, while also providing precise clock phases much finer than the delay of a typical gate in a given technology. The noise analysis of CPOs is based on a time-domain analytical model, which is expanded into a rapid behavioral simulator, and validated against measurement results for different diameter CPOs implemented in 130 nm technology. Next, complex CPO topologies that improve the high-frequency stability of loop-connected CPOs are presented and the potential of such structures in implementing timing distribution networks that utilize distributed feedback to enhance the accuracy of the timing source itself is evaluated. Architecture, design and measurement results for a 5.5 GHz low-jitter transmission-line stabilized pulsed wave oscillator implemented in 130 nm are also presented.

Second, Multi-Wire Phase Encoding (MWPE), a transition signaling strategy suitable for on-chip/on-interposer links connecting globally asynchronous modules is presented. These links are aimed at reducing the energy cost of moving data across a large scale SoC or NoC, which is known to be orders of magnitude higher than the cost of computation. MWPE, by encoding data in the time correlated switching of multiple signaling wires allows very high-bandwidth data transmission on band-limited and lossy on-chip wires, with PLL/DLL free data recovery. Theoretical and practical bandwidth limits for MWPE are derived and link performance in 22 nm FDX technology is evaluated. Finally, methods of implementing low cost time-domain linear filters that utilize precise clock phases to reduce timing noise originating from systematic noise sources are presented.

Contents

Chapter 1

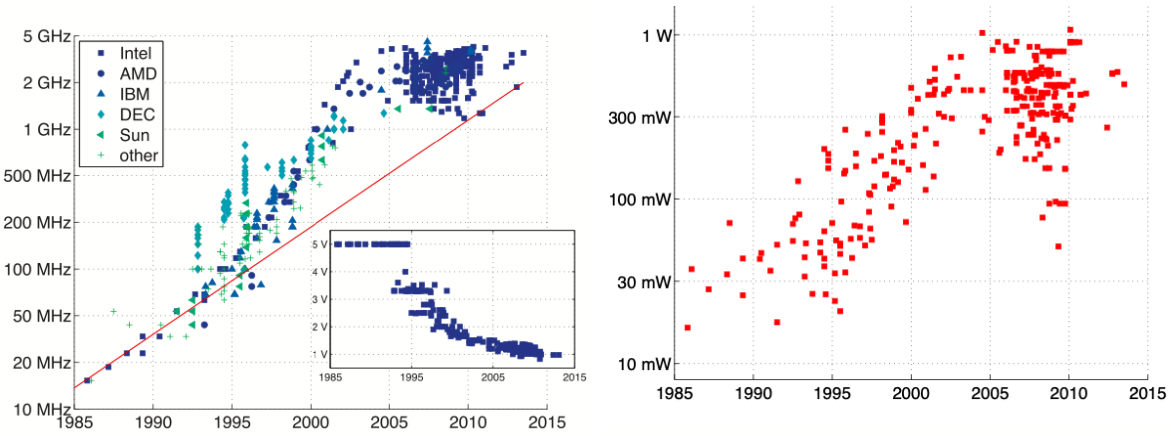
Introduction

1.1 Background and Motivations

Timing is an integral part of most VLSI design applications such as clocking [7, 8, 9], analog-to-digital and digital-to-analog conversion [10, 11], RF communication [12, 13], clock and data recovery [14], time-based metrology [15, 16] and neuromorphic computing [17, 18]. The performance, precision and silicon footprint of timing circuits driving these systems, as well as the energy cost associated with distributing timing and maintaining a notion of global coherence play a crucial role in quantifying the success of these designs. Thus, analysis and design of timing circuits optimized for varying system-level goals has been an active area of research for the last several decades.

With the exponential trend of shrinking transistor feature sizes in the last 50 years, and the consequences of scaling on the performance and power limits of integrated circuits, the architecture of large scale ICs and processors, in particular, has undergone a paradigm shift. As shown in Fig. 1.1, even though the speed of transistors has continued to increase, the clock frequency and power density of processor chips stalled around the mid 2000s. It is believed that the exponential power growth until the mid 2000s was

mainly due to the fact that supply voltages were *not scaled* at the constant field rate [19], and that clock frequencies were *scaled faster* than the constant field scaling in the quest for performance [1]. As power is $CV_{dd}^2 f$, and capacitance (C) scales with technology, the solution to bound the power density and associated thermal issues has been to limit the clock frequency f of uni-processors. The supply voltage V_{dd} has not been lowered below $0.8 - 1V$, as a lower V_{dd} demands lower transistor thresholds, which increases the leakage currents. Instead, multi-processor chips have emerged as a solution where each individual processor operates at a lower frequency, can be selectively enabled or disabled, and the multi-processor system achieves an overall higher performance measured by its throughput.



(a) Clock frequency and V_{dd} (insert plot) vs. year

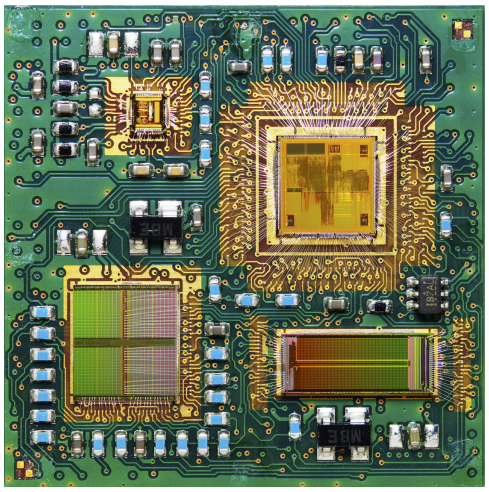
(b) Power density vs. year

Figure 1.1: Processor clock frequency and power density trends over the last 30 years [1]

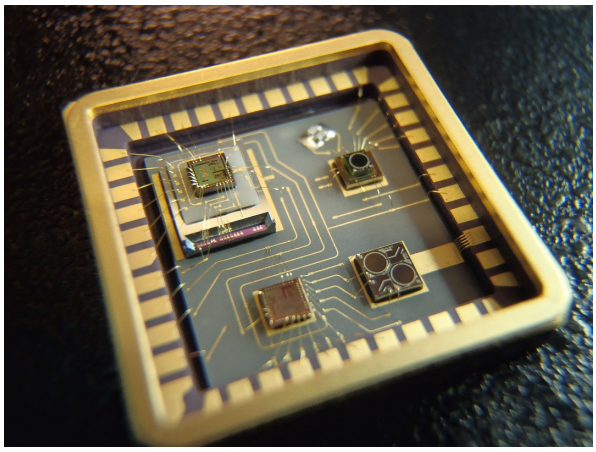
With the advent of large scale multi-processor chips as well as Systems-on-Chip (SoCs) that allow heterogeneous integration of various functionalities, the problem of global interconnect delays has become increasingly important. The average length of the longest global interconnect in a chip of area A is approximated to be $\sqrt{A}/2$, and the delay of global interconnects to be $\propto A/\lambda^2$, where λ is the feature size [20]. Thus, as feature sizes

shrink and chip areas grow, there is an explosion in the delay of global interconnects. The problem of global interconnect delays not scaling, increased manufacturing variances and yield degradation of large chips led to the concept of integrating several smaller die on a common substrate. The underlying substrate and integration technology used for multi-die integration has also evolved drastically over the last 30 years. What started with Multi-Chip Modules (MCMs) (Fig. 1.2a)in the 1990s, led to System-in-Package (SiP) (Fig. 1.2b) modules in the 2000s and 2.5D (Fig. 1.2c) and 3D (Fig. 1.2d) integrated circuits in the present day. These integration techniques have become increasingly efficient in terms of the connection pitch, bandwidth limits and latency. Current 2.5D integrated circuits use fine-pitched silicon interposers as the substrate, and bond each die using fine pitched microbumps with pitches ranging between $20\mu m - 50\mu m$ [21].

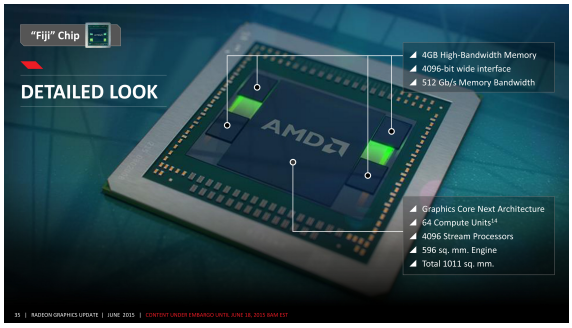
Network-on-Chip (NoC) architectures have emerged as a promising interconnection fabric for such integration techniques [22], as opposed to bus based communication. While early research on NoCs goes back to the 2000s [23, 24, 25], their applicability has become even more relevant with the advent of advanced integration techniques. As shown in Fig.1.3, NoC architectures include point-to-point links between router modules, making the interconnection network scalable. Router modules communicate with their neighboring compute modules and essentially decouple the task of designing communication and computation circuits. A natural question that arises is the mechanism of timing coherence of the modules and the most efficient intra-module and inter-module communication strategies. Fully synchronous communication schemes are generally ruled-out as distribution of global low-jitter clock signals across large areas in not only expensive in terms of power, but manufacturing variances and dynamic changes in the operating environment of the circuits limit the maximum performance of such a scheme. Globally Asynchronous Locally Synchronous (GALS) schemes have shown promise [26, 27], where small digital blocks operate synchronously, but the operation of different blocks is not mutually



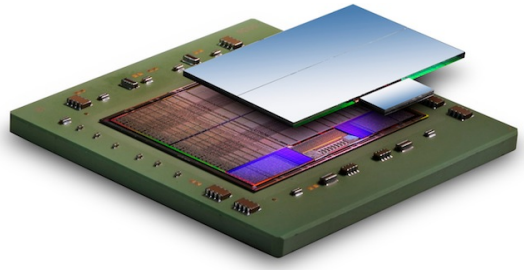
(a) Multi-chip module. Source: [2]



(b) Silicon-in-package. Source: [3]



(c) AMD Radeon^{TM} Fury graphics card with the "Fiji" GPU and High Bandwidth Memory. Source: [4]



(d) Xilinx Virtex-7 H580T 3D FPGA based on stacked silicon interconnect. Source: [5]

Figure 1.2: Evolution of integration techniques

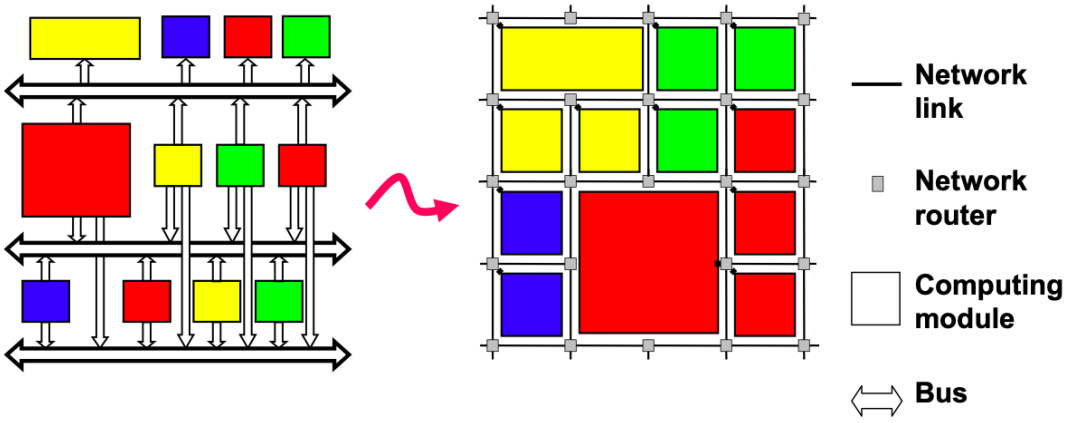


Figure 1.3: Comparison of bused-based and NoC based communication. Source: [6]

synchronized [28]. Often, an asynchronous wrapper acts as an interface between the synchronous and asynchronous environments, two common solutions being using FIFO buffers and pausible clock generators [29]. Thus, small-scale local synchronous modules with controlled variances are able to a achieve a high performance at relatively low power. The global communication incurs a small overhead due to these asynchronous wrappers, but the performance of local blocks is no longer bound by the tolerances necessary to ensure correct functionality given global static and dynamic variations. In fact, all the examples of systems involving timing circuits presented so far need customized solutions based on the physical scale of the problem being solved and the target specifications of the system.

The work presented in this thesis strives to provide an analytical framework and specialized circuit solutions to timing problems that are applicable to systems of largely varying physical scales. These non-standard solutions provide higher performance alternatives to the issues of multi-phase clock distribution and larger-scale on-chip or on-interposer communication and coherence. These solutions include (i) models, tools and circuits for precision timing generation in CMOS technology, (ii) techniques for building higher-order interconnected clock structures that use distributed feedback to improve the ensemble stability of the structure, and can be utilized as a low-noise timing source for small-scale synchronous modules, (iii) transmission-line stabilized low-noise wave oscillator circuits providing non-overlapping clock phases for RF applications as well as chip-scale timing distribution, (iv) transition signaling strategies that achieve very high-bandwidth communication with instantaneous recovery at low power, between globally asynchronous on-chip or on-interposer modules and (v) methods of implementing low cost time-domain filters that utilize precision clocks to reduce timing noise originating from systematic noise sources.

The underlying circuits for all these solutions are pulse mode and built using a logic

family of pulse gates [30]. Pulse logic is self-resetting and uses atomic pulses to encode both, data and the time of arrival of data. These gates utilize local negative feedback loops that make the pulse shape largely invariant and independent of the shape of the input pulse, and also lead to interesting dynamic behavior that is exploited in the design of closed loop timing circuits. Pulse logic has been utilized for the design of several high-performance asynchronous circuits including precision pipelines [31], Intel's Pentium 4 ALU [32] and radiation-hard serial links [33]. In order to better understand the anomalous stability of completely self-timed serial links designed for the Large Hadron Collider at CERN [34], simple looping pulse gate circuits or pulse oscillators were first studied in [35], and were found to exhibit superior figures of merit than other similarly powered and sized ring oscillators. The dynamics leading to the stability were evaluated and Allan deviation [36] was proposed to be a powerful metric to analyze the temporal properties of these oscillators.

Based on the temporal dynamics of pulse gates described in [35], the first part of this thesis presents an analytical model describing the behavior of simple pulse gate loops, termed Collective Pulse Oscillators or CPOs in the presence of impulse and power-law noise. The analytical model is extended to a tool that enables rapid behavioral analysis of the noise properties of CPOs and helps establish a general mathematical relationship between CPO parameters, power usage and the phase noise of the oscillator. Comparisons drawn with inverter based ring oscillators show that CPOs are more resilient to impulse noise, uncorrelated device noise as well as power-coupled noise. Measurement results for CPOs of different diameters fabricated in IBM 130 nm technology are also presented that validate the analytical noise model as well as behavioral/Hspice simulation results. Other advantages of CPOs over ring oscillators are that they can be constructed with either even or odd number of pulse buffers, thus providing even or odd number of phases. Further, they can be operated in multiple modes by injecting different numbers of pulses

at start-up, providing precise phases with resolution smaller than the buffer delay if the number of pulses does not divide the number of phase taps. In particular, operation at high frequencies is supported independent of the number of available phase outputs. It is also possible to design CPOs that provide identically timed image clock phases at multiple physical locations. Multi-pulse CPOs provide a mechanism to improve frequency stability and phase noise without increasing the total power density.

Two closely related works based on similar closed loop dynamics are by Fairbanks et al.[37], who observed that asynchronous tokens fired into a ring led to uniform distribution of the tokens around the ring, and Self Timed Ring Oscillators (STROs) [38], constructed using Muller-C gates [39], that utilized gate dynamics to yield integrated phase noise improvements. To the best of our knowledge, this is the first work that presents detailed time-domain analysis of the behavior of oscillators that exploit local time-variant gate delays to achieve precise phase resolutions and also improve the global frequency stability. The analysis is strongly supported by simulation and measurement results.

The second part of this thesis presents CPO structures that are more complex than simple loops of pulse gates. First, CPOs that make use of additional interconnections in order to improve the high-frequency noise properties of looping CPOs are studied, and interconnection strategies that scale with the size of the CPO are presented. These structures essentially make use of distributed timing feedback over short cross-links to improve the stability of strongly coupled high mass CPO rings and would serve as an excellent timing distribution solution over small-to-medium scale synchronous islands. Second, architecture, design and measurements for a transmission line stabilized pulsed rotary wave [40] CPO in IBM 130 nm are presented. This design utilizes pulse buffers as non-linear, full-swing, limiting amplifier stages to compensate for transmission line loss and maintain the wave shape of a unidirectional, low-duty cycle, fast-slope traveling pulse. It achieves further lower noise as it is coupled to a transmission line resonator and is an

ideal candidate for accurately capturing event timing at low injected jitter. Applications requiring timing distribution, or non-overlapping, low duty-cycle multi-phase clocks can effectively utilize this design.

The third part of this thesis presents Multi Wire Phase Encoding (MWPE), a signaling strategy suitable for on-chip/on-interposer links connecting globally asynchronous modules. These links are aimed at reducing the energy cost of moving data across a large scale SoC or NoC, which is known to be orders of magnitude higher than the cost of computation [41] [42] [1]. Most widely used high-performance I/O interfaces utilize PLL or DLL based data recovery at the receiver [43], [44]. These designs not only incur higher static power consumption, but also require hundreds to tens of thousands of clock cycles to achieve a lock, increasing the link overhead, especially if the communication is burst-mode or is switched from different sources.

MWPE in contrast, encodes data in the time correlated switching of multiple signaling wires as opposed to signal levels, and thus does not require a sampling clock to recover the transmitted data. This eliminates the static power consumption incurred due to a DLL or PLL at the receiver, and also makes the link 'fire-and-forget' as the data can be recovered instantaneously. Further MWPE can transmit data at effective rates (i) higher than the bandwidth limit of a single wire, (ii) higher than the rate of other known transition signaling techniques, and (iii) comparable to or higher than parallel conventional NRZ encoded links. This part of the thesis presents the MWPE protocol, deriving its theoretical bandwidth limits and evaluating its performance on practical on-chip channels. Link circuits for a 6-wire MWPE link are implemented in 22 nm FDX technology and the link is shown to operate at overall bandwidths $> 100Gbps$ with an energy overhead of only $0.5 - 0.8pJ/bit$.

The last section of this thesis introduces the potential of pulse-gate technology and the precise timing generated by CPO clocks for implementing low-cost time-domain linear

filters. These filters are designed to minimize systematic timing jitter in the arrival time of events by mixing the arrival time with the time of occurrence of a precise clock phase. Since CPOs can generate clock phases much finer than a gate delay, the mixing can be done at a very high resolution and correct for timing deviations as small as a phase fraction. A behavioral example showing the effectiveness of this technique to cancel timing jitter induced due to distortions on a wired channel is presented, showing a 40× reduction in time deviations after filtering.

1.2 Thesis Organization

The organization of the remainder of this thesis is as following: chapter 2 introduces pulse gates and Collective Pulse Oscillators (CPOs), describes the dynamics exhibited by closed loop pulse gate circuits and construction and properties of CPOs; chapter 3 presents a time-domain analytical model based on linear difference equations that describes the temporal noise progression in CPOs when injected by an impulse of noise. Three exact solutions are presented and a second order general solution is inferred; in chapter 4, Allan deviation is reviewed as a time-domain measure of oscillator frequency stability, a behavioral simulator to analyze bigger CPO loops is presented, the general analytical solution is verified against behavioral and Hspice simulations, the performance of CPOs when subjected to power-law noise is analyzed and a comparison is drawn between the noise resilience of CPOs and inverter-based ring oscillators; chapter 5 presents measurement results for a chip fabricated in IBM 130 nm consisting of programmable 8-gate and 40-gate CPOs. These CPOs are characterized using both phase noise and Allan deviation. The measurement results closely follow the theory set forth by the analytical model presented in chapter 3; chapter 6 presents complex CPO topologies that use 2-way and $p-way$ interleaving (where p is the number of pulses) to improve the high-frequency

stability of loop-connected CPOs. Architectures that allow interleaving to be scaled to higher dimension CPOs are presented and a tetrahedron clock structure is presented that further improves the high-frequency stability. Methods for voltage tuning and injection locking of CPOs are also presented in this chapter, followed by the architecture, design and measurement results for a transmission line stabilized pulsed rotary travelling wave oscillator; chapter 7 presents Multi-Wire Phase Encoding, including derivation of theoretical bandwidth bounds, performance of the encoding on practical on-chip channels, and characterization of the link performance using circuits constructed in 22 nm FDX technology; chapter 8 introduces the realization of time-domain linear filters that utilize precise clock phases generated by CPOs to reduce the magnitute of timing noise due to systematic noise sources. An example showing reduction of timing jitter due to on-chip channel distortions is presented; chapter 9 presents conclusions and future research directions.

1.3 Permission and Attributes

The contents of chapters 2, 3, 4 and 5 have previously appeared in the IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 28, no. 5, May 2020, doi: 10.1109/TVLSI.2019.2959532, P. Mukim et al., "Design and Analysis of Collective Pulse Oscillators". Some contents of chapter 6 have been previously presented at the IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Miami, FL, USA, 2019, doi: 10.1109/ISVLSI.2019.00051, P. Mukim et al., "Distributed Pulse Rotary Traveling Wave VCO: Architecture and Design". These materials are used here under the terms of the IEEE author re-use policy allowing primary author re-use in subsequent work. Chapter 7 has been submitted to the IEEE Transactions on Very Large Scale Integration (VLSI) Systems and is currently being reviewed.

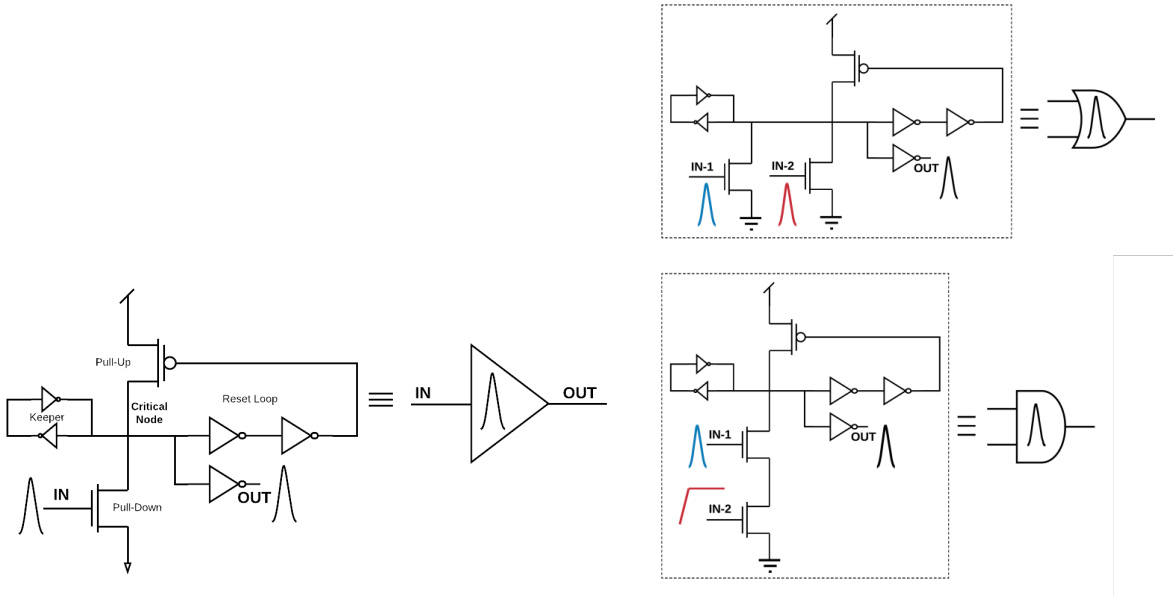
Chapter 2

Pulse Logic and Collective Pulse Oscillators

2.1 Introduction to Pulse Logic

The circuits analyzed and implemented in this work are based on a logic family of pulse gates [30], and this section provides an overview of the working of pulse gates. The simplest pulse gate, i.e., a pulse buffer is shown in Fig. 2.1a. A pulse buffer is a non-linear, shape-preserving amplifier that expects a pulse input and produces a pulse output. The critical node voltage is pulled down by the input pulse signal at IN, which pulls up the output node, OUT, and triggers the PMOS transistor to pull up the critical node, resetting the gate. The keeper loop restores charge on the critical node, and prevents it from floating when it is not actively driven by an input pulse or the resetting PMOS transistor. After a suitable delay, the input to the PMOS transistor returns to its steady (high) state. As shown in Fig. 2.1b, different logic functions such as ORing pulses, implementing guarded pulse generating by ANDing a pulse with a level and other AND/OR combinations can be realized by modifying the pull-down network of pulse

gates.

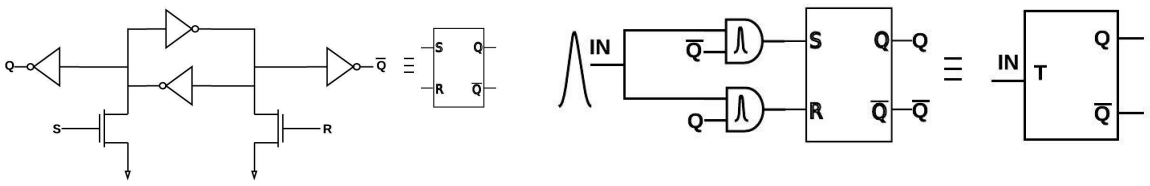


(a) Pulse buffer

(b) Logic functions realized in pull-down network of pulse gates

Figure 2.1: Pulse gate circuits

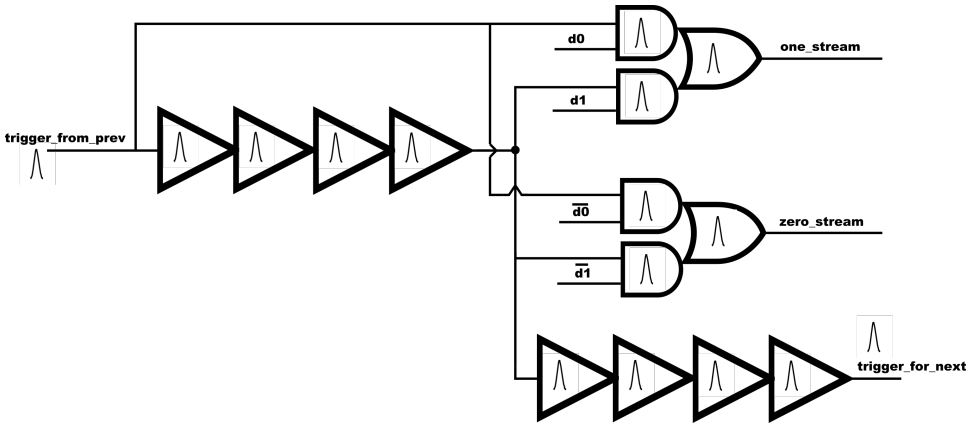
Pulse logic makes use of SR-latches to convert pulses to levels and store state. Fig. 2.2b shows an example of a pulse gate circuit where pulse gates and an SR-latch are used to implement a toggle latch. Assuming that initially the latch is reset and $Q = 0, \bar{Q} = 1$, arrival of a pulse at IN triggers the top pulse AND gate, setting the latch and making $Q = 1, \bar{Q} = 0$. Thus, every new event toggles the state of the SR-latch.



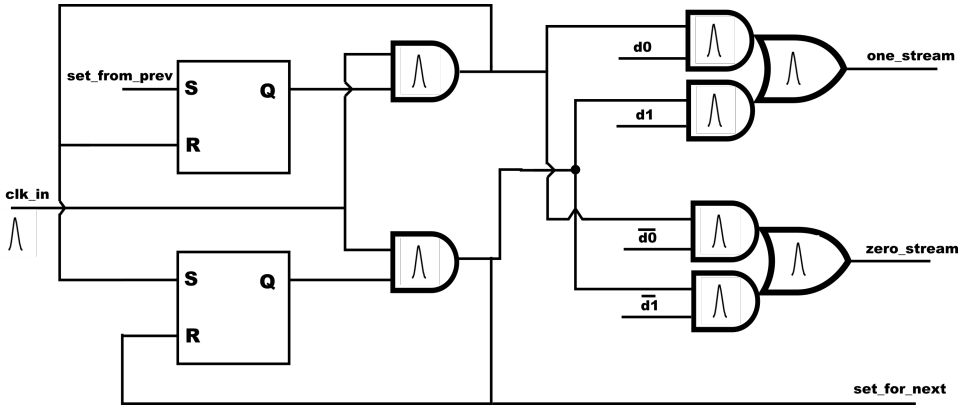
(a) SR Latch

(b) T Latch

Figure 2.2: Pulse latches



(a) Asynchronous 2-bit serializer



(b) Synchronous 2-bit serializer

Figure 2.3: 2-bit serializer implementations

A general rule followed while designing pulse gate circuits is that every parallel stack of pull-down transistors allows a single pulse input. Operations such as ANDing two pulses are prohibited. These circuits exhibit two-sided timing constraints and automated timing verification of pulse gate circuits is currently an active area of research [45]. Although pulse logic has been primarily used to implement asynchronous circuits, small synchronous blocks to be used in a globally asynchronous setting can also be easily implemented. Fig. 2.3 shows examples of 2-bit asynchronous and clocked serializers implemented using pulse gates. In either case, one-hot pulses on 2-channels represent data values as well as mark the arrival of data, but the module in the serializer of Fig. 2.3a generates a trigger signal for the next module, whereas in the serializer of Fig. 2.3b, the module communicates to the next module via an edge that it has processed the input clock pulse, and now it is the next module's turn to be evaluated on the next clock pulse. While pulse-mode signaling allows data and timing to be self-contained in the same signal, it also leads to interesting dynamic behavior in closed loop circuits or Collective Pulse Oscillators, which are described in the next section.

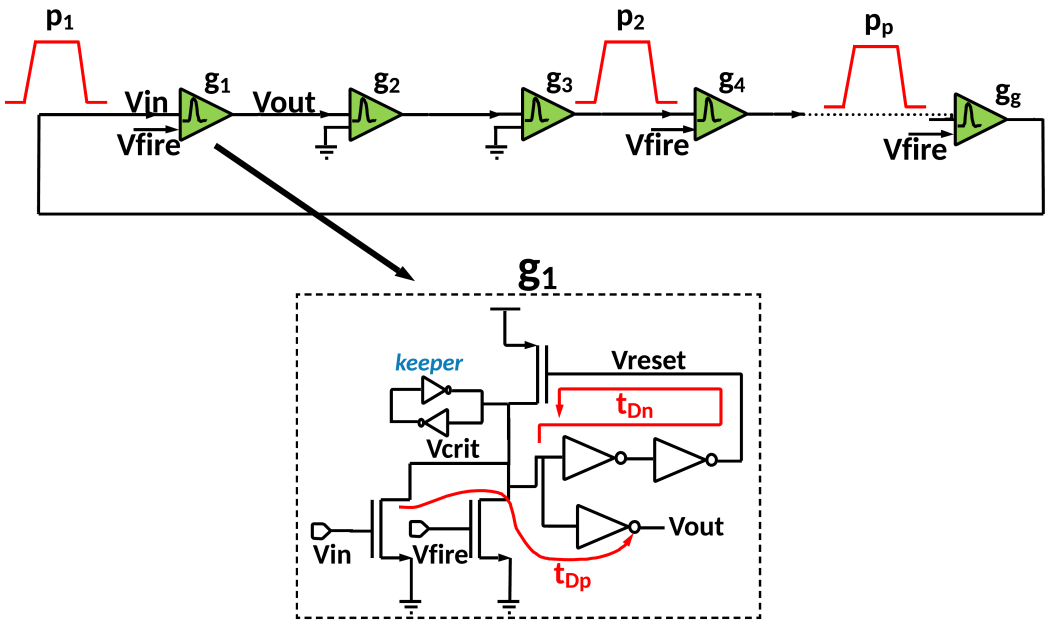
2.2 Collective Pulse Oscillators: System Description

CPOs can be used as Voltage controlled oscillators (VCOs) that are required in a broad range of digital, mixed-signal, and RF integrated circuit (IC) designs for logic timing, sampling, and frequency synthesis. Applications such as time to digital converters (TDCs)[46, 47], analog to digital converters (ADCs) [48, 49], clock and data recovery (CDR) [50] and microprocessor clocking [9] require low integrated timing noise and often utilize multiple oscillator phases. Ring oscillators, consisting of inverters or differential limiting amplifiers, are commonly used as VCOs due to their small footprint, broad tuning range and availability of multiple clock phases.

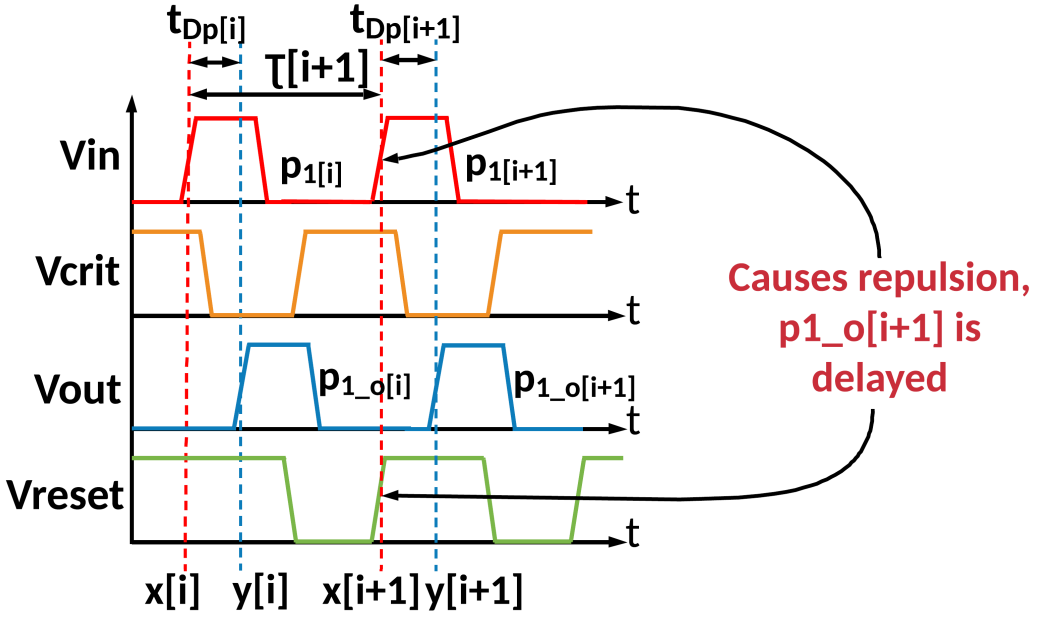
Widely-used techniques to improve phase noise of ring oscillators include transistor sizing [51, 52], jitter minimization [53], transmission line stabilization [54] and spatial-coupling [55, 56]. Phase-noise is improved by increasing the power or by coupling to a high-Q resonator. Increasing the power yields a phase noise improvement of $-10\log p$, where the oscillator power is $\propto p$. On the other hand, linear spatial coupling between p-identical oscillators yields a phase noise improvement of $-10\log p$ only near the carrier frequency [56]. At frequency-offsets far from the carrier or equivalently over small time-scales, the improvement in frequency stability is less than $\propto 1/\sqrt{p}$. This is due to the finite time associated with correction of noise perturbations in a weakly coupled system [57]. However, the possibility of achieving multiple clock phases with resolutions independent of the smallest gate delay for a given technology [55] and ease of low-skew low-jitter timing distribution [58] make their use viable for a variety of applications [59, 60, 61, 62].

For widely analyzed ring or LC oscillators, it is known that phase shifts due to noise persist indefinitely [63]. However, the behavior of CPOs is distinct in this regard. CPOs exhibit temporal degradation of phase error to a magnitude smaller than the initial injected phase error. This is achieved by partial retention of past state in the form of residual gate charge. Effectively, each gate of the CPO shows local negative timing feedback and corrects phase errors, leading to improvement of the global frequency stability. The self-correction of phase error occurs at time scales that are close to the oscillation frequency. As a result, the behavior of CPOs is similar to that of spatially-coupled oscillators with frequency stability improvement $\propto 1/\sqrt{p}$ across all time scales beyond the correction settling time.

CPOs are formed by connecting pulse gates in a loop and Fig. 2.4b shows the voltages at different nodes of a pulse buffer during pulse generation. Here, t_{Dp} is the delay of the forward path from the input to the output pulse and t_{Dn} is the delay of the reset

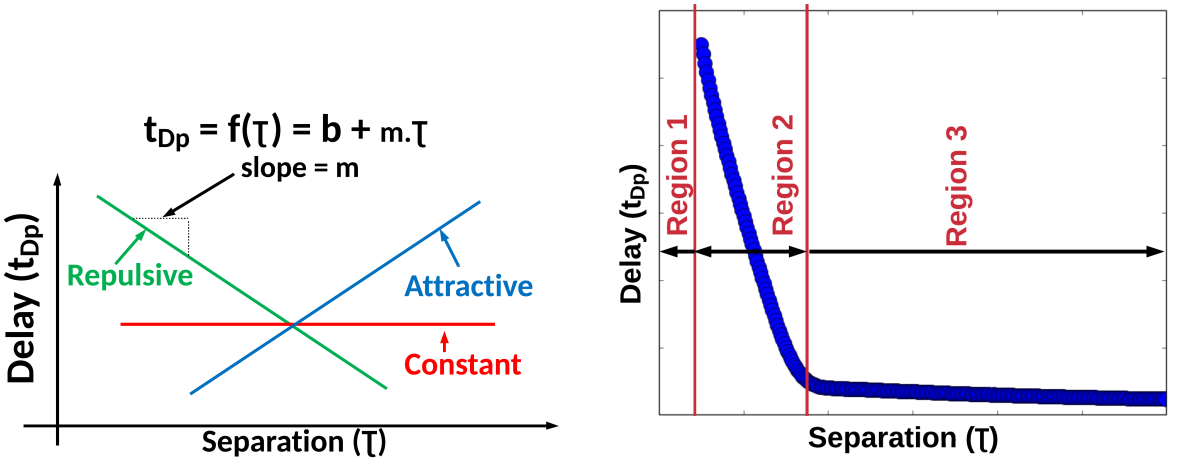


(a) CPO formed by ring of pulse gates



(b) Voltage at pulse gate nodes during pulse generation

Figure 2.4: CPO: Construction and Timing



(a) Different dynamics in amplifiers

(b) Different regions of operation for a typical pulse gate

Figure 2.5: Delay-separation dynamics

loop. The relative values of t_{Dp} and t_{Dn} lead to interesting dynamics. These dynamics are represented through a delay-separation curve that shows how the input-output gate delay (t_{Dp}) is modulated by the time separation (τ) between consecutive input pulses at that gate. Fig. 2.5a shows delay separation curves for repulsive, constant or attractive dynamics in amplifiers. Here m is the slope of the curve and b is the y-intercept. While this slope is negative in pulse gates, typical inverters have a very slightly positive slope, i.e. the delay of the gate decreases as pulses approach in time. This leads to the well-known 'settling' of ring oscillators to the lowest possible mode. Fig. 2.5b shows the delay separation curve for a typical pulse gate. This curve comprises of three regions:

- Region-1: If the separation $\tau[i+1]$ between consecutive pulses $p[i]$ and $p[i+1]$ is small such that V_{crit} is still pulled low when pulse $p[i+1]$ occurs, the pulse $p\_o[i+1]$ at V_{out} will coalesce with the pulse $p\_o[i]$. In effect, the input pulse $p[i+1]$ is rejected.

- Region-2: If the separation $\tau[i+1]$ between consecutive pulses $p[i]$ and $p[i+1]$ is

such that V_{crit} has been pulled-up substantially, but the gate is in its reset phase when pulse $p[i+1]$ occurs, i.e., V_{reset} is active (low), trying to pull-up V_{crit}, a distinct pulse $p\_o[i+1]$ at V_{out} will be generated. However, with both the pull-down and pull-up transistors active, the time to discharge V_{crit} increases, increasing the delay of the gate, effectively causing repulsion of $p\_o[i+1]$. This behavior leads to a high-slope delay-separation region.

- Region-3: If the separation $\tau[i+1]$ between consecutive pulses $p[i]$ and $p[i+1]$ is such that the reset phase has concluded when $p[i+1]$ occurs, the stored charge in the keeper hysteresis loop slightly delays the pulse $p\_o[i+1]$ at V_{out}. This leads to a low-slope delay-separation region.

By changing the relative delays of a pulse gate's forward and feedback paths (t_{Dp} and t_{Dn}), both b and m can be tuned. Methods for this are described in [64]. When pulse gates are connected in a loop as is the case with CPOs, pulses distribute uniformly in time around the loop due to the repulsive dynamics of the gates. Hence, a precise phase tap is available at each CPO gate. A CPO ring can be constructed with either even or odd number of pulse gates, making it possible to generate both even or odd number of clock phases. A g-gate CPO operating with p-pulses where g/p is an integer will exhibit g/p different phases, with each phase available at p gates. If g and p are relatively prime, each gate will produce a unique phase with period $(g/p) * t_{Dp}$ and phase resolution t_{Dp}/p. Thus a CPO can easily produce pulses with phase resolution finer than t_{Dp}.

For a CPO that has p pulses traveling around the loop, the oscillation frequency depends on the ratio of gates per pulse (g/p) and t_{Dp}, and is equal to $(g/p) * t_{Dp}$. The ratio g/p also sets the region of operation on the delay-separation curve. By increasing the number of pulses p while keeping the g/p ratio a constant, the rate of pulse arrival at each gate stays unchanged and the oscillator operates at the same frequency. However,

the total power increases $\propto p$, and the effective frequency stability is enhanced. The duty cycle of the oscillator is determined by the ratio of the width of pulses (set by the delay of the reset loop, t_{Dn}) and the oscillator period.

Since it is possible for a g gate CPO to run in multiple modes set by the number of pulses p, it is crucial to ensure a reliable start-up in the desired mode. The V_{fire} input of the pulse gate is used to inject a start-up pulse into the CPO. Starting a mode with p pulses can be done in two ways: (1) p pulses can be injected sequentially into the V_{fire} node of a single pulse gate with relative separations close to the expected period of the CPO, (2) the V_{fire} nodes of p uniformly-spaced gates can be injected with a pulse simultaneously.

As long as the separation of start-up pulses is large enough to avoid pulse coalescence, the CPO will start-up reliably. If the pulse injection period at start-up does not exactly match the period of a stably operating CPO or there is a skew between the fire inputs to the different gates, the repulsive dynamics in pulse gates will still lead to uniform distribution of pulses, and the CPO will rapidly settle to a stable state. Fig. 2.6 shows the start-up circuit and output waveforms for a $g = 8, p = 2$ CPO. The fire pulse is generated by converting an external edge to a pulse, and is injected into gates 1 and 5 of the CPO. Fig. 2.7 shows the period of pulse arrival obtained from Hspice simulations for a $g = 8, p = 2$ CPO in GFUS8RF technology at different gates after start-up. Since at start-up the V_{crit} nodes of the CPO do not store any residual charge, the gate delay and period of the CPO are different from the stable period. Once oscillations are sustained, the pulse arrival period settles to a stable value at all gates in a few cycles.

Widely used models for the design of oscillators like the Linear Time-Variant (LTV) model using Impulse Sensitivity Function (ISF) [63] and non-linear analysis techniques of stable oscillators in the presence of perturbations [65] assume that phase errors due to noise persist indefinitely, with their magnitude equal to the initial injected error. They

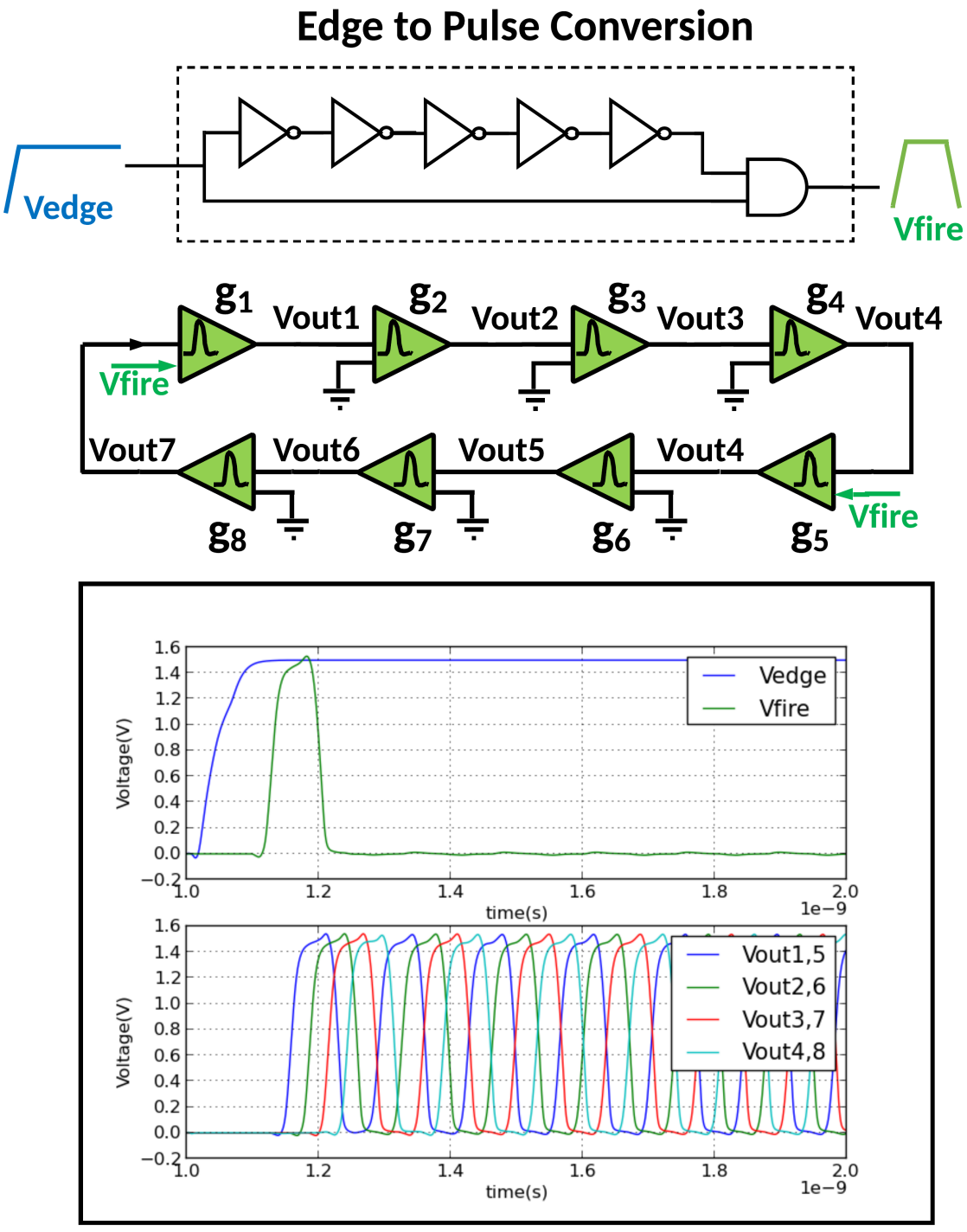


Figure 2.6: Start-up circuit and waveforms for $g = 8, p = 2$ CPO

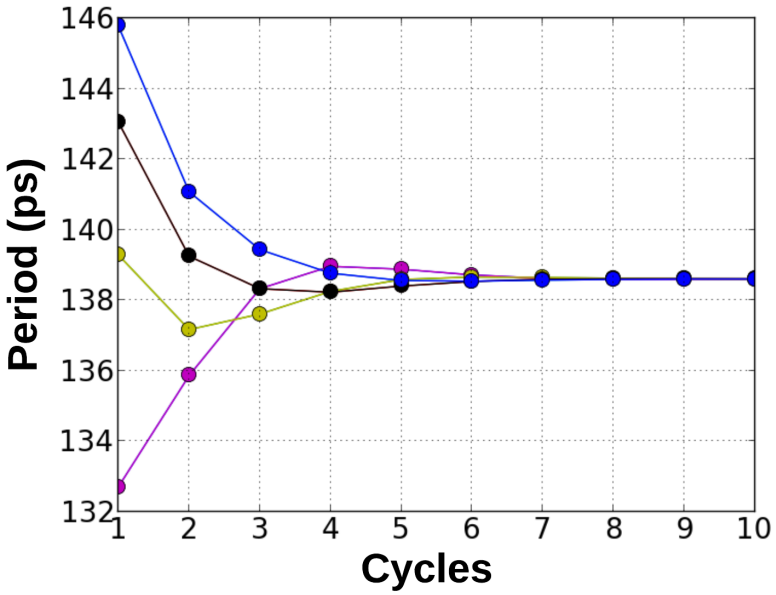


Figure 2.7: Period of pulse arrival at different gates of $g = 8, p = 2$ CPO after start-up settles to a constant value in less than 10 cycles

do not account for non-linear phase error corrections with time, a property fundamental to the operation of CPOs, and hence cannot be directly applied to analyze CPOs. To overcome this challenge, a new sequential time delay model is formulated and presented in chapter 3. It explains the noise properties of CPOs as a function of various oscillator parameters.

Chapter 3

Analytical Noise Modeling for CPOs

This chapter presents time-domain analytic solutions for the pulse arrival time for CPOs perturbed by a noise impulse. Although a bilinear approximation for the delay-separation curve is more accurate (Fig. 2.5b) for CPOs, the model assumes small perturbations for which the CPO operates in either one of the linear regions (Region 2 or Region 3). Analytic solutions are obtained by modeling the arrival time of pulses at each pulse gate via a set of linear difference equations. The goal is to derive exact solutions for a few lower order loops, that can be used to construct a general perturbation solution.

3.1 Exact Solutions for Pulse Arrival Time Under Impulse Noise Perturbation

3.1.1 $Gates(g) = 1$, $Pulses(p) = 1$

Consider a single gate CPO with the output of a pulse buffer looping back to its input, that has been started by an external event. Let x[k] be the arrival time of the k^{th} pulse at the gate. Then x[k] is given by Equation 3.1, where b and m are constants.

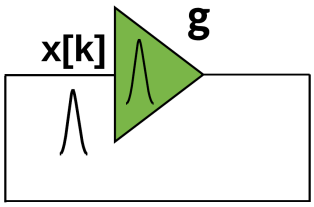


Figure 3.1: 1-pulse travelling in a 1-pulse-gate loop

$$x[k] = x[k-1] + b + m(x[k-1] - x[k-2]) \tag{3.1}$$

The arrival time of a pulse at the gate is the sum of the arrival time of the previous pulse and the delay of the gate. The delay here is represented as a linear function of the separation, that is set by the arrival times of the previous two pulses. In the case of ideal (noise-less) operation, the period $(x[k] - x[k-1] = x[k-1] - x[k-2])$ will be a constant given by $b/(1-m)$. If an impulse of timing error of magnitude ϵ is introduced into the system at $k = 1$, the initial conditions are given by Equation 3.2.

$$x[0] = 0, x[1] = \epsilon + \frac{b}{1-m} \tag{3.2}$$

Solving this linear difference equation, the k^{th} pulse arrival time is given by Equation 3.3.

$$x[k] = \frac{bk}{1-m} + \frac{\epsilon}{1-m} - \frac{\epsilon m^k}{1-m} \tag{3.3}$$

The solution consists of three distinct terms:

- The first term represents the ideal pulse arrival time, if no noise were introduced into the system.
- For $m < 0$ (the case for CPOs), the second term represents the residual phase error

after a sufficiently long interval. It can be observed that an injected impulse error of ϵ is reduced in magnitude to a residual error of $\epsilon/(1-m)$.

- For $m < 0$ and $|m| < 1$ (again the case for CPOs), the third term represents a rapidly diminishing transient that reduces the magnitude of the phase error to its final residual value. However, it can be seen that if $m > 0$ or if $|m| > 1$, the phase error can grow with time, making the oscillator unstable.

Since $m > 0$ or $|m| > 1$ have been shown to make the oscillator unstable, the subsequent analysis excludes these cases and is strictly for CPOs with $|m| < 1$. Next, we will evaluate the case of a single pulse travelling in a loop comprising of two pulse gates.

3.1.2 $Gates(g) = 2$, $Pulses(p) = 1$

Let x[k] and y[k] be the arrival times of the k^{th} pulses at gates 1 and 2 respectively. Then x[k] and y[k] are given by Equation 3.4 and Equation 3.5, where b and m are constants.

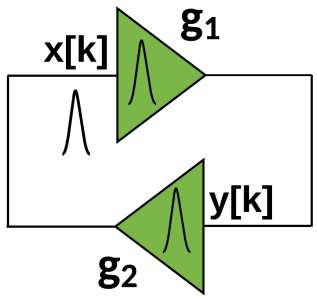


Figure 3.2: 1-pulse travelling in a 2-pulse-gate loop

$$y[k] = x[k] + b + m(x[k] - x[k-1]) \tag{3.4}$$

$$x[k] = y[k-1] + b + m(y[k-1] - y[k-2]) \tag{3.5}$$

In the case of ideal (noise-less) operation, the pulse arrival periods at the two gates ($x[k] - x[k-1] = y[k] - y[k-1] = y[k-1] - y[k-2]$) will be constants given by $2b/(1-2m)$, with each gate contributing to half of the delay. If an impulse of timing error of magnitude ϵ is introduced when a pulse arrives at gate 2 at $k = 1$, the initial conditions are given by Equation 3.6.

$$x[0] = 0, y[0] = \frac{b}{1-2m}, x[1] = \frac{2b}{1-2m}, y[1] = \epsilon + \frac{3b}{1-2m} \tag{3.6}$$

Solving this system of linear difference equations, the k^{th} pulse arrival times are given by Equation 3.7.

$$x[k] = \frac{2bk}{1-2m} + \frac{\epsilon}{1-2m} - \frac{\epsilon * 2^{-k}}{2(1-2m)} * o[k] \tag{3.7}$$

where $o[k]$ is given by Equation 3.8.

$$\begin{aligned} o[k] = \frac{1}{\sqrt{m(m+4)}} \times [(m(m+2-\sqrt{m(m+4)}))^k + (m(m-2+\sqrt{m(m+4)}))^k \\ + (m-2)((m(m+2-\sqrt{m(m+4)}))^k - (m(m+2+\sqrt{m(m+4)}))^k)] \end{aligned} \tag{3.8}$$

For CPOs with $m < 0$ and $|m| < 1$, this loop operates at a period that is slightly smaller than twice the period of the previous case, where $g = 1, p = 1$. The magnitude of the residual phase error is further reduced compared to the $g = 1, p = 1$ case. The transient term however, is a fairly complex function of time and the delay separation slope m as is expected of a 3^{rd} order system. This transient term represents the settling trajectory (and time) of the loop to a stable state, in response to the impulse injection.

3.1.3 $Gates(g) = 2$, $Pulses(p) = 2$

Let x[k] and y[k] be the arrival times of the k^{th} pulses at gates 1 and 2 respectively. Then x[k] and y[k] are given by Equation 3.9 and Equation 3.10, where b and m are constants.

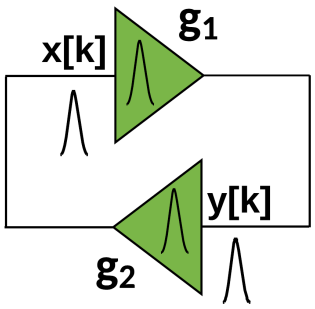


Figure 3.3: 2-pulses travelling in a 2-pulse-gate loop

$$y[k] = x[k-1] + b + m(x[k] - x[k-1]) \tag{3.9}$$

$$x[k] = y[k-1] + b + m(y[k] - y[k-1]) \tag{3.10}$$

In the case of ideal (noise-less) operation, the pulse arrival periods at the two gates ($x[k] - x[k-1] = y[k] - y[k-1]$) will be constants given by $b/(1-m)$. If an impulse of timing error of magnitude ϵ is introduced when a pulse arrives at gate 2 at $k = 1$, the initial conditions are given by Equation 3.11.

$$x[0] = 0, y[0] = 0, x[1] = \frac{b}{1-m}, y[1] = \epsilon + \frac{b}{1-m} \tag{3.11}$$

Solving this system of linear difference equations, the k^{th} pulse arrival times are given

by Equation 3.12 and Equation 3.13.

$$x[k] = \frac{bk}{1-m} + \frac{\epsilon}{2(1-m)} - \frac{\epsilon * 2^{-k} m^k}{2(1-m)} * o_x[k] \tag{3.12}$$

$$y[k] = \frac{bk}{1-m} + \frac{\epsilon}{2(1-m)} - \frac{\epsilon * 2^{-k} (-m)^k}{2(1-m)} * o_y[k] \tag{3.13}$$

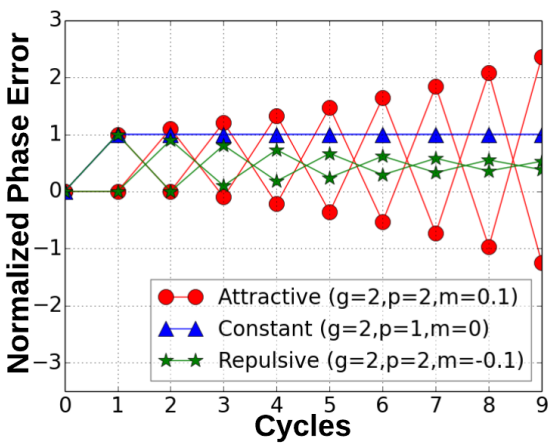
where $o_x[k]$ and $o_y[k]$ are given by Equation 3.14 and Equation 3.15 respectively.

$$o_x[n] = \frac{1}{\sqrt{1+m(m+6)}} \times ((-m-1-\sqrt{1+m(m+6)})^n - (-m-1+\sqrt{1+m(m+6)})^n) \tag{3.14}$$

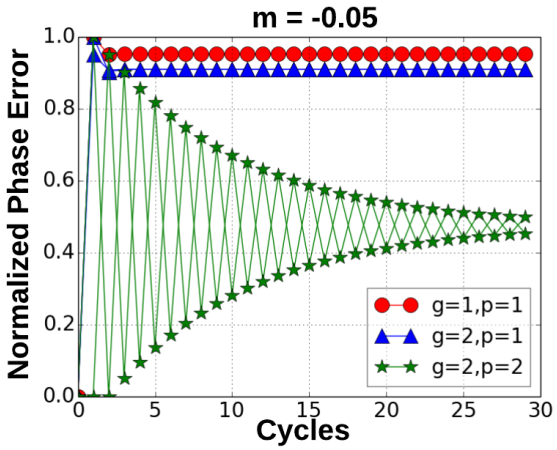
$$o_y[n] = -o_x[n] \tag{3.15}$$

The first term of the solution in this case exactly matches the first term of the case where $g = 1, p = 1$. This also matches the intuitive expectation, as fixing the g/p ratio and changing p should result in the same oscillation frequency. The magnitude of the residual phase error (the second term in Equation 3.12 and Equation 3.13) is exactly half, compared to the $g = 1, p = 1$ case. This illustrates that by doubling the effective mass of the ring (and its power consumption), while keeping its oscillation frequency constant, the magnitude of the residual phase error is made twice as small. Once again, the transient terms for the two gates are fairly complex, but also show a symmetry. The two gates act in conjunction such that their respective phase errors symmetrically approach the final residual phase error in the loop.

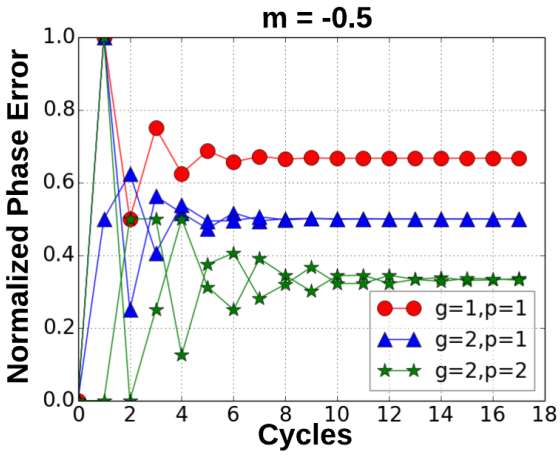
Based on the three exact solutions derived so far, the normalized phase error (obtained by subtracting the ideal noise-less arrival time from the derived solutions and normalizing it with respect to the magnitude of the injected impulse ϵ) is plotted in Fig 3.4. The following observations can be made from these plots:



(a) Different delay-separation dynamics



(b) Weakly repulsive dynamics



(c) Strongly repulsive dynamics

Figure 3.4: Normalized phase error obtained from analytical solutions

- Fig. 3.4a compares the normalized phase error for oscillators exhibiting attractive ($m > 0$), constant ($m = 0$) and repulsive ($m < 0$) dynamics. It can be seen that constant dynamics result in a constant phase error of magnitude equal to the impulse noise injection, as is the case with conventional oscillators. To model a conventional ring oscillator, the $p = 1$ mode was chosen, as inverter-based ring oscillators only involve circulation of a single event around the ring. The phase error for attractive dynamics increases in magnitude, making the oscillator unstable. With repulsive dynamics, the oscillator settles to a phase error smaller in magnitude than the initial injected impulse.

- Increasing the number of gates (g), number of pulses (p) or the magnitude of the (negative) delay-separation slope (m), reduces the magnitude of the residual phase error.

- The settling time, i.e., the time taken by the transient term to diminish in magnitude, is a complex function of the loop topology and operating slope. While it can be inferred that a multi-pulse ring tends to have a longer settling time, increasing the magnitude of the operating slope can increase or decrease the settling time, as can be seen in Fig. 3.4b and Fig. 3.4c. For the $g = 1, p = 1$ and $g = 2, p = 1$ CPOs increasing the magnitude of m from -0.05 to -0.5 increases the settling time. Whereas for the $g = 2, p = 2$ CPO, it causes the settling time to decrease.

3.2 Generalization of the solution to g-gates, p-pulses

Obtaining exact solutions for higher-order loops (order $>$ 3) is difficult. However, based on the three exact solutions derived, the fixed (noise-less) pulse arrival time and magnitude of the residual error can be inferred to have forms shown in Equation 3.16.

$$x[k] = \frac{gbk}{p(1-(g/p)m)} + \frac{\epsilon}{p(1-(g/p)m)} + \epsilon * transient_x[k,g,p,m] \tag{3.16}$$

$$or, x[k] = T_o k + \Phi[k] \tag{3.17}$$

The general solution of Equation 3.16 is re-written in Equation 3.17 as the sum of: (i) the nominal arrival time (with T_o as the nominal period of the CPO), represented by the first term of Equation 3.16 and (ii) phase deviation $\Phi[k]$, represented by the sum of second and third terms of Equation 3.16. The correctness of this general solution has been verified against both behavioral and Hspice simulations presented in chapter 4. The general analytic solution shows that:

- For a fixed number of gates g and (negative) m, CPOs operating in different modes set by the number of pulses p, see a reduction in period that is less than $\propto p$.
- The magnitude of both the residual phase error and the transient term is directly proportional to the magnitude of the injected noise impulse. This makes the settling time independent of the magnitude of noise injection.
- Increasing the number of gates g, the number of pulses p or the magnitude of the (negative) delay-separation slope m, reduces the magnitude of the residual phase error.
- For a fixed oscillation frequency (obtained by having g/p and m constant), the residual phase error reduction for an noise impulse is proportional to the number of pulses p, and hence the total power consumption. Qualitatively, this shows that for a loop, each gate's noise injection is scaled by the number of pulses which react as an ensemble to reduce the magnitude of the injected error.

These analytic solutions are based on impulse noise injection into a single CPO gate. To analyze the frequency stability with continuous power-law noise injection, Allan deviation, a time domain stability measurement metric is used and over-viewed in chapter 4.

Chapter 4

Behavioral and Hspice Simulation Results for Simple Loop Connected CPOs

This chapter first provides an overview of Allan deviation [66], a time-domain frequency stability metric for oscillators. Next, the analytic model presented in chapter 3 is expanded into a generic behavioral tool, enabling the analysis of more complex loops with arbitrary imposed noise. The behavioral model is validated against Hspice simulations and the effect of power-law noise on different parameter CPOs is analyzed. A comparison between CPOs and ring oscillators in terms of their response to impulse noise, uncorrelated white device noise and power coupled noise is also drawn.

4.1 Time Domain Measurement of Stability Using Allan Deviation

We use Allan deviation as the analysis metric as it a very useful aid in measuring the stability of CPOs at different time scales. It characterizes the fractional frequency fluctuations ($F[k]$) in oscillators given by Equation 4.1 where $\Phi[k]$ is the phase deviation (in seconds) as a discrete function of time and τ is the measurement interval.

$$F[k] = \frac{\Phi[k+\tau] - \Phi[k]}{\tau} \tag{4.1}$$

Noise in circuits generally exhibits a power law given by $S_F(f) \propto f^{\alpha}$, where $S_F(f)$ is the autospectral density of fractional frequency fluctuations $F[k]$ and the exponent α ranges from -3 to +2 typically. The well-analyzed noise sources in circuits, white FM and flicker FM, have α of 0 and -1 respectively. Allan deviation is the same as the ordinary standard deviation of fractional frequency fluctuation values for white FM noise, but has the advantage, for more divergent noise types such as flicker noise, of converging to a value that is independent on the number of samples [66]. Allan deviation is given by the square root of $\sigma^2(\tau)$ in Equation 4.2, where Φ_i is the i^{th} phase error value spaced by the measurement interval τ and N is the number of samples of phase error values averaged over τ.

$$\sigma^2(\tau) = \frac{1}{2(N-2)\tau^2} \sum_{i=1}^{N-2} [\Phi_{i+2} - 2\Phi_{i+1} + \Phi_i]^2 \tag{4.2}$$

Overlapping Allan deviation is a variant of the original Allan deviation that provides better statistical confidence [66]. Modified Allan deviation given by the square-root of Equation 4.3 is another variant that can additionally distinguish between noise behaviors having $\alpha \geq 1$ [36]. Here, the measurement interval $\tau = a\tau_o$, a is the averaging factor and

τ_o is the basic measurement interval.

$$Mod\,\sigma^2(\tau) = \frac{1}{2a^2\tau^2(n-3a+1)} \times \sum_{j=1}^{N-3a+1} \{ \sum_{i=j}^{j+a-1} [\Phi_{i+2a} - 2\Phi_{i+a} + \Phi_i] \}^2 \tag{4.3}$$

Different frequency domain noise profiles can be identified by measuring the slope of modified Allan deviation ($Mod\sigma^2(\tau)$) on a log-log scale. White FM and Flicker FM have a slope of -1/2 and 0 respectively. Results in this chapter use modified Allan deviation to characterize the frequency stability of CPOs at different time scales. The dynamic phase error correction properties and the settling time associated with them are represented on modified Allan deviation plots by an initial high-slope ($< -1/2$) region. For comparing the jitter of CPOs operating at different frequencies, the Allan deviation values are scaled by the oscillation period (T_o) and plotted as a function of number of clock cycles. This metric is termed "Jitter Stability" in this work and the k-cycle Jitter Stability, $J[k]$ is defined as follows:

$$Jitter\ Stability, J[k] = Mod\,\sigma(kT_o) * T_o \tag{4.4}$$

4.2 Behavioral System Simulator

The aim of building the behavioral simulator was to (i) verify the residual phase error expression in the general analytic solution of CPOs (Equation 3.16) when injected by impulse noise, (ii) analyze settling time behaviors of higher-order loops and, (iii) analyze the dependence of design parameters on frequency stability of CPOs under continuous

power-law noise injection. Fundamentally, the simulator is built upon the same abstract model as the analytic solutions. The details of the behavioral simulator are summarized as following:

- Pulse gates are abstracted as nodes that compute the arrival time of the output pulse based on the arrival time of the input pulse and the gate delay. The gate delay is modeled by linear and repulsive ($m < 0$) delay-separation dynamics and is given by:

$$t_{Dp} = b + m\tau \tag{4.5}$$

- Pulses are assumed to be traversing a simple loop of pulse gates as shown in Fig. 2.4.

- Pulse arrival times are the state variables initialized for the first two cycles, and then updated sequentially.

- Each gate can be injected by noise when a pulse arrives at its input. The noise injection is modeled by a perturbation of the pulse arrival time. This perturbation can either be an impulse, where only one gate is injected by noise at a given pulse arrival, or can be modeled as injection of white FM and/or flicker FM noise sources at each gate and at each pulse arrival.

- For a g-gate CPO, the maximum number of pulses p equals the number of gates g.

The functional flow of the simulator is described as following:

1. Given the number of gates g, number of pulses p and the magnitude of the delay separation slope m, a CPO loop is constructed and initialized. p (preferable equidistant) gates are chosen and their first pulse arrival times (iteration $i = 0$) are initialized to 0. The first pulse arrival times for the remaining gates are initialized

as well, based on the expected gate delays for the given specification of g, p and m. Iteration count i is incremented.

2. The pulse arrival times for the p gates chosen for first initialization in step 1 are initialized (if $i = 1$) or calculated (if $i > 1$). If $i = 1$, the initialization phase is completed, as the delays of these p gates can now be calculated based on the initialized input pulse arrival times for the first two iterations. Henceforth, pulse arrival times at all other gates can be calculated. It should be noted that even if the initialized arrival times are different from that of a stably operating (noise-less) loop, the repulsive dynamics will eventually distribute the pulses evenly around the loop. The subsequent noise analysis should be made after the loop is in its stable state.

3. The pulse arrival times for each of the next p gates (fed from the p gates chosen previously) are calculated for the current iteration.

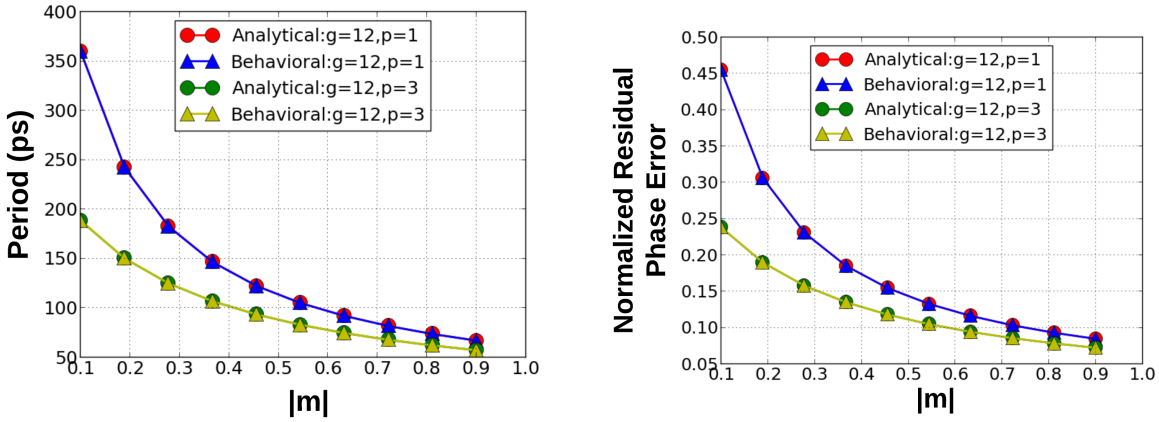
4. If the analysis being done is for an impulse-noise, any one of the arrival times calculated in the previous step is perturbed by a finite amount. This is done only once (in the second iteration, i.e. when $i = 1$). If the analysis is being done for continuous noise-injection, all the arrival times calculated in the previous step are perturbed by a finite amount. The perturbation values are based on the distribution of the noise-type whose effects are being analyzed.

5. Step 2 and Step 3 (in case of continuous noise injection) are repeated until the arrival times for all gates for the current iteration are updated.

6. i is incremented and Steps 2-5 are repeated for a desired iteration count I, until $i \geq I$.

4.2.1 Impulse-Noise Analysis

CPOs with varying parameters (g, p and m) were constructed and pulse arrival times under ideal (noiseless) operation were determined. Their behavior was then analyzed with impulse-noise injection. The ideal operating period (T_o) and residual phase error exactly match the first and second terms of the general analytic solution of Equation 3.16. This comparison is shown in Fig. 4.1. Since the general analytic solution does not predict the transient term, the behavioral simulator is utilized to analyze the transient behavior in terms of the settling time. The settling time is obtained by computing the time taken for the phase error of all gates of the CPO to reach 1% of the final residual phase error after injection of a noise-impulse.

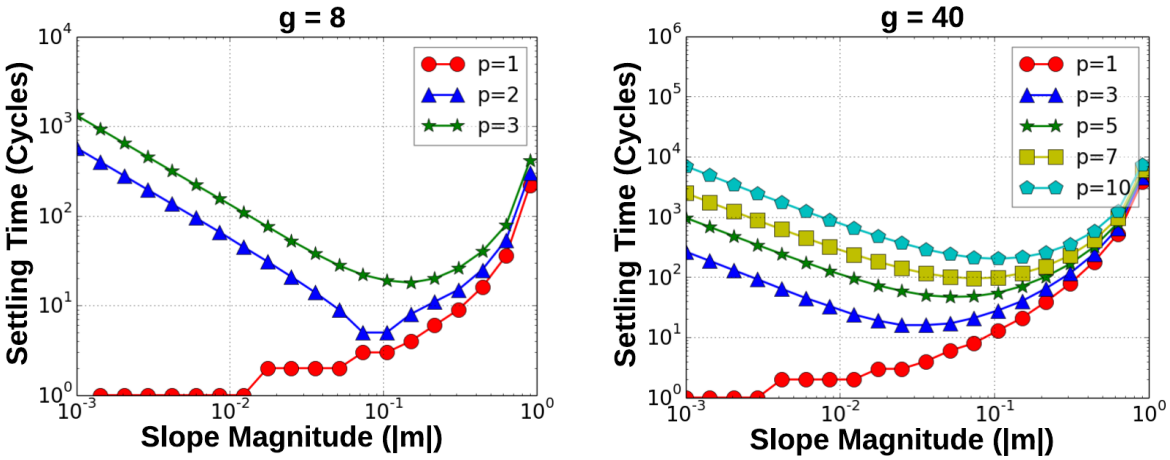


(a) Comparison of T_o

(b) Comparison of residual phase error

Figure 4.1: Comparison of values obtained from the general analytic solution and behavioral simulations

Settling times as a function of $|m|$ for 8-gate and 40-gate CPOs ($m < 0$) running in different modes are shown in Fig. 4.2a and Fig.4.2b. These plots indicate that for $p = 1$ CPOs, larger values of $|m|$ (for a constant g) lead to longer settling times. However, for multi-pulse CPOs ($p > 1$), the settling time is non-monotonic with respect to $|m|$; increasing $|m|$ leads to lower settling times until a minimum is reached, beyond which



(a) $g = 8$ CPO (b) $g = 40$ CPO

Figure 4.2: Effect of m on settling time with impulse-noise injection obtained from behavioral simulator

the settling time increases as $|m|$ increases. It can also be observed that for a fixed m, increasing the density of a CPO ring (by increasing p for a fixed g) or increasing the ring diameter (or g) for a fixed ring density (or p/g) both result in longer settling times. Interestingly, the residual phase error diminishes in magnitude for larger ring densities and diameters. Hence, from a design perspective, there exists a trade-off between the desired residual phase error and the amount of time taken to settle to the residual value.

4.2.2 Verification of Behavioral Simulations using Hspice

The analysis so far on the behavior of CPOs has been based on an abstract model constructed using the linear and repulsive delay-separation dynamics of pulse gates. To verify that this model accurately captures salient properties of CPOs, impulse-noise results obtained from the behavioral simulator were verified against Hspice simulations. The simulation setup used in Hspice is shown in Fig. 4.3. $g = 8$ and $g = 12$ CPOs were tested, in the $p = 2$ and $p = 3$ modes respectively. After starting the oscillators by firing 2 and 3 pulses, in about 10-cycles stable operation of the oscillators was achieved with

a constant non-time-varying period. To model an impulse time perturbation, a small current impulse (1% of the peak output current) was injected into the input node of a single pulse gate. The pulse arrival times at each gate were calculated at the rising-edge zero-crossings of the pulses when the pulse voltage was at half the maximum voltage ($V_{dd}/2$). Phase errors were obtained by subtracting the pulse arrival times after noise injection and those in an ideal noise-less simulation.

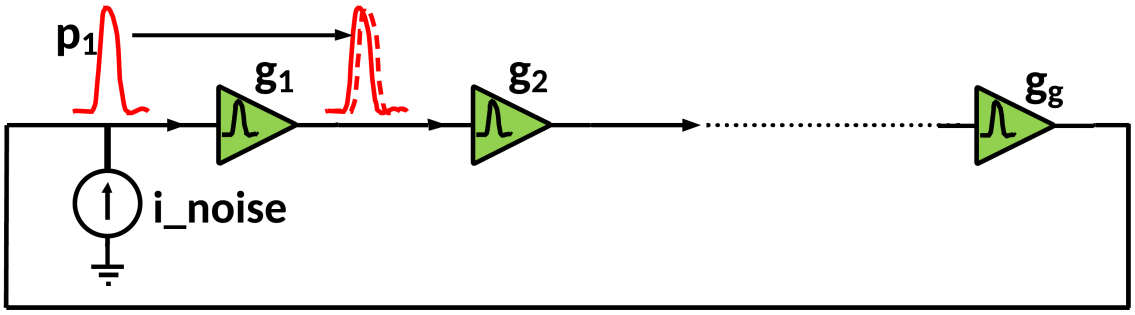
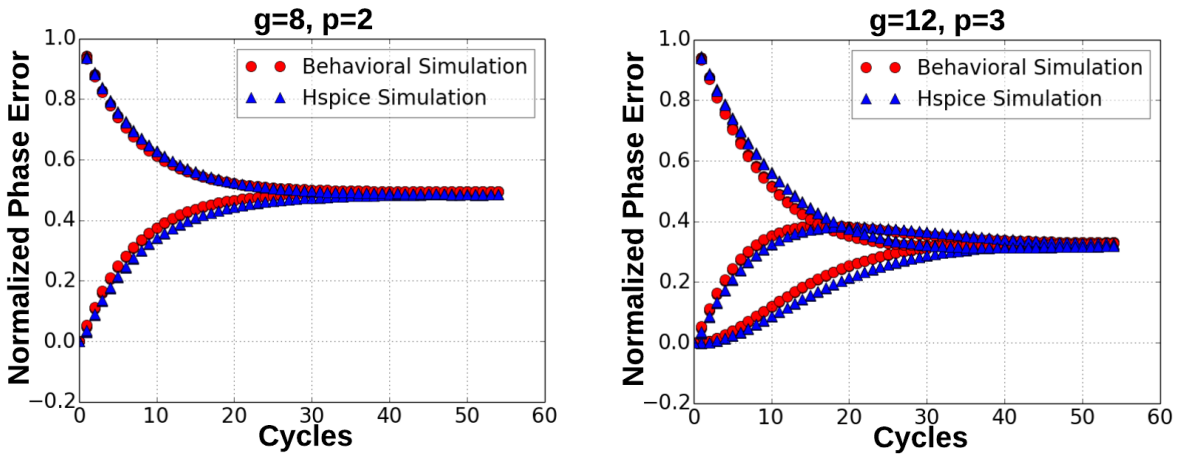


Figure 4.3: Impulse noise simulation setup in Hspice

Phase errors obtained using Hspice and the behavioral simulator for 2 and 3 equidistant gates of the $g = 8$ and $g = 12$ CPOs respectively, normalized with respect to the magnitude of the initial injected error are shown in Fig. 4.4a and Fig. 4.4b. The delay-separation slope m in the behavioral simulations was set to the same value as obtained from Hspice simulations. It can be seen that for both simulations there is a very close match between the trajectory of phase-error correction, settling time as well as the magnitude of the residual phase error. This validates that the abstract model and anaytical solutions accurately capture the behavior of CPOs, and can be used as a substitute to relatively slow Hspice simulations.



(a) $g = 8, p = 2$ CPO (b) $g = 12, p = 3$ CPO

Figure 4.4: Comparison of normalized residual (impulse) phase error obtained from behavioral and Hspice simulations

4.2.3 Power-Law Noise Analysis

Discrete-time noise sequences generated by the algorithm presented in [67] were used for simulating power-law noise sources. For analyzing the resulting phase data using Allan deviation, the IEEE Standard Allan deviation tool Stable-32[68] was used. Frequency stability over different time-scales was analyzed for CPOs of various gate counts g, operating in various modes (set by p) and values of delay-separation slopes m. Fig.4.5, Fig. 4.6 and Fig. 4.7 show modified Allan deviation plots across the design space with both white FM and flicker FM noise injection. The magnitude of the flicker noise component was set to be $1/10^{th}$ of the white noise component. The "conventional" oscillator in these plots corresponds to a single-event oscillator ($p = 1$) that does not exhibit any dynamics ($m = 0$). The error bars depict the 95% confidence limits.

Fig.4.5 shows the modified Allan deviation of CPOs of different diameters (or g), but same pulse density (or p/g) and m. For larger values of p the frequency stability substantially improves beyond $\approx 3 - 4$ ns, whereas the initial instability and higher slope region correspond to the larger settling time associated with multi-pulse CPOs.

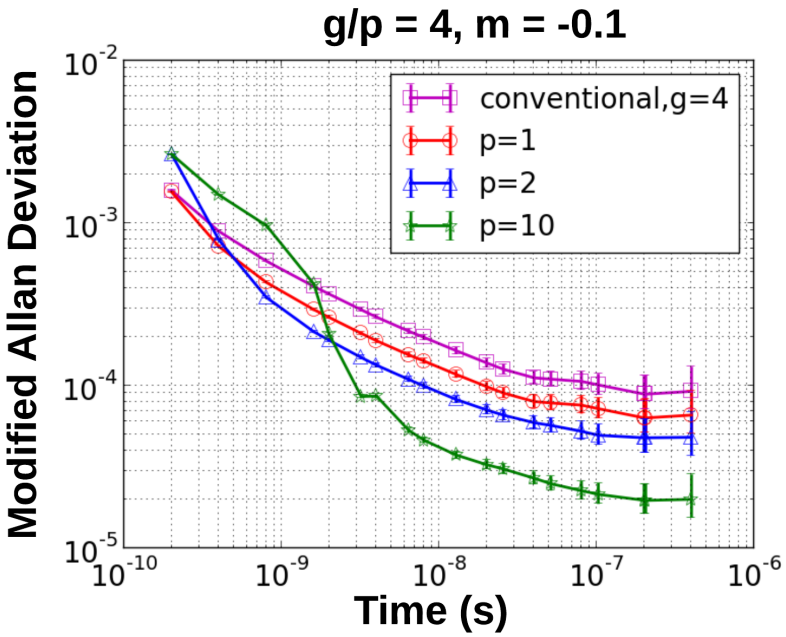


Figure 4.5: Improvement in frequency stability of CPOs as a function of p for fixed g/p and m obtained from behavioral simulations

The single-pulse CPO does not exhibit any initial instability, due to its shorter settling-time. The CPOs in this simulation naturally operate at identical frequencies (= 5 GHz), and hence the improvement in frequency stability and jitter stability (Equation 4.4) is the same. Comparing the jitter stability values obtained from this experiment, it was inferred that beyond the high-slope region, $J[k]$ is $\propto \sqrt{g}/(p(1+(g/p)m))$. The denominator is identical to the residual phase error derived in Equation 3.16. The $\sqrt{g}$ term in the numerator can be explained by the increase in the total (uncorrelated) noise power injection as the CPO diameter is increased. This indicates that equal pulse density, equal frequency CPOs will exhibit an improvement in low-frequency phase noise that equals $-10\log p$, where the oscillator power is $\propto p$. A unique feature of this class of CPOs is that any added power is distributed in space, and thus frequency stability is improved without increasing the power density.

Fig.4.6 shows the modified Allan deviation for $g = 40, p = 5$ CPOs operating at

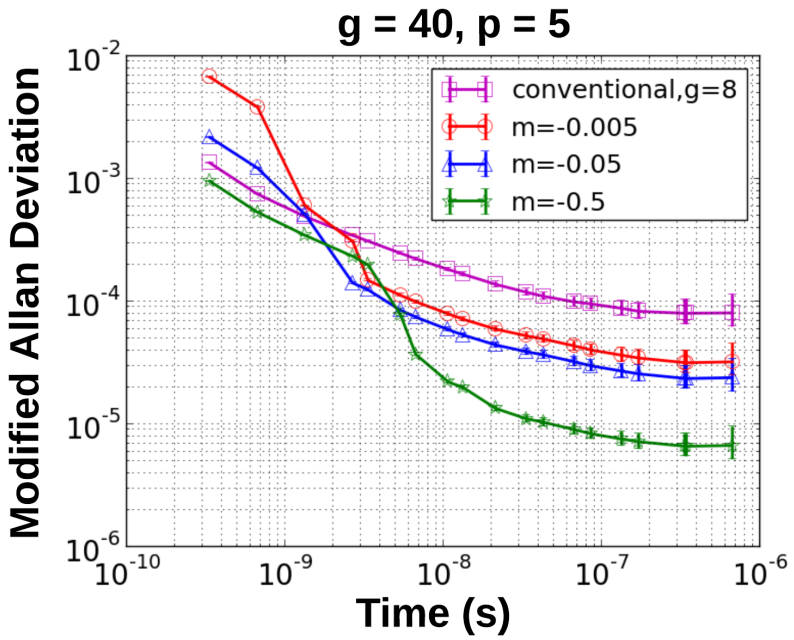
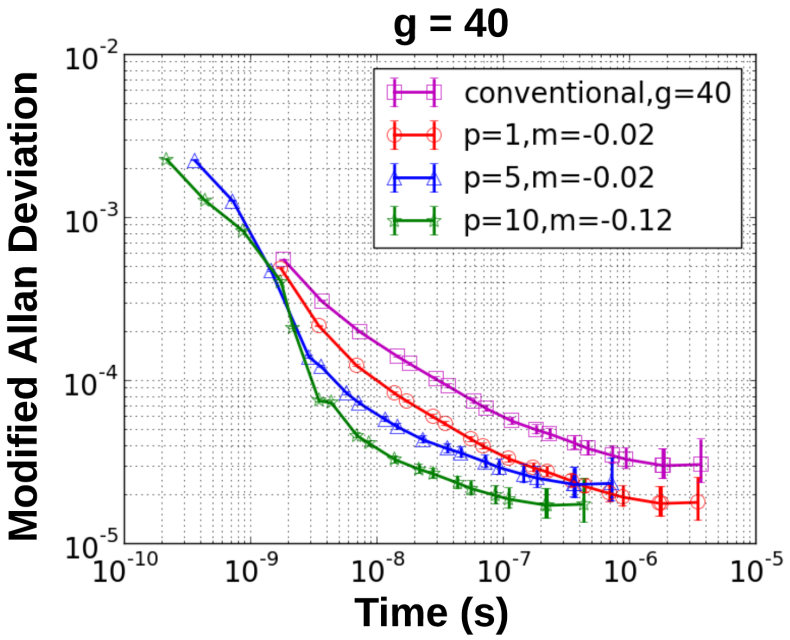


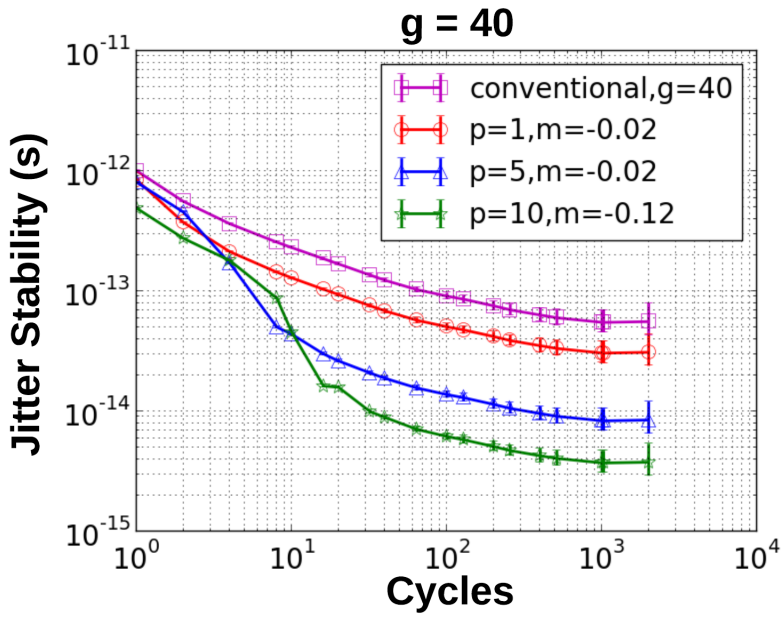
Figure 4.6: Improvement in frequency stability of CPOs as a function of m for fixed g, p and T_o obtained from behavioral simulations

different values of m. Compared to the conventional oscillator, the frequency stability of the $m = -0.005$ and $m = -0.05$ CPOs shows a slight deterioration for ≈ 3 ns. This is due to the relatively smaller corrections made by CPOs with smaller $|m|$, which increases the time for multi-pulse rings to reach an equilibrium, leading to a larger degradation in the frequency stability over short time-scales. Whereas, for the $m = -0.5$ CPO, the disequilibrium between multiple pulses quickly falls to a small value, and as the errors get smaller, the corrections also get smaller, increasing the time to settle to the final (smaller) residual error. In this simulation, the CPOs were made to operate at the same frequency (3 GHz) by tuning their b values. Hence, the improvement in frequency stability and jitter stability of the compared CPOs is again the same. The $J[k]$ values computed in this experiment match the relation between $J[k]$ and CPO parameters inferred from the previous experiment.

Fig. 4.7 shows the frequency and jitter stability of $g = 40$ CPOs running in different



(a) Improvement in frequency stability



(b) Improvement in jitter stability

Figure 4.7: Improvement in stability of CPOs as a function of p for fixed g obtained from behavioral simulations

modes and at different values of m. These values of m were chosen to simulate designs nearly identical to the fabricated 40-gate CPO, whose results are presented in chapter 5. These CPOs all operate at different frequencies (550 MHz, 580 MHz, 2.8 GHz and 4.6 GHz), and hence the jitter stability plots aid better in comparing their stability as a function of oscillation cycles. The larger initial instability and longer high-slope regions for CPOs running in a high-p mode correspond to their longer settling times. This simulation also validates the relation between $J[k]$ and CPO parameters.

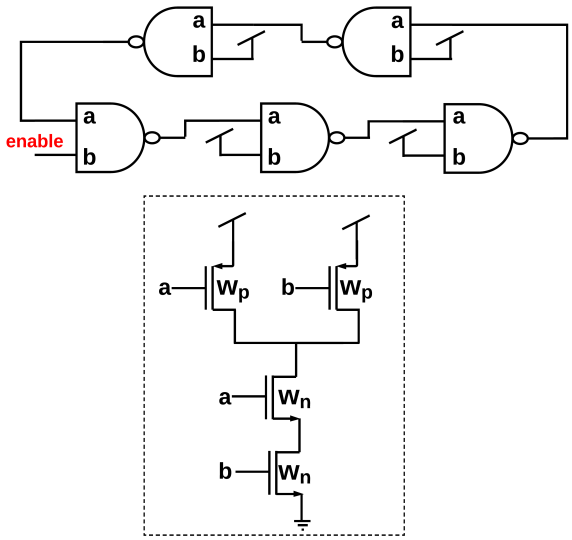
Results obtained from the behavioral simulations have helped us understand better the frequency stability of CPOs at different time-scales, and its dependence on design parameters. The main conclusions that can be drawn are that (i) the phase error correction properties of CPOs improve their frequency stability across time-scales beyond the settling-time of the correction, (ii) jitter stability of CPOs is $\propto \sqrt{g}/p(1+(g/p)m)$, and (iii) CPOs identical in terms of frequency, pulse density (p/g) and m provide phase noise improvement of $-10\log p$, where the power consumption is $\propto p$ and distributed in space. Although for the results presented in this section, CPOs shown better long-term stability than ring oscillators, they also consume more power due to the more complicated design of pulse gates. Analyzing this power-performance trade-off is beyond the scope of the behavioral simulator, and hence is done through Hspice simulations presented in the next section.

4.3 Hspice Based Comparison of CPOs and Ring Oscillators

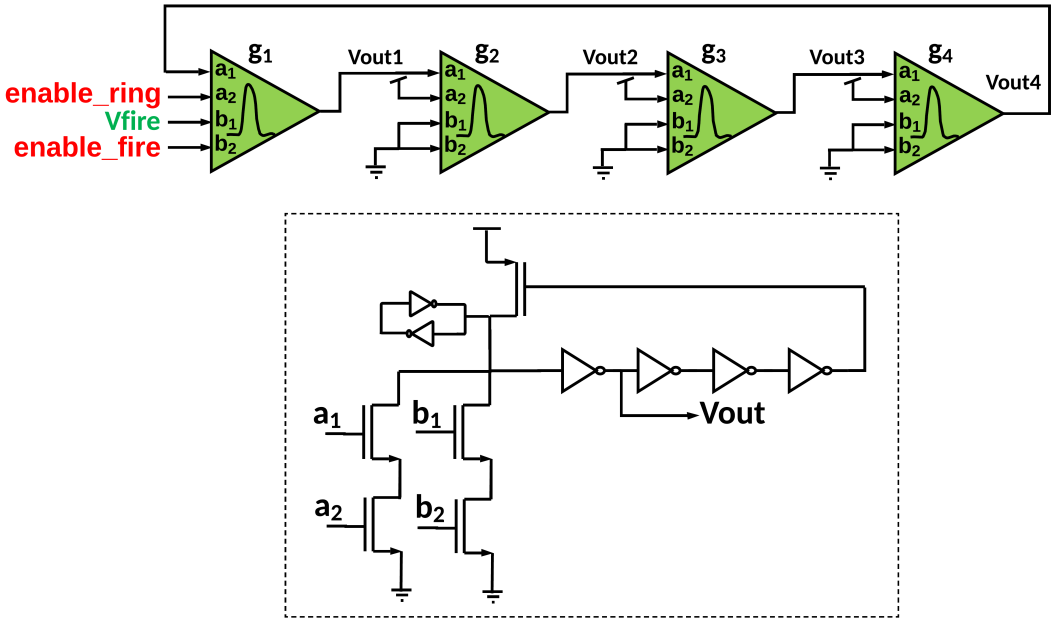
To compare the Figure-of-Merit (FoM) of ring oscillators and CPOs based on frequency, power and device noise, as well as their response to power-coupled and impulse-

noise, Hspice simulation results are presented. Comparisons are made between two 5-Nand gate rings with different transistor strengths and $g = 4$ CPOs with different delay-separation slopes (m), as well as a $g = 8, p = 2$ CPO. The value of m for the CPOs was tuned by changing the relative delay of the feed-forward and feedback paths in the pulse gate and the feedback path was built using 4-inverters for better controllability of m. Nand-gate rings were chosen for the comparisons as they allow ring oscillators to be enabled/disabled easily. The ring was built using all Nand gates instead of one Nand gate and four inverters to ensure that the phases of the ring are uniformly spaced. This makes the comparison of ring oscillators to CPOs fair as CPOs also have additional inputs to start and enable/disble the ring. The schematics of the ring oscillator and CPOs used for comparison are shown in Fig.4.8. The pull-down network of the pulse gates in this CPO contains additional inputs that allow the oscillator or the firing of a start-up pulse to be disabled by making the nodes enable_ring and enable_fire low respectively.

The effect of device noise was evaluated by transient noise Hspice simulations conducted at a temperature of 45^oC and supply voltage of 1.5 V in GFUS8RF 130 nm technology. Phase noise estimation with the Shooting Newton engine in Hspice or Spectre is based on the assumption that the oscillator follows a Linear-Time-Variant model [69], and injected phase errors persist indefinitely with a magnitude that equals the initial injected phase error. This assumption is not valid for CPOs as they exhibit temporal phase error correction, which in fact improves their frequency stability. This makes the simulated phase noise estimation for CPOs inaccurate. Fig. 4.9 shows the simulated phase noise for a $g = 4, p = 1$ CPO and $w_p = w_n = 8\mu m$ 5-Nand-gate ring oscillator. RMS phase jitter J_{RMS}, obtained by integrating the phase noise, is also shown on this



(a) 5-Nand-gate ring



(b) $g = 4, p = 1$ CPO

Figure 4.8: Schematics of compared oscillators

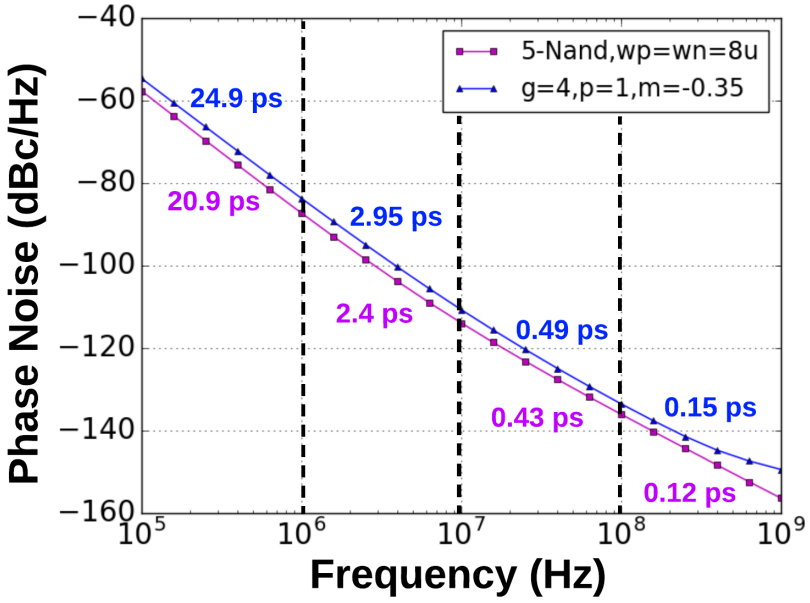


Figure 4.9: Simulated phase noise and RMS phase jitter obtained by integrating phase noise

plot. The relationship between phase noise and RMS jitter is as follows [70]:

$$J_{RMS} = \frac{1}{2\pi f_o}\sqrt{2\int 10^{L(f)/10} df} \tag{4.6}$$

Here f_o is the nominal oscillation frequency and $L(f)$ is the single-sideband phase noise. While phase noise simulation results predict the CPO phase noise to be $\approx 3dB$ higher than that of the 5-Nand-gate ring oscillator and consequently higher integrated RMS phase jitter for the CPO, the jitter obtained from time-domain simulations (Fig. 4.10) clearly shows the opposite trend. The k-cycle RMS phase jitter was calculated from $N = 100$ runs of transient noise simulations based on the k-cycle phase error $\Phi[k]$ as follows:

$$J_{RMS}[k] = \sqrt{\frac{\sum_{i=1}^{N} \Phi_i[k]^2}{N}} \tag{4.7}$$

This validates the inadequacy of phase noise simulations and hence, the comparisons in

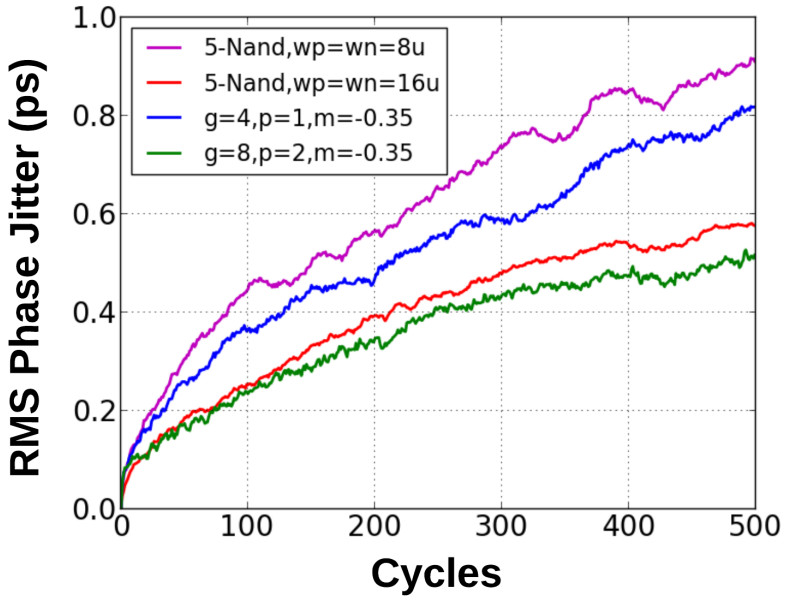


Figure 4.10: RMS phase jitter obtained from transient noise simulations

this section are based on time-domain simulations. The frequency, power and 500-cycle RMS phase jitter obtained from transient noise simulations for the oscillators are listed in Table 4.1. The table also includes the FoM improvement of CPOs with respect to that of ring oscillators. The improvement for the $g = 4, p = 1$ CPOs was calculated against the $w_p = w_n = 8\mu m$ Nand-gate ring, while that of the $g = 8, p = 2$ CPO was calculated using the $w_p = w_n = 16\mu m$ Nand-ring as follows:

$$\begin{aligned} \text{FoM improvement} = 20\log(\frac{f_{CPO}}{f_{RO}}) - 10\log(\frac{P_{CPO}}{P_{RO}}) \\ -20\log(\frac{J_{RMS}[k=500]_{CPO}}{J_{RMS}[k=500]_{RO}} \end{aligned} \tag{4.8}$$

Here f_{CPO}, P_{CPO}, $J_{RMS}[k = 500]_{CPO}$ are the frequency, power and 500-cycle RMS phase jitter of the CPO and f_{RO}, P_{RO}, $J_{RMS}[k = 500]_{RO}$ are those of the ring oscillator against which the CPO is compared. While the CPOs operate at a higher frequency and power than the ring oscillators, the jitter in CPOs strongly depends on the value of m. It can

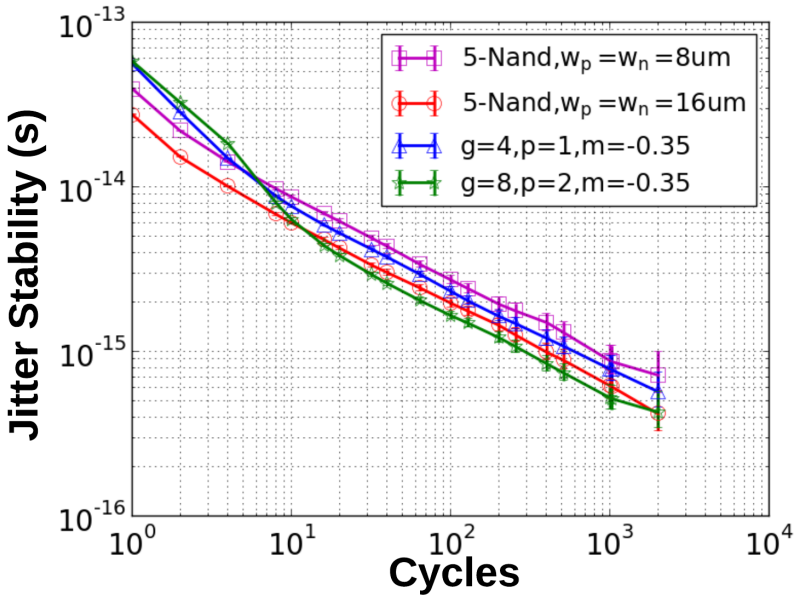


Figure 4.11: Jitter stability obtained from transient noise simulations

be seen that larger magnitudes of m lead to lower jitter as expected and the FoMs shows a trend similar to the trend in jitter. These results suggest that by operating CPOs at relatively high values of $|m|$, the phase correction properties lead to similar FoMs as ring oscillators. Jitter stability plots for the 5-Nand-gate rings, $g = 4, p = 1$ and $g = 8, p = 2$ CPOs also obtained from transient noise simulations are shown in Fig. 4.11. The higher powered Nand-gate rings and CPOs show jitter stability values $\approx \sqrt{2}$ smaller than their respective lower powered counterparts. For both CPOs, the jitter stability values are higher for ≈ 10 cycles, while the far-out values are better than the corresponding ring oscillators as expected.

Fig. 4.12 shows the magnitude of residual phase error in the CPOs and ring oscillators after the injection of an impulse of noise current, of magnitude $2\mu A$. Although the ring oscillators see a smaller initial timing deviation for the same magnitude of injected noise, the CPOs show significantly smaller residual error values. The $g = 4, p = 1$ CPO has a residual error 38% better than the $w_p = w_n = 8\mu m$ Nand-gate ring, while the $g = 8, p = 2$

Table 4.1: Performance Comparison of CPOs and Ring Oscillators (obtained from Hspice simulations)

Oscillator	Frequency (GHz)	Power (mW)	$J_{RMS}[k = 500]$ (ps)	FoM improvement (dB)
5-Nand Ring, w_p=w_n=8μm	5.91	3.62	0.91	-
g=4, p=1, m=-0.17	6.75	4.53	1.43	-3.76
g=4, p=1, m=-0.22	6.89	4.79	1.04	-1.06
g=4, p=1, m=-0.25	6.99	5.04	0.94	-0.24
g=4, p=1, m=-0.3	7.12	5.37	0.89	0.06
g=4, p=1, m=-0.35	7.22	5.69	0.82	0.68
g=4, p=1, m=-0.43	7.42	6.81	0.69	1.6
5-Nand Ring, w_p=w_n=16μm	5.92	7.26	0.58	-
g=8, p=2, m=-0.35	7.22	11.38	0.51	0.88

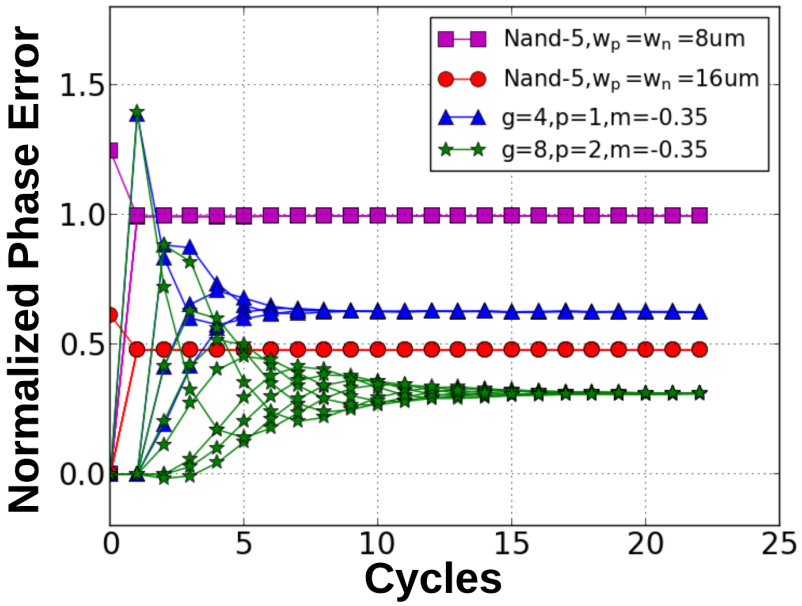
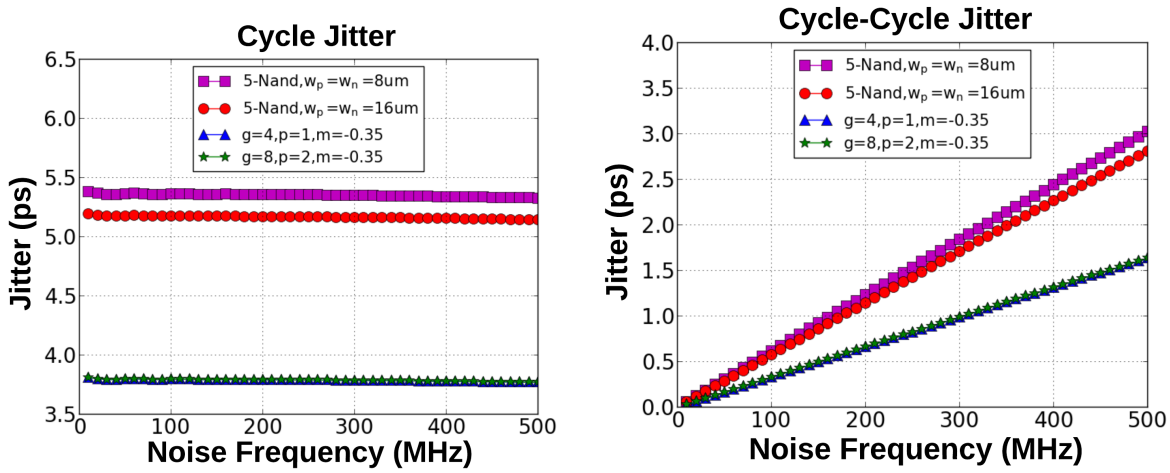


Figure 4.12: Simulated normalized phase error of CPOs and ring oscillators with impulse noise injection

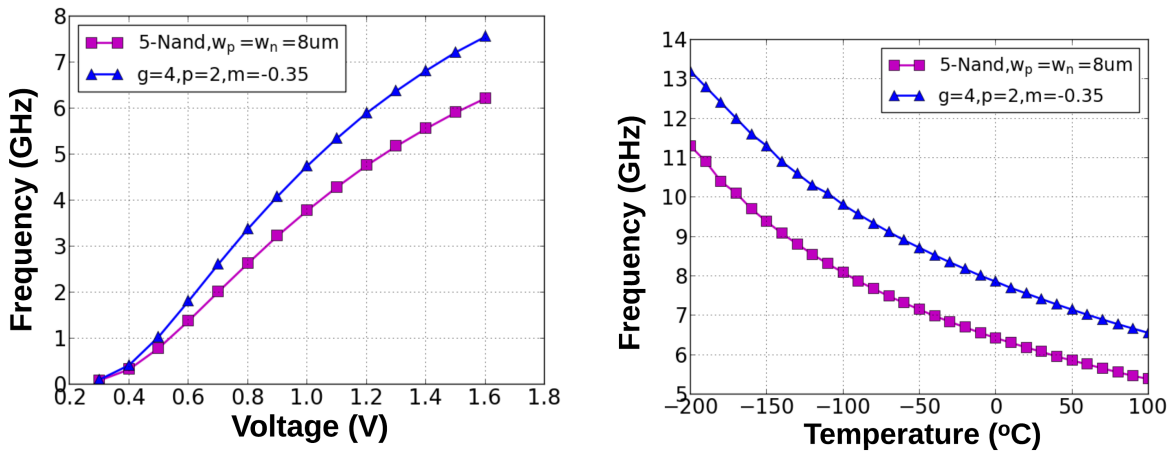


(a) Cycle jitter

(b) Cycle-cycle jitter

Figure 4.13: Comparison of CPOs and ring oscillators with power-coupled noise

CPO has a residual error 35% better than the $w_p = w_n = 16\mu m$ Nand-gate ring. Fig.4.13 compares the performance of CPOs and ring oscillators under the influence of sinusoidal power-coupled noise sources of amplitude 75mV ($5\%V_{dd}$). The cycle-jitter and cycle-cycle-



(a) Variable voltage

(b) Variable temperature

Figure 4.14: Frequency for 5-Nand-gate ring oscillator and $g = 4, p = 1$ CPO

jitter [71] for the $g = 4, p = 1$ CPO are 29% and 46% smaller than the $w_p = w_n = 8\mu m$ Nand-gate ring. These results indicate that CPOs designed to operate at high values of $|m|$ offer significant improvements over ring oscillators in terms of both impulse noise and power-coupled noise rejection. The jitter values for the higher powered $g = 8, p = 2$ CPO in Fig. 4.13 are almost identical to that of the $g = 4, p = 1$ CPO indicating that the lower jitter values with power noise are a result of the corrections due to a larger value of $|m|$, and not multiple pulses. Finally, Fig. 4.14 shows the effect of supply voltage and temperature on the frequency of ring oscillators and CPOs, indicating that the tuning range of CPOs largely resembles that of ring oscillators.

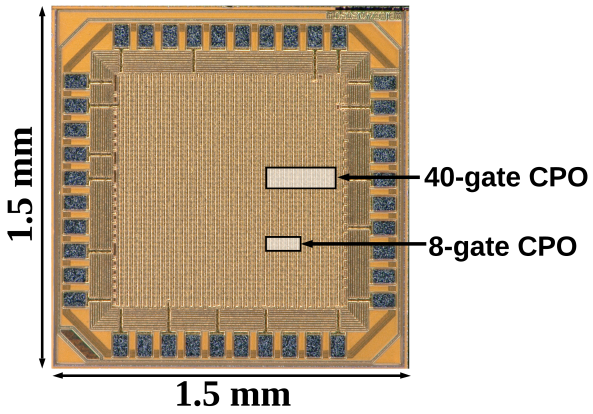
Chapter 5

Measurement Results for Simple Loop Connected CPOs

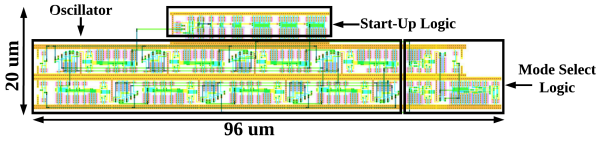
5.1 Measurements on Chip Fabricated in 130nm

Collective Pulse Oscillator with 8-gates and 40-gates fabricated in the GFUS8RF (130 nm) process have been tested. The chip micrograph is shown in Fig. 5.1a and CPO layouts are shown in Fig. 5.1b, 5.1c. The $g = 8$ CPO can be run in either the $p = 1$ or $p = 2$ mode, and the $g = 40$ CPO can run in 10 modes corresponding to $p = 1 - 10$. The pulse gate topology in the fabricated designs is similar to Fig. 4.8b, with 4-inverters in the feedback loop and pull-down transistors to enable firing of a start-up pulse as well as enable the ring itself. To generate a start-up pulse, an external rising edge was driven into the chip and converted to a pulse. The pulse was used to drive the fire inputs of multiple gates of the two CPOs. For the $g = 8$ CPO, the fire input on gate-1 and gate-5 (Fig. 4.8b, node b_1) was driven by the fire pulse, with a separate enable_fire signal (node b_2) for gate-1 and gate-5. The CPO mode was set by making one ($p = 1$ mode) or both ($p = 2$ mode) enable_fire signals active by external control signals. The enable_ring signal

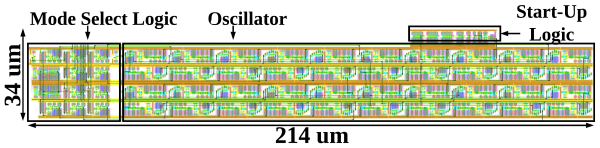
(node a_2) was used to enable or disable oscillations in the ring. Similarly, for the $g = 40$ CPO, every 4^{th} gate was driven by the fire pulse, with independent fire_enable signals for each of the 10 gates to run the CPO is different modes and a ring_enable signal to enable the CPO.



(a) Micrograph of fabricated chip



(b) Layout of fabricated $g = 8$ CPO



(c) Layout of fabricated $g = 40$ CPO

Figure 5.1: Chip in GFUS8RF (130 nm)

The CPOs were characterized by both phase noise and frequency/jitter stability measurements. Phase noise measurements were made using a Keysight PXA Signal Analyzer (N9030B). For obtaining the Allan deviation values, CPO output waveforms captured on an Agilent Infiniium oscilloscope were sampled at 80 GSamples/s. The zero-crossings of the rising edge of the waveforms (at $V_{dd}/2$) were computed and using the resultant phase data, Allan deviation and jitter stability values were calculated in IEEE Stable-32

[68]. The error bars on these plots depict the 95% confidence values. Power supply was regulated by an on-board Analog Devices LT3042 regulator. No on-chip regulator was used for these measurements. The measured frequency, power, tuning range, phase noise @ 1 MHz and 10 MHz offsets, number of phases and figure of merit (FoM) calculated at an offset frequency of 10 MHz are summarized in Fig. 5.2. It also includes the normalized jitter stability at $k = 100$ cycles for different modes of the two CPOs. These values were computed by dividing the $J[k = 100]$ for the $p = 1$ mode of each CPO by the $J[k = 100]$ for the other modes of the same CPO. The FoM is calculated as follows:

$$FoM = 20\log(\frac{f_o}{\Delta f}) - P.N. - 10\log(\frac{P_o}{1mW}) \tag{5.1}$$

where f_o is the oscillation frequency, Δf is the frequency offset at which the phase noise (P.N.) is measured and P_o is the oscillator power consumption in mW.

Oscillator, Mode	Frequency f_o (GHz), $V_{dd} = 1.5V$	Power (mW), $V_{dd} = 1.5V$	Frequency range (GHz), $V_{dd} = 0.6 - 1.6V$	$1MHz$ Phase Noise $@f_o(dBc/Hz)$	$10MHz$ Phase Noise $@f_o(dBc/Hz)$	$\frac{J[k=100],g,p=1}{J[k=100],g,p}$	# of Phases	FoM (dB)
g=8, p=1	2.86	5	0.52-3.02	-94.6	-115.6	1.0	8	157.7
g=8, p=2	4.87	8.5	0.88-5.11	-96.8	-118.8	2.92	4	163.2
g=40, p=1	0.58	5	0.09-0.62	-109.4	-127.7	1.0	40	156.0
g=40, p=2	1.15	9.8	0.18-1.22	-106.9	-126.5	1.98	20	157.8
g=40, p=3	1.71	13.5	0.28-1.81	-104.8	-125.1	3.02	40	158.5
g=40, p=4	2.26	18.5	0.37-2.38	-103.6	-124.2	4.06	10	158.6
g=40, p=5	2.78	22.5	0.45-2.93	-102.2	-122.6	4.73	8	158.0
g=40, p=6	3.29	27	0.54-3.47	-100.9	-122.7	5.97	20	158.7
g=40, p=7	3.80	31.9	0.62-4	-99.6	-120.3	6.42	40	156.8
g=40, p=8	4.27	36	0.71-4.49	-100.9	-122.6	9.48	5	159.6
g=40, p=9	4.51	38.2	0.8-4.73	-100.5	-122.7	10.08	40	160.0
g=40, p=10	4.64	39.3	0.86-4.85	-100.5	-121.9	9.6	4	159.3

Figure 5.2: GFUS8RF (130 nm) Measured Results

For the $g = 8$ CPO, the $p = 2$ mode operates at a higher frequency and power than the $p = 1$ mode as expected. The modified Allan deviation and jitter stability for the two modes of the $g = 8$ CPO are plotted in Fig. 5.3 and Fig. 5.4. This CPO shows better jitter stability in the $p = 2$ mode than the $p = 1$ mode beyond $\approx 2 - 3$ cycles. This is attributed to both smaller residual phase error for multi-pulse CPOs,

Table 5.1: Performance Comparison of fabricated g=8, p=2 CPO with similar previous works

Work	Technology	Frequency (GHz)	Power (mW)	Phase Noise (dBc/Hz), offset frequency (MHz)	FoM (dB)
This work	130 nm	4.87	8.5	-96.8, 1	163.2
[57]	90 nm	3.16	13	-103.4, 1	162.25
[38]	65 nm	5.0	1	-101, 4	162.94
[72]	135 nm	4.7	14.8	-97.5, 1, 4	159.24
[73]	180 nm	0.9	65.5	-106.1, 0.6	151.46

as well as operation of the 2-pulse and 1-pulse CPOs in Region 2 and Region 3 of the delay-separation curve of Fig. 2.5b, respectively and hence at different magnitudes of m. An increase in the delay-separation slope (of relatively small magnitude) results is both smaller settling time (Fig. 4.2) and smaller jitter stability, which improves both short-term and long-term behavior as can be seen for the $g = 8$ oscillator. The value of $J[k = 100]$ is 2.9 times smaller in the $p = 2$ mode compared to the $p = 1$ mode. If the two modes operated at identical (small) slopes, the improvement would have been ≈ 2 times instead. Consequently, the phase noise in the $p = 2$ mode is $\approx 3dB$ better than in the $p = 1$ mode, resulting in an improvement in FoM of $\approx 5dB$.

These results show that multi-pulse CPOs can not only support higher operating frequencies, but also exhibit lower phase noise resulting in an improved FoM. Fig. 5.5 shows the measured phase noise for the $g = 8, p = 2$ CPO compared against simulation values. As explained in Section 4, since phase error correction is not taken into account by phase noise simulators, the measured values at 1 MHz and 10 MHz phase offsets are indeed lower than the simulated values. Whereas the simulated value at 100 MHz offset is better than the measured value due to higher settling time associated instabilities at large frequency offsets and high-frequency power-coupled noise during measurements.

For the $g = 40$ CPO operating in different modes, the frequency scaling with respect

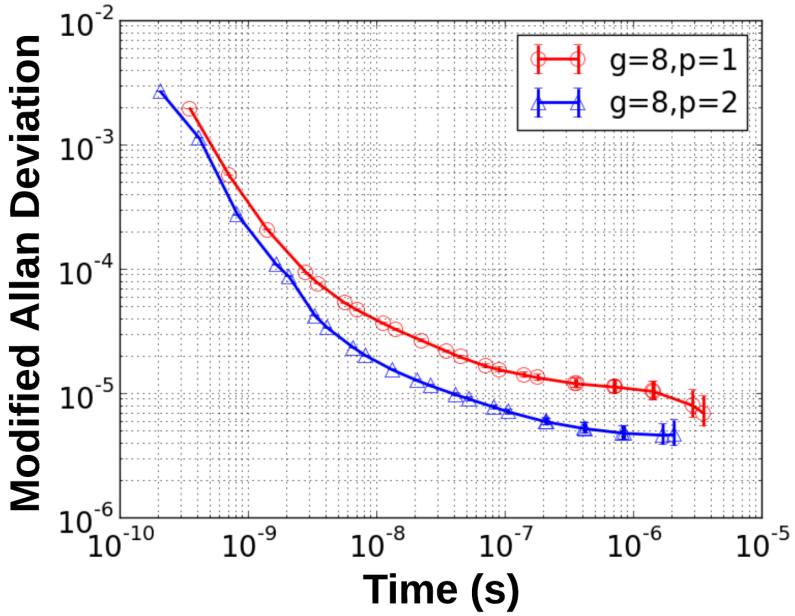


Figure 5.3: Measured $Mod\ \sigma(\tau)$ for $g = 8$ CPO

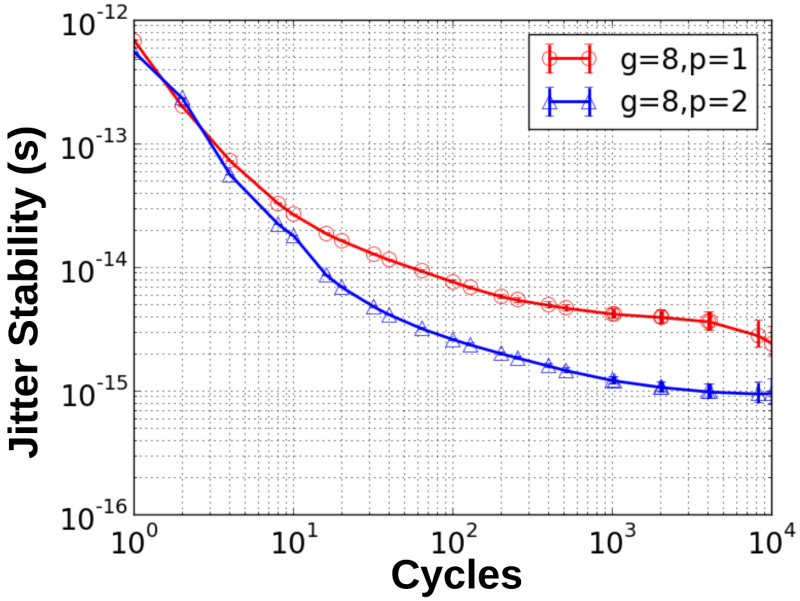


Figure 5.4: Measured jitter stability $J[k]$ for $g = 8$ CPO

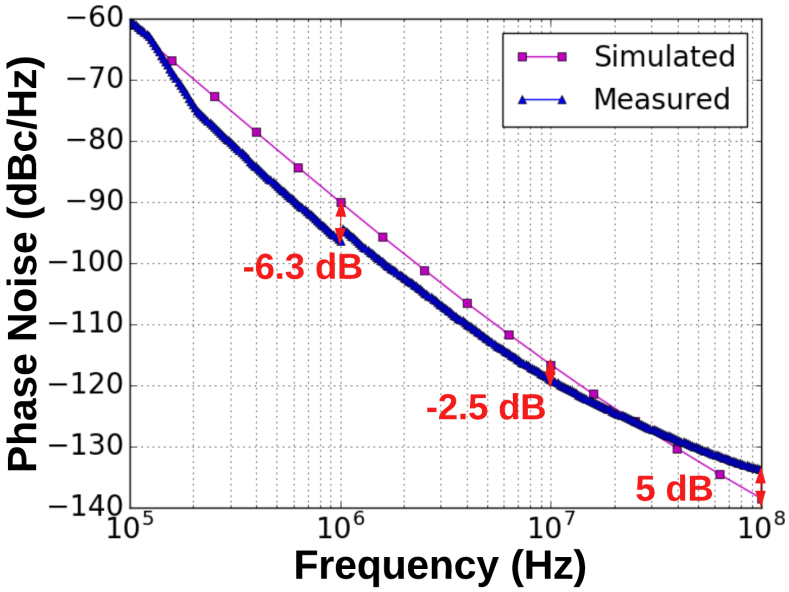


Figure 5.5: Comparison of simulated and measured phase noise for $g = 8, p = 2$ CPO

to the number of pulses, p is less than proportional to the increase in p, which matches our analytic predictions. The modified Allan deviation and jitter stability for three modes ($p = 1, p = 5$ and $p = 10$) are plotted in Fig. 5.6 and Fig. 5.7. The $p = 1$ and $p = 2$ modes of the $g = 8$ CPO mode present nearly identical frequency (same g/p ratio) comparison points to the $p = 5, p = 10$ modes of the $g = 40$ CPO. As expected, the $g = 40$ CPO outperforms the corresponding modes of the $g = 8$ CPO at longer time-scales. While at shorter time-scales, the $g = 8$ CPO modes show better jitter stability as they involve interactions between a smaller number of pulses and exhibit shorter settling times.

As can be seen from Fig. 5.2, the phase noise of the $g = 40, p = 5$ is $\approx 7.5dB$ better than the $g = 8, p = 1$ CPO. This confirms the inference made from behavioral simulations about the improvement in phase noise to be $\propto \sqrt{p}$ for power $\propto p$. Fig. 5.2 also shows the jitter stability for the $g = 40$ CPO in different modes at $k = 100$ cycles, compared against the $p = 1$ mode of this CPO. The improvement is close to the mode value p for modes $p = 1 - 6$, which is expected of CPOs operating at a small magnitude of m. The

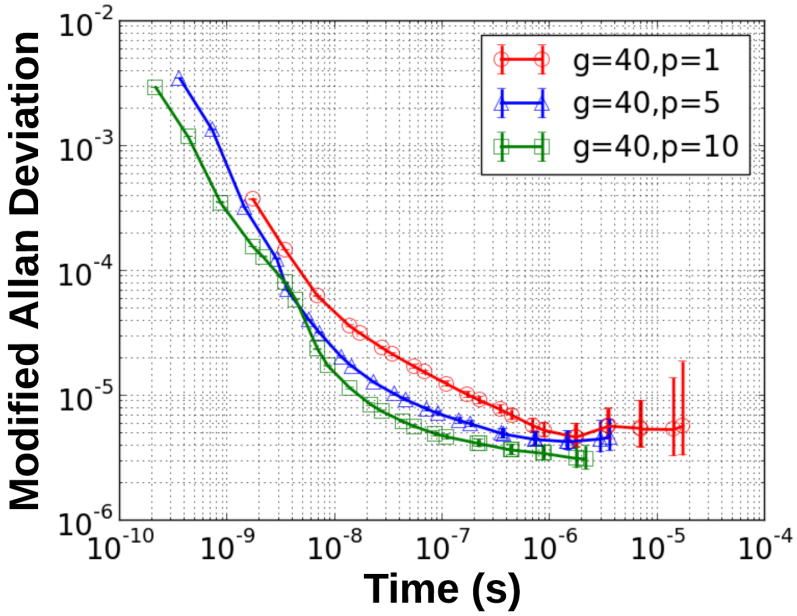


Figure 5.6: Measured $Mod\ \sigma(\tau)$ for $g = 40$ CPO

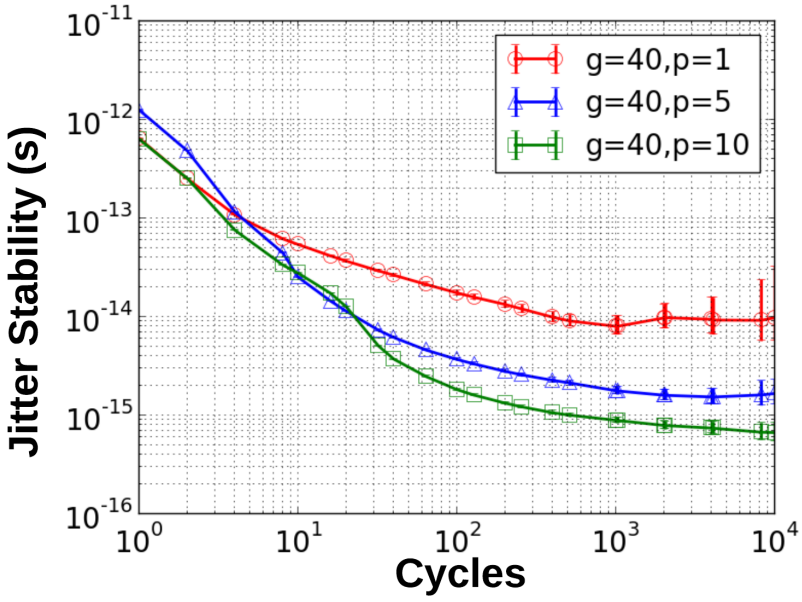


Figure 5.7: Measured jitter stability $J[k]$ for $g = 40$ CPO

improvement in jitter stability dips slightly for the $p = 7$ mode, followed by a significant improvement for modes $p = 8 - 10$. One factor leading to this shift in the trend of jitter stability values is the switch in CPO operation from Region 3 to Region 2 of the delay-separation curve of Fig. 2.5b. The frequency and power consumption rise as the mode (p) is increased, the phase noise degrades slightly and the FoM across all modes lies in a $3 - 4dB$ range. Our model currently does not capture the effects of wire-length mismatches, substrate coupling and power noise, which could all contribute to some non-monotonicity in phase noise as can be seen for the $p = 7 - 10$ modes of this CPO. This CPO can also provide upto 40 phases in certain modes with phase resolution as small as $5.56ps$ ($p = 9$ mode) in 130 nm technology. Fig. 5.8 shows oscilloscope waveform for $p = 1$ mode of the $g = 40$ CPO.

Figure 5.8: Oscilloscope waveform for $g = 40, p = 1$ CPO

5.2 Conclusions

Collective Pulse Oscillators present a new design space for high-performance multi-phase ring oscillators that can provide precise phase resolutions, not limited by the smallest achievable gate delay for a given technology. Pulse gates forming CPOs exhibit time-variant gate dynamics that cause pulses to distribute uniformly around the ring, enabling the existence of precise phases in even or odd numbers as well as mirror phase taps. Further, these dynamics result in temporal phase error corrections that can be utilized to improve the overall frequency stability of the oscillator beyond the first few cycles.

The analysis and results presented in chapters 3, 4, 5 show that to first order, the properties of CPOs are solely governed by the local gate delay-separation dynamics. For equal frequency, equal pulse density CPOs, scaling power by a factor of p improves the frequency stability and phase noise by a factor of $1/\sqrt{p}$ or $-10\log p$. A unique feature of such multi-pulse CPOs is that phase noise is improved by adding power that is distributed in space, and hence phase noise improvement is obtained without increasing the power density. CPOs achieve device noise based FoMs similar to that of ring oscillators. However, CPOs are more resilient to noise that shows correlation among different gates, such as power noise. Finally, for systems dominated by noise profiles that are impulsive, CPOs present a potential to achieve frequency stability improvements $\propto 1/p$, significantly improving the power-vs-noise trade-off.

Chapter 6

Complex CPO Topologies and Transmission Line Stabilized CPOs

Results presented in chapters 3, 4 and 5 have shown that simple loop connected CPOs show lower high-frequency stability for large p or $|m|$. The correction properties of CPOs lead to longer settling times for ring of higher mass or density, which degrades the very high-frequency behavior of these oscillators. This chapter presents alternative CPO topologies that help with improving the short-term stability of CPOs. These structures link multiple small CPO links such that one or multiple gates perform time-averaging of the arrival time of pulses generated by 2 or more gates. The stability of these topologies is evaluated via the behavioral simulator described in chapter 4, and interconnection networks that scale even as the dimension of the CPO becomes large are presented. These networks can serve the purpose of timing distribution and synchronization over small-to-medium scale synchronous domains, where the distribution network contains feedback loops to improve the ensemble timing stability of the network. Methods for making voltage controlled and injection locked CPOs are also presented in this chapter. Finally, transmission line stabilized CPOs are presented that have overall lower noise,

generate full-swing clock phases with a low duty cycle and can be useful for a variety of digital and RF applications.

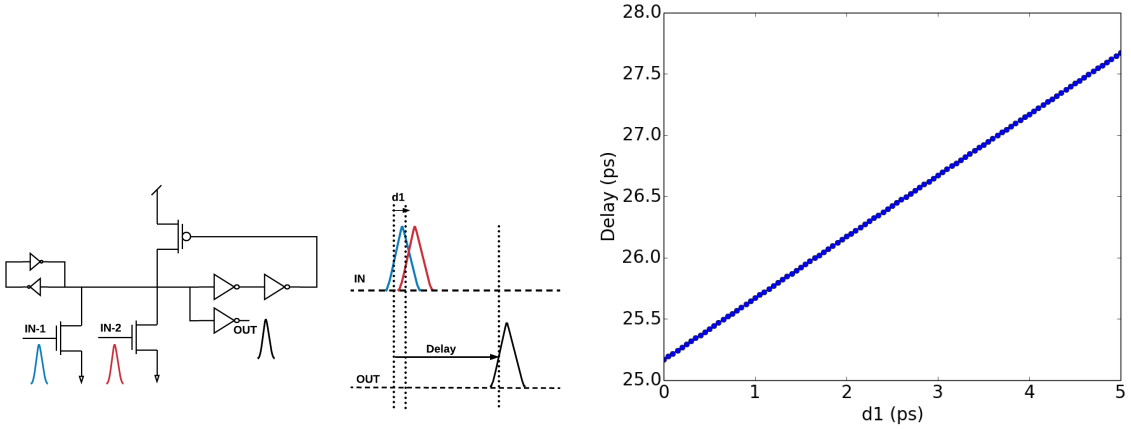
6.1 Interconnected CPO Topologies

As the mass or density of CPO rings is made higher, or if the delay separation slope causes the CPO response to be heavily under-damped or over-damped, the CPO stability at high frequencies degrades. To solve this issue, we use Pulse-OR gates as low-pass averaging filters that take as inputs pulses generated by independently running loops of a relatively small diameter and pulse density. Fig. 6.1a shows how the delay of a pulse-OR gate varies based on the relative delay between the two inputs to the gate. The pulse-OR gate with equal sized pull-down transistors basically does an unweighted averaging of the arrival times of the two input pulses, and if $y[k]$ is the k^{th} arrival time of a pulse at the output of the OR-gate, $x_1[k]$ and $x_2[k]$ are the k^{th} arrival times of pulses at the input of the OR-gate, the delay of the gate is obtained from the following equation:

$$y[k] = \frac{x_1[k] + x_2[k]}{2} + b + m(\frac{x_1[k] + x_2[k]}{2} - \frac{x_1[k-1] + x_2[k-2]}{2}) \tag{6.1}$$

Similar to the equations in chapter 3, b is a constant and m is the slope of the delay separation curve. The arrival time of output pulses at the gate has been modeled as the average arrival time of input pulses plus the delay of the gate, which in turn depends on the average arrival time of input pulses for the current and previous time-steps. It should be noted that this model assumes that the delay separation slope m is unaffected by the difference $x_1[k] - x_2[k]$, which is a reasonable assumption if the difference is small. However, if the difference gets to be large, the effective m of the gate increases, as the drive of the NMOS is lowered, causing greater repulsion and increase in the delay of the

gate.



(a) Timing of input pulses

(b) OR gate delay as a function of relative input timing

Figure 6.1: Pulse OR timing

The behavioral simulator presented in chapter 4 was expanded to model the behavior of pulse-OR gates as described by equation 6.1. First, two CPO rings of diameter $g = 4$ were interleaved using the topologies shown in Fig. 6.2. The single interleaved architecture contains a single interconnection between the two rings, and the "all-gates-interleaved" architecture contains two-way links between each gate in either loop. For

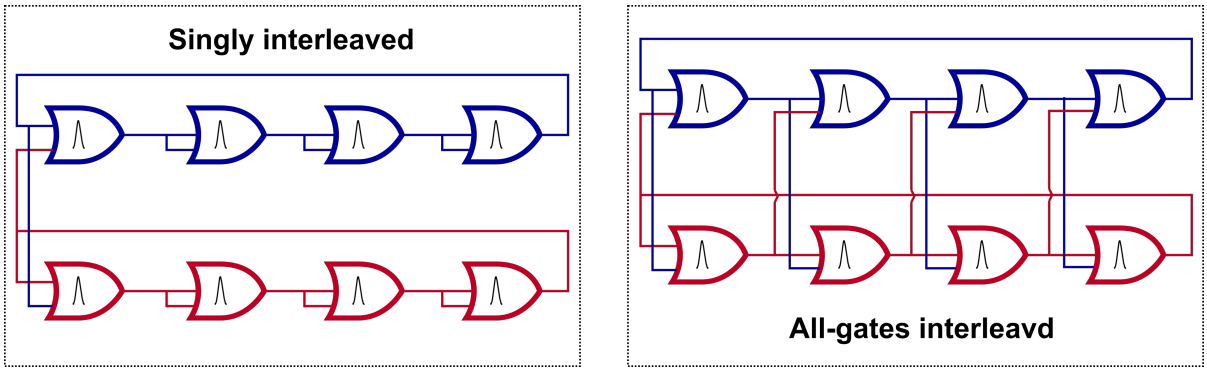


Figure 6.2: Structure of g=8 interleaved clocks

each of these structures, m was set to -0.1, and each design was simulated with white

noise injection. The resulting jitter stability is shown in Fig. 6.3. Results for $g = 4, p = 1$ and $g = 8, p = 2$ are also included to serve as a reference. It can be seen that each interleaved ring shows faster settling and an improved short-term behavior compared to the simple 8-gate loop. The ring with all gates interleaved performs better than the singly interleaved ring as expected, as it does averaging 4 times as often as the singly interleaved ring. The long-term stability however, for interleaved and non-interleaved rings is identical, which also follows intuition as the noise is random and uncorrelated and by interleaving, no additional power is added to the system.

The next question that arises now is how can rings of larger dimensions be interleaved. Fig. 6.4 and Fig. 6.5 show topologies that we term "vertically" and "horizontally" interleaved. In vertical interleaving with k small loops, one or more gates of each loop are interleaved with one or more gates of all other loops. Whereas, in horizontal interleaving, gates in each small loop are only interleaved with a gate in one other loop. As the dimension of these structures grows, vertical interleaving becomes unfeasible due to the limited fan-in/fan-out of gates. However, horizontal interleaving can be easily scaled to higher-dimensional structures. In order to evaluate the relative stability of these topologies, they were constructed and behaviorally simulated, and Fig. 6.6 shows the jitter stability plot with white noise injection. As expected, rings with more interconnections show faster settling responses and lower high-frequency noise. Interestingly, the performance of the 3-way horizontally interleaved ring is very close to the 3-way vertically interleaved ring, even though it only involves 2-way averaging. This points to the possibility of using horizontally interleaved rings to implement multi-pulse CPOs that are highly stable in the long term by virtue of a smaller residual phase error term given by a larger value of p, and also show high short-term stability by virtue of these horizontal interconnections. It is true that CPOs are costly in terms of power as they scale, but these structures can readily be used as timing distribution networks over small-to-medium distances over

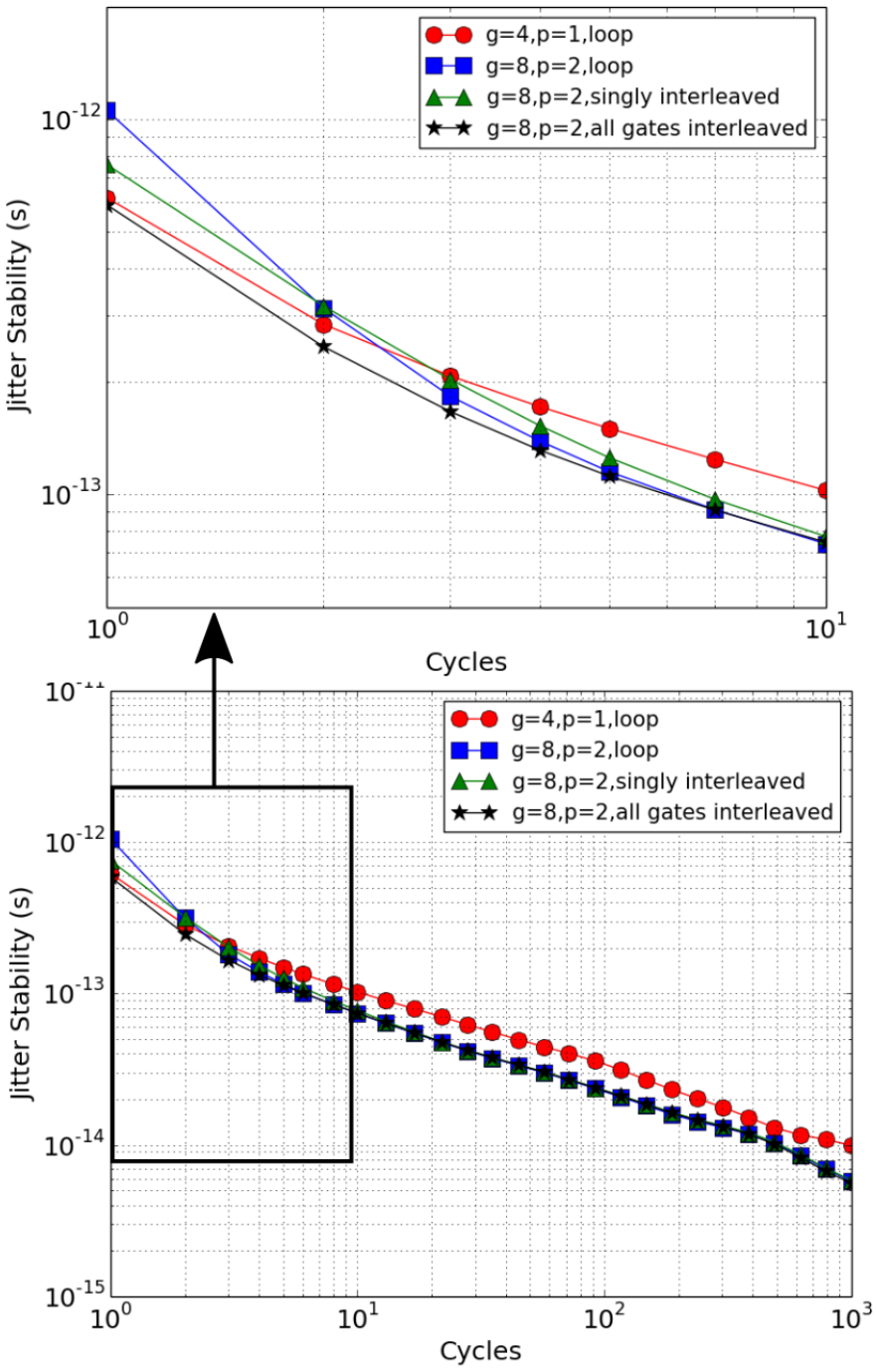


Figure 6.3: Comparison of stability of g=8 interleaved rings

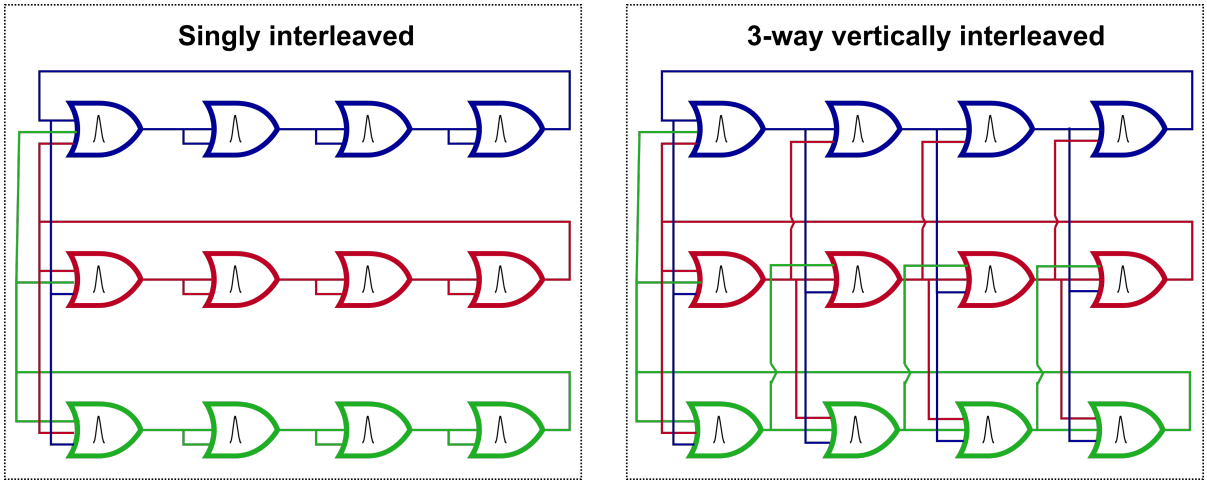


Figure 6.4: Structure of g=12 vertically interleaved clocks

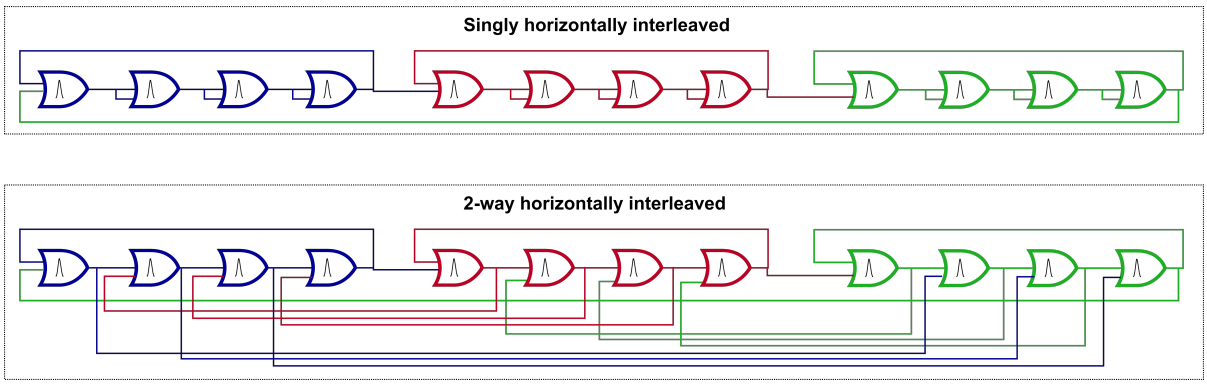


Figure 6.5: Structure of g=12 horizontally interleaved clocks

which pulses can travel reliably and maintain their integrity. Such a distribution network containing feedback loops would in-turn act to improve the ensemble timing stability of pulses circulating in the network. Each gate in these horizontally interleaved clocks only needs to communicate with two of its nearest neighbours (that are g/p gates away) and thus physical placement and routing for these structures can be rather regular. Fig. 6.7 shows an example where 4 small loops of 4 gates each can be placed to create a horizontally interleaved structure. Larger loops placed as a mesh can operate similarly. There are other platonic solid geometries of interconnection networks possible which can further improve the short-term timing stability as the scale of the network grows. One such structure is a tetrahedron clock shown in Fig. 6.8 that uses four 3-input OR gates and has 12-pulses circulating within 6 small-loops forming the tetrahedron. The delay elements here are pulse buffers and the number of buffers should be chosen based on the desired ring frequency and mode of operation along the delay separation curve. The tetrahedron ring can be started by injecting start-up pulses in each of the four OR-gtaes. Fig. 6.9 shows the jitter stability of the tetrahedron clock implemented using 3 pulse buffers in every arc, and using a total of 40 pulse gates, with $g/p = 4$. For comparison, the jitter stability of a simple looping and 1-way horizontally interleaved rings with $g = 48, p = 12$ are also plotted. Fig. 6.9 shows that the 2-way interleaved ring substantially improves the high-frequency noise present in an un-interleaved ring of 48 gates and 12 pulses. However, the tetrahedron ring causes even faster settling and lower noise at high frequencies. This is due to the fact that the results of time averaging are distributed to the elements of the ring more quickly. The long-term stability of the tetrahedron ring is slightly degraded, compared to the 2-way horizontally interleaved $g = 48, p = 12$ ring. Thus, both 2-way horizontally interleaved rings and tetrahedron rings are promising candidates for building timing distribution networks containing an array of such structures.

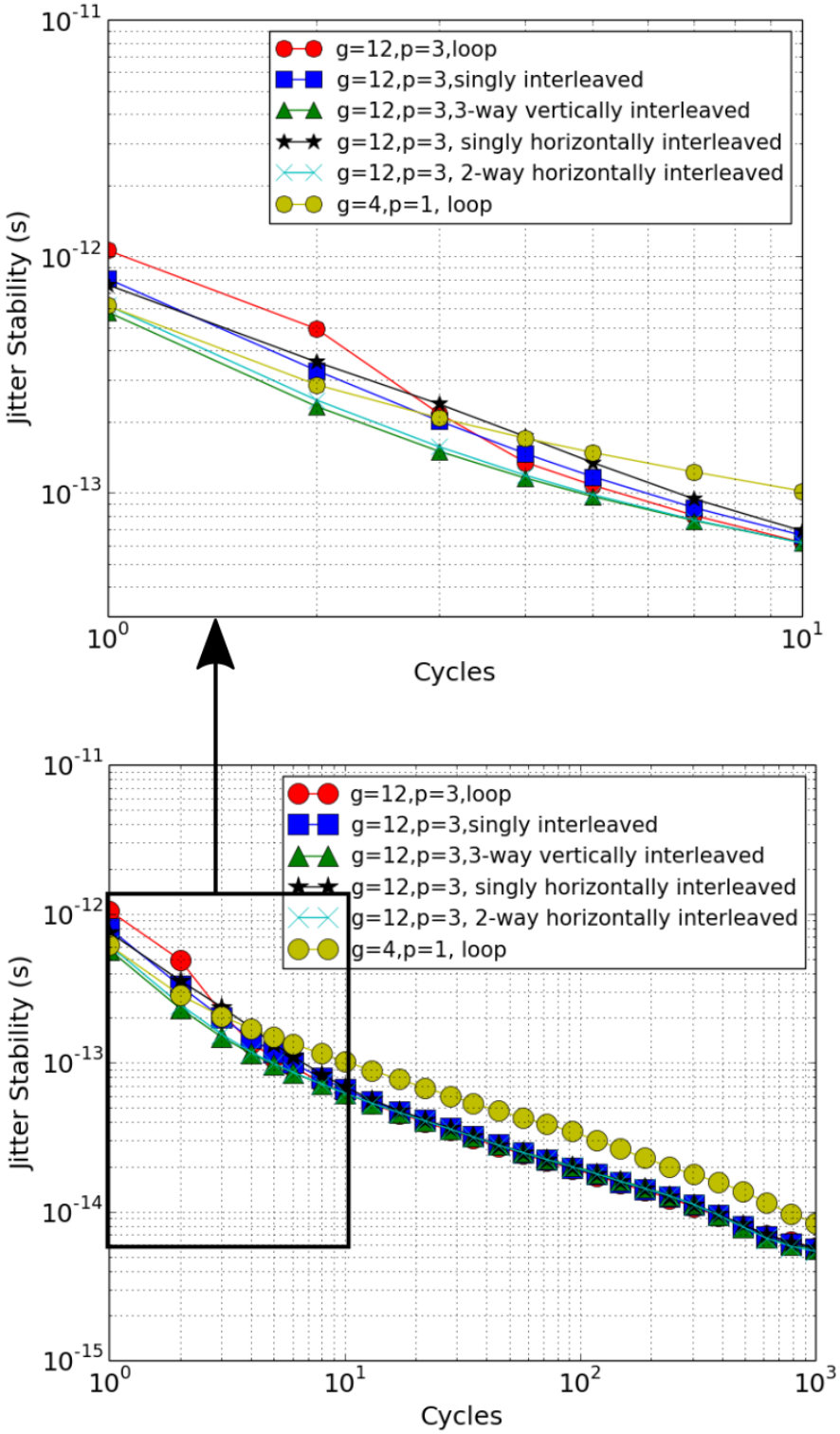


Figure 6.6: Comparison of stability of g=12 interleaved rings

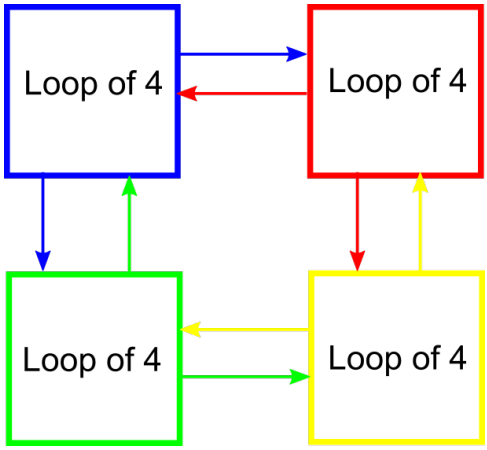


Figure 6.7: Physical placement of horizontally interleaved clocks

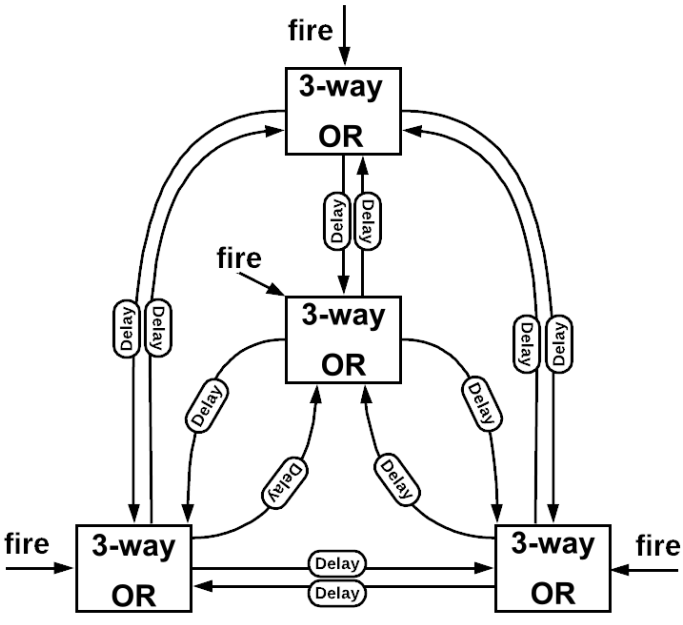


Figure 6.8: Tetrahedron clock structure

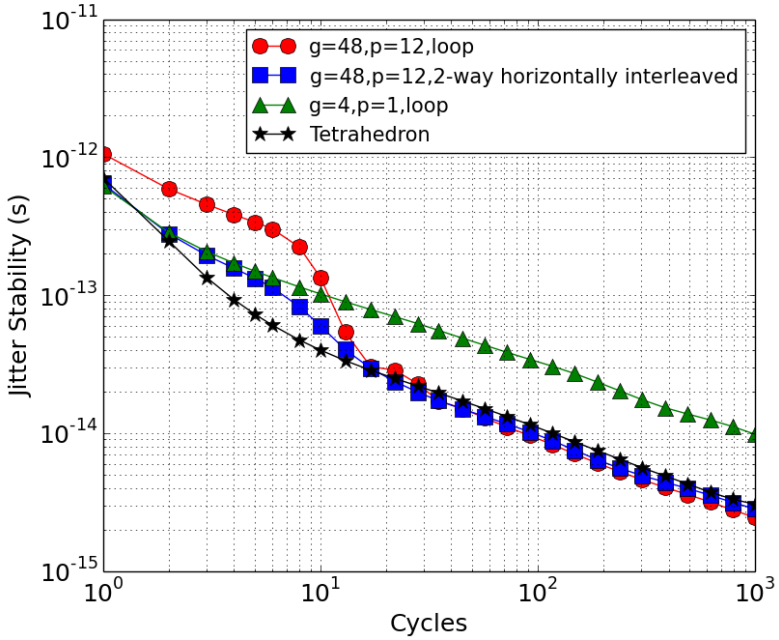


Figure 6.9: Comparison of stability of g=48 interleaved ring tetrahedron clock

6.2 Voltage Controlled and Injection Locked CPOs

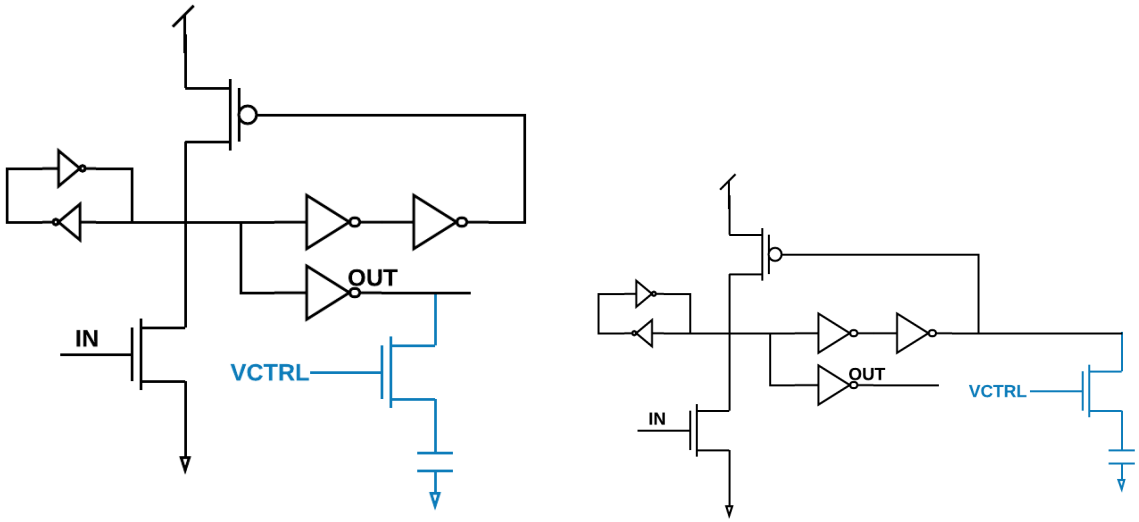
CPOs can be used as VCOs using three tuning mechanisms, each of which results in a different K_{VCO}:

1. By varying the supply voltage. In 130 nm technology, for a $5GHz$ CPO, supply tuning results in a $K_{VCO} = 4.5GHz/V$

2. By changing the forward path delay as shown in Fig. 6.10a. This results in a smaller K_{VCO} of $\approx 400MHz/V$

3. By changing the feedback path delay as shown in Fig. 6.10b. This results in an even smaller K_{VCO} of $\approx 100MHz/V$. The frequency change in this case is due to modified delay separation dynamics due to a modified pulse-width, and this technique would be ideal for low-noise tuning if a large tuning range is not required.

Using one of these tuning mechanisms, CPOs can be used as VCOs for Phase-Locked-Loops. While PLLs were not analyzed or built in this work, injection locked CPOs have been fabricated in 130 nm. Injection locking was achieved by firing external pulses into a single gate of the ring, with an injection strength $\approx 1/10th$ of the usual drive of gates for driving feedback pulses. Fig. 6.11 shows the jitter stability measurements for an 8-gate CPO, running 2-pulses and injection locked to fundamental and sub-harmonic frequency sources. This is a proof-of-concept that multi-pulse CPOs can indeed be locked easily to achieve low noise across low and high frequencies.



(a) Tuning forward path delay (b) Tuning feedback path delay

Figure 6.10: Method of tuning pulse gate delay

6.3 Transmission Line Stabilized CPOs

Transmission-line stabilized clock distribution offers a viable method to produce low skew, multi-phase VCOs. In particular, low jitter, low skew timing distribution is required for mixer first receiver architectures and N-Path filters[74][75]. Key features needed in such designs are full swing, non-overlapping and less than 25% duty cycle clock drivers.

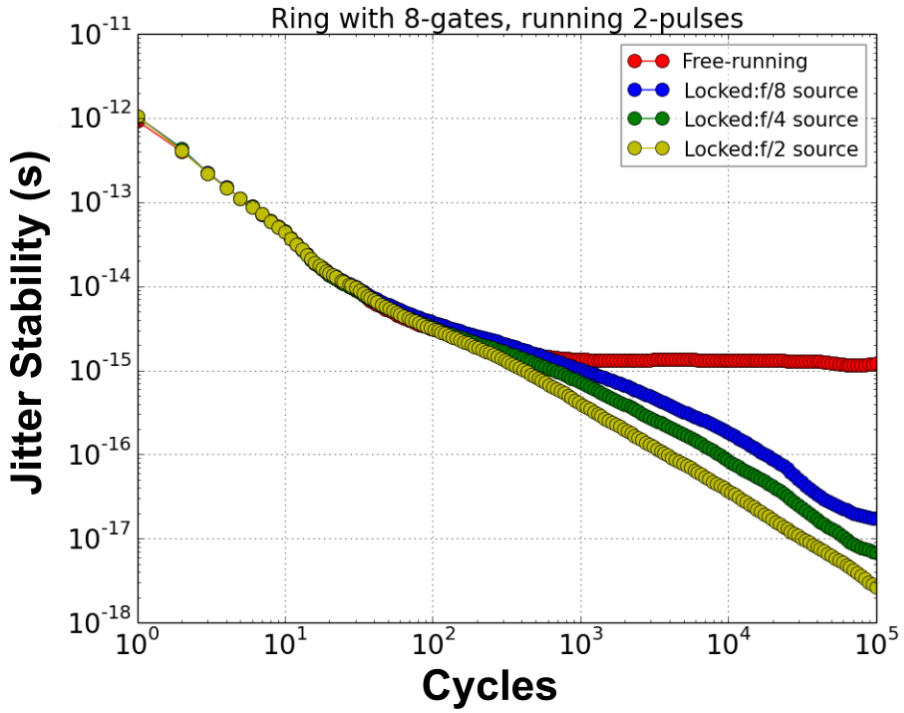


Figure 6.11: Measured jitter stability of an injection locked $g = 8, p = 2$ CPO

Transmission line systems such as salphasic distribution[76] and distributed transmission line amplification[77] provide sinusoidal or semi-sinusoidal clocks. They achieve fast clocking edges using additional buffering, unfortunately adding to the delivered jitter and power requirements. Rotary traveling wave oscillators (RTWO), which have been well analyzed [78][54] overcome these challenges. However, such designs provide limited signal swings and do not provide non-overlapping clock phases.

Previous studies on pulsed wave oscillators using linear transmission line elements demonstrated good phase noise properties[79]. These oscillators are implemented using off-chip, low-loss elements. A different implementation [80] demonstrated an on-chip transmission line pulsed wave oscillator. That design used distributed pulse regenerating amplifiers to counter comparatively high loss on-chip transmission line elements to obtain an oscillator that operates at 3.02 GHz but required 68.7 mW of power consumption. In this section, systematic design and architecture of a low-power, full-swing pulsed rotary wave VCO is presented. The presented design operates at about 80% higher frequency while consuming substantially less power than in [80], resulting in a $\approx 3\ dB$ figure of merit improvement. Like CPOs, it also utilizes pulse buffers as non-linear, full-swing, limiting amplifier stages to maintain the wave shape of a unidirectional, low-duty cycle, fast-slope traveling pulse. Since the event sampling accuracy directly relates to the slope of the clock edge, the presented high-slope pulsed rotary wave VCO is an ideal candidate for accurately capturing event timing at low injected jitter. Further, applications requiring timing distribution, or non-overlapping, low duty-cycle multi-phase clocks can effectively utilize these designs. Although the presented oscillator operates at 5.32 GHz, it provides 12-clock phases, and hence can be potentially used for capturing or synchronizing events at a much higher effective rates.

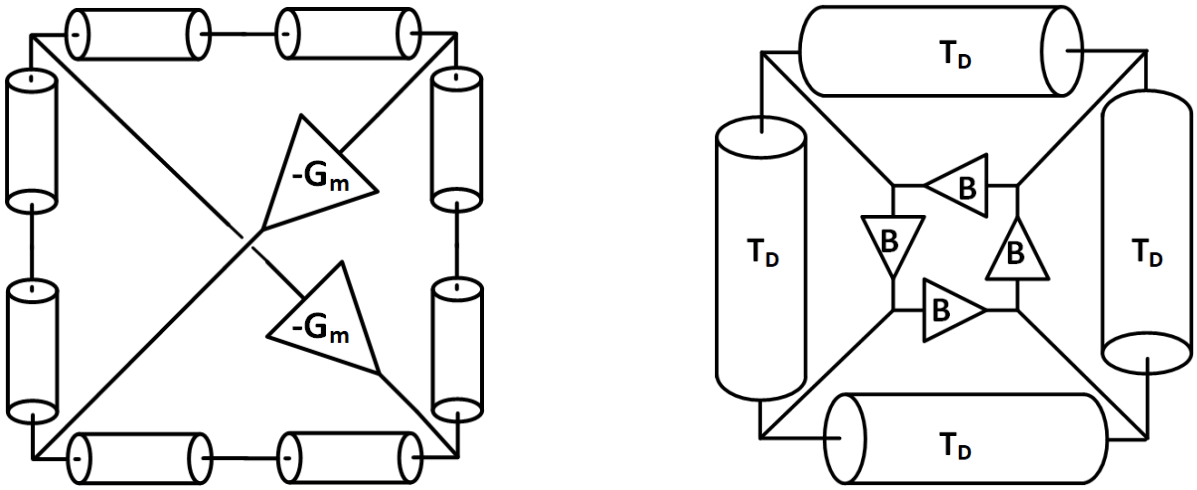
In this section, (1) A pulsed RTWO fabricated in GFUS 130 nm (8RF) technology is presented that preserves the slope of the traveling pulse and provides 12-low-duty cycle

clock phases. (2) A method to reliably start-up the oscillator is presented. The start-up methodology aids in both - lowering the power consumption, as well as maintenance of wave shape by minimizing reflections. (3) Design constraints and methods for tuning of parameters such as oscillator frequency, phase-stability and power are discussed. (4) Techniques for stable multi-pulse (overtone) behavior are described that enable a unique architectural feature: the same phase is available at multiple physical locations of the oscillator. This feature can be utilized for timing distribution and synchronization.

6.3.1 Oscillator Architecture

Conventional RTWOs have differential amplifiers along a transmission line loop (Fig.6.12a). The amplifiers are always connected to diagonally opposite ends of a transmission line. It is not possible to get non-overlapping clock output taps from the propagating wave in the loop. Also, no two locations along the loop have the same phase. On the other hand, the pulse RTWO (Fig.6.12b) allows propagation of a unidirectional pulse around the transmission line loop. Pulse regenerating amplifiers (labeled "B") are used to compensate for losses and create a pulse traveling wave. Multiple low-duty cycle, non-overlapping clock phases can be tapped from the transmission line. By propagating more than one pulse around the loop, it is possible to obtain identical phases at different loop locations. There are easy to satisfy circuit conditions which enforce uniform clock pulse spacing for multi-pulse clocks [81] .

Consider the segment shown in Fig.6.13. A pulse V_1 at the input of the segment goes through the transmission line and comes out as V_{1T} after delay T_D. The amplifier also generates an output pulse V_{1B}, after delay T_B. In stable operation, the amplifier output compensates for the loss in the corresponding transmission line segment. An output waveform V_2 is produced, that is the superposition of the two waves V_{1B} and V_{1T}. The



(a) Conventional RTWO (b) Pulse RTWO

Figure 6.12: Rotary Traveling Wave Oscillator architectures

shape of the output waveform is dependent on the relative phase difference of the two superposing pulses ($T_D - T_B$).

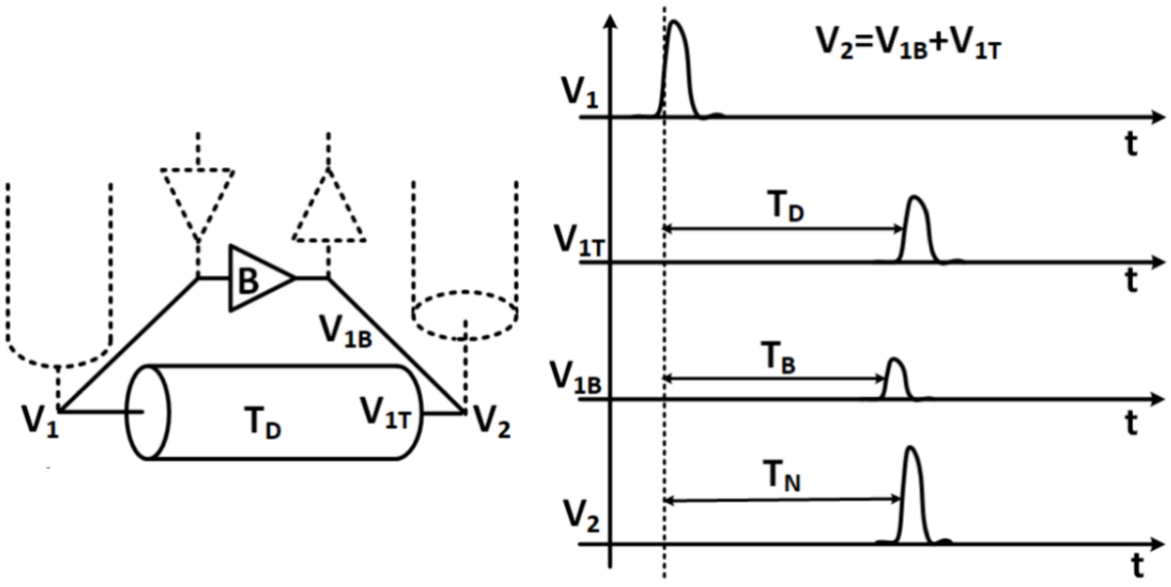
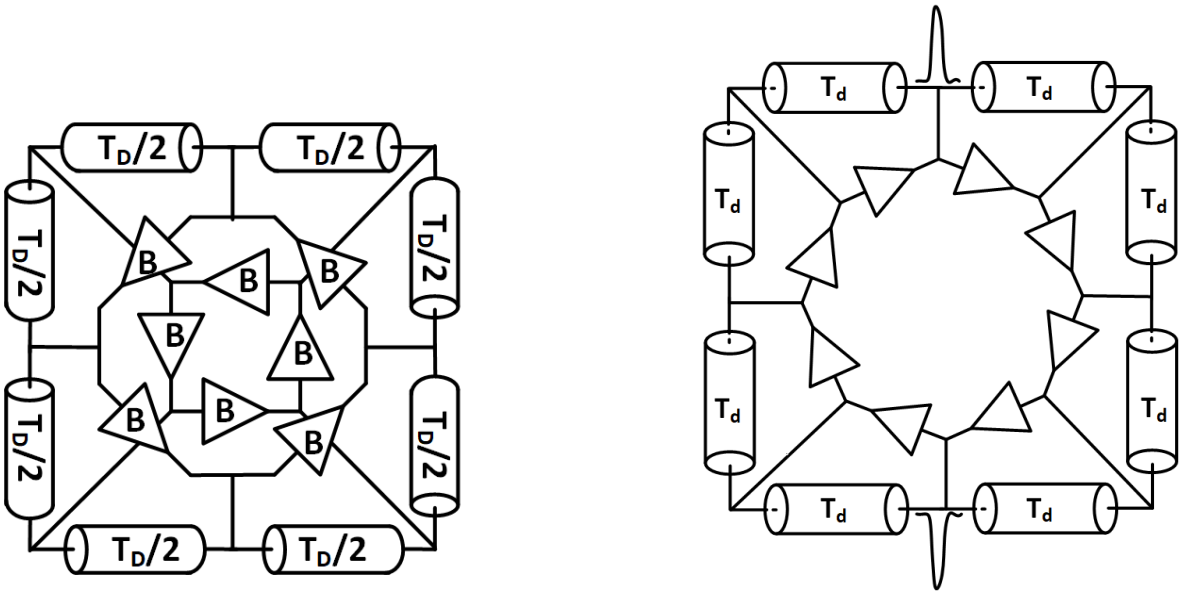


Figure 6.13: Pulse wave along a distributed transmission line

The pulse RTWO consists of one or more copies of the pulse amplifier, connected

in parallel to a transmission line segment, with drive that is enough to compensate for loss and time delay that meets the phasing constraints for regeneration (Fig.6.13). The pulse amplifier stages, by design, have a forward propagation delay that is roughly 1/4 of their reset time. The local transmission line segments are timed so that $T_D \approx T_B$, the condition where most of the amplifier power is in-phase with the transmission line wave. The portion of the output wave traveling in the reverse direction arrives at the input of the previous stage when the pulse amplifier is in its reset phase. It arrives too early to re-trigger the amplifier, thus suppressing the reflected wave. The pulse then propagates in a single direction (either clockwise or counter clockwise), set by the physical connectivity between amplifiers and transmission line segments. Fortunately, the required timing conditions are easy to meet, allowing stable operation over a fairly wide range of voltages and device variations.

The pulse RTWO architecture allows for several design alternatives to meet specific goals. For example, it is possible to interleave the amplifiers (Fig.6.14a), such that the transmission line is driven more frequently by smaller amplifiers. This results in a more uniform drive and availability of more phase taps from the buffer outputs. Another topology can have multiple pulses stored in the transmission line (Fig. 6.14b), allowing identical phases to be present at more than one physical location on the ring. This feature can aid in clock distribution networks where identical clock phases can be utilized without the extra jitter and skew associated with timing distribution. Multi-pulse stability and uniform timing distribution is maintained by the interaction between pulses in pulse amplifier circuits[81].



(a) Interleaved pulse RTWO (b) 2-Pulse RTWO

Figure 6.14: Pulse RTWO architectures

6.3.2 Oscillator Design

Pulse Amplifier The oscillator utilizes pulse gates similar to the ones used in CPOs. The output pulse shape is set by the internal reset loop of the pulse buffer and as long as the input pulse does not persist longer than the reset time, it is relatively insensitive to the input pulse shape or amplitude. After the pulse output, but before the reset has concluded, the input is insensitive to further pulses and the output is low impedance to ground, suppressing the reverse wave in the transmission line as shown in Fig. 6.16. The pulse RTWO uses the pulse amplifier shown in Fig. 6.15, that contains additional inputs V_{fire}, V_{bp} and V_{bn} that aid in oscillator start-up, as explained in the next section. V_{tune} is an analog voltage used for fine control of the oscillator frequency. The supply voltage (V_{dd}) is used as a coarse frequency control knob.

Startup Mechanism At startup, all the inputs of the pulse amplifiers are low-impedance to ground and there is no energy stored in the transmission line. In order to sustain os-

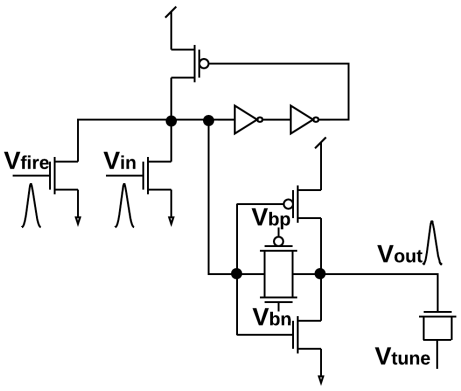
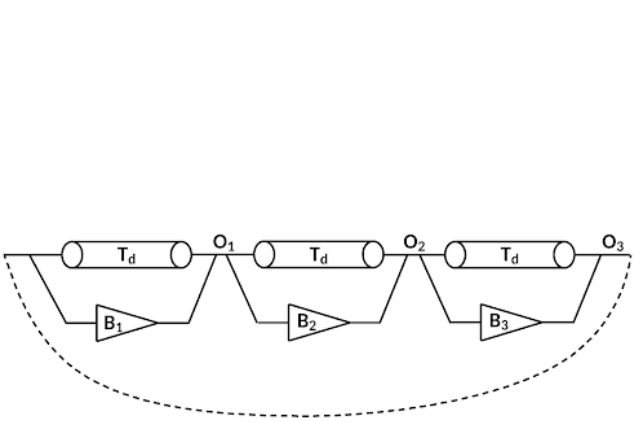
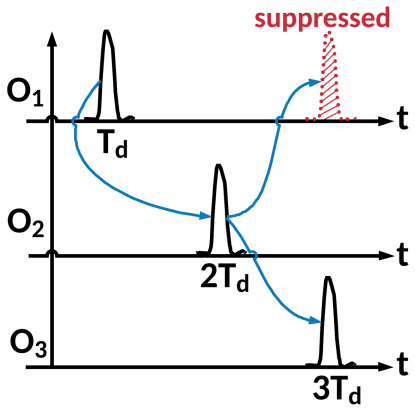


Figure 6.15: Pulse amplifier with added resistive feedback, fire input and fine tuning input



(a) Pulse RTWO representation



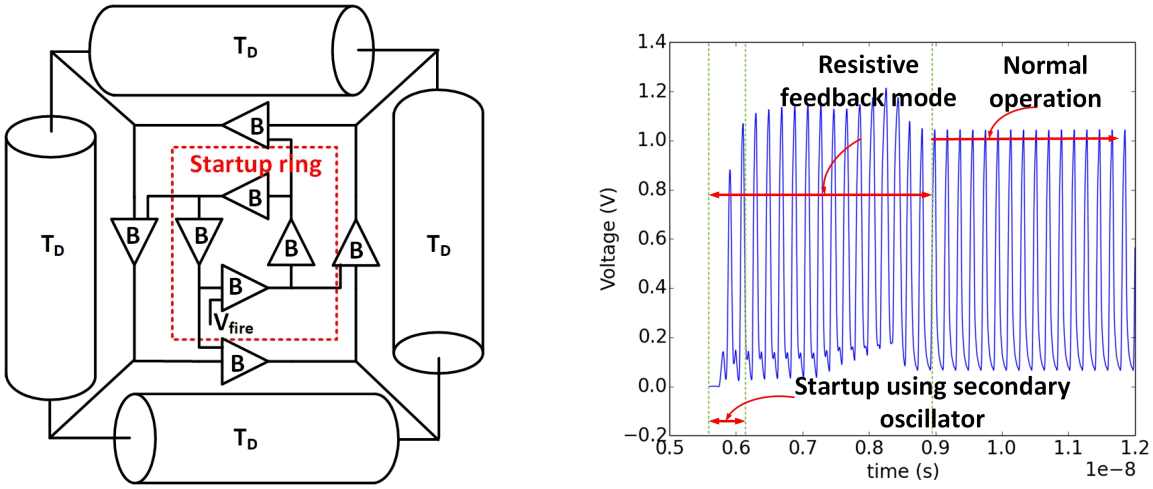
(b) Reverse wave suppression

Figure 6.16: Direction control for pulse RTWO

cillations in the transmission line, pulses must have sufficient amplitude to trigger the amplifiers. Unfortunately, a single pulse amplifier connected to a transmission line segment cannot create a pulse of sufficiently high amplitude. The output impedance of the amplifier is much higher than the transmission line, in fact it is just sufficient to compensate line loss. The following techniques ensure reliable startup:

Resistive Feedback Mode The pulse amplifier is modified with the addition of a transmission gate as shown in Fig. 6.15. Minimum sized PMOS and NMOS are used for the T-gate to ensure that the startup circuitry adds minimum capacitance to the output driver. The feedback resistance thereby achieved is $R_{on-p}||R_{on-n}$ and is relatively high (few KΩ). During startup, V_{bp} is set to *GND* and V_{bn} to V_{dd} externally in the current designs. This connects the input and output of the output inverter through the *T-gate*'s R_{on}, forcing the pulse amplifier into resistive feedback mode and charging the transmission line to $\approx V_{dd}/2$. In this mode, small pulses are amplified so that start-up pulses rapidly grow to the limiting amplitude. After oscillation is established, the amplifier is set to the pulse amplifier mode by removing the feedback.

Secondary Start-up Oscillator A "starter" oscillator with oscillation frequency equal to or slightly faster than RTWO is built via a ring of pulse transmission buffers. This oscillator uses internal delay instead of the stable transmission line. A pulse is then injected into the secondary oscillator at V_{fire} to start it. The outputs of the secondary oscillator drive the V_{fire} inputs of the RTWO amplifiers with correctly matched phases as shown in Fig. 6.17a. Once the RTWO boots up, the secondary oscillator is disabled. The secondary oscillator has series stacked NMOS at V_{in}. One of the series transistors is switched from V_{dd} to *GND* to disable it. The RTWO output in different startup modes is shown in Fig. 6.17b.



(a) Start-up circuit (b) Transients at start-up

Figure 6.17: Start-up of pulse RTWO

Transmission Line Design

The transmission line structure utilized in the fabricated design is shown in Fig.6.18. It comprises of $5\mu m$ copper signal lines shielded by $4\mu m$ ground lines spaced $3\mu m$ from the signal lines, with a ground plane underneath at a distance of $4\mu m$. The copper traces are $3\mu m$ thick. For the pulse RTWO fabricated in GFUS 130 nm (8RF) technology, the total transmission line length is $24mm$ built in a serperntine topology as shown in Fig. 6.19. It occupies a total area of $0.41mm^2$. The transmission line is divided into 12 equal segments of $2mm$ each with a buffer tap (B1-B12) at every $2mm$ segment, connected across a $4mm$ segment. The buffers as well as the secondary start-up oscillator are placed at the centre of the transmission line loop.

6.3.3 Simulation Results for a Fully Parasitic-Extracted Design

Post-layout simulation results for the fabricated pulse RTWO are summarized in Table 6.1. The oscillator operates reliably across the different process corners with frequency ranging between 4.65 GHz and 5.4 GHz at the nominal supply voltage. The change in

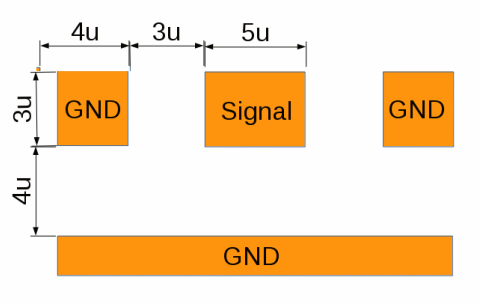


Figure 6.18: Transmission line structure

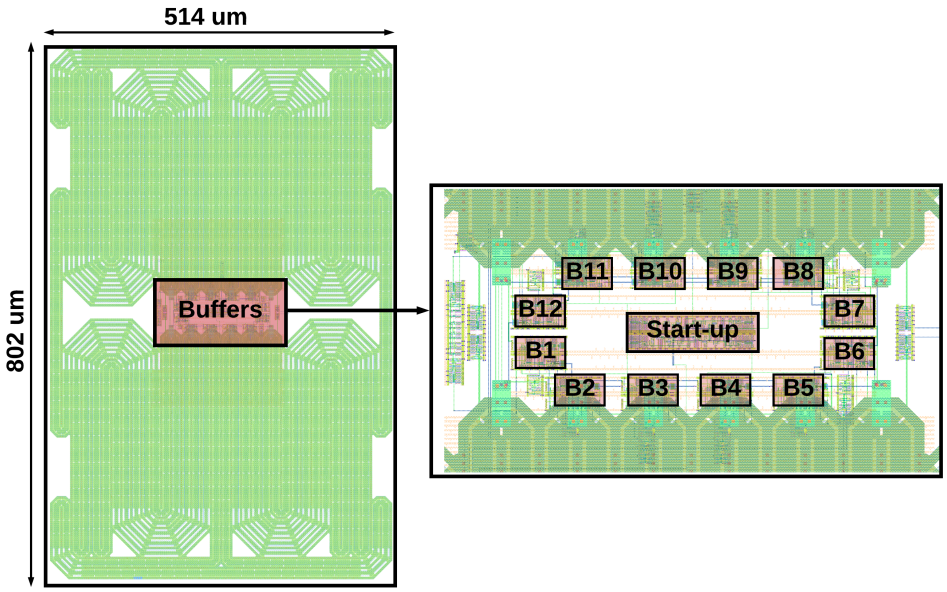


Figure 6.19: Oscillator layout: transmission line loop and buffers

buffer delay as a result of process variations causes the phasing relationship between the buffers and transmission line segments to vary, resulting in change in the shape of the propagating pulse and the oscillation frequency. Consequently, the phase noise is also affected and can see a worst-case degradation of $\approx 2.5dB$.

Table 6.1: Simulation results for a fully parasitic extracted design at $V_{dd} = V_{tune} = 1.5$V

Corner	Frequency (GHz)	Power (mW)	Pulse Width (pS)	P.N. @ 10 MHz (dBc/Hz)
TT	4.98	41.0	57.3	-127.5
FF	5.40	42.1	43.8	-126.9
SS	4.65	39.6	75.5	-125.2
FS	4.94	41.7	57.1	-127.7
SF	5.00	40.0	58.8	-127.4

Fig. 6.20 shows the timing of 4 of the 12 total phases of the oscillator under nominal operating conditions. The pulse exhibits a 27% duty cycle. At the typical corner, the simulated oscillator frequency can be tuned in the range of 4.41 GHz to 5.09 GHz for $V_{dd} = 1.15$V to 1.6V, providing a coarse $K_{VCO} = 1.51$ GHz/V. Further, at $V_{dd} = 1.5$V, varying V_{tune} in the range of 0 to 0.6V provides a fine frequency control between 4.61 GHz to 4.93 GHZ, providing a fine $K_{VCO} = 543.4$ MHz/V.

6.3.4 Design Constraints and Parameters

The fabricated RTWO has 12 output phases and a cross-interleaved topologyas shown in Fig. 6.21. Design choices were based on performance projections of transient noise simulations on the fully parasitic-extracted design.

Period: Fast rising edges minimize the conversion of voltage noise to time deviations [63]. Setting $T_D \approx T_B$ leads to constructive superposition and maximizes the rising edge slope of the propagating wave. Thus, the period of the oscillator is largely dependent on

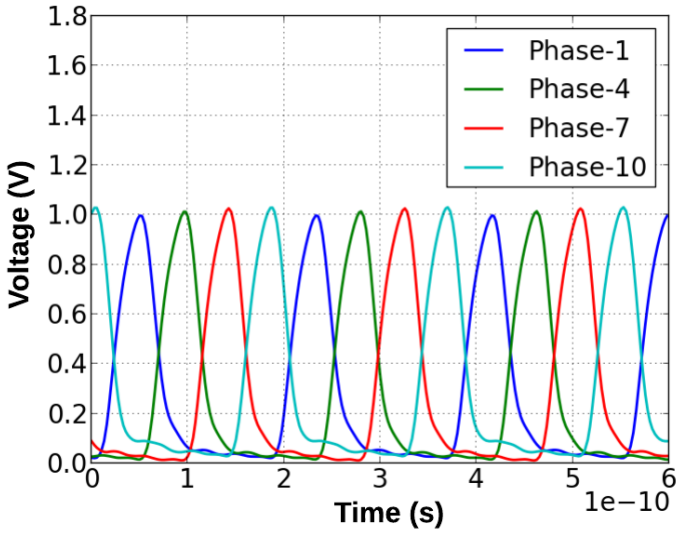


Figure 6.20: Waveform showing 4 of 12 clock phases, duty cycle = 27%

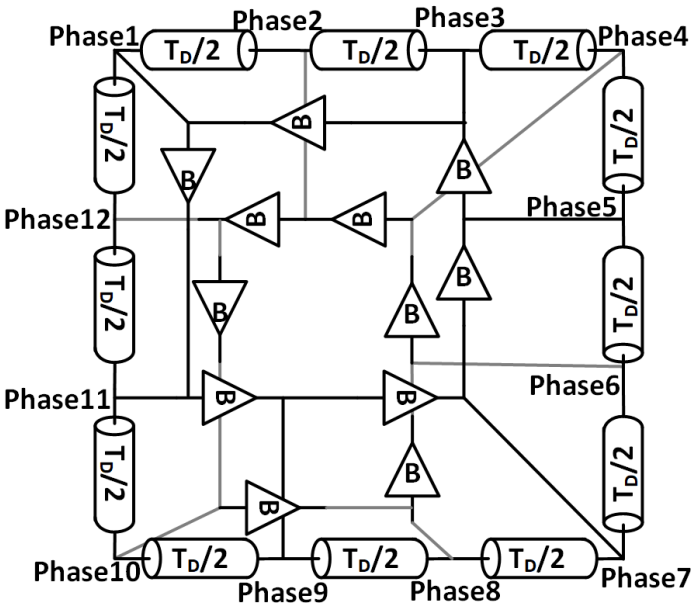


Figure 6.21: Interleaved 12 phase pulse RTWO

the total transmission line length, with gain elements added at the appropriate intervals. The phasing of the transmission line and amplifier impacts both power and noise as can be seen from the simulation results.

Buffered Segment Lengths: The maximum functioning transmission line length is determined by simulating the performance of minimum sized buffers and analyzing the performance across a number of possible single segment lengths. For the fabricated pulse RTWO $2mm$ was chosen to be the transmission line segment length in order to achieve low cycle-cycle jitter. The forward pulse buffer delay is greater than the propagation delay of a $2mm$ transmission line segment, thus the design needs interleaving to support both correct phasing and the attenuation characteristics of the transmission line.

Power: In simulation, a $24mm$ transmission line required 2 parallel minimum sized buffers to reliably maintain oscillations. This assumes the transmission line is divided into 12 segments with a buffer every $2mm$, connected across a $4mm$ segment. Hence a multiplier of 2 was chosen in the final design, which also causes the power consumption to be lower.

6.3.5 Measurement Results

The pulse RTWO fabricated in GFUS 130 nm (8RF) operates at 5.32 GHz ($V_{dd} = 1.5V$) with a $24mm$ transmission line length, $0.41mm^2$ area and 12 output phases. Fig. 6.22 shows the die micrograph of the fabricated oscillator chip ($1.5mm \times 1.5mm$). Voltage controlled frequency responses obtained by tuning V_{dd} and V_{tune} are shown in Fig. 6.23. The tunable frequency range of the oscillator is 4.35 GHz to 5.4 GHz with coarse and fine K_{VCO} of 1.7 GHz/V and 470 MHZ/V respectively for V_{dd} = 1V to 1.6V and V_{tune} = 0 to 0.6V. The corresponding power consumption lies in the range of 14.7 mW

to 48.8 mW.

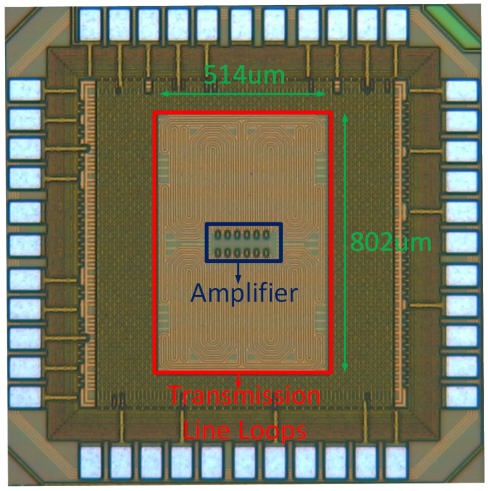
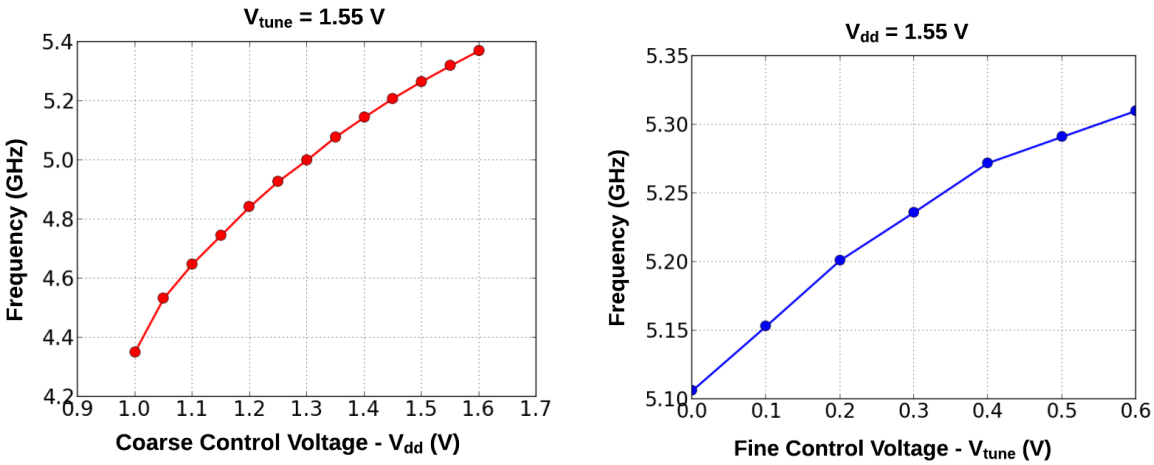


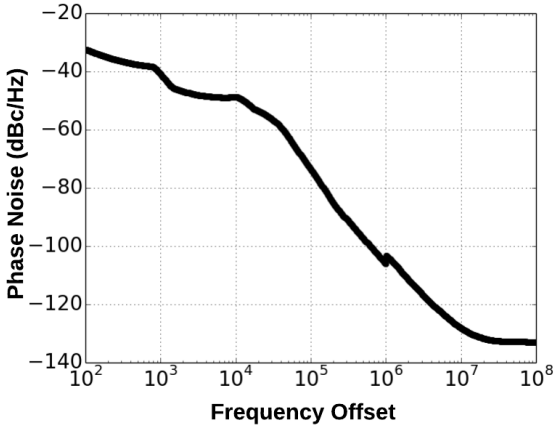
Figure 6.22: Die micrograph



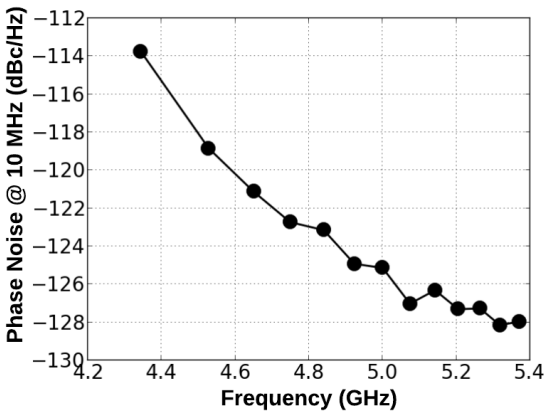
(a) Coarse tuning (b) Fine tuning

Figure 6.23: Measured voltage controlled frequency

Phase noise measurements were done using Keysight PXA signal analyzer N9030B, 3 Hz-26.5 GHz. Fig. 6.24a shows the phase noise of the pulse RTWO operating at $V_{dd} = V_{tune} = 1.55$V for offset frequencies between 100 Hz and 100 MHz. Fig. 6.24b shows the 10 MHz offset phase noise across the operational frequency range of the pulse RTWO. The improvement in phase noise for higher frequencies is due to the operation of



(a) Measured Phase Noise at $V_{dd} = 1.55V, V_{tune} = 1.55V$



(b) Phase Noise as a function of operating frequency

Figure 6.24: Phase noise measurements

Table 6.2: Comparison of measured pulse RTWO with other free running RTWOs

Ref	Node	Phase Noise	Frequency (GHz)	Power (mW)	Range (GHz)
This Work	130nm	-128.15dBc/Hz@10MHz	5.32	48.8	4.35-5.4
[78]	130nm	-134dBc/Hz@3MHz	2.00	30.2	1.7-2
[54]	110nm	-140.8dBc/Hz@3MHz	3.05	52.8	3.05-3.65
[80]	130nm	-131.75dBc/Hz@10MHz	2.93	68.7	2.53-3.02
[82]	110nm	-134dBc/Hz@3MHz	3.50	90.0	2.6-4.25

the oscillator at a higher value of V_{dd}, resulting in stronger compensation of loss by the buffers and a higher amplitude traveling pulse wave. As shown in table 6.2, although the fabricated RTWO operates at a higher frequency, it consumes substantially less power, provides a wide tuning range and has a comparable phase noise with other free running RTWOs. In addition, it provides 12 output phases with $\approx$ 26% duty cycle, 15.66 ps phase resolution and cycle jitter of less that 500 fs.

6.4 Conclusions

This chapter first presented complex CPO topologies that improve the high-frequency stability of simple looping CPOs. 2-way horizontally interleaved rings as well as tetrahedron rings were shown to be stable at low and high frequencies, and can potentially be utilized for distributing timing as well. Next, methods for voltage tuning CPOs and measurements of an injection locked CPO were presented, showing the viability of using CPOs as PLL VCOs. Finally, an on-chip low-power pulse RTWO that utilizes distributed pulse buffers for compensating transmission line loss. It involves propagation of a high slope pulse around the transmission line loop. The design achieves low phase noise at low power by ensuring that the superposition of the compensating buffers' output and traveling wave maximizes the wave slope. It provides a technique for low duty-cycle, full-swing, multi-phase clocking, that avoids additional power costs associated with timing distribution and is useful for applications ranging from clock data recovery to ADCs.

Chapter 7

Multi-Wire Phase Encoding (MWPE)

7.1 Introduction

It has been well established that the energy cost of moving data across a large scale System-on-Chip (SoC) or Network-on-Chip (NoC) is orders of magnitude higher than the cost of computation [41] [42] [1]. Most widely used high-performance I/O interfaces are based on either a source synchronous architecture [43], where the clock is transmitted on a separate channel along with parallel data channels, or an embedded clock architecture [44], where the clock is recovered from the transmitted data edges. These techniques utilize Delay Locked Loops (DLLs) or Phase Locked Loops (PLLs) at the receiver, that not only increase the static power consumption, but also require hundreds to tens of thousands of clock cycles to achieve a lock. This increases the link overhead, especially if the communication is burst-mode or is switched from different sources.

Transition signaling protocols, in contrast, encode data as signal transitions as opposed to signal levels, and thus do not require a sampling clock to recover the transmitted

data. This eliminates the static power consumption associated with a DLL or PLL in the link receiver, and also makes the link 'fire-and-forget' as the data can be recovered instantaneously. However, currently known transition signaling techniques such as 1-of-n Level Encoded Transition Signaling (LETS) [83], m-of-n RTZ encoding [84], relative order based encoding [85] and relative time based encoding [86] achieve this at the cost of substantially lower effective data bandwidth compared to conventional Non-Return-to-Zero (NRZ) level encoded links. Further, their use has been limited to short-distance and low-speed (< 2 Gbps) bus-based communication.

One differentiating work is a two phase Level Encoded Dual Rail (LEDR) link proposed by Ginosar et al. that has been shown to operate at bandwidths up to 67 Gbps in 65 nm technology [87]. However, this technique achieves a high data bandwidth by transmitting pulses as narrow as $15pS$, and may not be viable for multi-mm communication over band-limited, lossy and dispersive on-chip wires. Double-edge pulse-width modulation (DPWM) [88] on a single wire is another technique that encodes data in the width of pulses and has been shown to operate at 10 Gbps in 45 nm CMOS SOI process. In this chapter, we propose a general transition signaling protocol called Multi-Wire Phase Encoding (MWPE), that is designed to exploit timing correlations between multiple signaling wires to transmit data at effective rates (i) higher than the bandwidth limit of a single wire, (ii) higher than the rate of other known transition signaling techniques, and (iii) comparable to or higher than parallel conventional NRZ encoded links. On-chip links or on-interposer links [89] for SoCs and NoCs involving communication between multiple locally clocked but globally asynchronous domains would be the ideal target applications for this technique.

The organization of this chapter is as follows: The MWPE protocol, theoretical bandwidth limits and bandwidth comparisons are presented in Section 7.2; The link architecture and performance limits of a 6-wire MWPE link on practical on-chip channels

are presented in Section 7.3, circuits for the 6-wire MWPE link are implemented in 22 nm FDX technology and the link is characterized in terms of power, performance and reliability against PVT variations in section 7.4.

7.2 Multi-Wire Phase Encoding: Protocol and Theoretical Bandwidth Limits

Assume T_{min} to be the minimum pulse-width that can be reliably transmitted on a given channel, chosen based on length, loss and dispersion limits. This dictates that consecutive transitions on any particular wire should be apart in time by at least T_{min}, for a multi-wire link, transitions on *different* wires have no such constraint. MWPE on an n-wire link involves switching a single wire or multiple wires at intervals that are a *fraction* of T_{min} and encoding data in the identity of switching wire(s). We define Single-Transition or S-MWPE and Multi-Transition or M-MWPE as follows:

Single-Transition Multi-Wire Phase Encoding (S-MWPE) on n-wires involves switching *one* of the n-wires on every phase of a k-phase clock, such that transitions on different wires are apart by $\Delta T = T_{min}/k$, and $2 \leq k \leq n-1$. Fig. 7.1 shows an example of an S-MWPE link where $n = 4$ wires and $k = 2$ phases. For the next word transmission at time ΔT, any wire except wire-1 (w_1) could switch without violating T_{min}, as w_1 switched in the previous phase. For $k = 2$, this constraint is released in the next phase. Thus, on each phase there are 3 available wires that can transit, enabling transmission of $log_2 3$ bits of data at a rate $= \Delta T = T_{min}/k$.

By the same argument, an n-wire k-phase S-MWPE link can transmit $log_2(n-k+1)$ bits of data at a rate $= \Delta T = T_{min}/k$, resulting in an effective data transmission

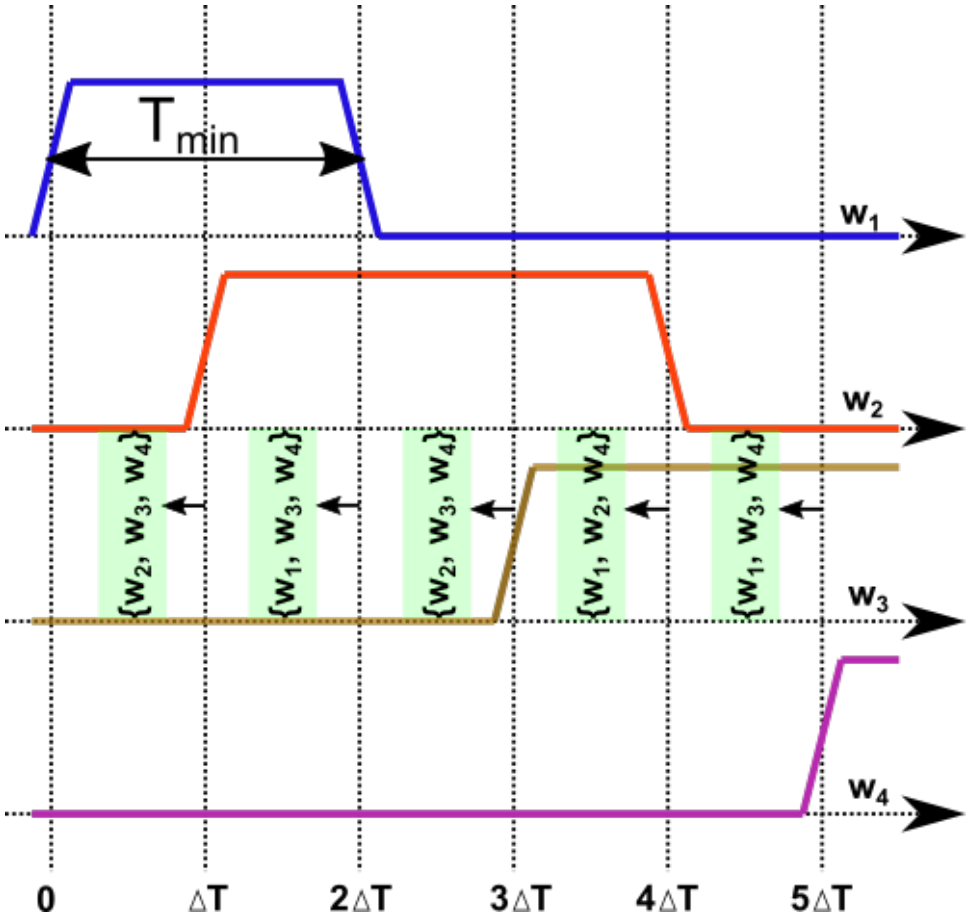


Figure 7.1: Transitions on an $n = 4, k = 2$ S-MWPE link

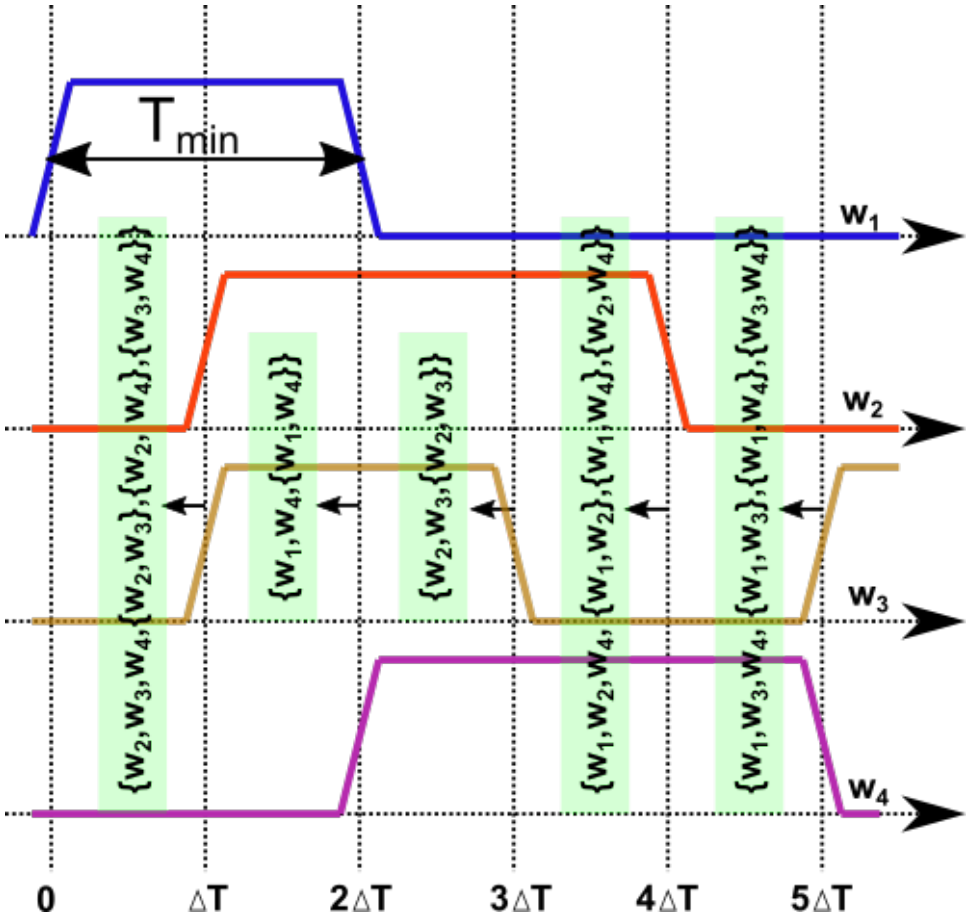


Figure 7.2: Transitions on an $n = 4, k = 2$ M-MWPE link

bandwidth of:

$$BW_{S-MWPE} = \frac{k \times log_2(n-k+1)}{T_{min}} \tag{7.1}$$

$k \geq 2$ ensures that ΔT is indeed less than T_{min}, and $k \leq n-1$ ensures that at least 2 wires are available to be transitioned on every phase and at least $log_2 2 = 1$ bit of data can be transmitted by means of transitions on each of the k-phases.

The second type of MWPE is **Multi-Transition Multi-Wire Phase Encoding (M-MWPE)**: For an n-wire link, it allows switching *one or multiple* of the n-wires on every phase of a k-phase clock, such that transitions on different wires are apart by $\Delta T = T_{min}/k$, $2 \leq k \leq n-1$. Fig. 7.2 shows an example of an M-MWPE link where $n = 4$ wires and $k = 2$ phases. In every clock phase, either 1 wire can switch (leaving 3 available wires to switch in the next time step) or 2 wires can switch (leaving 2 available wires to switch in the next time step, allowing at least 1 bit of data transmission). Let the state where 1 wire switched in the previous time step be called S_1, and the state where 2 wires switched in the previous time step be called S_2. When the state is S_1, there are 6 combinations of transitions possible on the 3 available wires: 3 single wire transitions + $\binom{3}{2}$ double wire transitions) and thus, $log_2 6 = 2.58$ bits can be transmitted. On the other hand, when the state is S_2, there are 3 combinations of transitions possible on the 2 available wires: 2 single wire transitions + 1 double wire transition) and $log_2 3 = 1.58$ bits can be transmitted. To estimate the *average* number of transmitted bits, we need to know the probability of being in each state. When in S_1, 3 of the 6 combinations result in the next state being S_1, and when in S_2, 2 of the 3 combinations result in the next state being S_1. Therefore, the probability of being in S_1 =

$$p_1 = p_1 \times (3/6) + (1-p_1) \times (2/3) \tag{7.2}$$

Solving, $p_1 = 4/7$ and probability of being in $S_2 = p_2 = 1 - p_1 = 3/7$. The average data transmission bandwidth for $n = 4, k = 2$ M-MWPE is then:

$$\frac{p_1 * log_2 6 + p_2 * log_2 3}{\Delta T} = \frac{4.3}{T_{min}} \tag{7.3}$$

The average bandwidth for other combinations of n, k can be obtained similarly and will be of the form shown in equation 7.4:

$$BW_{M-MWPE} = \frac{k \times \sum_{i=1}^{S}(p_i \times log_2 c_i)}{T_{min}} \tag{7.4}$$

where S in the total number of states for that encoding, p_i is the probability of being in state i, and c_i is the number of switching combinations available when in state i.

Algorithm 1 describes the steps for estimating the bandwidth of any n, k M-MWPE link. First, all possible states are calculated, where the length of each state is $k - 1$ (as every transitioning wire is busy for $k - 1$ time steps), and the elements of each state are ordered to denote how many wires transitioned in the previous $k - 1$ time steps. A maximum of $n - 2$ wires can be busy at any given time (to allow transmission of at least one bit in the next cycle). All permutations of lists of length $k - 1$ whose elements lie between 1 and $n - 2$, with sum of all elements less than or equal to $n - 2$ is the set of valid states for the given n, k. Next, the number of transitions possible for every state are counted. The count depends on: 1) the number of available wires to switch given the history of the current state and, 2) the maximum number of wires that can switch in the current time step given the history of the current state, such that at least 2 wires are available to switch in the next time step. The maximum number of wires allowed to switch is the minimum of these two values, and $count_{ij}$ stores the number of combinations of transitions for state i, when j wires switch. Then, for each $count_{ij}$ value, the next

state is formed by the history of the current state except the first element, as it will no longer be busy, and the number of wires that switch in the current state (j). Now, the probability p_i of being in each state S_i can be calculated by scanning through all the $count_{mj}$ values for each state S_m. If the next state for a $count_{mj}$ value is the same as the state for which the probability is being calculated (S_i), the ratio of this $count_{mj}$ value to the sum of all $count_{mj}$ values for the current state S_m is multiplied with the probability p_m of being in the state being iterated (S_m) and added to the total probability p_i of being in state S_i. Linear equations for the probability of each state create S (= number of states) linear equations in S variables. Solving these equations determines the state probabilities. The bandwidth for the encoding is calculated by calculating the average number of bits transmitted in every time step ($= T_{min}/k$).

The receiver decodes data by determining which of the n-wires switched, as well as maintainting the history of the previous $k-1$ transitions. It should be noted that ΔT cannot be arbitrarily small, as the received signal will show timing deviations caused by channel dispersion, inter-symbol interference, crosstalk, device noise and PVT variations. Ensuring that ΔT is greater than the total expected timing noise of the received signal will result in successful data reception.

Table 7.1 compares the effective data bandwidths for conventional level based NRZ encoding, other transition signaling protocols and Phase Encoding. 1-of-n Level Encoded Transition Signaling (LETS) [83] is similar to S-MWPE in that it is one-hot, but the transitions on any two wires are constrained to be apart by T_{min}, and not a fraction of T_{min}. m-of-n RTZ codes [84] involve transitions on $\binom{n}{m}$ wires every bit period and are denser than 1-of-n LETS codes, but the bit-period is twice of T_{min} as it is a Return-to-Zero protocol and transmits data on only one edge. Order codes [85] further improve the code density by encoding data in the relative order of transitions on n-wires, resulting in $n!$ available codes transmitted over a bit period $= T_{min} + (n-1)\Delta T$. Although counter

Algorithm 1 Calculating Bandwidth for M-MWPE

function CALCULATE_BANDWIDTH(n, k, T_{min})
  $states \leftarrow$ CALCULATE_STATES(n, k)
  for all $S_i \in states$ **do**
    $count_i \leftarrow$ CALCULATE_COUNT(n, k, S_i)
    $c_i = \sum_{all j} count_{ij}$
    $nextState_i \leftarrow$ MAP_NEXT_STATE($S_i, count_i$)
  end for
  $P \leftarrow$ CALCULATE_PROB($states, count, nextState$)
  $bandwidth_{M-MWPE} = \dfrac{\sum(p_i \times log_2 c_i)}{T_{min}/k}$
  return $bandwidth_{M-MWPE}$
end function
function CALCULATE_STATES(n, k)
  $states \leftarrow \{$all $S_i \in \{1, n-2\}^{k-1}$ such that $\sum_{j=1}^{k-1} S_{ij} \leq n-2\}$
  return $states$
end function
function CALCULATE_COUNT(n, k, S_i)
  $avail\_switch \leftarrow n - \sum_{j=1}^{k-1} S_{ij}$, $next\_switch \leftarrow n - 2 - \sum_{j=2}^{k-1} S_{ij}$
  $max\_switch \leftarrow min(avail\_switch, next\_switch)$
  for $j := 1 : max\_switch$ **do**
    $count_{ij} \leftarrow \binom{avail\_switch}{j}$
  end for
  return $count_i$
end function
function MAP_NEXT_STATE($S_i, count_i$)
  for all $count_{ij} \in count_i$ **do**
    $nextState_{ij} \leftarrow \{\{$ all $S_{ij} \in S_i$ such that $j \geq 2\}, j\}$
  end for
  return $nextState_i$
end function

30: **function** CALCULATE_PROB($states, count, nextState$)
31: **for all** $S_i \in states$ **do**
32: **for all** $count_m \in count$ **do**
33: **for all** $count_{mj} \in count_m$ **do**
34: **if** $nextState_{mj} == S_i$ **then**
35: $p_i \leftarrow p_i + p_m \cdot \frac{count_{mj}}{\sum_{allj} count_{mj}}$
36: **end if**
37: **end for**
38: **end for**
39: **end for**
40: $\sum p_i = 1$
41: Solve $S \times S$ matrix ($S = |states|$) to get all p_i
42: $P \leftarrow \{p_i\}_{i=1}^{S}$
43: **return** P
44: **end function**

intuitive, the efficacy of order codes actually reduces as n increases, because the time to transmit a symbol is also $\propto n$.

Encoding	Effective Bandwidth
Conventional level-based NRZ	$\frac{n}{T_{min}}$
1-of-n NRZ LETS [83]	$\frac{log_2(n)}{T_{min}}$
m-of-n RTZ [84]	$\frac{\log_2 \binom{n}{m}}{2 \times T_{min}}$
Order Encoding [85]	$\frac{log_2(n!)}{T_{min} + (n-1)\Delta T}$
Proposed S-MWPE	$\frac{k \times log_2(n-k+1)}{T_{min}}$
Proposed M-MWPE	$\frac{k \times \sum_{i=1}^{S}(p_i \times log_2 c_i)}{T_{min}}$

Table 7.1: n-wire Bandwidth Comparisons

It is clear from Table 7.1 that the proposed S-MWPE protocol outperforms the other

transition signaling protocols in terms of the effective bandwidth for sufficiently large values of k. Fig. 7.3 shows a comparison of the per-wire bandwidth of conventional NRZ encoding and MWPE as a function of the number of wires, n. For $n \geq 6$, S-MWPE can outperform the bandwidth of parallel conventional NRZ encoding without requiring clock-based decoding. In practice, the choice of a large n should be made with careful consideration to the encoder/decoder complexity and wire length matching requirements to preserve the phase of the transmitted signal. Fig. 7.4 shows the bandwidth comparison of different M-MWPE encodings. M-MWPE results in a higher effective bandwidth than certain S-MWPE cases, for example when $n = 4, k = 2$, S-SWPE achieves a bandwidth that is 80% of an equivalent NRZ encoded link, whereas, M-MWPE achieves a bandwidth that is 110% of NRZ. But from a switching point of view, M-MWPE is also prone to larger crosstalk induced time deviations, as it allows simultaneous switching of wires. In the next section, the performance of S-MWPE and M-MWPE protocols on practical on-chip channels will be evaluated to determine the actual achievable bandwidth given channel distortion. The analysis will be focused to $n = 6, k = 2$ MWPE links as they offer reasonably high per-wire and total bandwidths with low encoding/decoding complexity.

7.3 Link Architecture and Performance on Practical Channels

6-wire, 2-phase S-MWPE and M-MWPE links were implemented in 22 nm FDX technology and the link circuits as well as coupled transmission lines using W-models [90] were simulated in Hspice. The link architecture is shown in Fig. 7.5. At the transmit end, a 4-phase clock is utilized to read from a shift register block, random 6-wire transition encoded data, that follows the protocol rules described in section 2. The data for each of

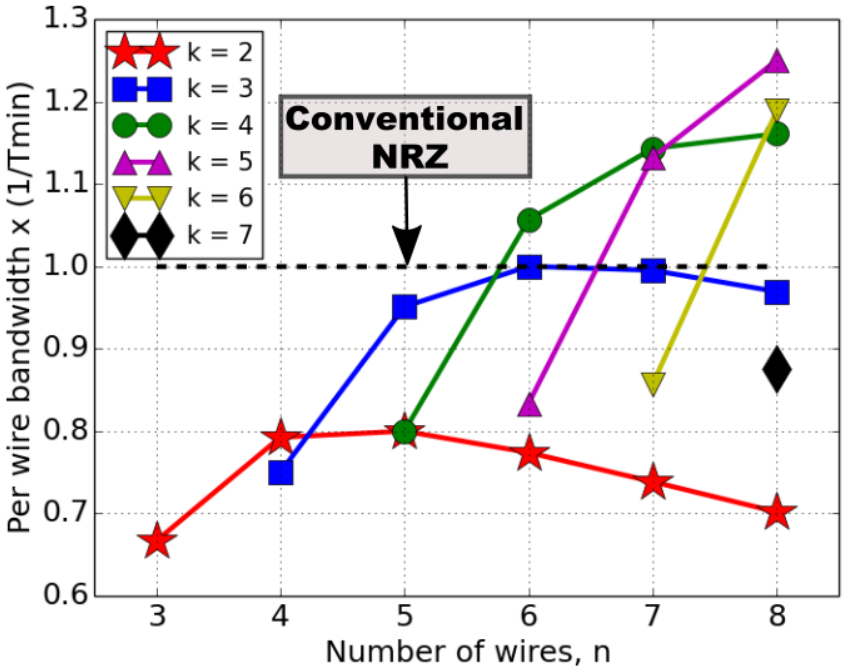


Figure 7.3: Bandwidth comparisons of S-MWPE with conventional NRZ links

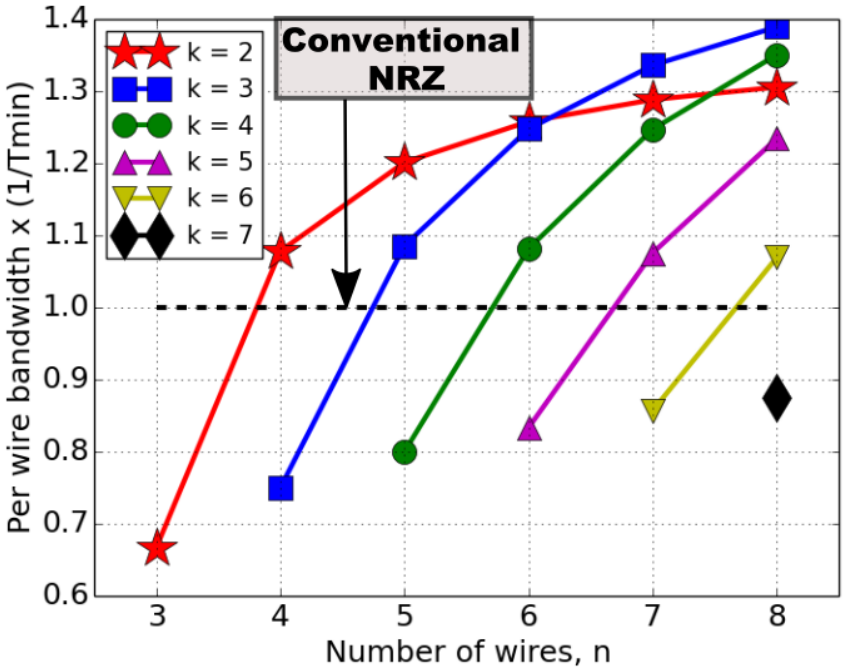


Figure 7.4: Bandwidth comparisons of M-MWPE with conventional NRZ links

the 6-wires is one-hot encoded and transmitted on 4-wires: 2-wires switch on phase P1 and 2-wires switch on P3, and a high value would indicate that the wire is selected to switch on one of the four clock phases. The data switched on phase P1 drives the Q-phase encoder block, which in turn uses clock phases P3 and P4 to switch the transmission lines. Similarly, the data switched on phase P3 drives the I-phase encoder block, which uses clock phases P1 and P2 to switch the transmission lines. The encoder uses interleaved clock phases to ensure settling of the read wire select values before they are used to switch the line. The line drivers drive 6-coupled single-ended transmission lines. The lines are terminated at the receiver, where a comparator first amplifies the received data, converts rising and falling edges to pulses, which are finally retimed to occur at $k \times \delta T$ intervals by the retimer block. The output of the retimer consists of pulses on 2-channels for each wire, whose presence indicate that the corresponding wire transitioned in the respective clock phase.

In order to evaluate the maximum allowable bandwidths for $n = 6, k = 2$ S-MWPE and M-MWPE links on practical on-chip channels, the link was driven by random data (following the rules specified by the encoding protocol) with the minimum pulse width on any wire (T_{min}) ranging between $40pS$ and $80pS$. Timing deviations in the received data due to channel dispersion, inter-symbol interference and crosstalk effects were then extracted to analyze the reliability of the link at different lengths. The first channel that was simulated is shown in Fig. 7.6. It consists of 6-unshielded local chip wire microstrip lines with the parameters shown in the figure. The outer two wires have a larger C_o/C_c value than the inner two wires and the values shown in the figure are the ones that lead to minimum C_o/C_c. A channel comprising of 6-shielded coplanar waveguides as shown in Fig. 7.7 was also simulated that shows significantly smaller crosstalk effects, but consumes nearly double the wiring resources. While eye-diagrams are a useful metric to evaluate the reliability of NRZ encoded links involving a sampling receiver clock, their

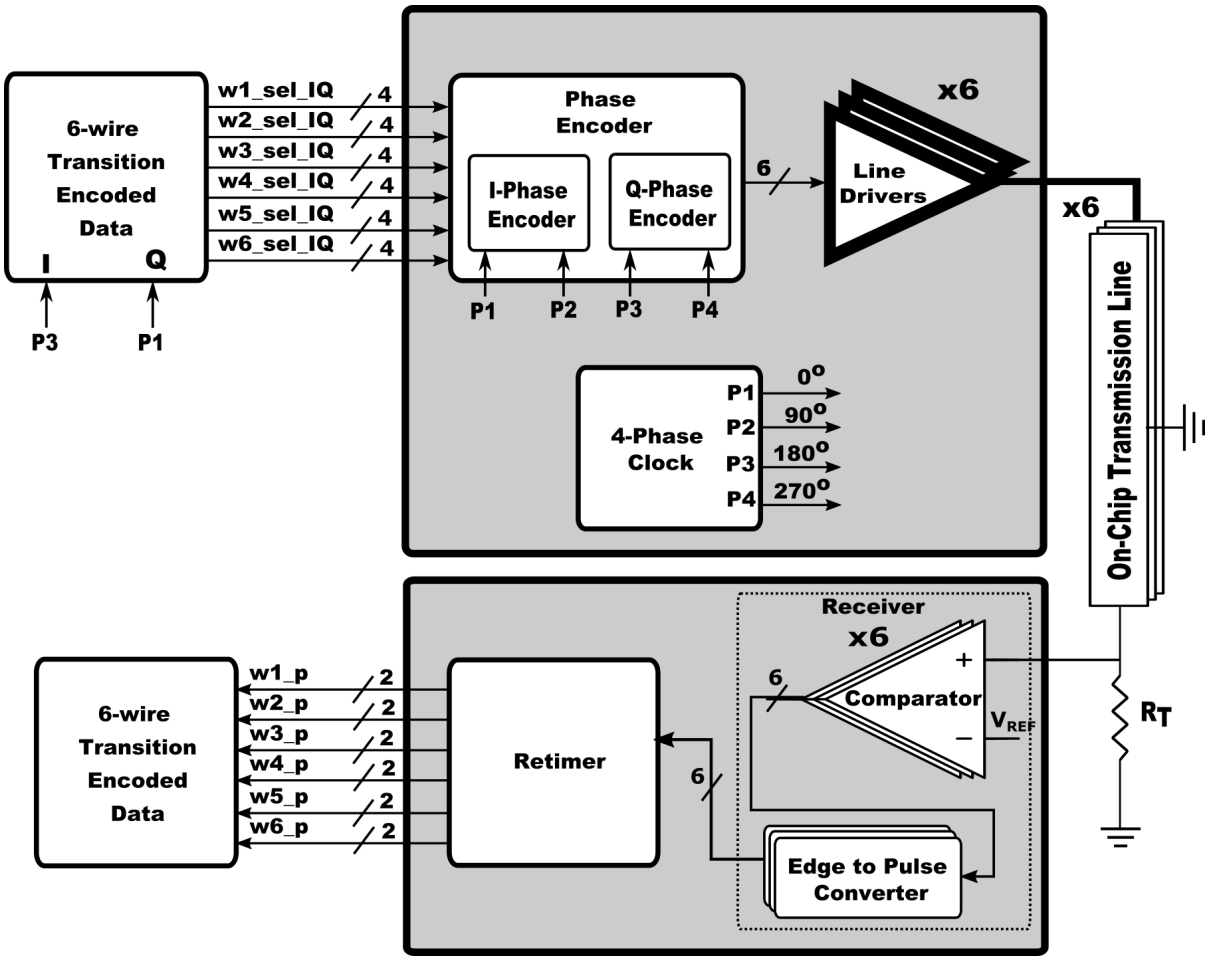


Figure 7.5: 6-wire MWPE Link Architecture

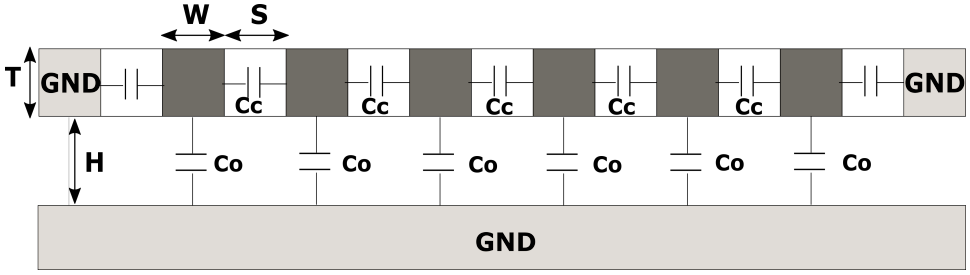


Figure 7.6: 6-wire coupled microstrip line topology; $W = S = 3\mu m, H = 1\mu m, T = 90nm$; $C_o = 230pF/m, L_o = 165nH/m, C_c = 56pF/m, R_o = 1.93K\Omega/m$

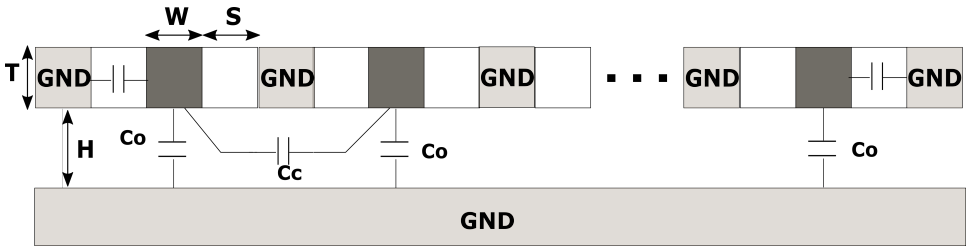


Figure 7.7: 6-wire shieldied coplanar waveguide line topology; $W = S = 3\mu m, H = 1\mu m, T = 90nm$; $C_o = 230pF/m, L_o = 160nH/m, C_c = 8.5pF/m, R_o = 1.93K\Omega/m$

direct use is not appropriate for evaluating MWPE links. MWPE links do not make use of any sampling clock at the receiver, and their reliability only depends on the timing deviations of *local successive* transitions, as opposed to global timing deviations across all transitions. Thus, *jitter*, as defined in Fig. 7.8 to be the deviation in the timing of consecutive transitions from the ideal timing of consecutive transitions (ΔT) is used to assess the reliability of MWPE links.

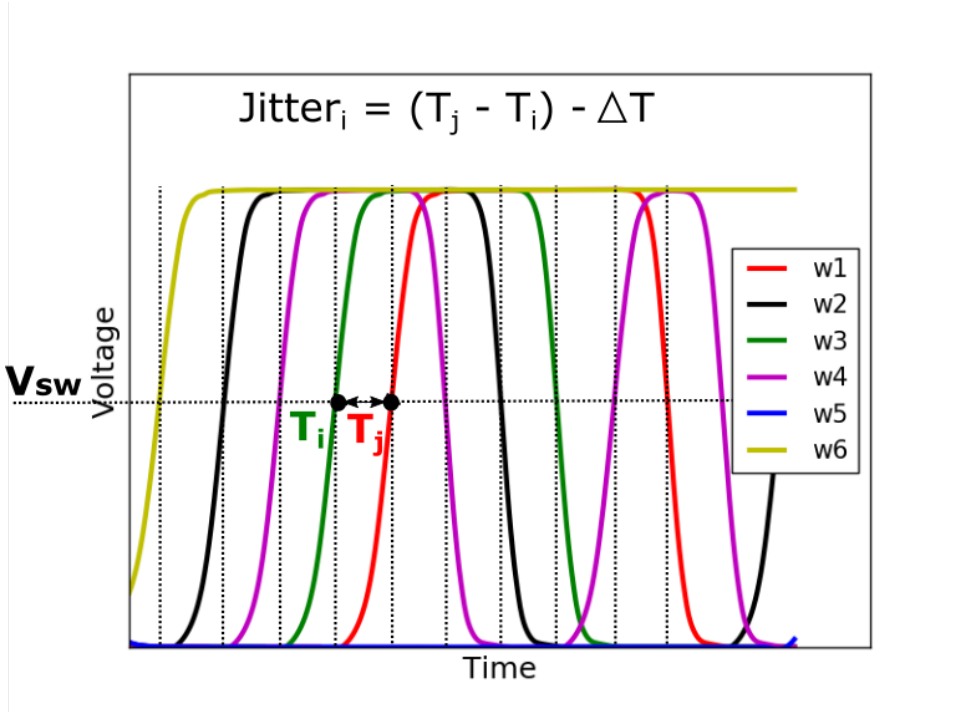


Figure 7.8: Jitter for a Phase Encoded link computed from timing of relative transitions

Fig. 7.9 shows the jitter for $n = 6, k = 2, T_{min} = 60pS$ S-MWPE links implemented using the unshielded and shielded transmission line topologies of Fig. 7.6 and Fig. 7.7, with line lengths of $4mm$. The following constraint is used to determine if the link operation would fail with the given jitter: the retiming circuit at the receiver is designed to reliably trigger at *minimum* consecutive transition times of $(2/3)\Delta T$, is known to not re-trigger if the spacing between consecutive transition is between $0 - (1/3)\Delta T$, and the window between $(1/3)\Delta T - (2/3)\Delta T$ is the "unreliability" window where the retimer may trigger but produce pulses too narrow to drive the subsequent logic, and hence the window should be avoided. Thus for the $S - MWPE$ link with a single transitioning wire on every phase, the link would not fail as long as the time between consecutive transitions is $\geq (2/3)\Delta T$, or if the jitter is $< -\Delta T/3 = -10pS$. As shown in Fig. 7.9, the S-MWPE link has enough margin when shielded signal wires are used, and is very close to failure with unshielded wires for wire lengths of $4mm$.

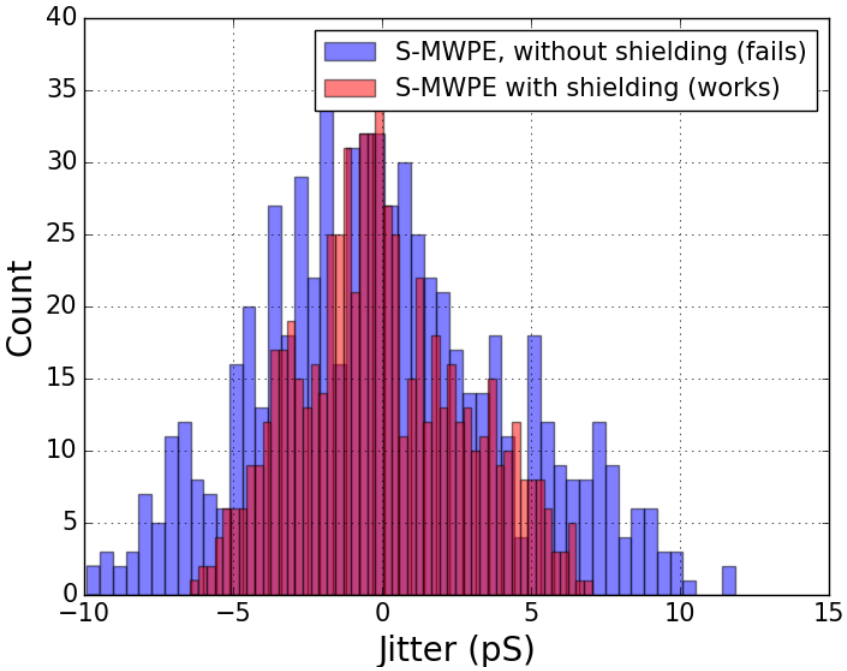


Figure 7.9: Jitter for $n = 6, k = 2, T_{min} = 60$ps S-MWPE links of length $4mm$

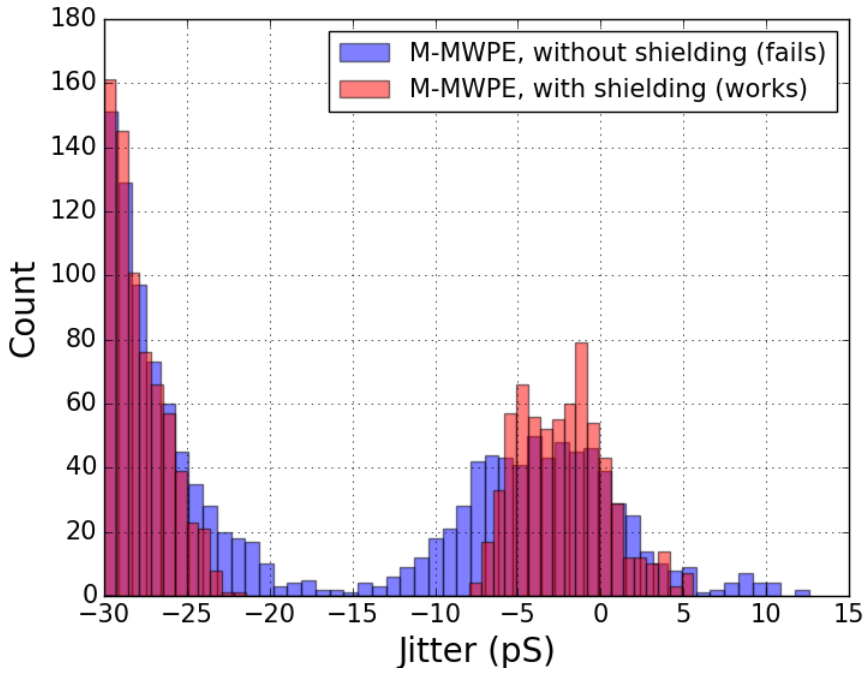


Figure 7.10: Jitter for $n = 6, k = 2, T_{min} = 60$ps M-MWPE links of length $4mm$

Fig. 7.10 shows the jitter for $n = 6, k = 2, T_{min} = 60pS$ M-MWPE links implemented using the same unshielded and shielded transmission line topologies. The reliability of M-MWPE links depends on one additional constraint: since multiple wires can transition in the same clock phase, the jitter for same phase transitions should be $< (1/3)\Delta T$ to avoid retriggering the retiming circuit, as the retiming circuit is designed to trigger *once* every clock phase (working described in the next section). The M-MWPE link has enough margin when shielded signal wires are used, and is fails with unshielded wires for wire lengths of $4mm$ as shown in Fig. 7.10. Fig. 7.11 shows the maximum wire lengths over which data can be reliably transmitted using shieldied and unshielded S-MWPE and M-MWPE links with varying minimum pulse widths T_{min}. When T_{min} is $40pS$, the unshielded M-MWPE link fails even for line lengths of $0.5mm$. Unshielded S-MWPE links are reliable for longer distances than M-MWPE links as they exhibit lower worst-

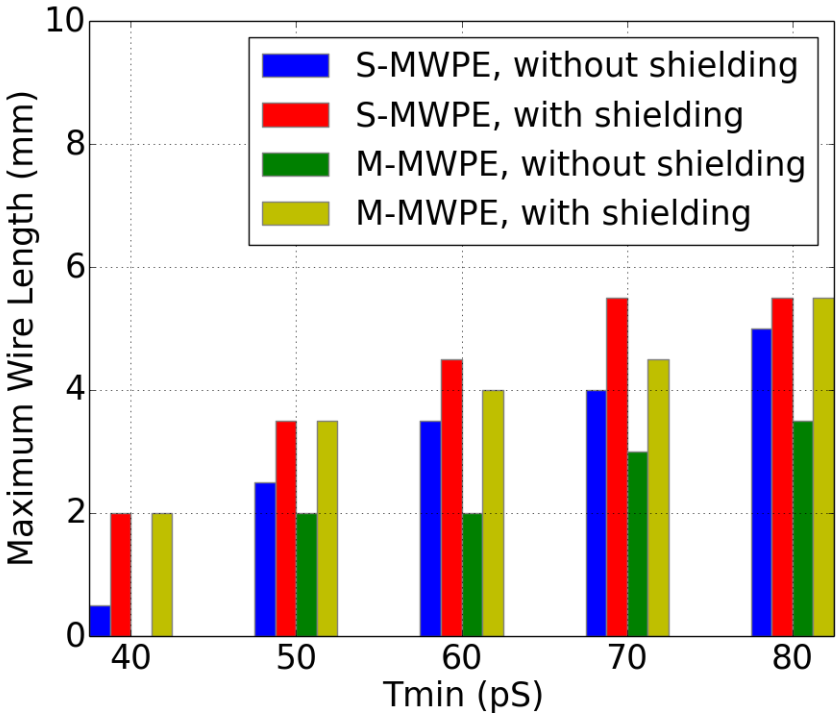


Figure 7.11: Maximum link length for which $n = 6, k = 2$ MWPEs are successful for the given channel for different T_{min}

case crosstalk induced timing deviations. Inter-symbol interference, loss and dispersion effects limit the minimum T_{min} that can be used. Note that, as these arguments are worst-case, restricted M-MWPE codes avoiding the worst-case crosstalk transition-pairs might well achieve higher performance, however, selection of such codes is beyond the scope of this work. The next section presents link circuits that are implemented in 22nm FDX technology to evaluate the maximum performance achievable in this technology, the power cost of moving data using MWPE and factors that govern the tolerance of the link under PVT variations.

7.4 Link Implementation

The 4-phase clock, phase encoder and retimer circuits were implemented using pulse logic. In 22 nm FDX technology, the F04 delay of an inverter implemented using low-V_t transistors is $\approx 5pS$, and for a pulse gate, the propagation delay is $\approx 7pS$, reset time is $\approx 20pS$. Thus, the retimer can retrigger at minimum intervals of $20pS$, and we use a $\Delta T = 30pS$ in order to meet the timing constraints described Section 7.3. The clock used in this implementation is a 12-gate, 1-pulse Collective Pulse Oscillator (CPO) designed to have a period $= T_{min} = 60pS$, and the 4-phases used in the link are tapped from every third gate of the ring.

The phase encoder circuit shown in Fig. 7.12 is a pulsed multiplexer utilizing Pulse AND-OR gates. The I-phase encoder MUXes either clock phase P1 or P2, and the Q-phase encoder MUXes either clock phase P3 or P4 to the input of a toggle T-latch, based on the values (levels) of the $w1\_sel\_I[0:2]$ and $w1\_sel\_Q[0:2]$ signals respectively. The T-latch thus produces a transition on wire-1 in one of the clock phases. The encoder circuit of Fig. 7.12 is replicated 6 times for the 6 wires and drives the line driver shown in Fig. 7.13. The driver is a single-ended make-before-break driver driving a terminated

transmission line of $Zo \approx 27\Omega$, the signal swing on each line being $\approx 200mV$.

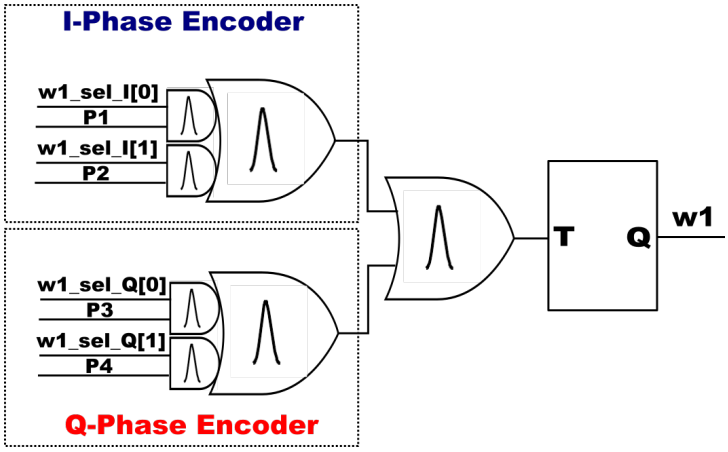


Figure 7.12: Phase Encoder Circuit

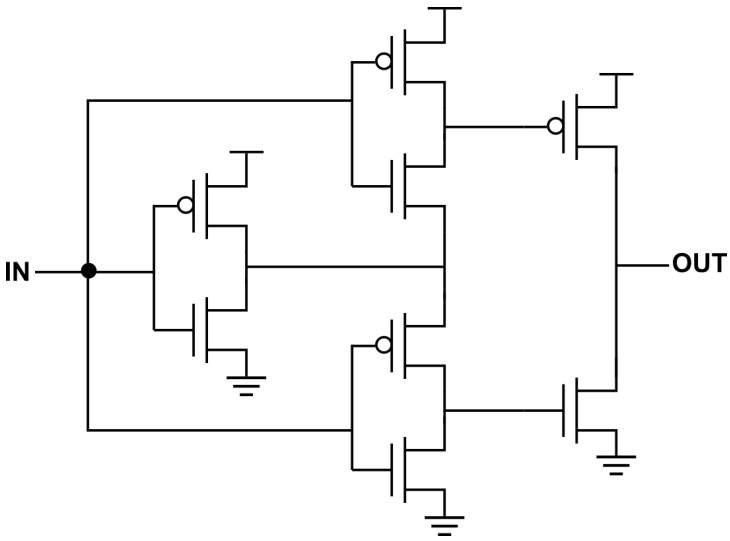


Figure 7.13: Driver Circuit

At the receiver, the signal is amplified and rising and falling transitions are converted to pulses (Fig. 7.14), which are retimed by the circuit shown in Fig. 7.15 to occur at intervals $= 2 \times \Delta T = T_{min}$. The retimer combines transition events (represented by pulses) on the 6-wires using a tree of Pulse-OR gates, whose output drives a T-latch (or a 1-bit counter) to indicate the clock phase in which the wire transitioned. In parallel,

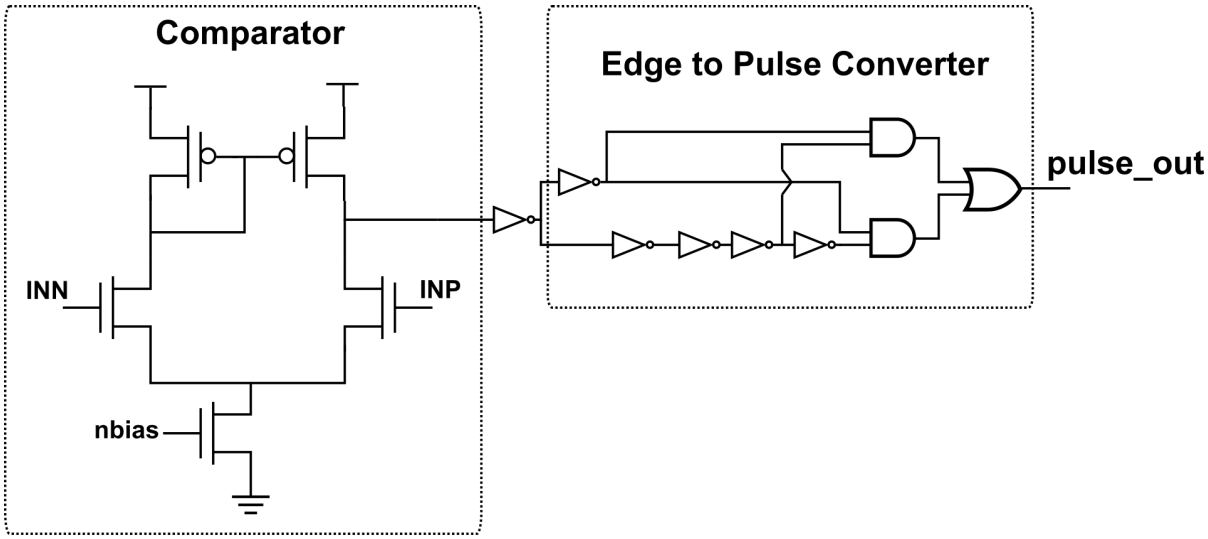


Figure 7.14: Receiver Circuit

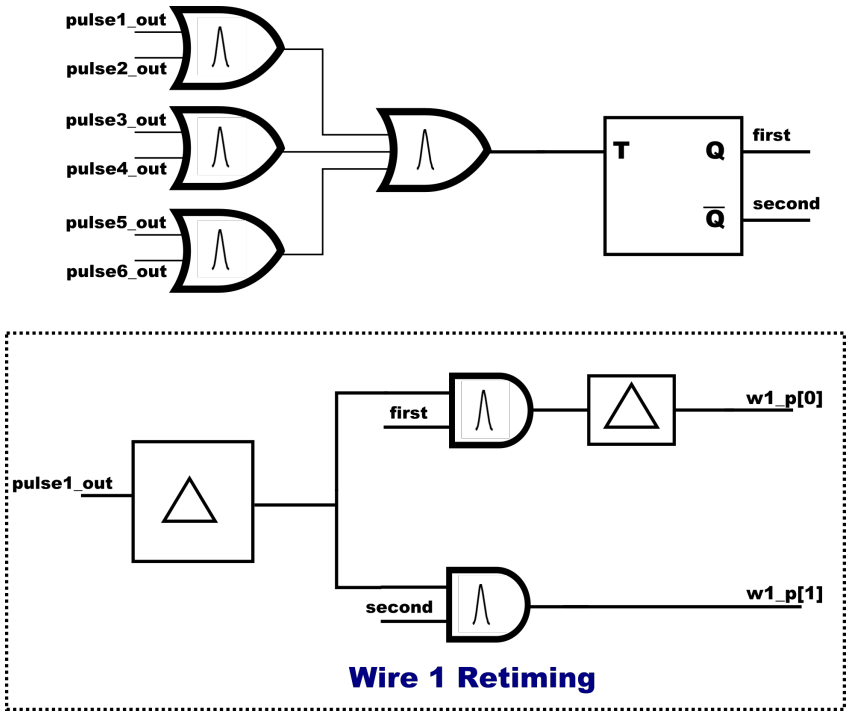


Figure 7.15: Retimer Circuit

delayed versions of transitions on all wires (encoded as pulses) are split into two channels based on the phase in which they occurred using Pulse-AND gates, and the output of the channel containing phase-1 pulses is delayed by one clock phase to be aligned with the pulses on the second channel. In general, for a $k-phase$ MWPE link, a $k-bit$ counter can be used to split events on each wire to $k-$channels identifying their phase of occurrence. Thereafter, the data can be decoded by evaluating groups of k-consecutive transitions. The following two-sided timing constraints should be met for the retimer circuit to work as intended:

Reset-time constraint: Minimum separation between consecutive events on different clock phases should be longer than the reset time of the OR-gates used in the combining tree to ensure all distinct events are captured.

Setup-time constraint:The T-latch should be set before the next event arrival, and therefore the delay of the T-latch should be less than the minimum expected separation between events on distinct clock phases.

Hold-time constraint:The delay of the T-latch should not be smaller than the width of pulses, else both Pulse-AND gates in the retiming logic would trigger.

Due to these constraints, the maximum allowable jitter due to channel distortions in Section 7.3 was limited to $\Delta T/3$.

The same circuits can be used for both S-MWPE and M-MWPE links. The S-MWPE link achieves a total bandwidth = $77.4Gbps$ ($log_2 5 = 2.32$ bits transmitted every $30pS$) and the M-MWPE link achieves $125.8Gbps$ (3.77 *average* bits transmitted every $30pS$) (Table 7.1) with total power consumption of $60.3mW$ according to the power breakdown shown in Fig. 7.16. At the receiving ends, the transmission lines are biased at $V_{dd}/2 = 0.4$V using termination resistors = $2Z_o$. The power reported here includes the static current flow through the termination resistors. However, by using a low output impedance op-amp regulator, the biasing can be set with a much smaller

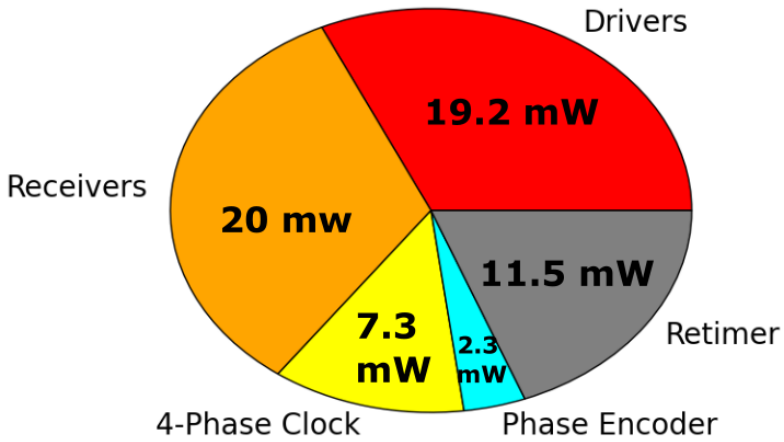


Figure 7.16: Power Breakdown

static power. An $n = 6, k = 2, T_{min} = 60pS$ S-MWPE link allows data movement across wires up to $3.5 - 4.5mm$ long (see Fig. 7.11 at an energy cost of $0.78pJ/b$, whereas an $n = 6, k = 2, T_{min} = 60pS$ M-MWPE link allows data movement across wires up to $2 - 4mm$ long at an energy cost of $0.48pJ/b$. By using transmit pre-emphasis and/or receive equalization, the transmission length could be further extended. Finally, these encodings are also robust against PVT variations, as long as the retiming circuit can be guaranteed to work even at the slowest corner as the transmitter works at the fastest corner and vice versa. Fig. 7.17 shows how the delay of a pulse gate varies with local and global process variations, $\pm 10\% V_{dd}$ variations and temperature variations between $0 - 100^oC$. With the tunable body-bias offered by 22 FDX technology, the n and p body biases of the transmitter circuits, particularly the clock, and the retiming circuit at the receiver end can be tuned to lower the TX/RX speed disparity and further improve the link robustness.

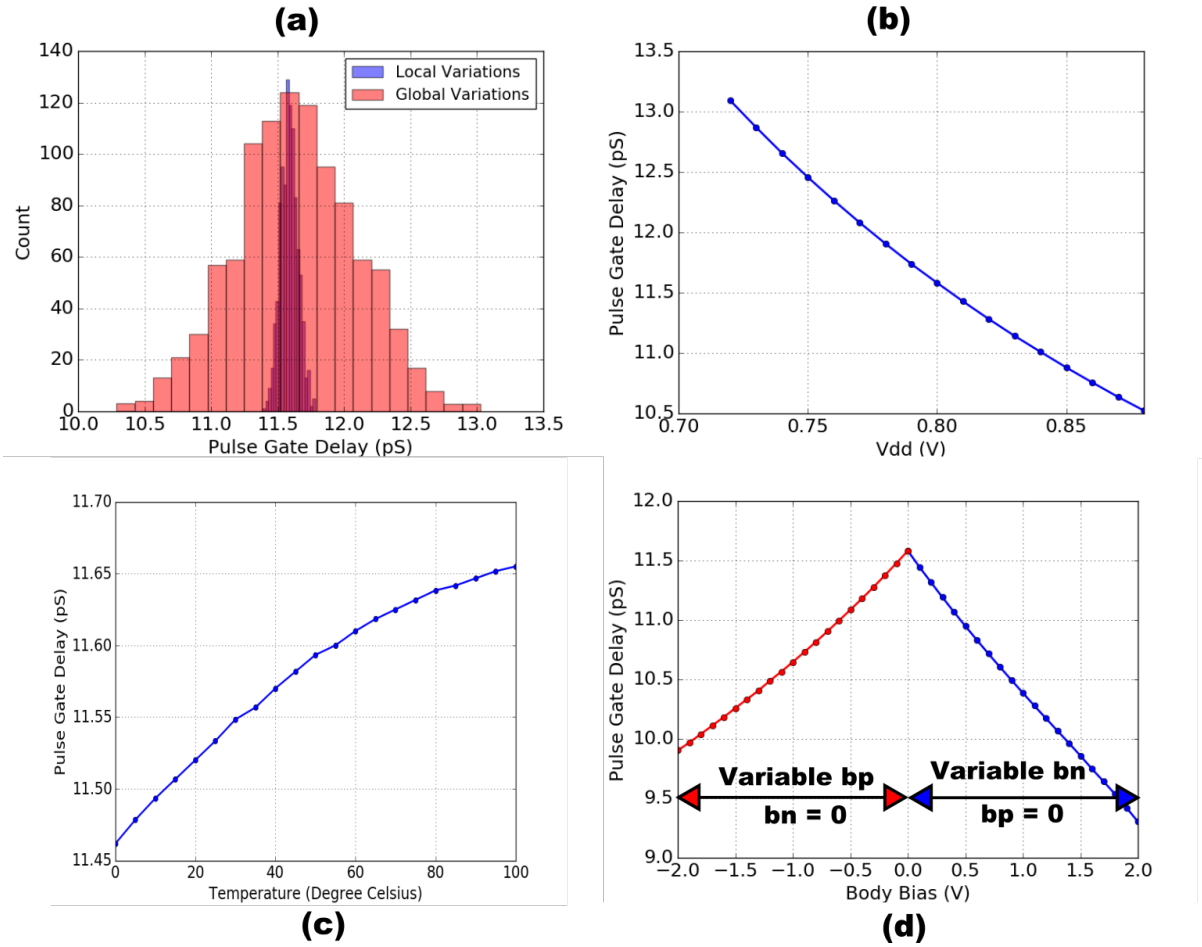


Figure 7.17: Pulse Gate delay variance due to (a) Process variations, (b) Voltage variations, (c) Temperature variations, (d) Body bias tuning

7.5 Conclusions

A general class of Multi-Wire Phase Encoding (MWPE) protocols, that encode data in the timing correlation of transitions on multiple signaling wires as opposed to edge levels were presented in this work. They offer the advantage of very high bandwidth data transmission on lossy and dispersive on-chip channels at rates higher than the effect bandwidth of the channel and comparable to or higher than conventional level encoded NRZ links. Further, these links do not use PLL/DLL based recovery mechanisms, leading to instantaneous data recovery and low idle-state power dissipation. In this chapter, theoretical bandwidth bounds for Single-Transition (S-MWPE) and Multi-Transition (M-MWPE) multi-wire phase encodings were first derived, and the performance of these encodings on practical on-chip/on-interposer channels was evaluated. Construction of link circuits for moving MWPE data across wires $2-5mm$ long in 22 FDX technology showed that the energy cost of these links is as low as $0.5-0.8pJ/b$.

Chapter 8

Pulse-based Time Domain Linear Filtering

8.1 Introduction

It was seen in chapter 6 that by averaging the arrival time of two events, the high-frequency stability of interleaved CPOs subjected to white noise was improved. The same concept can also be applied to create general time-domain filters that work to reduce *deterministic jitter* in a signal by mixing it with a signal that is driven by a precise source. Essentially, the precise source is set to occur at such a time that mixing it with the noisy signal cancels the deterministic jitter. In the design of such time-domain filters, we exploit the property of CPOs to generate precise clock phases, which can have a resolution much smaller than the delay of a gate when the number of gates and pulses is relatively prime.

Fig. 8.1 shows an 8-gate clock initialized with 3-pulses, thus producing 8-clock phases. For this clock, the delay of every gate is $b/(1 - 2.67m)$ (equation 3.16) and the phase resolution is a third of this gate delay. In this figure p_1 to p_8 are denoted as the outputs

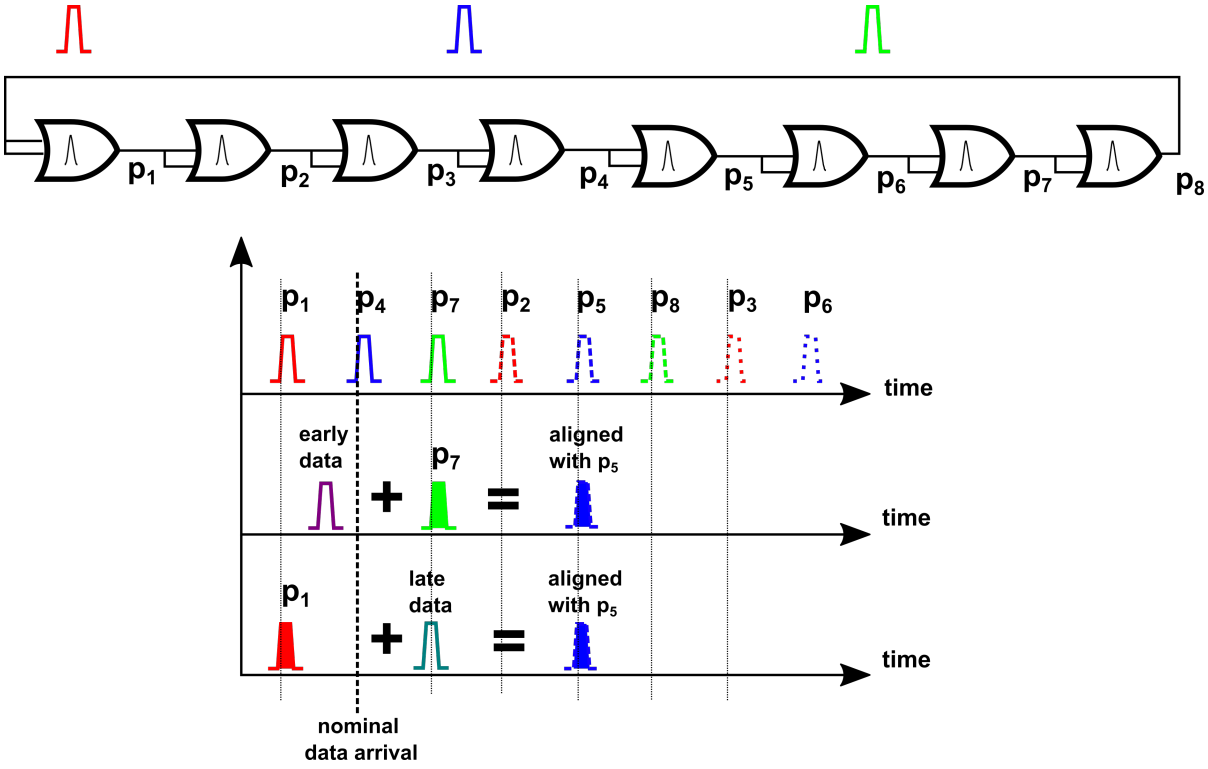


Figure 8.1: $g = 8, p = 3$ CPO with 8 phases used for filtering

of the 8-sequential gates forming the loop, and the sequence of phases is $p_1 \rightarrow p_4 \rightarrow p_7 \rightarrow p_2 \rightarrow p_5 \rightarrow p_8 \rightarrow p_3 \rightarrow p_6$. The figure shows an example where a data pulse is required to arrive at a nominal time that aligns with phase p_4. However, if the data pulse arrives earlier than the occurrence of p_4, mixing it with the next phase, i.e., p_7 will completely cancel the offset in the data pulse's timing if appropriate mixing ratios are selected. After mixing, the output pulse will be aligned in time with the pulse arrival time of the next gate, i.e., phase p_5. Similarly, if the data pulse arrives later than the occurrence of p_4, mixing it with the previous phase, i.e., p_1 will completely cancel the offset in the data pulse's timing if appropriate mixing ratios are selected. After mixing, this output pulse will also be aligned in time with phase p_5. In other words, this circuit interpolates the time between a noisy (but deterministic) signal and a precise signal to

regenerate the nominal arrival time. The pulse gate circuit shown in Fig. 8.2 implements

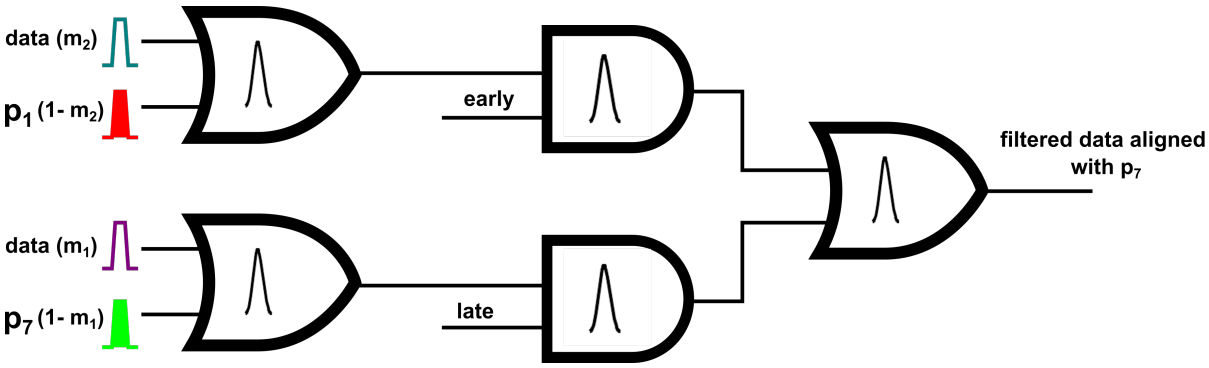


Figure 8.2: Filtering circuit implemented using pulse gates

this behavior where m_1 and m_2 are the mixing ratios. Thus, for input pulses arriving at $x_1[k]$ and $x_2[k]$, the arrival time of the output pulse of the gate $y[k]$ for a mixing ratio of m_1 is given by:

$$y[k] = b + m_1 x_1[k] + (1 - m_1) x_2[k] \tag{8.1}$$

where b is a constant. The delay separation dynamics of pulse gates are not accounted for in this equation as the intent here is to only use the time-averaging properties of a pulse-OR gate, and not it's temporal correction properties.

8.2 Inter-symbol interference induced timing jitter reduction

This technique was applied to the problem of cancelling inter-symbol interference induced timing jitter for a $1mm$ long on-chip shielded wire with the dimensions shown in Fig. 8.3. It was driven with PRBS8 NRZ data at $16.67Gbps$ or $T_{min} = 60pS$ at $200mV$ swing, amplified at the receiver and data edges were converted to pulses. Fig. 8.4 and Fig. 8.5 show the maximum jitter in the timing of rising and falling edges as a function

of the 2-bit history (b_2b_1) of the channel. These plots show the data dependence of jitter before filtering and the fact that the jitter is a strong function of the last bit before the transition and a relatively weak function of the second last bit.

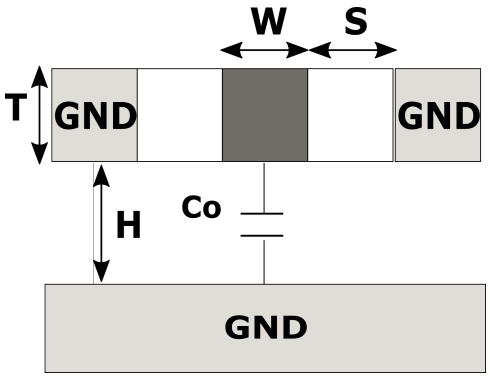


Figure 8.3: Shieldied coplanar waveguide with $W = S = 3\mu m, H = 1\mu m, T = 90nm$; $C_o = 230pF/m, L_o = 160nH/m, R_o = 1.93K\Omega/m$

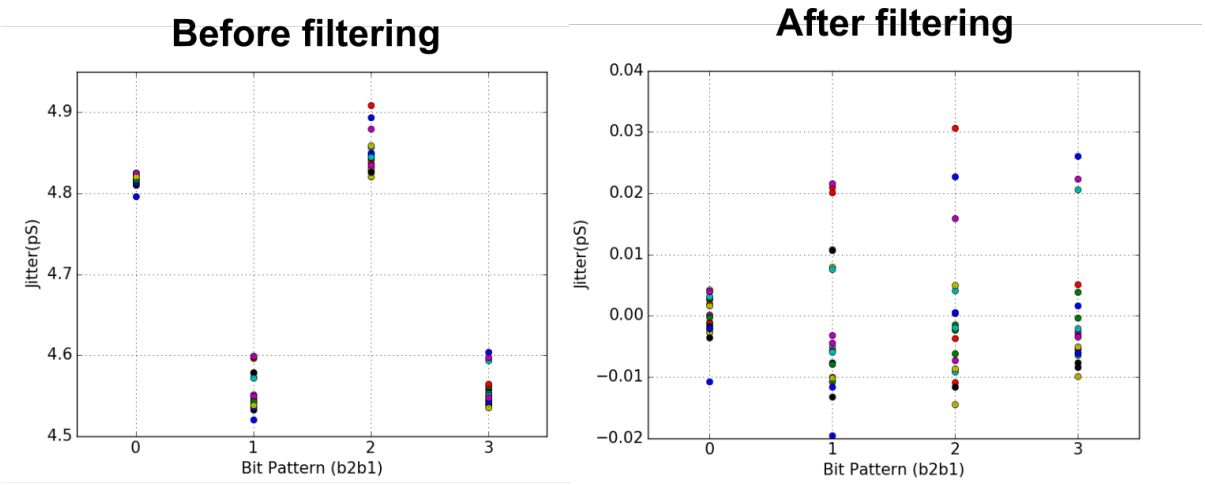


Figure 8.4: Rising edge jitter reduction after 2-tap filtering

In order to cancel this data dependent jitter, the average jitter in the case of rising and falling edges for each set of 2-bit histories was first computed. Then each rising/falling edge was mixed with a clock phase occurring $\pm 5pS$ from the nominal arrival time of these edges, and the mixing ratios were calculated such that the average jitter for each

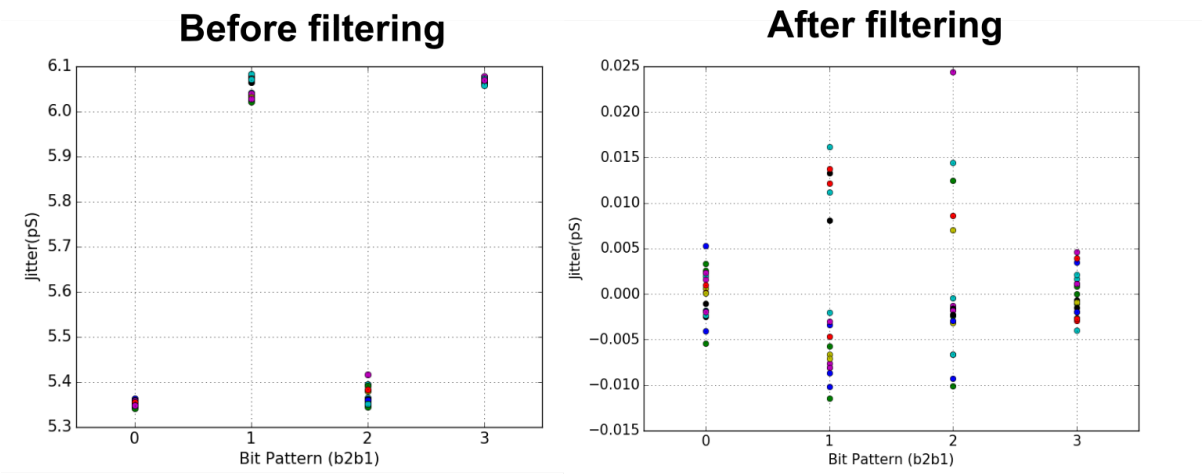


Figure 8.5: Falling edge jitter reduction after 2-tap filtering

pattern is completely cancelled. Equation 8.2 shows the math for calculating the mixing ratio α, given the average jitter j_{avg}, and the clock phase resolution, ΔT.

$$\alpha \times j_{avg} + (1-\alpha) \times \Delta T \times (-1 \times sign(j_{avg})) = 0 \implies \alpha = \frac{sign(j_{avg}) \times \Delta T}{j_{avg} + sign(j_{avg}) \times \Delta T} \quad (8.2)$$

As shown in the "after filtering" plots of Fig. 8.4 and Fig. 8.5, this two tap filter reduces the channel jitter by $40 \times -100\times$. For a channel that shows greater dispersion than the example presented here, the same technique can be applied to implement a filter with more taps. Fig. 8.6 shows the implementation of the 2-tap filter described above using pulse gates. The latches on the top implement a 2-bit shift register, triggered on the arrival of every new transition and the pulse-OR gates implement history dependent mixing.

Compared to conventional multi-tap Decision Feedback Equalizer circuits of the form shown in Fig. 8.7, the time-domain filter presented here is not only cheaper in terms of area and power consumption, but eliminates the need for an adder, which often is the component that limits the performance of a multi-tap DFE. Instead, the presented

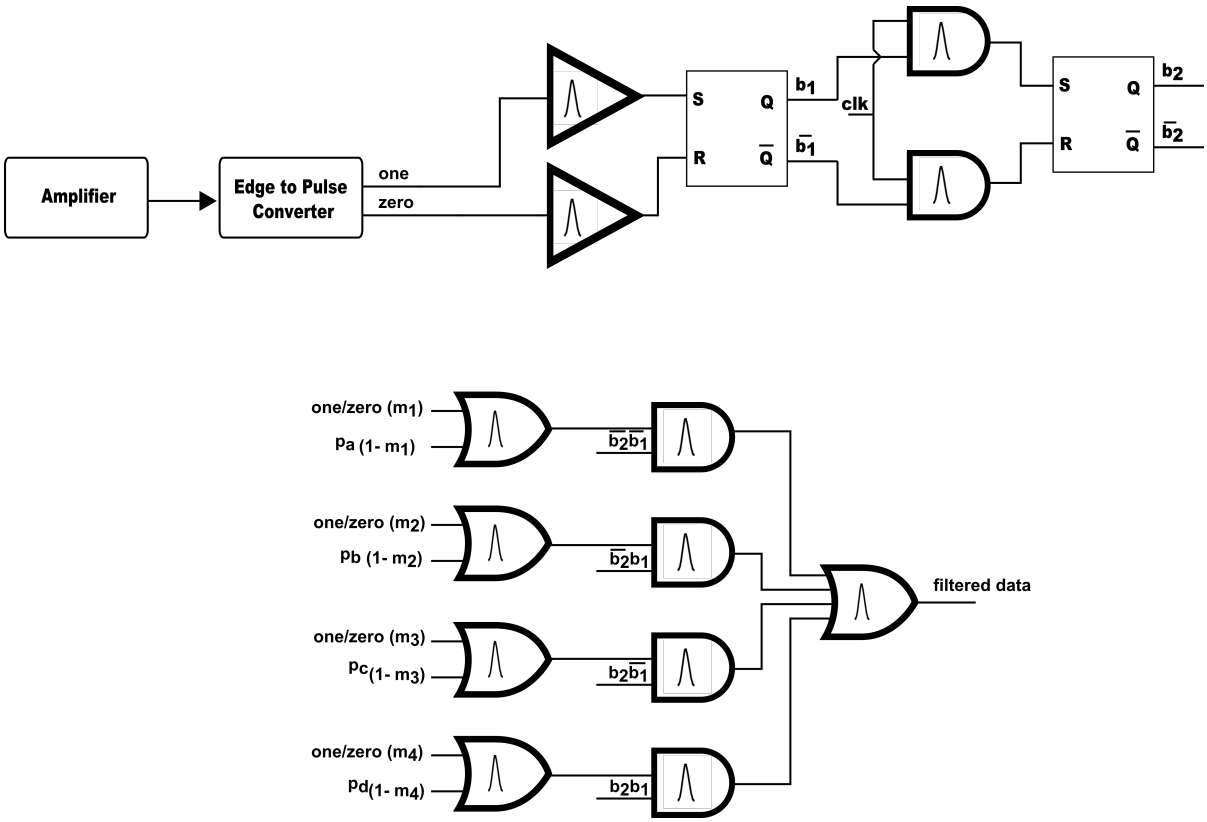


Figure 8.6: 2-tap filtering circuit

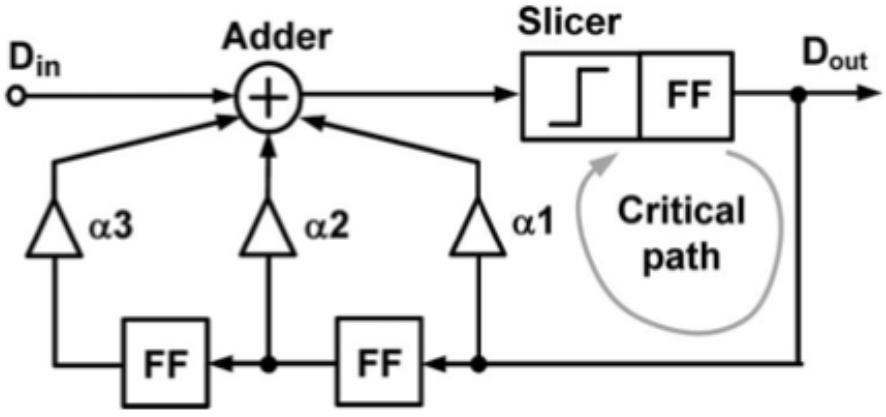


Figure 8.7: Conventional multi-tap Decision Feedback Equalizer

time-domain equalizer uses non-linear history-based switching to cancel data dependent jitter. Apart from the benefit of precise high-resolution clock phases supplied by CPOs, utilizing pulse logic for the filtering circuit makes its operation event driven and the circuit can achieve a higher performance due to a shorter critical path. This technique can be applied for crosstalk cancellation, jitter reduction for self-timed links and a host of other problems where the noise source is deterministic and where it is possible to draw a correlation between data and jitter. The work presented in this chapter is merely an introduction to the design of time-domain linear filters, and the actual realization of this technique and analysis of the impact of circuit non-idealities on the efficacy of this technique will be a part of our future work.

Chapter 9

Conclusions and Future Work

This thesis covered the analysis and design of a variety of timing circuits that have applications in problems requiring precision timing sources, low-jitter low-power timing distribution, energy efficient data movement for current chip-scale integration strategies, and time-domain filtering. All the solutions presented in this work utilize pulse logic where data signals and their timing are conveyed by atomic pulses, enabling the design of both high-performance event-driven feed-forward circuits and closed-loop circuits with negative timing feedback that exhibit precise timing. The negative timing feedback arises from partial retention of charge from a past state in pulse gates, leading to temporal noise correction effects which have been utilized for the design of Collective Pulse Oscillators (CPOs).

The behavior of CPOs is distinct from other liming-amplifier loop or resonator based oscillators in that CPOs exhibit temporal degradation of any injected noise, and thus were analyzed using a new time-domain analytical noise model proposed in this work. The behavior of CPOs is similar to a network of coupled oscillators, and to the best of our knowledge this is the first work that presents detailed time-domain analysis of a network coupled using repulsive dynamics. Based on this analytical model, a behavioral tool was

developed to enable rapid analysis of noise properties of larger and more complex coupled structures. Several CPO designs were also fabricated where measurement results closely matched our proposed theory and showed figures of merit that were close to the current state-of-the-art.

While the noise correction properties of CPOs lead to higher timing stability at longer time-scales, the high-frequency stability shows a slight degradation due to the finite settling-time of the feedback loop. Methods for improving the high-frequency stability of CPOs were also presented in this work and the applicability of inter-connected CPO topologies to serve as timing distribution networks with feedback was evaluated. A transmission-line stabilized pulsed rotary traveling wave oscillator (RTWO) was presented that exhibits lower over-all phase noise and provides multiple full-swing, low duty-cycle, physically distributed clock phases. Further, the direction of the propagating wave in a pulse RTWO is unambiguous and methods for reliable start-up have been presented, with measurement results for a 5.5 GHz design fabricated in 130 nm.

The second part of this thesis studies the problem of on-chip/on-interposer data communication and proposes a signaling strategy that enables high-bandwidth data movement on lossy and band-limited wires with PLL/DLL free data recovery. The signaling technique is termed Multi-Wire Phase Encoding (MWPE), where data is encoded in the time-correlated switching of multiple signaling wires as opposed to signal levels. Theoretical bandwidth limits for MWPE are derived and shown to exceed the bandwidths of other known transition signaling techniques, and in certain cases MWPE offers bandwidths even higher than equivalent parallel NRZ encoded data links. We evaluate the performance of MWPE on practical on-chip channels and provide performance and power estimates of MWPE links implemented in 22 nm FDX. These links are shown to be extremely energy efficient ($0.5-0.8pJ/bit$ at bandwidths $> 100Gbps$) and additionally allow for instantaneous data capture, a property useful for communication that is burst-mode

or involves switching between multiple sources. Finally, the last part of this thesis introduces the design of time-domain linear filters to cancel timing noise that is deterministic in nature. These filters utilize the precise high-resolution clock phases provided by CPOs to implement low-cost adder-free realizations.

There are several opportunities for future research projects in this area, some being direct follow-ons to the work presented in this thesis and others being new applications of pulse-mode signaling:

- Chapter 6 presented interconnected CPO topologies that improve the high-frequency noise properties of loop connected CPOs and easily scale with the growth of the CPO dimension. This makes their use viable as timing distribution networks with feedback that acts to improve the ensemble timing stability of the entire structure. Circuit simulation of such a network would help with knowing the scale over which this technique can be applied, and its performance in terms of accuracy of the delivered timing vs. power in comparison to completely feed-forward timing distribution strategies.

- The performance of CPOs when used as VCOs for PLLs has not been evaluated in this work. The use of CPOs within multiplying DLL or injection locked PLL configurations is particularly interesting as multi-pulse CPOs provide multiple injection points which can potentially be utilized for achieving faster locks. Further, such techniques using injection for faster phase-error cancellation pay the cost of lower high-frequency accuracy. The inherent self-correction and engineered filtering effects in CPO gates can also be used to spread the effect of large phase corrections over longer timescales and effectively smoothen out the high-frequency noise.

- The precise high-resolution phases provided by CPOs can be utilized for direct and very accurate timing measurements instead of techniques that use free-running

Vernier clocks to create small time differentials.

- For the Multi-Wire Phase Encoding (MWPE) protocol presented in chapter 7, there exists a space for exploring mappings of binary data to MWPE that lead to efficient circuit realizations capable of doing the conversion real-time. Further, systematic identification of codes leading to worst-case channel behavior and eliminating their usage will help increase the transmission distance and wire density for this technique.

- Chapter 8 only provides an introduction to the concept of time-domain filtering using pulse clocks and logic. The detailed analysis and realization of these filters provides exciting future research opportunities.

- While this thesis has presented direct timing strategies that utilize the benefits of pulsed signalling in different ways, these circuits also offer a solid backbone for implementing neuromorphic designs, especially Spiking Neural Networks. Pulse logic can also serve as an effective CMOS interface to superconducting logic.

Bibliography

[1] M. Horowitz, *1.1 Computing's energy problem (and what we can do about it)*, in *Digest of Technical Papers - IEEE International Solid-State Circuits Conference*, 2014.

[2] *http://developer.axis.com/old/products/mcm/index.html*.

[3] *https://wpo-altertechnology.com/system-in-package/*.

[4] *http://vrworld.com/2015/09/07/amd-r9-fury-x-potential-supercomputing-monster/*.

[5] *https://www.xilinx.com/products/silicon-devices/3dic.html*.

[6] A. Kolodny, *Networks on chips: Keeping up with rent's rule and Moore's law*, in *International Workshop on System Level Interconnect Prediction, SLIP*, 2007.

[7] E. G. Friedman, *Clock distribution networks in synchronous digital integrated circuits*, *Proceedings of the IEEE* (2001).

[8] P. J. Restle, T. G. McNamara, D. A. Webber, P. J. Camporese, K. F. Eng, K. A. Jenkins, D. H. Allen, M. J. Rohn, M. P. Quaranta, D. W. Boerstler, C. J. Alpert, C. A. Carter, R. N. Bailey, J. G. Petrovick, B. L. Krauter, and B. D. McCredie, *A clock distribution network for microprocessors*, *IEEE Journal of Solid-State Circuits* (2001).

[9] N. A. Kurd, J. S. Barkatullah, R. O. Dizon, T. D. Fletcher, and P. D. Madland, *A multigigahertz clocking scheme for the Pentium® 4 microprocessor*, *IEEE Journal of Solid-State Circuits* (2001).

[10] S. Lee, A. P. Chandrakasan, and H. S. Lee, *A 1 GS/s 10b 18.9 mW time-interleaved SAR ADC with background timing skew calibration*, *IEEE Journal of Solid-State Circuits* (2014).

[11] L. Angrisani and M. D'Arco, *Modeling timing jitter effects in digital-to-analog converters*, *IEEE Transactions on Instrumentation and Measurement* (2009).

[12] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, *Millimeter wave mobile communications for 5G cellular: It will work!*, *IEEE Access* (2013).

[13] J. F. Buckwalter, X. Zheng, G. Li, K. Raj, and A. V. Krishnamoorthy, *A monolithic 25-Gb/s transceiver with photonic ring modulators and ge detectors in a 130-nm CMOS SOI process*, *IEEE Journal of Solid-State Circuits* (2012).

[14] J. Poulton, R. Palmer, A. M. Fuller, T. Greer, J. Eyles, W. J. Dally, and M. Horowitz, *A 14-mW 6.25-Gb/s transceiver in 90-nm CMOS*, in *IEEE Journal of Solid-State Circuits*, 2007.

[15] F. Villa, R. Lussana, D. Bronzi, S. Tisa, A. Tosi, F. Zappa, A. Dalla Mora, D. Contini, D. Durini, S. Weyers, and W. Brockherde, *CMOS imager with 1024 SPADs and TDCS for single-photon timing and 3-D time-of-flight*, *IEEE Journal of Selected Topics in Quantum Electronics* (2014).

[16] J. P. Jansson, V. Koskinen, A. Mantyniemi, and J. Kostamovaara, *A multichannel high-precision CMOS time-to-digital converter for laser-scanner-based perception systems*, *IEEE Transactions on Instrumentation and Measurement* (2012).

[17] E. Painkras, L. A. Plana, J. Garside, S. Temple, F. Galluppi, C. Patterson, D. R. Lester, A. D. Brown, and S. B. Furber, *SpiNNaker: A 1-W 18-core system-on-chip for massively-parallel neural network simulation*, *IEEE Journal of Solid-State Circuits* (2013).

[18] F. Akopyan, J. Sawada, A. Cassidy, R. Alvarez-Icaza, J. Arthur, P. Merolla, N. Imam, Y. Nakamura, P. Datta, G. J. Nam, B. Taba, M. Beakes, B. Brezzo, J. B. Kuang, R. Manohar, W. P. Risk, B. Jackson, and D. S. Modha, *TrueNorth: Design and Tool Flow of a 65 mW 1 Million Neuron Programmable Neurosynaptic Chip*, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2015).

[19] R. H. Dennard, J. Cai, and A. Kumar, *A perspective on today's scaling challenges and possible future directions*, *Solid-State Electronics* (2007).

[20] K. C. Saraswat and F. Mohammadi, *Effect of Scaling of Interconnections on the Time Delay of VLSI Circuits*, *IEEE Transactions on Electron Devices* (1982).

[21] P. Vivet, C. Bernard, F. Clermidy, D. Dutoit, E. Guthmuller, I. M. Panadès, G. Pillonnet, Y. Thonnart, A. Garnier, D. Lattard, A. Jouve, F. Bana, T. Mourier, and S. Chéramy, *3D advanced integration technology for heterogeneous systems*, in *2015 International 3D Systems Integration Conference, 3DIC 2015*, 2015.

[22] D. Stow, I. Akgun, and Y. Xie, *Investigation of cost-optimal network-on-chip for passive and active interposer systems*, in *2019 ACM/IEEE International Workshop on System Level Interconnect Prediction, SLIP 2019*, 2019.

[23] W. J. Dally and B. Towles, *Route packets, not wires: On-chip interconnection networks*, in *Proceedings - Design Automation Conference*, 2001.

[24] D. Wingard, *MicroNetwork-based integration for SOCs*, in *Proceedings - Design Automation Conference*, 2001.

[25] L. Benini and G. De Micheli, *Networks on chip: A new paradigm for systems on chip design*, in *Proceedings -Design, Automation and Test in Europe, DATE*, 2002.

[26] A. J. Martin and M. Nyström, *Asynchronous techniques for system-on-chip design*, *Proceedings of the IEEE* (2006).

[27] M. N. Horak, S. M. Nowick, M. Carlberg, and U. Vishkin, *A low-overhead asynchronous interconnection network for GALS chip multiprocessors*, in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2011.

[28] M. Krstić, E. Grass, F. K. Gürkaynal, and P. Vivet, *Globally asynchronous, locally synchronous circuits: Overview and outlook*, 2007.

[29] J. Muttersbach, T. Villiger, and W. Fichtner, *Practical design of globally-asynchronous locally-synchronous systems*, *Proceedings - International Symposium on Asynchronous Circuits and Systems* (2000).

[30] M. Miller, *Realization and Formal Analysis of Asynchronous Pulse Communication Circuits*. PhD thesis, University of California Santa Barbara, 2015.

[31] B. D. Winters and M. R. Greenstreet, *A negative-overhead, self-timed pipeline*, in *Proceedings - International Symposium on Asynchronous Circuits and Systems*, 2002.

[32] G. Hinton, M. Upton, D. J. Sager, D. Boggs, D. M. Carmean, P. Roussel, T. I. Chappell, T. D. Fletcher, M. S. Milshtein, M. Sprague, S. Samaan, and R. Murray, *A 0.18-μm CMOS IA-32 processor with a 4-GHz integer execution unit*, *IEEE Journal of Solid-State Circuits* (2001).

[33] M. Miller, G. Hoover, and F. Brewer, *Pulse-mode link for robust, high speed communications*, *Proceedings - IEEE International Symposium on Circuits and Systems* **5** (2008) 3073–3077.

[34] M. P. Miller, G. Magazzu, and F. D. Brewer, *5 Gbps Radiation-Hardened Low-Power Pulse Serial Link*, *IEEE Transactions on Nuclear Science* **63** (2016), no. 1 203–212.

[35] A. Dalakoti, *Collective Pulse Dynamics: A New Timing Circuit Strategy.* PhD thesis, University of California Santa Barbara, 2018.

[36] D. Allan and J. Barnes, *A modified Allan variance with increased oscillator characterization*, in *35th Annual Frequency Control Symposium*, pp. 470–475, 1981.

[37] S. Fairbanks and S. Moore, *Analog micropipelines for high precision timing*, in *10th International Symposium on Asynchronous Circuits and Systems*, pp. 41–50, 2004.

[38] O. Elissati, A. Cherkaoui, A. El-Hadbi, S. Rieubon, and L. Fesquest, *Multi-phase low-noise digital ring oscillators with sub-gate-delay resolution*, *AEU - International Journal of Electronics and Communications* **84** (2018) 74–83.

[39] T. Y. Wuu and S. B. Vrudhula, *A Design of a Fast and Area Efficient Multi-Input Muller C-element*, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (1993).

[40] J. Wood, T. C. Edwards, and S. Lipa, *Rotary traveling-wave oscillator arrays: A new clock technology*, *IEEE Journal of Solid-State Circuits* (2001).

[41] A. Boroumand, S. Ghose, Y. Kim, R. Ausavarungnirun, E. Shiu, R. Thakur, D. Kim, A. Kuusela, A. Knies, P. Ranganathan, and O. Mutlu, *Google workloads for consumer devices: Mitigating data movement bottlenecks*, in *ACM SIGPLAN Notices*, 2018.

[42] S. Borkar and A. A. Chien, *The future of microprocessors*, *Communications of the ACM* (2011).

[43] B. Casper, J. Jaussi, F. O'Mahony, M. Mansuri, K. Canagasaby, J. Kennedy, E. Yeung, and R. Mooney, *A 20Gb/s forwarded clock transceiver in 90nm CMOS*, in *Digest of Technical Papers - IEEE International Solid-State Circuits Conference*, 2006.

[44] J. Lee and B. Razavi, *A 40-Gb/s Clock and Data Recovery Circuit in 0.18-μm CMOS Technology*, in *IEEE Journal of Solid-State Circuits*, 2003.

[45] D. McCarthy, M. Miller, and F. Brewer, *Automated Timing Constraint Generation for Pulse Gate Circuits*, *IWLS Workshop, Lausanne, Switzerland* (2019).

[46] J. Yu, F. F. Dai, and R. C. Jaeger, *A 12-Bit vernier ring time-to-digital converter in 0.13 μ CMOS technology*, in *IEEE Journal of Solid-State Circuits*, 2010.

[47] A. Liscidini, L. Vercesi, and R. Castello, *Time to digital converter based on a 2-dimensions Vernier architecture*, in *Proceedings of the Custom Integrated Circuits Conference*, 2009.

[48] M. Park and M. H. Perrott, *A VCO-based analog-to-digital converter with second-order sigma-delta noise shaping*, in *Proceedings - IEEE International Symposium on Circuits and Systems*, 2009.

[49] W. Yu, J. Kim, K. Kim, and S. Cho, *A time-domain high-order MASH $\Delta\Sigma$ ADC using voltage-controlled gated-ring oscillator*, *IEEE Transactions on Circuits and Systems I: Regular Papers* (2013).

[50] S. H. Lee, M. S. Hwang, Y. Choi, S. Kim, Y. Moon, B. J. Lee, D. K. Jeong, W. Kim, Y. J. Park, and G. Ahn, *A 5-Gb/s 0.25-μm CMOS jitter-tolerant variable-interval oversampling clock/data recovery circuit*, in *IEEE Journal of Solid-State Circuits*, 2002.

[51] B. Razavi, *A study of phase noise in CMOS oscillators*, *IEEE Journal of Solid-State Circuits* **31, no. 3,** (1996), no. 3 331–343.

[52] A. A. Abidi, *Phase noise and jitter in CMOS ring oscillators*, in *IEEE Journal of Solid-State Circuits*, 2006.

[53] T. Weigandt, B. Kim, and P. Gray, *Analysis of timing jitter in CMOS ring oscillators*, in *IEEE International Symposium on Circuits and Systsems - ISCAS*, pp. 27–30, 1994.

[54] K. Takinami, R. Walsworth, S. Osman, and S. Beccue, *Phase-noise analysis in rotary traveling-wave oscillators using simple physical model*, *IEEE Transactions on Microwave Theory and Techniques* (2010).

[55] J. G. Maneatis and M. A. Horowitz, *Precise Delay Generation Using Coupled Oscillators*, *IEEE Journal of Solid-State Circuits* (1993).

[56] H. C. Chang, X. Cao, U. K. Mishra, and R. A. York, *Phase noise in coupled oscillators: Theory and experiment*, *IEEE Transactions on Microwave Theory and Techniques* (1997).

[57] M. M. Abdul-Latif and E. Sánchez-Sinencio, *Low phase noise wide tuning range N-push cyclic-coupled ring oscillators*, *IEEE Journal of Solid-State Circuits* (2012).

[58] L. Hall, M. Clements, W. Liu, and G. Bilbro, *Clock distribution using cooperative ring oscillators*, in *Proceedings of the Conference on Advanced Research in VLSI*, 1997.

[59] Y. W. Lin and S. S. Hsu, *A Sierpinski Space-Filling Clock Tree Using Multiply-by-3 Fractal-Coupled Ring Oscillators*, *IEEE Journal of Solid-State Circuits* (2017).

[60] G. Taylor and I. Galton, *A reconfigurable mostly-digital delta-sigma ADC with a worst-case FOM of 160 dB*, *IEEE Journal of Solid-State Circuits* (2013).

[61] S. D. Vamvakos, C. Boecker, E. Groen, A. Wang, S. Desai, S. Irwin, V. Rao, A. Bottelli, J. Chen, X. Chen, P. Choudhary, K. C. Hsieh, P. Jennings, H. Lin, D. Pechiu, C. Rao, and J. Yeung, *A 8.125-15.625 Gb/s SerDes using a sub-sampling ring-oscillator phase-locked loop*, in *Proceedings of the IEEE 2014 Custom Integrated Circuits Conference, CICC 2014*, 2014.

[62] J. Carnes, I. Vytyaz, P. K. Hanumolu, K. Mayaram, and U. K. Moon, *Design and analysis of noise tolerant ring oscillators using Maneatis delay cells*, in *Proceedings of the IEEE International Conference on Electronics, Circuits, and Systems*, 2007.

[63] A. Hajimiri, S. Limotyrakis, and T. H. Lee, *Jitter and phase noise in ring oscillators*, *IEEE Journal of Solid-State Circuits* **34** (1999), no. 6 790–804.

[64] A. Dalakoti, M. Miller, and F. Brewer, *Pulse Ring Oscillator Tuning via Pulse Dynamics*, *35th IEEE International Conference on Computer Design* (2017).

[65] A. Demir, A. Mehrotra, and J. Roychowdhury, *-Phase noise in oscillators: a unifying theory and numerical methods for characterization*, *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* (2000).

[66] W. Riley and D. A. Howe, *Handbook of Frequency Stability Analysis*. 2008.

[67] N. Kasdin and T. Walter, *Discrete simulation of power law noise (for oscillator stability evaluation)*, in *Proceedings of the 1992 IEEE Frequency Control Symposium*, 1992.

[68] W. J. Riley, *IEEE Stable 32 Software for Frequency Stability Analysis*.

[69] *Virtuoso Spectre Circuit Simulator and Accelerated Parallel Simulator RF Analysis User Guide Version 12.1.1*. May, 2013.

[70] C. Azeredo-Leme, *Clock Jitter Effects on sampling: A tutorial*, *IEEE Circuits and Systems Magazine* (2011).

[71] F. Herzel and B. Razavi, *A study of oscillator jitter due to supply and substrate noise*, *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* **46** (1999), no. 1 56–62.

[72] S. Shieh Ali Saleh and N. Masoumi, *Wide-tuning-range, low-phase-noise quadrature ring oscillator exploiting a novel noise canceling technique*, *AEU - International Journal of Electronics and Communications* (2012).

[73] Z. Q. Lu, J. G. Ma, and F. C. Lai, *A low-phase-noise 900-MHz CMOS ring oscillator with quadrature output*, *Analog Integrated Circuits and Signal Processing* (2006).

[74] C. K. Luo and J. F. Buckwalter, *A 0.25-to-2.25 GHz, 27 dBm IIP3, 16-path tunable bandpass filter*, *IEEE Microwave and Wireless Components Letters* (2014).

[75] E. A. Klumperink, H. J. Westerveld, and B. Nauta, *N-path filters and mixer-first receivers: A review*, in *Proceedings of the Custom Integrated Circuits Conference*, 2017.

[76] V. L. Chi, *Salphasic Distribution of Clock Signals for Synchronous Systems*, *IEEE Transactions on Computers* (1994).

[77] B. Kleveland, C. H. Diaz, D. Vock, L. Madden, T. H. Lee, and S. S. Wong, *Monolithic cmos distributed amplifier and oscillator*, in *1999 IEEE International Solid-State Circuits Conference. Digest of Technical Papers. ISSCC. First Edition (Cat. No.99CH36278)*, pp. 70–71, 1999.

[78] Z. Bai, X. Zhou, R. D. Mason, and G. Allan, *Low-phase noise clock distribution network using rotary traveling-wave oscillators and built-in self-test phase tuning technique*, *IEEE Transactions on Circuits and Systems II: Express Briefs* (2015).

[79] X. Li, O. O. Yildirim, W. Zhu, and D. Ham, *Phase noise of distributed oscillators*, *IEEE Transactions on Microwave Theory and Techniques* (2010).

[80] M. Miller, A. Dalakoti, P. Mukim, and F. Brewer, *Low Phase Noise Pulse Rotary Wave Voltage Controlled Oscillator*, *14th IEEE International SoC Design Conference* (2017).

[81] P. Mukim, A. Dalakoti, D. Mccarthy, C. Segal, M. Miller, J. F. Buckwalter, and F. Brewer, *Design and Analysis of Collective Pulse Oscillators*, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (2020).

[82] K. Takinami and R. Walsworth, *Phase error calibration technique for rotary traveling wave oscillators*, in *IEEE Journal of Solid-State Circuits*, 2010.

[83] P. B. McGee, M. Y. Agyekum, M. A. Mohamed, and S. M. Nowick, *A level-encoded transition signaling protocol for high-throughput asynchronous global communication*, in *Proceedings - International Symposium on Asynchronous Circuits and Systems*, pp. 116–127, 2008.

[84] W. J. Bainbridge, W. B. Toms, D. A. Edwards, and S. B. Furber, *Delay-insensitive, point-to-point interconnect using M-of-N codes*, in *Proceedings - International Symposium on Asynchronous Circuits and Systems*, pp. 132–140, 2003.

[85] C. D'Alessandro, D. Shang, A. Bystrov, A. Yakovlev, and O. Maevsky, *Multiple-rail phase-encoding for NoC*, in *Proceedings - International Symposium on Asynchronous Circuits and Systems*, vol. 2006, pp. 107–116, 2006.

[86] C. D'Alessandro, N. Minas, K. Heron, D. Kinniment, and A. Yakovlev, *NoC communication strategies using time-to-digital conversion*, in *Proceedings - NOCS 2007: First International Symposium on Networks-on-Chip*, pp. 65–74, 2007.

[87] R. Dobkin, R. Ginosar, and A. Kolodny, *Fast asynchronous shift register for bit-serial communication*, in *Proceedings - International Symposium on Asynchronous Circuits and Systems*, 2006.

[88] W. Wang and J. F. Buckwalter, *A 10-Gb/s, 107-mW double-edge pulsewidth modulation transceiver*, *IEEE Transactions on Circuits and Systems I: Regular Papers* **61** (2014), no. 4 1068–1080.

[89] N. E. Jerger, A. Kannan, Z. Li, and G. H. Loh, *NoC Architectures for Silicon Interposer Systems*, in *Microarchitecture (MICRO), 2014 47th Annual IEEE/ACM International Symposium on*, 2014.

[90] *HSPICE User Guide: Signal Integrity Modeling and Analysis, Version Q-2020.03.* May, 2013.