DESIGN OF CMOS ADAPTIVE-SUPPLY SERIAL LINKS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Jaeha Kim
December 2002

© Copyright by Jaeha Kim 2003
All Rights Reserved

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

Mark A. Horowitz
(Principal Advisor)

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

William J. Dally

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

John G. Maneatis

Approved for the University Committee on Graduate Studies:

Abstract

There is a need for IC chips that can support very high input/output (I/O) bandwidths. The key to high bandwidth is high per-pin I/O data rate and low power operation to enable a large number of pins to be integrated. This dissertation explores how adaptive power-supply regulation and parallelism can help minimize the link power dissipation while achieving high performance.

To maximize the energy-efficiency, the supply voltage is adaptively regulated to the minimum required for the desired frequency. The adaptive supply uses a buck regulator for efficient voltage step-down, and this regulator uses a novel digital sliding controller that monitors the link performance and adapts the voltage to process and temperature variations. Since the dynamics of the sliding controller do not depend on its operating frequency, the controller can be operated off of the adaptive supply, achieving the overall efficiency of 89-95% over the entire operating range (over $40\times$ change in power).

The analog sections of the I/O circuits are modified to extend their operation to very low voltages. The input signals to the transmitter output stage are level-shifted to make the effective threshold voltage of the output devices zero and to mitigate the output current vanishing as the supply voltage approaches Vth. The receiver stage uses an integrating stage with no sampling switches and a charge-injection-based comparator that can operate at very low supply. Overall, the link is operational down to 0.9V with Vth of 0.55V.

The timing for the links is controlled by either PLL or DLL circuitry that locally generates the needed multiphase clocks for the parallelized transceiver architecture. The area of these circuits are reduced by using the adaptive supply as the global loop to coarse-tune the frequency and using the local loops to fine-tune over a narrow range. In this architecture, the PLL design requires 52% less power and 41% less area than the DLL design with about

the same jitter. The clock recovery PLLs use bangbang control and its nonlinear effects are carefully analyzed.

Prototype chips were fabricated in a 0.25μm CMOS technology. The adaptive-supply link operates from 0.65 to 5.0Gb/s. At 3.1Gb/s, the complete link dissipates only 113mW.

Acknowledgments

Looking back on my days at Stanford, I realize that I am tremendously blessed by God to have had teachers, friends, colleagues, and family members close by to provide me with guidance and support.

First of all, I would like to thank my advisor, Prof. Mark Horowitz, who has been my great mentor and source of knowledge and wisdom. Mark has the Midas touch that turns any idea into an even better one, and I enjoyed and benefited from discussing new ideas with him as he respects and listens to even the hopeless ones. Mark has also been my best English teacher, always knowing better words to express my ideas. I am grateful for his guidance and am privileged to have worked with him.

I would also like to thank my associate advisor, Prof. Bill Dally, for his keen insights that have inspired me in many ways, for serving on my orals committee, and for reading this thesis. I am grateful to Dr. John Maneatis for his thorough proofreading and his great expertise in PLL design. I am also grateful to Prof. John Gill who served on my orals committee, and to my undergraduate advisor, Prof. Wonchan Kim, who gave me continual support and encouragement during my course of study.

This thesis builds upon the work done by many former members of the Horowitz group. I am grateful for their guidance, support, and criticism as I carried out my research. I thank Gu-Yeon Wei who provided the direct foundation of this work, and Stefanos Sidiropoulos, Ken Yang, John Maneatis, Ramin Farjad-Rad, David Harris, and Ricardo Gonzalez who paved the early work that guided me in this direction. Thanks also go to other senior members in the group: Evelina Yeung, Bennett Wilburn, Birdy Amrutur, Dan Weinlader, Ken Mai, Ron Ho, Ken Chang, Bill Ellersick, and Hema Kapadia, for their friendship and hearty welcome when I first joined the group.

I cherish the fond memories with my colleagues at Stanford. I thank Dean Liu for his friendship and selfless support, and Haechang Lee and Elad Alon for helping on my chips and correcting my countless English mistakes in this thesis. I am fortunate to be surrounded by caring people such as Azita Emami-Neyestanak, Vladimir Stojanovic, David Lie, Ed Lee, Samuel Palermo, Eric Kim, and countless others in CIS and CSL. Especially, I appreciate the collaboration and dedication of those who managed the National liaison program, including Greg Gordon, Ali Tabatabaei, Hirad Samavati, Joel Dawson, Shwe Verma, and Lalit Nathawad. Last but not least, I cherish the good memories I have with my Korean friends, Sangeun Han, Wonjoon Choi, Jinyung Namgoong, Donghyun Kim, Youngjune Gwon, and Wonjong Rhee, to name just a few.

This research would not have been possible without the generous support of the Department of Energy (ASCI), National Semiconductor, Vitesse Semiconductor, Nassda Corporation, and Synopsys Inc. I also thank the staff members of CIS and CSL: especially Charlie Orgish, Joe Little, and Jason Conroy for their technical support, and Penny Chumley, Terry West, Taru Fisher, Deborah Harper, Pamela Elliot, and Ann Guerra for their administrative support.

I give my heartfelt thanks to my parents, who have always been proud of me becoming an engineer, for their endless love and prayer. I also thank my parents-in-law who have not spared anything to support my research and well-being. I cannot find the best words to thank my lovely wife, Sookyung Kim, who has given me continual love and encouragement throughout my years at Stanford, either distant or nearby. I dedicate this thesis to her as I would not have completed it without her.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The success of modern integrated circuits (ICs) is in large part due to the low-cost realization of a large-scale electronic system on a tiny semiconductor chip. Among several IC technologies, complementary metal-oxide-semiconductor (CMOS) technology has been the main driver of the exponential growth in IC's computing performance [4]. The speed of CMOS logic gates has improved by 13% per year and the number of integrated transistors has increased by 50% per year. As a result, the chip's computing capability has grown by 70% per year [5].

As the computing capability of an IC chip rises, the demand for communication bandwidth between chips also grows [6]. However, the signalling bandwidth between chips was not keeping up with the processor performance, as shown in Figure 1.1. This resulted in a widening gap between on-chip computation speed and off-chip communication bandwidth, which compromises the overall system performance.

In order to bridge this gap, it is necessary to both increase the signalling rate of each pin and to increase the number of signal pins on the chip. However, circuits that operate at high speed generally have high power consumption, prohibiting a large number of links from being integrated on a single chip. To meet both the power constraint imposed by the chip package and the input/output (I/O) bandwidth requirement of the system, it is therefore critical to reduce the power dissipation of high-speed links.

This dissertation focuses on improving the power-efficiency of the links while maintaining their high performance. The two main approaches taken are the adaptive regulation

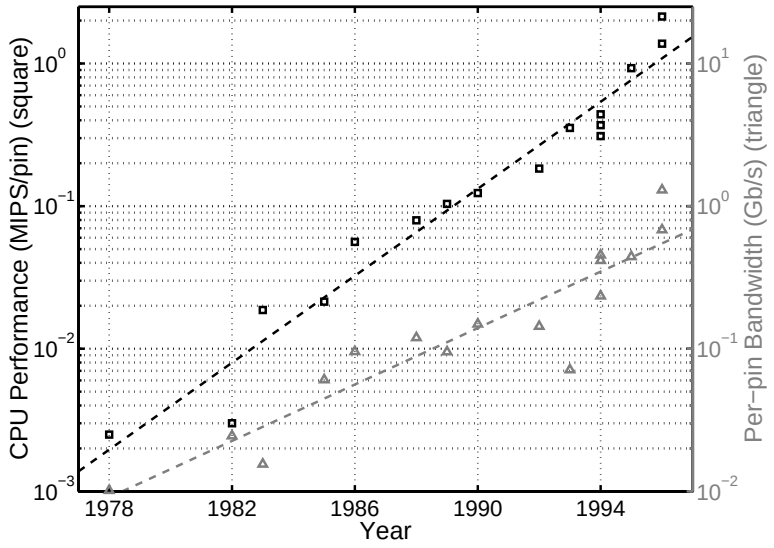


Figure 1.1: The widening gap between the processor performance and off-chip pin bandwidth, from D. Burger et al. [1]

of supply voltage and the use of parallelism. To apply these approaches, supporting circuits such as an efficient adaptive power-supply regulator, a low-voltage parallelized transceiver, and low-cost per-pin multiphase clock generation and recovery circuits are designed and presented.

This chapter begins by reviewing the basic principles of high-speed links. Then it discusses the details of adaptive power-supply regulation and parallelized architecture as means of achieving low power dissipation. The chapter ends with an overview of an adaptive-supply serial link, which provides an outline of the rest of this dissertation.

1.1 High-Speed Links

High-speed links achieve high data rates by using: 1) careful termination on the channel, 2) synchronous transmission, and 3) high-bandwidth transmitters and receivers.

First, terminating the channel to a matched impedance suppresses reflection which can cause interference and limit the signalling rate. Figure 1.2 illustrates the cases when the channel is properly terminated and when it is not. When the channel is not properly terminated, the signal that arrives at the receiving end can bounce back to the opposite direction.

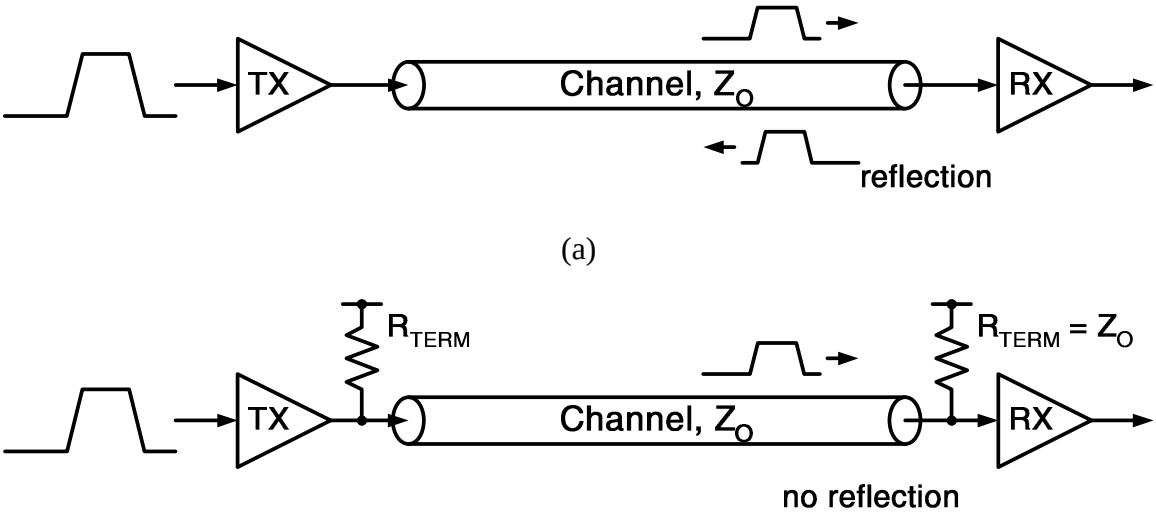


Figure 1.2: Channels for high-speed links: (a) without termination, (b) with termination

This signal will remain on the line until it is attenuated by the channel loss and thus will interfere with subsequent signals launched onto the channel. For reliable operation, one solution is to send the next signal only after the current signal has disappeared from the channel. However, the signalling rate is then severely limited by the channel characteristics– it may take several round trips for the signal to be fully attenuated.

When the channel is properly terminated with the characteristic impedance, as in Figure 1.2(b), the transmitted signal is fully absorbed by the termination at the receiving end. Knowing that the signal will not get reflected and cause interference, we can send the next signal even before the current signal reaches the receiver. In this way, more than one signal can reside on the channel simultaneously and be transported in a pipelined fashion. The signalling rate is greatly improved without being limited by the channel latency.

Second, the adoption of synchronous transmission to distinguish one signal from another also led to improvement in the signalling rate. In the past, asynchronous transmission was commonly used in chip-to-chip signalling because it allowed two chips with different speeds to communicate easily. For example, as shown in Figure 1.3(a), when a sender wishes to send data across the channel, it launches the signal on the channel and asserts a control signal, **REQ**. When this request signal propagates to the receiver, the receiver reads

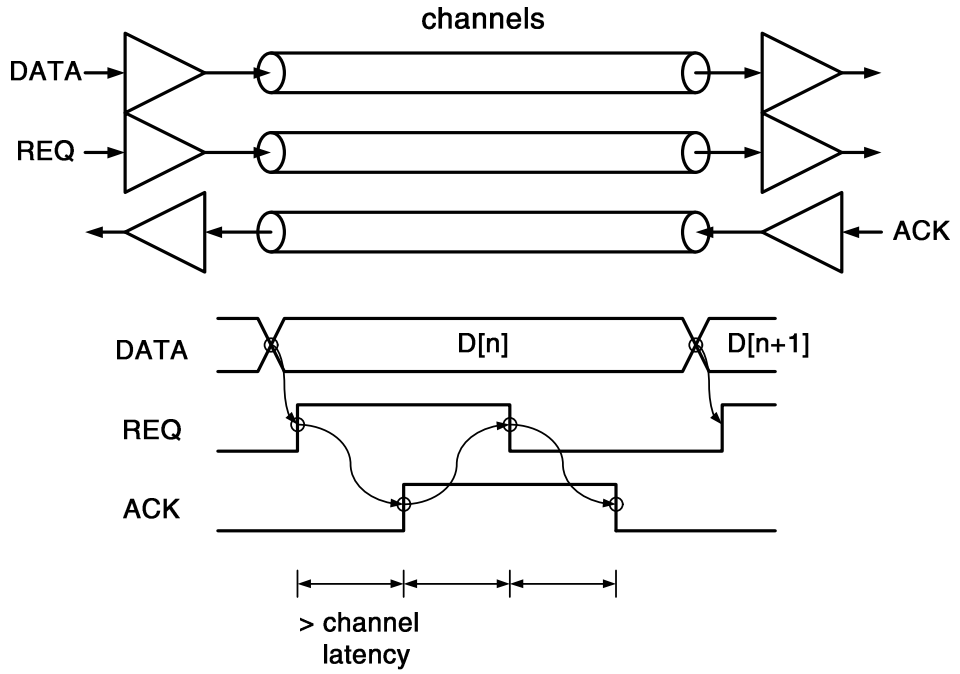
in the data on the channel and launches another control signal, **ACK**, back to the sender. Upon the arrival of this signal, the sender knows that the transmission is completed and de-asserts **REQ**. Every time a signal is transmitted, the sender and the receiver go through this whole cycle of handshaking. The throughput is therefore limited by the channel latency.

Figure 1.3(b) illustrates the link with synchronous transmission. The channel is properly terminated and signals are transmitted successively one after another. Different signals are distinguished only by their positions in time, assuming a uniform time spacing between the signals. Although the throughput is now greatly improved, synchronous transmission calls for precise control over the timing of the signals. Phase-locked loops and delay-locked loops are commonly employed to achieve this high level of timing accuracy [7],[8],[9].

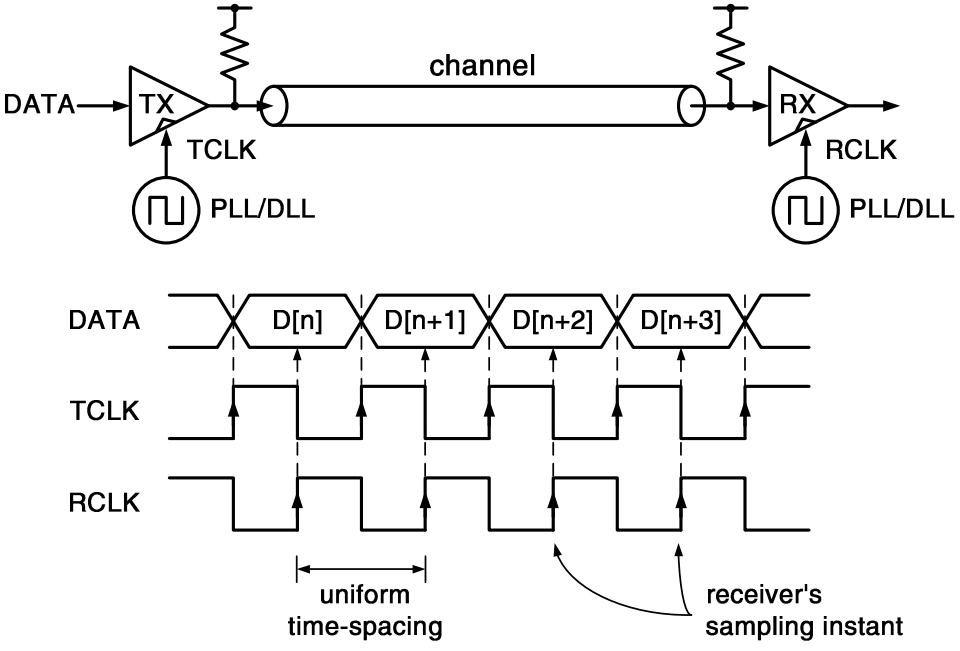
Third, high-bandwidth transmitters and receivers, along with the aforementioned schemes, can achieve high signalling rates that are no longer bounded by the channel latency. In particular, the transceiver circuits implemented in CMOS technology can benefit from the exponential growth in performance. One of the major speed limits in CMOS high-speed links is the bandwidth of the clock buffer chain, which sets the maximum clock frequency that can be propagated to the transmitter and receiver [10]. For example, with no loss in amplitude, a chain of CMOS inverters can propagate pulses that are as short as 3-4 fanout-of-4 (FO4) inverter delays,[1] and thus the minimum clock period is limited to 6-8 FO4 inverter delays. The FO4 inverter delay in picoseconds will decrease as the device feature size shrinks, as plotted in Figure 1.4, and thus the maximum clock frequency will grow higher. Since the bitrates of CMOS high-speed links are usually some multiples of the clock frequency due to various parallel techniques, the bitrates are expected to improve along with CMOS process scaling.

In summary, high-speed links can offer signalling rates that will continue to scale with on-chip transistor performance. However, the increase in IC's computing capability results not only from the transistor speed improvement, but also from the increased number of transistors on a chip, which increases by 50% per year. Therefore, to meet the off-chip bandwidth requirement, the number of high-speed links required on a chip is likely to increase as well.

[1]The delay of an inverter driving a load that is four times larger than its input load. The FO4 inverter delay is a good metric of the CMOS circuit performance scaling with process generations and can be estimated as $500ps/\mu m \cdot L_{gate}$.



(a)



(b)

Figure 1.3: Signalling conventions: (a) asynchronous transmission, (b) synchronous transmission

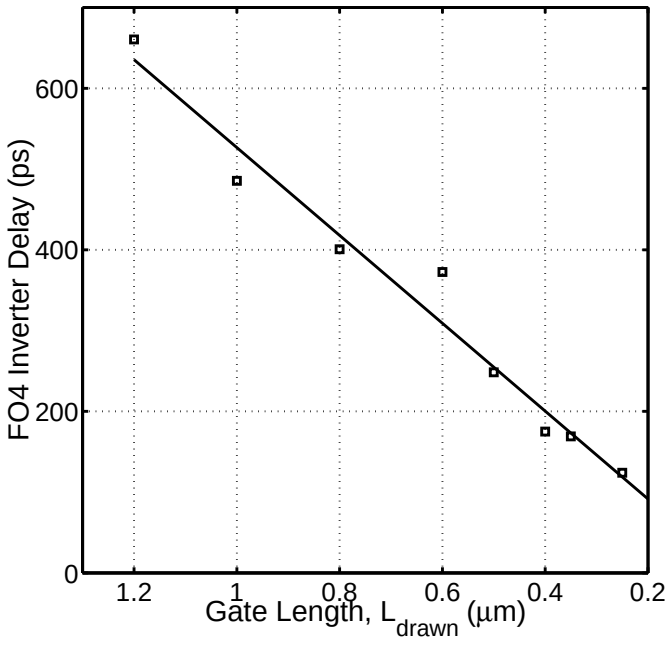


Figure 1.4: Scaling of a fanout-of-4 inverter delay over CMOS process generations

1.2 Demand for Low Power Dissipation

Emerging applications such as multiprocessor-based servers and high-end network routers demand much higher bandwidth between IC chips [11],[12],[13],[14],[15]. These applications require not only high bitrate per pin, but also a large number of links with different sources and destinations. This calls for more links to be integrated on each chip. The number is currently around a hundred, but it is likely to increase to many hundreds and even thousands in the next 10 years.

One of the limits to increasing the number of I/O pins and hence the off-chip bandwidth is the power dissipated by high-speed links. The fast signalling rate of a high-speed link comes at the expense of increased power dissipation, needed mainly to switch the internal capacitive nodes constantly at high frequencies.

The increased power dissipation will become a critical factor because of the power limits of chip packages. The power dissipation of each link will determine the maximum number of links that can be integrated onto a chip and thus the maximum off-chip bandwidth. Figure 1.5 shows the slow increase in the maximum power allowed on high-end

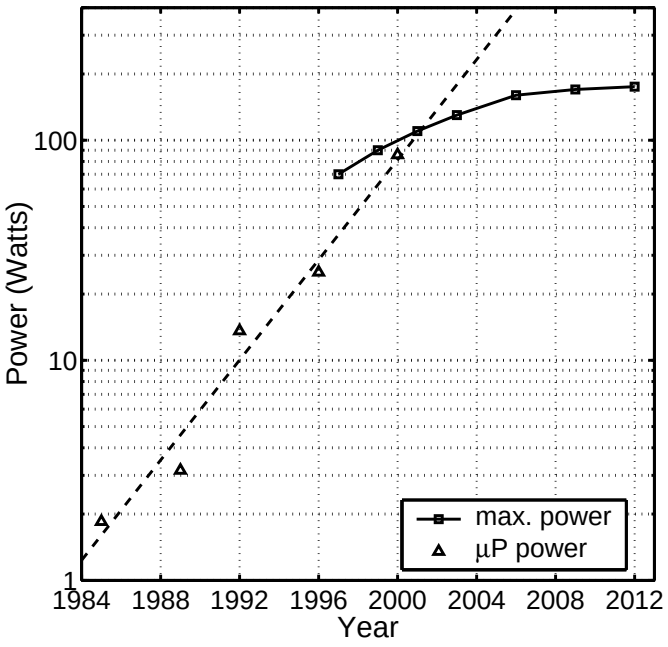


Figure 1.5: Power crisis: the maximum power allowed on high-end packages and the increasing trend of the microprocessor core power

packages [16] and the rapid increase in the microprocessor core power [17]. Simply increasing the number of high-speed links to meet the off-chip bandwidth needs will cause the I/O power to increase at a similar rate with that of the processor core and will soon reach the limits of the package. Therefore, high-speed links will have to be efficient both in speed and power to address the increasing demand for total off-chip bandwidth.

A figure-of-merit would be convenient to quantify the speed-power efficiency of a link. A metric that is often used is power-per-bitrate (W/bps), or equivalently, bitrate-per-power (bps/W), which measures the ratio between the power and bitrate of a high-speed link. This is in fact an energy metric that measures the total energy required to transfer each bit across the channel (J/bit), and like other energy metrics, tends to favor low-speed, low-power designs. However, the low-speed links motivated by the power-per-bitrate metric cannot accommodate the bandwidth needs of the applications because the number of pins available in a package has been growing only at a moderate rate of 11% per year [16].

Another metric, energy-per-bitrate (J/bps) is more suitable to account for this pin constraint. The energy-per-bitrate metric favors the faster link if the two links dissipate the

same energy per bit. The goal of this dissertation is to design high-speed links with low energy-per-bitrate ratios in order to maximize the total off-chip communication bandwidth in power- and pin-limited environments.

1.3 Approaches for Low Energy-per-Bitrate Ratio

The energy-per-bitrate metric is in fact analogous to the energy-delay product, which is a metric to assess trade-offs between speed and power in digital systems [18],[19],[20]. Numerous ways of lowering the energy-delay product have been proposed in literature and we can extend these techniques to improve the energy-per-bitrate ratio of high-speed links. This dissertation addresses largely two of them, adaptive power-supply regulation and the use of parallelism.

1.3.1 Adaptive Power-Supply Regulation

Lowering the supply voltage is often the most effective way of reducing power dissipation, especially when the peak performance is not needed [21]. This is because as the supply voltage is reduced, the speed of CMOS circuits drops linearly, while the power dissipation drops cubically, as shown in Figure 1.6. Therefore, when the maximum performance is not needed, a large reduction in power is possible by lowering the supply voltage. For example, as marked in Figure 1.6, when 60% of the peak performance is sufficient for system needs, the power can be reduced by a factor of 3 by lowering the supply voltage by 36%. The idea of scaling both the frequency and the voltage has demonstrated large power savings in embedded microprocessors and digital signal processors (DSPs) [22],[23],[24],[25].

Adaptive power-supply regulation maximizes this power saving by lowering the supply voltage to the minimum that meets the speed requirements. An adaptive power-supply regulator continuously monitors the on-chip circuit speed and regulates the supply voltage accordingly. For this purpose, the adaptive power-supply regulator uses a reference circuit which resides on the same chip with the supported digital system and whose speed can track that of the system over process, voltage, and temperature variations [26],[27],[28],[29], [30],[31].

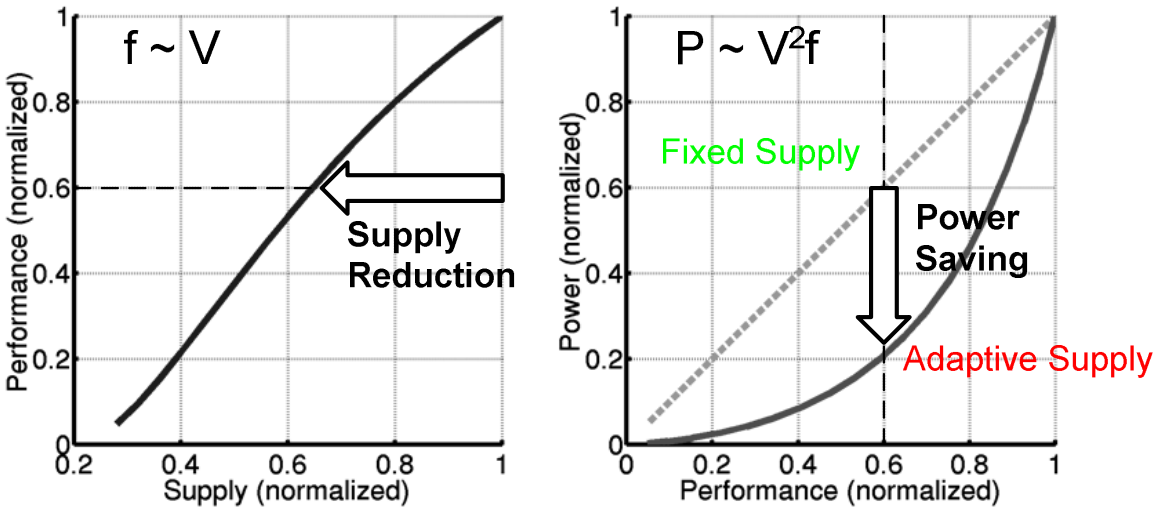


Figure 1.6: Scalings of (a) performance and (b) power with respect to supply voltage in CMOS circuits

Wei et al. [32] demonstrated that adaptive power-supply regulation can be extended to high-speed links to achieve the best energy-efficiency. In addition to power saving, the adaptive supply was found to have several other benefits. First, the adaptive supply makes the circuit performance more predictable and thus requires less design margins for the remaining uncertainties. The circuit speed is continuously being monitored and regulated via supply voltage control, so die-to-die variations are compensated and only within-die variations remain. Second, the adaptive supply can be used to scale many link parameters that need to vary proportionally with the clock frequency. For example, the adaptive supply served as a global bias adjusting the bandwidth of the delay-locked loop, the slew rate of the transmitted signal, and the input bandwidth of the receivers. Without an adaptive supply, scaling of these parameters requires separate biasing circuits.

As the supply voltage is reduced, the power dissipation of a high-speed link will drop significantly, but the performance of the link will also degrade. Therefore, it is important to find an optimal balance between bitrate and power. The lowest power-per-bitrate solution, or equivalently, the lowest energy solution is to operate at the lowest possible voltage,

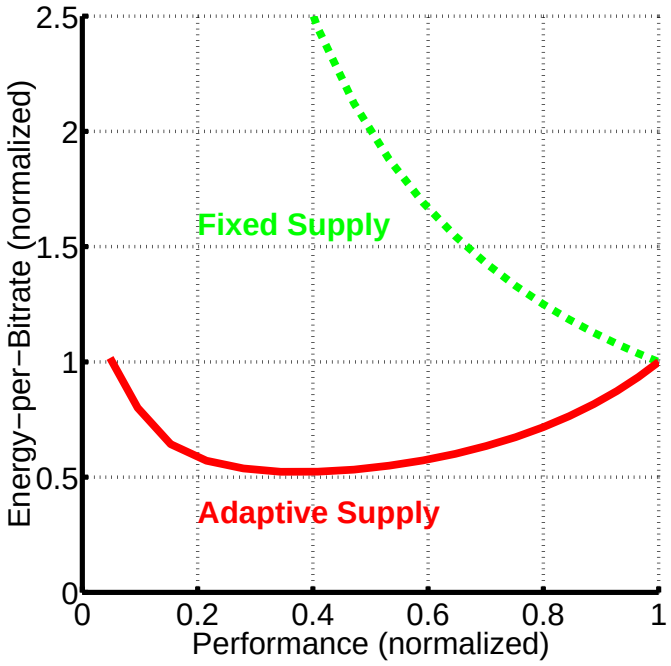


Figure 1.7: Energy-per-bitrate ratio scalings: cases for the adaptive supply and the fixed supply

which is clearly not a solution of interest. The power versus frequency scaling in Figure 1.6(b) suggests that initially at the highest supply, lowering voltage results in a large reduction in power compared to a small loss in speed, but the benefit diminishes as the voltage continues to drop. When the voltage becomes too low, a small saving in power can incur a large loss in speed. We can therefore expect that the optimal operating point should lie somewhere in the middle.

The energy-per-bitrate metric can help find this optimal point, as shown in Figure 1.7. For most digital systems, the minimum energy-delay product is found at voltages between $1.5Vth$ and $3Vth$ depending on the carrier velocity of the device, where Vth is the threshold voltage [20]. If the power of a high-speed link scales similarly to that of a digital system, the lowest energy-per-bitrate points for links are expected to be in a similar range. It should be noted that when the supply voltage is fixed, it is most efficient to operate the link at its highest bitrate because scaling frequency only does not provide enough power savings to justify low-speed operation.

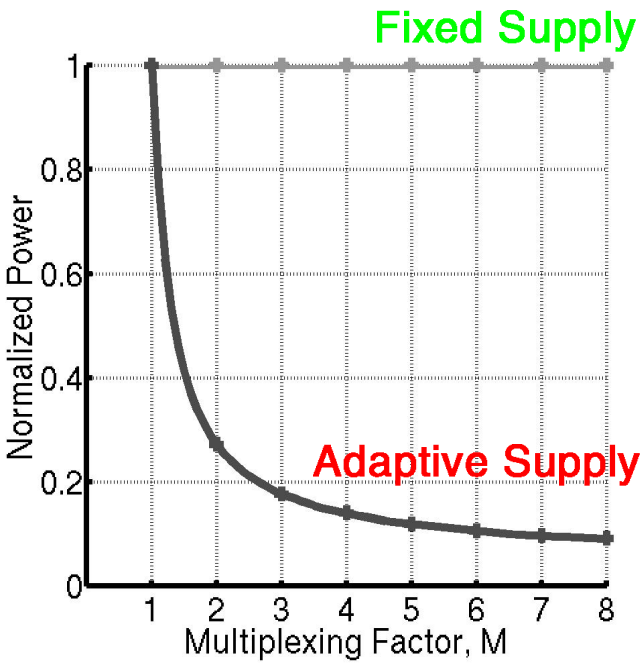


Figure 1.8: Power saving via parallelism

1.3.2 Parallelized Architecture

Chandrakasan et al. [33] found that parallelism can significantly reduce the power dissipation of a digital system. For a certain desired throughput, having M subsystems operating in parallel relaxes the frequency requirement f of each subsystem by a factor of M, while increasing the effective switched capacitance C_{eff} and the occupied area roughly by a factor of M. If the supply voltage V is dynamically scaled to match the frequency f, V will be lower for larger M. Since the power of an active digital system is dominated by dynamic switching power, expressed as $C_{eff}V^2f$, combining the scaling of V and f, the total power dissipation will decrease with M, as plotted in Figure 1.8.

High-speed links can be parallelized in many ways. One example is to use time-division multiplexing, as shown in Figure 1.9. A set of parallel transmitters and receivers rotating their duties in time can together attain a high bitrate while each operates at a lower frequency [2],[34]. The timing of the active period of each transmitter and receiver is controlled by equally-spaced, multiple phases of the lower frequency clock. Another method of exploiting parallelism is multilevel signalling. Multiple signal levels represent more

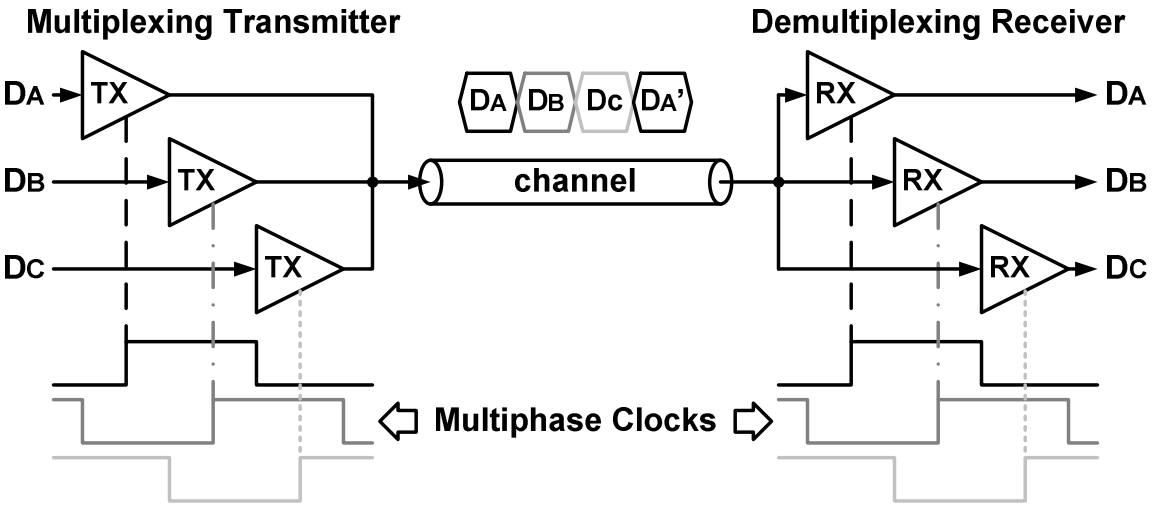


Figure 1.9: An example of exploiting parallelism in links: time-division multiplexing

than one bit per symbol period, thus achieving a higher bitrate than the clock frequency [35]. However, the penalty in power and area of multilevel signalling is larger than that of time-division multiplexing because $2^M - 1$ levels are required to represent M bits per cycle.

1.4 Adaptive-Supply Serial Link Overview

This dissertation describes an energy-efficient high-speed serial link that is suitable for a high degree of integration on a single chip. The energy-per-bitrate ratio of the link is improved by using an adaptively-regulated supply voltage and parallelized transceivers. Figure 1.10 illustrates the overall architecture of this adaptive-supply serial link. The adaptive power-supply regulator determines the optimal voltage that minimizes power dissipation for a given operating frequency and supplies it to the link circuitry, including the transmitters, the receivers, and the clock recovery and generation circuits. Both the transmitter and the receiver are parallelized for low-power, high-speed operation. The clock generation circuit precisely controls the timing at which each bit is transmitted, and the clock recovery circuit maintains the optimal timing for the receiver to sample the incoming data stream. Both clocking circuits generate multiphase clocks for the parallelized transceiver and special efforts are made to keep uniform spacing between phases.

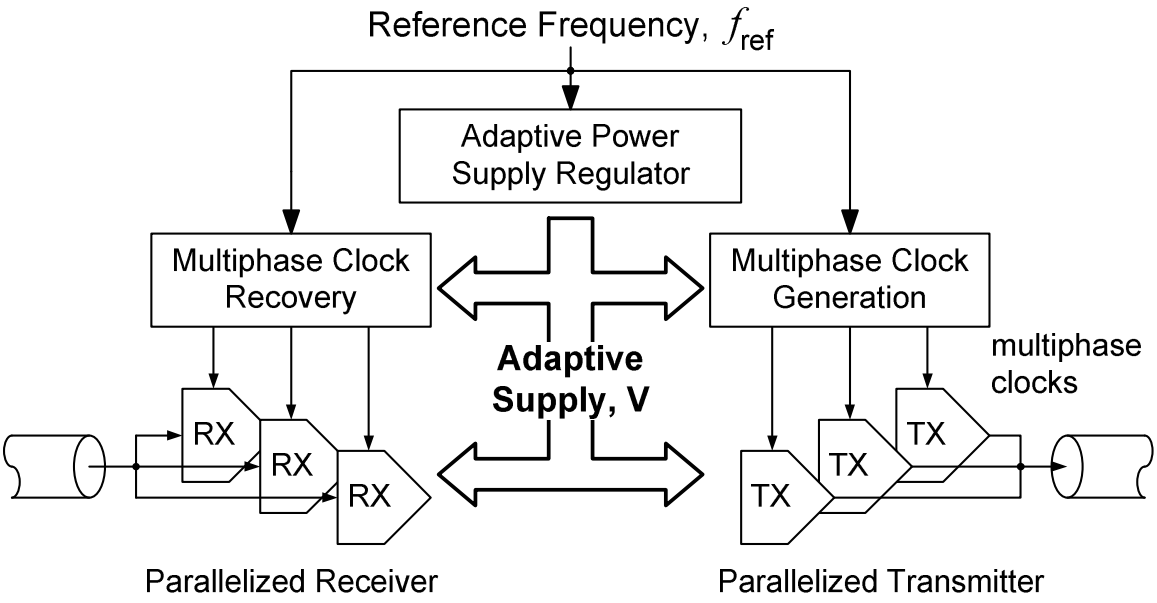


Figure 1.10: Adaptive supply serial link overview

Chapter 2 presents a power-efficient implementation of the adaptive power-supply regulator. Like most previous work in literature [26],[27],[28],[29], this design uses a buck converter for efficient dc-to-dc conversion. Thus, the chapter focuses on the design of the controller that stabilizes the feedback loop. It describes the first digital implementation of a nonlinear sliding controller for adaptive power-supply regulation [30],[31]. Unlike linear controllers, sliding controllers are generally insensitive to parameter variations, which enables the digital sliding controller to operate at variable frequency and variable voltage. Therefore, the controller achieves high efficiencies in power over a wide set of operating conditions.

The parallelized transmitter and receiver circuits are discussed next in Chapter 3. The parallelism is exploited in the form of time-division multiplexing as shown in Figure 1.9. Since the parallelized architecture helps reduce power only when it can operate at low voltage, a design that can operate at a supply as low as 1.6 times the threshold voltage is presented. Although the presented low-voltage techniques incur overhead in power and area, they will gain more importance as the ratio between nominal supply and threshold voltage continues to scale down and low-voltage operation becomes a requirement, not an option.

Chapter 4 discusses the timing circuits dedicated for each pin. The multiplexing transmitter and receiver require precisely-generated multiphase clocks to control their timing. Generating these multiphase clocks locally at each I/O pin is the best option for keeping static phase offsets and clock jitter low, despite the potential overheads in power and area. Chapter 4 presents a dual-loop architecture that exploits the adaptive power-supply regulator as the global frequency regulation loop. It is found that the adaptively-regulated supply can significantly reduce the power and area of the per-pin timing generation and recovery circuits.

Adaptive supply and parallelized architecture can greatly improve the speed and power efficiency of a high-speed link, but transistor mismatch poses a limit to further improvement. For example, the static phase offsets of the multiphase clocks becomes worse as the number of phases increases and the supply voltage decreases. Chapter 5 presents the measurement data from an experimental adaptive-supply link chip fabricated in 0.25μm CMOS and discusses the trade-offs among speed, power, and quality of the link [36]. The data is then projected to the case of 0.05μm CMOS technology, to understand how these trade-offs will scale.

Chapter 2

Digital Sliding Controller for Adaptive Power-Supply Regulation

The role of the adaptive power-supply regulator is to keep the supply voltage as low as possible to minimize the power dissipation of the I/O circuits. This minimum supply voltage is determined by the required performance level, i.e. the desired bitrate of the link. Although the speed of the CMOS circuits is roughly proportional to the supply voltage, the exact mapping between the bitrate and the supply voltage may vary depending on the process and temperature conditions and is unknown prior to fabrication.

An adaptive power-supply regulator therefore uses a feedback control, which monitors the circuit speed and adjusts the supply voltage dynamically, as shown in Figure 2.1. The input to the adaptive power-supply regulator is the desired operating frequency, f_{ref}, which directly governs the performance of most synchronous systems including high-speed links. The DC/DC converter then generates the voltage V as the output, so that the frequency of the reference circuit f equals the input frequency f_{ref}.

The reference circuit models the critical path of the supported system and tracks its V-to-f relationship over process and temperature variations. In other words, the delay of the reference circuit must closely track the critical path delay of the system at all possible operating points. In CMOS circuits, this reference circuit is often implemented simply as a chain of inverters, relying on the fact that the delays of various CMOS logic gates scale approximately in proportion to the inverter delay [10]. However, for some complex digital

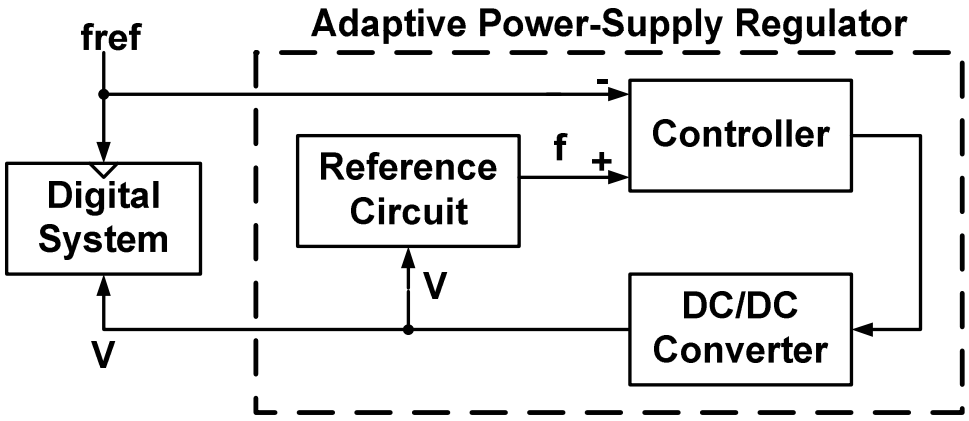


Figure 2.1: A general adaptive power-supply regulator

systems, it is not always easy to find such a good reference circuit.[1]

Fortunately in high-speed links, a chain of inverters can accurately model the critical path. It has been demonstrated that the I/O circuits can be made sufficiently fast so that only the maximum clock frequency allowed on the chip limits the link's performance [10],[32],[37]. The maximum clock frequency is limited by the time needed for the clock buffers to swing their outputs high and low every cycle without a loss in amplitude, which can be expressed as some constant number of inverter delays at a given operating condition. Figure 2.2 plots the attenuation in swing amplitude versus the clock period normalized to one fanout-of-4 inverter delay when the clock signal is passed through a clock buffer chain. Here the clock buffers are inverters with a fanout of 4, optimized for the shortest delay driving a large load [38]. The graph suggests that the minimum on-chip clock period at each operating condition is a fixed number of inverter delays (about 6-8) operating at that condition. Therefore, a string of inverters is the best candidate for the reference circuit modeling the link's performance.[2]

For efficient DC/DC conversion, switching regulators are most desirable [39],[40]. One example is the buck converter shown in Figure 2.3. The power transistors drive a full-swing

[1]For example, the wire delay does not scale with the inverter delay and makes it difficult to choose a reference circuit especially when there are multiple contending critical paths in the system. Depending on the amount of wire delay that each critical path contains, the path delays may scale differently with supply voltage and the dominant critical path may not be consistent.

[2]It is a question whether the timing requirements of the analog circuits in links will also scale with the inverter delay. Chapter 3 discusses the transceiver circuits that satisfy this property even at low supply voltages.

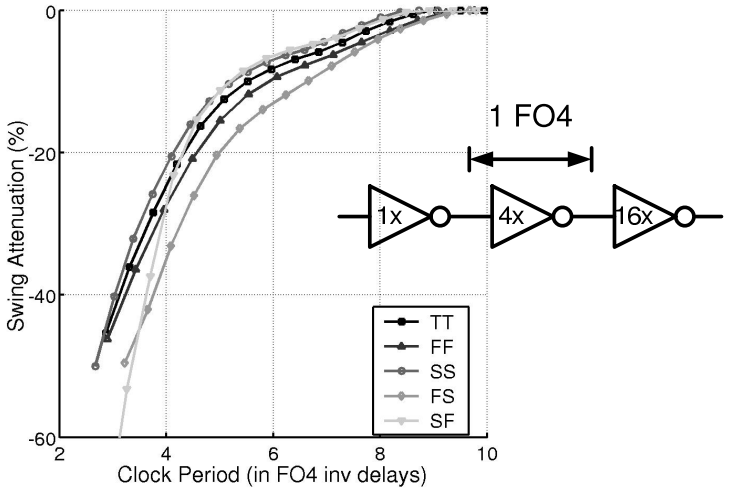


Figure 2.2: Minimum on-chip clock period

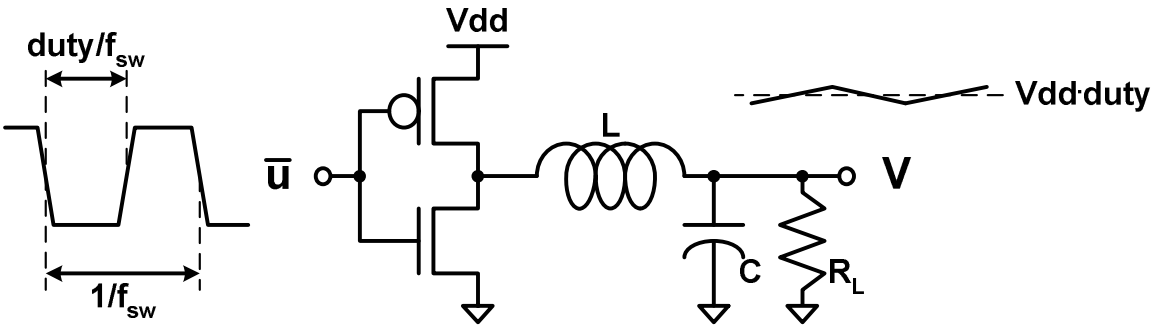


Figure 2.3: A buck converter

periodic pulse, which is then filtered by the inductor-capacitor (LC) filter. If the switching frequency of this pulse is sufficiently higher than the cut-off frequency of the filter, then the output voltage V becomes the time-average of the input pulse, or $Vdd \cdot duty$, where Vdd is the pulse swing and $duty$ is the duty-cycle. Since this filter is ideally lossless, over 90% conversion efficiency is typically achieved.

The controller stabilizes the feedback loop by applying an appropriate control input u to the buck converter that produces the desired output. The controllers for adaptive power-supplies differ from those for conventional switching power-supplies in the sense that the controlled variable is the frequency of the reference circuit, rather than the voltage. In most

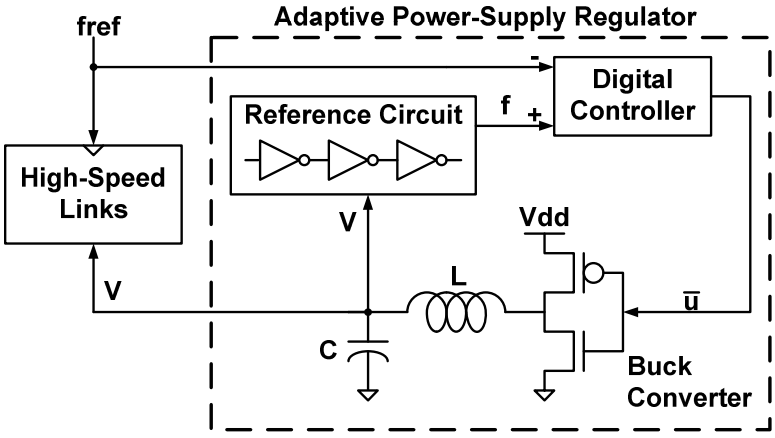


Figure 2.4: Block diagram of the adaptive power-supply regulator for high-speed links

cases, the reference circuit is a digital circuit and its frequency is conveniently measured in digital format, for example, the output of a counter [29] or a phase detector array [31]. Also, the buck converter takes a discrete input value, either Vdd or 0. For these reasons, digital controllers are more suitable for interfacing between the reference circuit and the buck converter and most controllers for adaptive power-supplies were in fact implemented in digital circuits. Digital controllers are also robust, portable, and capable of operating at low voltages.

Figure 2.4 shows the block diagram of the adaptive power-supply regulator for a high-speed I/O system. The reference circuit is implemented as either a delay line or a ring oscillator made of CMOS inverters and the buck converter performs the DC/DC conversion. The rest of this chapter focuses on the design of the digital controller that closes the feedback loop.

Section 2.1 examines the problem of the prior-art digital controllers for adaptive supplies: the fixed controller power dissipation. To overcome this problem, a digital controller that uses a nonlinear control mechanism, called sliding control, is proposed. For switching supplies, sliding controllers are known for robust stability and fast transient response and they are usually implemented in analog circuits [41],[42]. Section 2.2 introduces sliding control theory and discusses some design issues of these analog sliding controllers.

Unfortunately, implementing the same controller in digital circuits is not straightforward because it needs information on the time derivative of the output variable. Since the

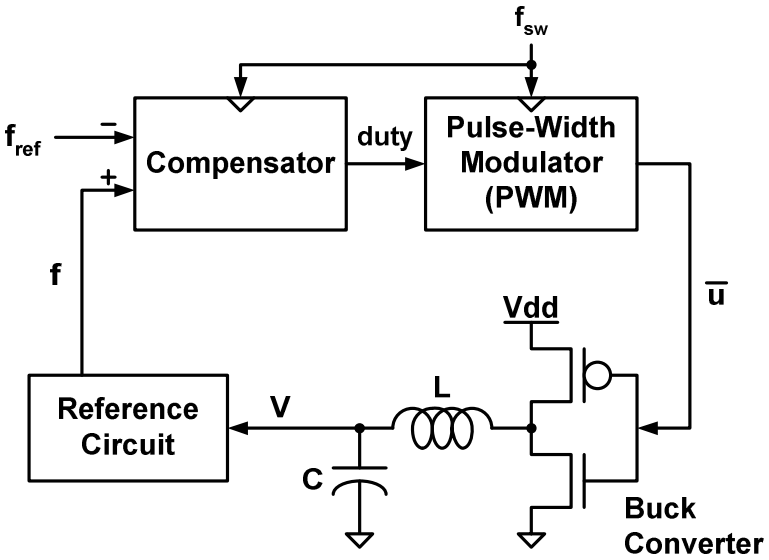
delay of the reference circuit is measured only in finite steps, the delay step is usually too coarse to accurately estimate the change in the delay between the sampling instants. Section 2.3 proposes an equivalent, but different form of sliding control that is more suitable for digital implementation. Since the sampling time step is much finer than the delay step, this new sliding control measures the elapsed time for a fixed change in the delay. This greatly improves the resolution in estimating the derivative, although some discrepancy with the original sliding controller still remains due to finite quantization steps in time and delay measurements.

Section 2.4 presents the circuit implementation of this new sliding controller. The sensor circuits for both types of reference circuit, delay line and ring oscillator, are discussed. Section 2.5 analyzes the measurement results from the prototype chip fabricated in 0.25-μm CMOS process.

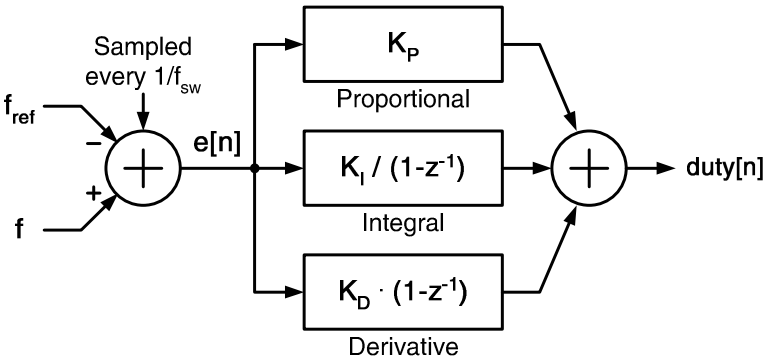
2.1 Prior Art in Digital Controller Design

A number of digital controllers for adaptive power-supplies have been published in literature, but most of them have a drawback that the controller's power dissipation does not scale with the reference frequency, f_{ref}. These controllers have fixed timing requirements internally, e.g. that they must operate at an externally-fixed frequency, and thus demand a fixed supply voltage to meet this timing requirement. Since the power delivered to the load scales as $V^2 f_{ref}$, the power efficiency can be severely degraded at low f_{ref}'s if the controller power remains unchanged.

For example, many controllers use pulse-width modulators (PWMs) and regulate the output voltage by adjusting the duty-cycle of a fixed-frequency pulse that drives the buck converter [26]-[29],[22],[25]. Figure 2.5(a) shows the simplified block diagram of the PWM-based controller. This type of controller is inherently a sampled-data system, where the error in frequencies, $f - f_{ref}$, is sampled at the PWM's operating frequency and the control action is decided upon this value. Since the continuous-time loop behavior including stability depends strongly on the sampling frequency, the controller's operating frequency is usually fixed by an external source. If the frequency is allowed to vary, the feedback loop may go unstable due to the resulting change in loop parameters.



(a)



(b)

Figure 2.5: PWM-based controllers: (a) general block diagram, (b) PID controller

Wei et al. [29] recognized this problem in the controller that used the lead-lag compensator, i.e. the proportional-integral-derivative (PID) controller shown in Figure 2.5(b). When the sampling frequency varies, loop parameters such as the loop gain and the pole/zero frequencies also vary undesirably, causing the feedback loop to lose stability. However, operating the controller at a fixed supply voltage and frequency causes the controller power to scale from 54.3mW only to 32.1mW when the load power drops from 1.38W to 279mW. The power efficiency degrades from 92.9% to 81.9% mainly because of the increased overhead of the controller power (from 4.55% to 9.44%). To mitigate this problem, Wei proposed a scheme that compensates for the variations in the loop parameters due to the varying sampling rate, but the resulting complexity was high.

The digital sliding controller proposed in the following sections can reliably operate over variable frequency and voltage, allowing its power to scale as a constant fraction of the load power. Therefore, the power efficiency is not limited by the controller power, but only by the inherent loss of the buck converter.

2.2 Sliding Control

Sliding control is a nonlinear control mechanism widely used in switching power supplies because of its robust stability and fast transient response [41],[42]. Sliding controllers for switching power-supplies are typically implemented in analog circuits and Figure 2.6 shows the block diagram of such an analog sliding controller. The controller regulates the output voltage V via a buck converter so that V matches the reference voltage V_{ref}.

The basic operation of this sliding controller is as follows. Unlike PWM-based controllers, the sliding controller chooses the buck converter input to be either 0 or Vdd at each instant, rather than determining the duty-cycle of a fixed frequency pulse. The switching frequency of the buck converter input u is therefore not fixed by an external source and can vary with the operating point. The buck converter is switched either to 0 or to Vdd based on the output of the compensator, which is the voltage error, $V - V_{ref}$, scaled by $1/\tau$ plus the time derivative of the error. The compensator dampens the resonant filter response of the buck converter and thus reduces overshoot during transient.

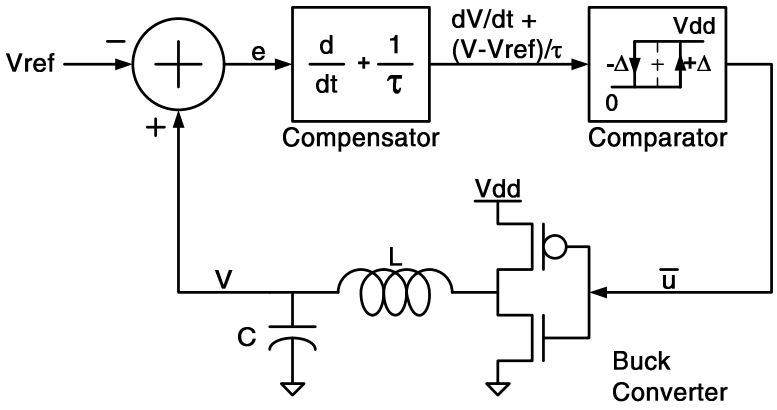


Figure 2.6: Analog sliding controller with a buck regulator

When the loop acquires lock, the derivative dV/dt becomes negligible and the controller reduces to a simple one that switches u based merely on the voltage error, $V - V_{ref}$. Hysteresis in the comparator controls the magnitude of the output voltage ripple and the switching frequency by forming a limit cycle at steady-state.

This control mechanism is highly nonlinear due to the comparator action. The comparator output only takes one of two discrete values, 0 or Vdd, depending on the criteria outlined above. The feedback loop behavior is therefore insensitive to parameter variations as long as they do not change the comparator outputs. It will be shown later that this robustness of sliding control allows the digital sliding controller to operate at variable frequency and supply voltage.

The nonlinearity induced by the comparator makes it inappropriate to apply linear system theory for analysis. Such a nonlinear system can be analyzed only by observing the state variables of the system directly. Fortunately, for second-order systems such as buck converters, the state of the system is a two-dimensional quantity which can be plotted easily on a graph. Visualization of the system behavior is thus possible by plotting the time trajectories of the state, called phase portraits [43]. By using these phase portraits, the next subsection analyzes the analog sliding controller in Figure 2.6.

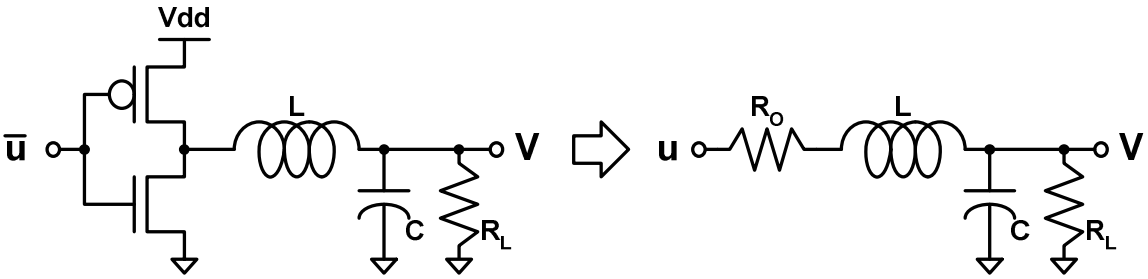


Figure 2.7: Linear model of the buck converter

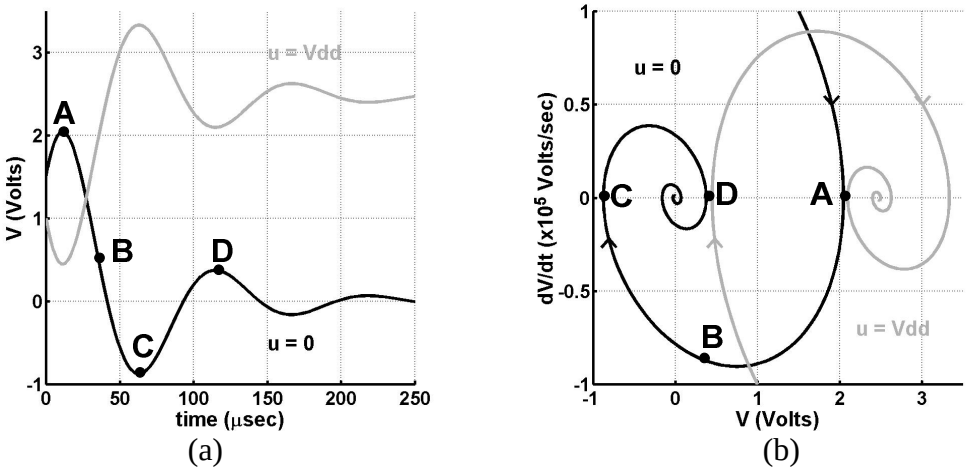


Figure 2.8: (a) Transient responses of the buck converter, (b) phase portraits corresponding to the responses in (a)

2.2.1 Phase Portrait Analysis

The linear model of the buck converter is shown in Figure 2.7, where the power transistors are replaced with a resistor. With two energy storage elements, the inductor (L) and the capacitor (C), the buck converter is a second-order system. The series resistance of the power transistor, Ro, is usually kept low for high efficiencies and the buck converter shows an underdamped response to an input step as shown in Figure 2.8(a). The output voltage V initially oscillates at the resonant frequency $\omega_n \cong 1/\sqrt{LC}$ but eventually settles to the input value u. Figure 2.8(a) shows two transient responses of the buck converter when the input u is either 0 or Vdd (=2.5V).

Another way of displaying the buck converter response is to plot the time trajectories of

the system state, as shown in Figure 2.8(b). As a second-order system, the state of the buck converter can be completely described by two scalar variables: the output voltage V and its derivative dV/dt. With V on the x-axis and dV/dt on the y-axis, each state corresponds to a point on this graph and the progression of the state over time will project to a curve. These time-trajectory curves of the state are called *phase portraits*. For phase portraits, the time variable is implicit.

Given an initial state, the system is determined to follow the phase portrait originating from the corresponding point on the state space. For example, the buck converter response starting at point **A** in Figure 2.8(b) visits the points **B**, **C**, and **D** in sequence following the phase portrait of $u = 0$. These points are also marked in the equivalent transient response in Figure 2.8(a). With a set of phase portraits plotted on the state space, it is easy to predict the system behavior starting at any state. Figure 2.9 plots some phase portraits of the buck converter when the input u is either 0 or Vdd.

The sliding controller controls the output voltage by imposing a boundary line on the state space, as in Figure 2.9. Recall that the comparator in Figure 2.6 switches the buck converter to 0 when the quantity $dV/dt + (V - V_{ref})/\tau$ is positive and to Vdd when it is negative, ignoring the hysteresis at this moment. The equation $dV/dt + (V - V_{ref})/\tau = 0$ corresponds to the linear boundary curve shown in Figure 2.9. Therefore, we can equivalently say that the sliding controller switches the buck converter input u to 0 when the state is located above the boundary line and to Vdd when it is below the boundary line. The boundary line divides the state space into two regions and the state of the overall sliding control system follows the phase portrait with the input u that corresponds to the region of the current state.

Notice in Figure 2.9 that these phase portraits are pointing toward the boundary line. Therefore, from any starting point, the state of the system will first reach the boundary line, either from top down or from bottom up. After reaching the boundary, the state is then confined to the boundary line and slides on it toward the settling point, $(V_{ref}, 0)$, thus the name *sliding control*. The boundary line is also called the *sliding surface*.

Once the state is confined to the sliding surface, the system behaves according to the equation $dV/dt + (V - V_{ref})/\tau = 0$, which is equivalent to the behavior of a first-order

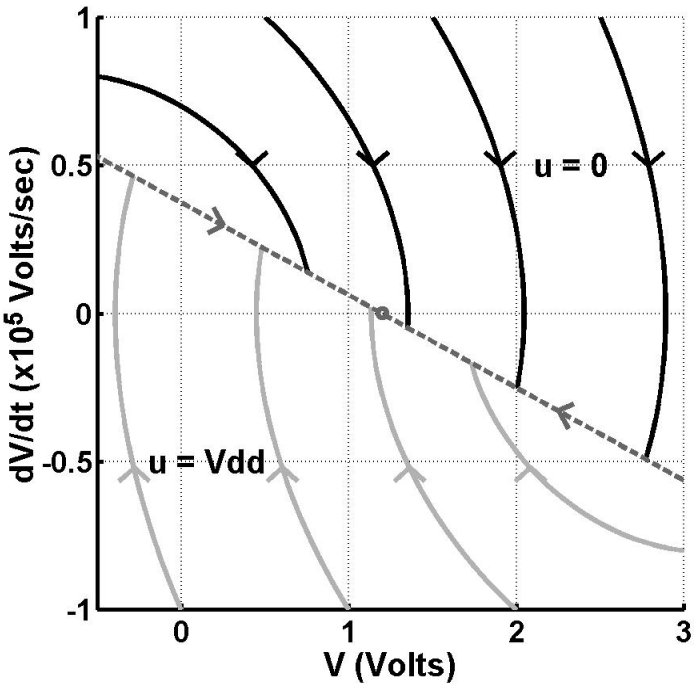


Figure 2.9: Phase portraits of the sliding controller, for V_{ref}=1.2V and Vdd=2.5V

linear system with the time constant τ. Thus, τ is a design parameter that sets the input-tracking bandwidth of the regulator. The other parameters have little influence on the system behavior. As long as the phase portraits on both regions direct toward the boundary line, the system will behave as described above. There is a certain range of τ that satisfies this so-called *sliding condition* and a rule of thumb is that the tracking bandwidth $1/\tau$ can be as large as twice the resonant frequency of the buck converter, ω_n, to satisfy this condition for V ranging from 0 to Vdd.

As mentioned earlier, the hysteresis in the comparator controls the switching frequency of the buck converter and the magnitude of the output voltage ripple. The comparator in Figure 2.6 switches to 0 when the compensator output is greater than $+\Delta$, and to Vdd when it is less than $-\Delta$. In state space, it means that the boundary line splits into two lines and the steady state becomes a limit cycle formed between these two lines, as shown in Figure 2.10.

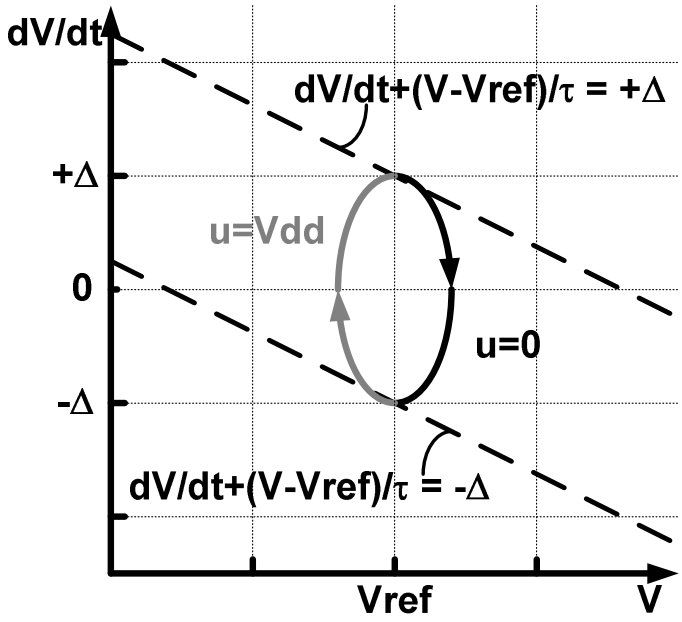


Figure 2.10: Phase portraits of the limit cycles during lock

By exploiting the geometry of this phase portrait, we can derive the analytical expression for the switching frequency f_{sw} and the peak-to-peak voltage ripple ΔV_{pp} (see Appendix A). These expressions are approximations which hold for sufficiently small $V_H \equiv \Delta/\omega_n$ and for $V_H \ll V_{ref} \ll Vdd - V_H$:

$$\begin{aligned} f_{sw} &\cong \tfrac{\omega_n}{2V_H} \cdot \tfrac{V_{ref}\cdot(Vdd-V_{ref})}{Vdd}, \\ \Delta V_{pp} &\cong \tfrac{V_H^2}{2} \cdot \tfrac{Vdd}{V_{ref}\cdot(Vdd-V_{ref})}. \end{aligned} \tag{2.1}$$

The larger the hysteresis, the lower the switching frequency and the larger the voltage ripple. The switching frequency and the voltage ripple also vary as a function of the reference voltage V_{ref}. At $V_{ref} = Vdd/2$, the switching frequency is at maximum and the voltage ripple is at minimum. From the above equations, we can also derive the relationship between f_{sw} and ΔV_{pp}:

$$f_{sw} \cdot \Delta V_{pp} \cong \frac{\omega_n \cdot V_H}{4} = \frac{\Delta}{4}. \tag{2.2}$$

2.3 Digital Sliding Controller

In the previous section, sliding control for switching power-supplies was analyzed. The sliding controller uses information of the output voltage V and its derivative dV/dt in order to regulate V to the reference V_{ref}. These controllers are implemented in analog circuits where V and dV/dt are measured directly as analog quantities [41],[42].[3]

Adaptive power-supplies, on the other hand, regulate the frequency or the delay of the reference circuit. In other words, the controlled variable is the frequency f, instead of the voltage V. The sliding controller for adaptive power-supplies then requires the information on f and its derivative df/dt. However, unlike in switching power-supplies, this derivative information is not directly available. The controller needs to estimate the derivative df/dt by measuring the change in f (Δf) and the change in t (Δt) separately.

A common way of estimating the derivative is to measure the difference in f between the present and previous samples of f that are Δt apart in time. However, this approach requires a very high resolution in frequency detection. As will be discussed later, the sampling period has to be short enough to avoid undesirable effects due to the loop delay. During this short period, the change in frequency would be very small, such that it is very hard to distinguish the information from noise.

An alternative is to measure the elapsed time Δt for a fixed change in frequency Δf. This approach instead requires a high resolution in time measurement, but it is readily available since the sampling frequency has to be high anyway. Common digital frequency detection circuits such as counters or phase detector arrays measure the frequency f in finite steps, making this approach a good fit. Details on the sensor circuits that measure the frequency f and its change will be discussed later in Section 2.4.

When the derivative is estimated using the alternative approach, we can reformat the sliding control law to make the digital implementation simple. As described in the next subsection, the control law in fact reduces to checking a single counter.

[3] dV/dt is measured as the current flowing through the capacitor ($i_C = C \cdot dV/dt$) or the inductor ($i_L = i_C + V/R_L$).

2.3.1 Reformulation of Sliding Control

For the sake of discussion, we will divide the loop dynamics into two cases: when the feedback loop is in transient and when it is in lock. While the loop is in transient, the comparator hysteresis has little effect on the loop behavior and can be ignored. Then the sliding law can be simply put as:

$$df/dt + (f - f_{ref})/\tau \underset{<}{\overset{>}{or}} 0, \tag{2.3}$$

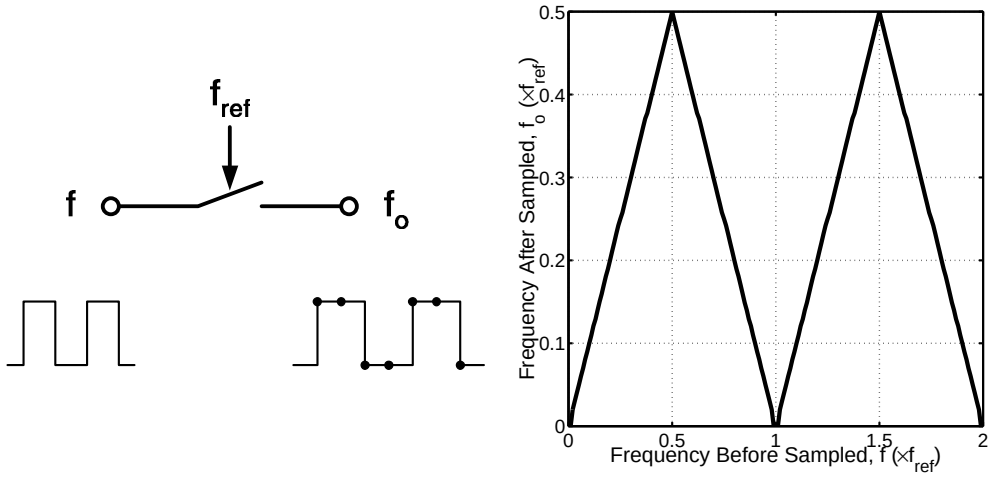
The control decision on whether to set the buck converter input u to 0 or Vdd depends on the polarity of the left-hand side of Eq(2.3). Rearranging this equation after substituting the derivative df/dt with its estimated value $\Delta f/\Delta t$ yields:

$$\Delta t \underset{<}{\overset{>}{or}} -\tau \Delta f/(f - f_{ref}), \tag{2.4}$$

where Δf is the fixed magnitude change in frequency and Δt is the elapsed time during that change. Note that Δf can be either positive or negative but Δt is always positive. Therefore, the comparison in Eq(2.4) is trivial if the right-hand side is negative, i.e. $sgn(\Delta f)/(f - f_{ref}) > 0$. This case corresponds to when the frequency f is moving farther away from the locking point f_{ref}, which is an uninteresting case. Except this case, the control law is:

$$\Delta t \underset{<}{\overset{>}{or}} N_\tau/(f - f_{ref}), \tag{2.5}$$

where N_τ is a design constant defined as $N_\tau = \tau \cdot |\Delta f|$. Now the sliding control law has become comparing the elapsed time Δt with the quantity $N_\tau/(f - f_{ref})$. This quantity may seem cumbersome to measure, as it is a constant divided by a difference in two frequencies. But suppose that a clock with a frequency $|f - f_{ref}|$ is available and the elapsed time Δt is measured with a counter triggered by this clock. Then Δt is equal to $N/|f - f_{ref}|$ and the



(a) (b)

Figure 2.11: (a) Sampling the f-frequency clock at a rate f_{ref}, (b) the frequency of the sampler output

sliding control law above finally reduces to:

$$N \begin{array}{c} > \\ or \\ < \end{array} N_\tau, \tag{2.6}$$

checking whether the count N has reached the threshold value, N_τ.

The clock with a frequency $|f - f_{ref}|$ can be obtained easily. As illustrated in Figure 2.11(a), when a rectangular wave with frequency f is sampled at a frequency f_{ref}, the resulting samples form a rectangular wave with a frequency varying as plotted in Figure 2.11(b). For $f_{ref}/2 < f < 3f_{ref}/2$, the output frequency is equal to the difference in two frequencies, $|f - f_{ref}|$. With some constraints on the frequency range being measured, a clock with a frequency $|f - f_{ref}|$ is generated with a simple flip-flop. More details will be discussed in Section 2.4.

Once the loop reaches lock, the analog sliding controller controls the magnitude of the voltage ripple ΔV_{pp} and the switching frequency f_{sw} via the comparator hysteresis, as in Eq(2.1). However, for digital controllers that have coarse resolution in frequency detection, the use of hysteresis is difficult. An alternative way to limit switching speed is to guarantee a minimum pull-down time for the buck converter [28]. This approach better suits a digital

implementation since it again measures time instead of frequency.

When the frequency of the reference circuit f has settled to a steady state, the change in f is small and thus the quantity $df/dt + (f - f_{ref})/\tau$ can be approximated simply as $(f - f_{ref})/\tau$. In this digital scheme, the frequency f will not make a change as large as $|\Delta f|$ and the derivative term is regarded as zero. Therefore, the control decision on the buck converter input u is based merely on the error term, $f - f_{ref}$. The digital sliding controller switches u to 0 when $f > f_{ref}$ and to Vdd otherwise.

In the guaranteed minimum pull-down time scheme, the buck converter input u does not switch from 0 to Vdd unless a certain amount of time has passed, even if the error $f - f_{ref}$ has turned negative. On the other hand, when the error turns positive, the buck converter starts pulling down immediately ($u = 0$) regardless of the pull-up time.

The enforced minimum pull-down time $T_{u=0}$ controls the voltage ripple and the switching frequency as the comparator hysteresis Δ does in analog sliding controllers. The relationship between $T_{u=0}$ and Δ is (see Appendix A):

$$T_{u=0} = \frac{2LC\Delta}{V_{ref}}, \tag{2.7}$$

where L and C are the inductance and the capacitance of the buck converter, respectively, and V_{ref} is the supply voltage that corresponds to f_{ref}. Since f_{ref} is roughly proportional to V_{ref}, the above equation can be expressed in terms of f_{ref},

$$T_{u=0} = K \cdot \frac{2LC\Delta}{f_{ref}} = \frac{N_\Delta}{f_{ref}}, \tag{2.8}$$

where K is the proportionality constant between f_{ref} and V_{ref} (Hz/V) and N_Δ is a constant defined as $2LC\Delta \cdot K$. Thus, to ensure the minimum pull-down time $T_{u=0}$, the digital controller can simply observe the counter triggered at the reference frequency f_{ref} and check if the count has reached the threshold N_Δ while the buck converter is driven to 0.

The decision chart shown in Figure 2.12 summarizes the new sliding control law formulated for the digital implementation. The digital controller has two modes, the transient mode and the steady-state mode, and each of them is associated with a design parameter, N_τ and N_Δ, respectively. N_τ governs the tracking bandwidth while the loop is in transient

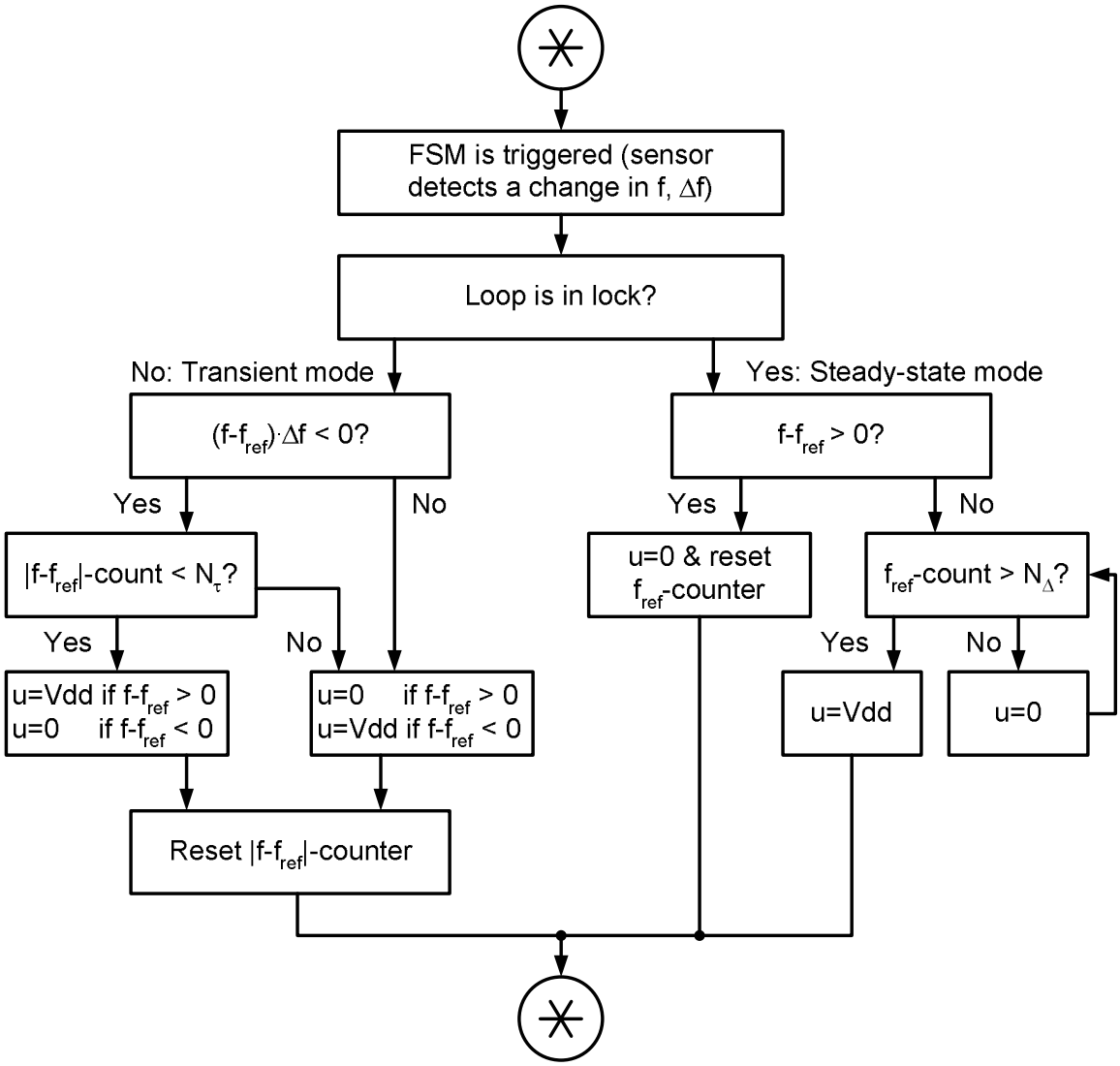
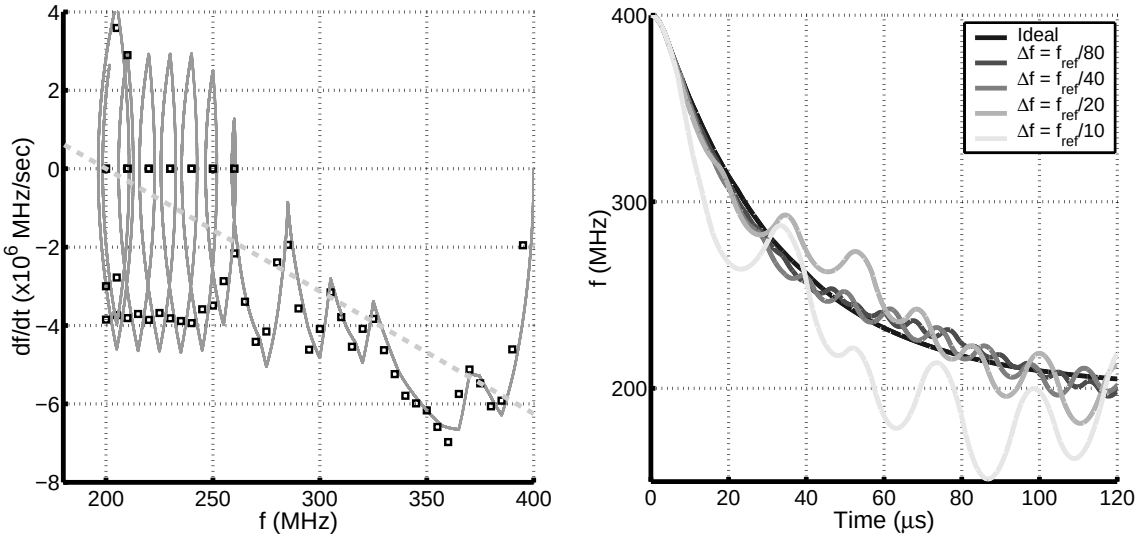


Figure 2.12: Decision chart illustrating the digital sliding control law



(a) (b)

Figure 2.13: Effects of finite resolution in f: (a) actual phase portrait when $\Delta f = f_{ref}/40$, (b) transient responses for various Δf's.

and N_Δ controls the switching frequency and the voltage ripple while in lock. Each of these parameters is proportional to the analog sliding control parameter τ and Δ, respectively, and thus Eq(2.1) and Eq(2.2) apply to the digital sliding controller as well. The new form of sliding control law enables the control decisions to be made simply by examining whether the counters have reached their thresholds.

2.3.2 Limitations due to Quantization Effects

The quantity $\Delta f/\Delta t$ will well estimate the ideal derivative df/dt if the values Δf and Δt are sufficiently small. However, in digital controllers, there are finite resolutions in measuring f and t. The coarse resolution can introduce measurement errors that are as large as the quantization step and thus cause the controller to deviate from its ideal behavior. This section discusses the effects of finite resolutions in frequency f and time t, respectively.

The finite resolution in f changes the first-order transient response of the ideal sliding controller. Figure 2.13(a) shows the phase portrait of the digital sliding controller when Δf is equal to $f_{ref}/40$. The solid curve is the trajectory of the buck converter state, whereas the

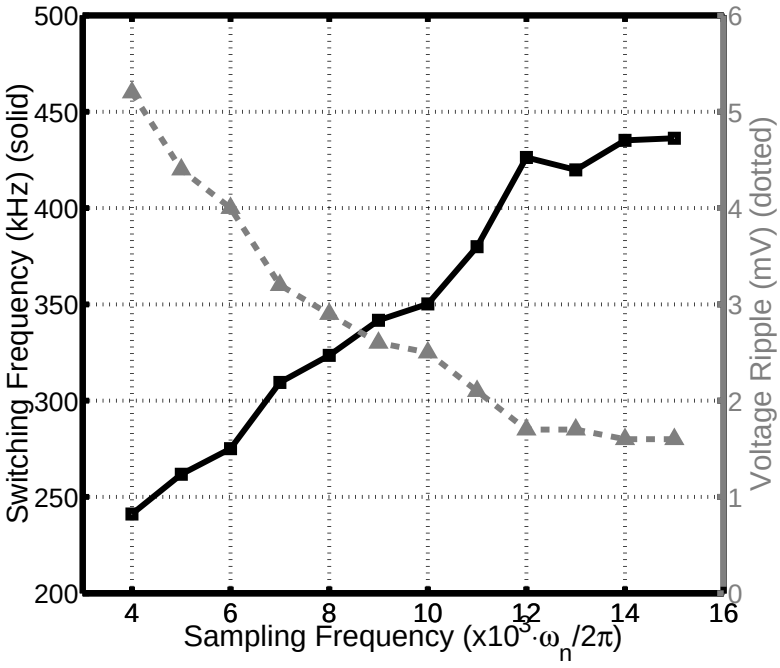


Figure 2.14: Effects of the finite time step Δt

points are the estimations made by the digital controller. The trajectory of an ideal sliding controller would lie on the dotted boundary line. The estimation error is large when the solid curve has a steep slope, i.e. when df/dt varies a lot while f moves by Δf. $\Delta f/\Delta t$ can only reflect the average value of df/dt during that interval and thus the feedback loop cannot react quickly to the current value of df/dt. As a result, the phase portraits are not tightly confined to the boundary line, especially when f is close to lock.

Figure 2.13(b) shows the deviation of the transient response from an ideal first-order exponential decay as Δf varies from $f_{ref}/80$ to $f_{ref}/10$. The wandering phase portraits in Figure 2.13(a) appear as small ripples, which grow larger as the frequency step becomes coarser. The plot suggests that for a response reasonably close to an exponential decay, Δf less than $f_{ref}/40$ is desirable.

The finite time step Δt can also cause error, especially in the steady-state behavior such as the switching frequency and the voltage ripple. These parameters are controlled by guaranteeing the minimum pull-down time $T_{u=0}$, which is measured only as an integer multiple of Δt. Thus, the coarse granularity in time can effectively increase $T_{u=0}$ by Δt

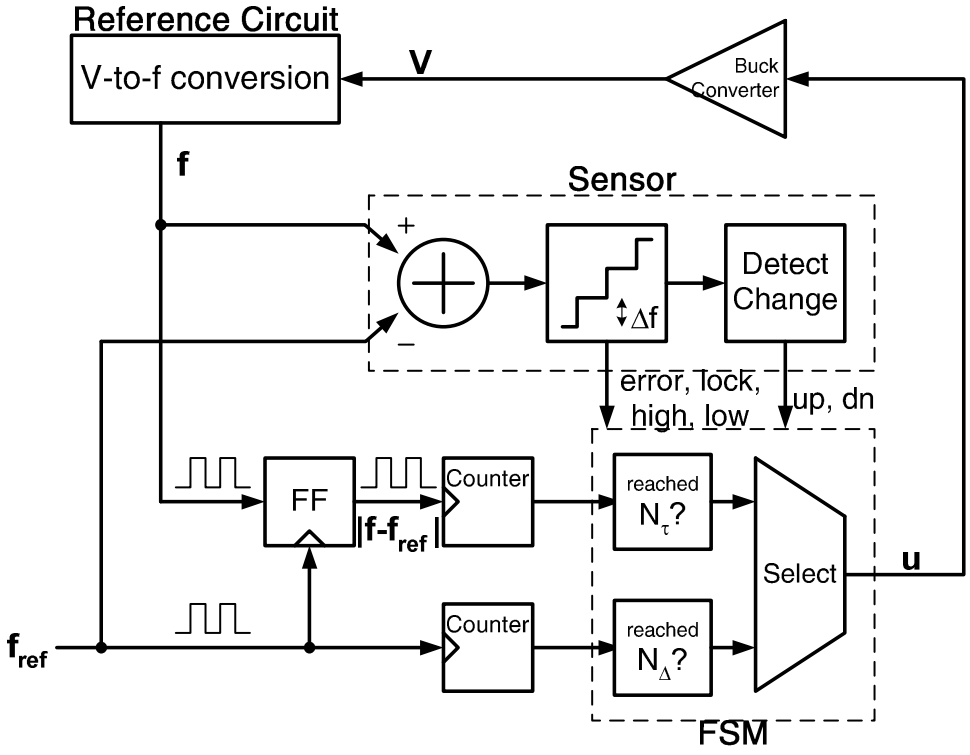


Figure 2.15: Architecture of the digital sliding controller

in the worst case. According to Eq(2.8), the increased $T_{u=0}$ is equivalent to an increased hysteresis in the comparator, and the values of switching frequency and voltage ripple will be altered likewise. Figure 2.14 plots the change in switching frequency and voltage ripple versus the sampling frequency. For negligible quantization effects, the time step Δt must be smaller than $1/12000$ of the resonant period, $2\pi/\omega_n$.

2.4 Circuit Implementation

The overall architecture of the digital sliding controller that implements the reformulated sliding control law is illustrated in Figure 2.15. The major components of the controller are the sensor that measures the frequency error of the reference circuit $f - f_{ref}$, the counters that measure the elapsed times, and the finite-state machine (FSM) that makes control decisions based on the operating mode.

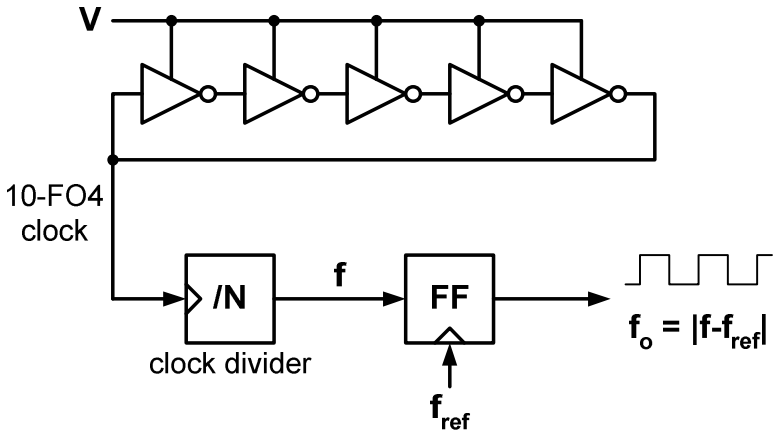


Figure 2.16: Circuits that generate $|f - f_{ref}|$-frequency clock

The sensor measures the delay or the frequency of the reference circuit f in quantized steps and triggers an event whenever it detects a change in this quantized value. The sensor provides the FSM with information on whether the error $f - f_{ref}$ is positive or negative (**error**), whether the loop is in lock (**lock**), whether the change Δf is positive or negative (**up**, **dn**), and whether f is too high or too low (**high**, **low**). These are the sufficient information for the FSM to make control decisions according to the sliding control law.

Between these events triggered by the sensor, the counters measure the elapsed time. One counter is triggered off of the $|f - f_{ref}|$ frequency clock and the other is triggered off of the reference clock. The $|f - f_{ref}|$ frequency clock is generated by the circuit shown in Figure 2.16 with the principle explained in Section 2.3.1. A ring oscillator matched to the reference circuit and running at the voltage V generates the clock with the frequency f.[4] This clock is then sampled by a flip-flop triggered at f_{ref}. If $f_{ref}/2 < f < 3/2 \cdot f_{ref}$, the resulting output will have frequency $|f - f_{ref}|$.[5]

When an event occurs, the finite-state machine takes an appropriate action depending on the loop's situation. First, since the $|f - f_{ref}|$ clock generation poses constraints on the

[4]Although Figure 2.16 show "f" originating from the reference circuit, the clock with the frequency f may not be available explicitly. Another ring oscillator operating off of voltage V thus generates this clock.

[5]Metastability of this sampling flip-flop is not such a major concern since this $|f - f_{ref}|$ clock is used only in the transient mode when f and f_{ref} are different and since only the average frequency information is relevant, but not the instantaneous clock level. However, in the actual implementation, multiple flip-flops were cascaded to reduce the chance of metastability [44].

frequency f, the FSM drives the buck converter hard toward lock when f is too high or too low as indicated by the sensor (**high**, **low**). Otherwise, the FSM chooses the buck converter input u based on the sliding control law outlined in Figure 2.12. During transient mode, i.e. with **lock** signal not asserted, the FSM examines if the $|f - f_{ref}|$-counter has reached N_τ. Once the loop is in lock, the FSM checks if the f_{ref}-counter has reached N_Δ before switching u from 0 to Vdd.

The next two subsections describe sensor implementations for two types of reference circuit: a delay line and a ring oscillator. For adaptive supply links, the reference circuit is the replica of the voltage-controlled oscillator (VCO) used in the local phase-locked loop (PLL), which will be described in more detail in Chapter 4. For modern microprocessors, a clock cycle is about 20 inverter delays [45] and a delay line can also be used as the reference circuit. In both cases, the robustness of sliding control allows the sensor to take samples on f at the varying rate, f_{ref}, as long as f_{ref} is high enough to avoid the quantization issues. Therefore, the digital sliding controller can operate at the variable frequency f_{ref} and at the regulated voltage V, in contrast to the previous controllers which must operate at the fixed frequency and voltage. The power dissipation of the digital sliding controller thus scales with that of the supported system load and the power efficiency can be kept high over a wide range of f_{ref} without being limited by the controller power overhead.

2.4.1 Sensor for a Reference Delay Line

The circuit details of the sensor when the reference circuit is a delay line are shown in Figure 2.17. In this example, the delay line is a string of fanout-of-4 inverters supplied off of the voltage V. The reference circuit essentially performs the V-to-f conversion and the delay line indicates the frequency f via the distance that a signal propagates down the line for a given amount of time. The farther the signal has traveled, the shorter the delay of each inverter and the higher the frequency f. The sensor in Figure 2.17 basically measures this travel distance by tapping the outputs of the individual buffers on the chain and sampling them with an array of binary phase detectors.

This example assumes that the critical path delay of the system is equivalent to 20 fanout-of-4 inverter delays and the frequency f is thus considered matched to f_{ref} if the

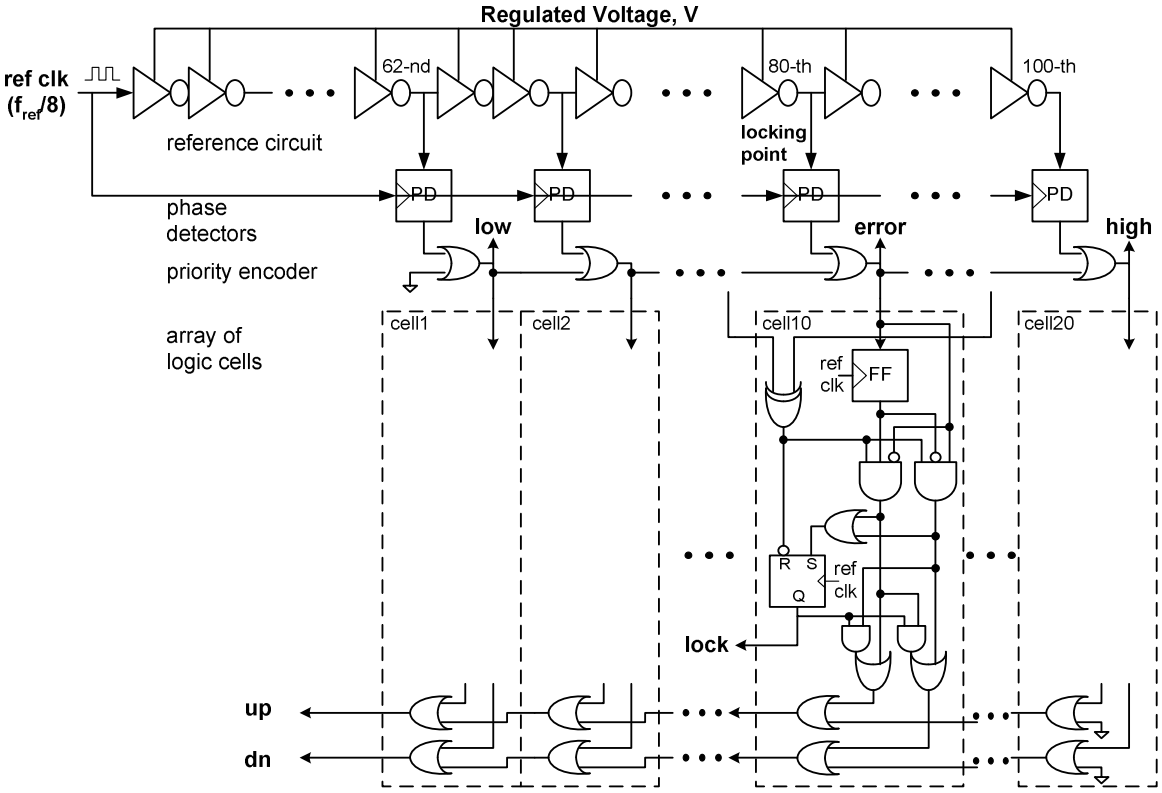


Figure 2.17: Sensor circuit implementation when the reference circuit is a delay line

signal propagates through 20 stages on the delay line during one reference clock cycle $1/f_{ref}$. However, to increase the resolution in frequency measurement, the sensor sees if the travel distance is 80 stages long for 4 cycle periods. In Figure 2.17, the reference clock is first divided by 8 and fed into the delay line. At the rising edge of this divided clock, the array of phase detectors samples the voltages on the line and detects the position of the clock's falling edge. Twenty phase detectors are placed at the output of every other inverter in between the 62nd and the 100th. This configuration provides the quantized measure of f from $3/4 \cdot f_{ref}$ to $5/4 \cdot f_{ref}$ with a uniform resolution of $f_{ref}/40$ ($= |\Delta f|$).

With this particularly long delay line, there can be multiple clock edges residing on the line simultaneously, especially when the voltage V is low and the total delay of the line is much longer than the reference clock period. In this case, the priority encoder implemented as a series of OR gates ensures that only the nearest edge to the input is detected.

The information that the sensor needs to provide to the FSM is obtained as follows. The

polarity of the error, $f - f_{ref}$, is obtained from the phase detector that samples the 80-th inverter output (**error**). Since the falling edge is being propagated, if the 80-th inverter output is still at 1 at the sampling moment, it means that the error $f - f_{ref}$ is negative. Likewise, The first phase detector that samples the 62-nd inverter output indicates whether the frequency is too low or $f < 3/4 \cdot f_{ref}$ (**low**). The last phase detector that samples the 100-th inverter output indicates whether the frequency is too high or $f > 5/4 \cdot f_{ref}$ (**high**).

The sensor triggers the FSM when it detects any change in the quantized measure of f or equivalently, when any of the phase detector outputs change. The array of logic cells shown in Figure 2.17 detects this change in the frequency f. There is one logic cell for each phase detector and each logic cell stores the previous output of the phase detector and compares it with the present output. When a certain cell detects a change, we consider that the frequency f has crossed the quantization boundary corresponding to the cell's position. Since the clock edge being detected is the falling edge, f is crossing the boundary upward if the phase detector output changes from 1 to 0 (**up**) and downward if it changes from 0 to 1 (**dn**). Since the priority encoder ensures that only one cell can detect change at a time, the **up** and **dn** signals from each logic cell can be collected simply through the series of OR gates to provide the aggregate outputs. Cascading the OR gates as in Figure 2.17 makes the internal critical path delay of the sensor scale with the supply voltage V. For example, if the voltage V is low so that the clock edge does not propagate far on the delay line, the **up** and **dn** signals also have the short paths to come back.

It is possible that the frequency f dithers across the same quantization boundary. For example, a certain phase detector output can change from 0 to 1 and later back to 0. This is especially the case when the loop is in lock and f dithers around the reference point f_{ref}. In this case, the frequency f has not really changed compared to the last boundary-crossing event and therefore the sensor should report zero change in f. In order to do this, the S-R flip-flop of each logic cell stores the information whether the cell had the last crossing event. The S-R flip-flop is set when the phase detector output changes and reset when the neighbor phase detector outputs are equal. If the crossing event occurs at the same position as the last one, the cell asserts both **up** and **dn** to indicate that the crossing event has happened but the change in f is zero.

The information on whether the loop is in lock mode is obtained from the S-R flip-flop

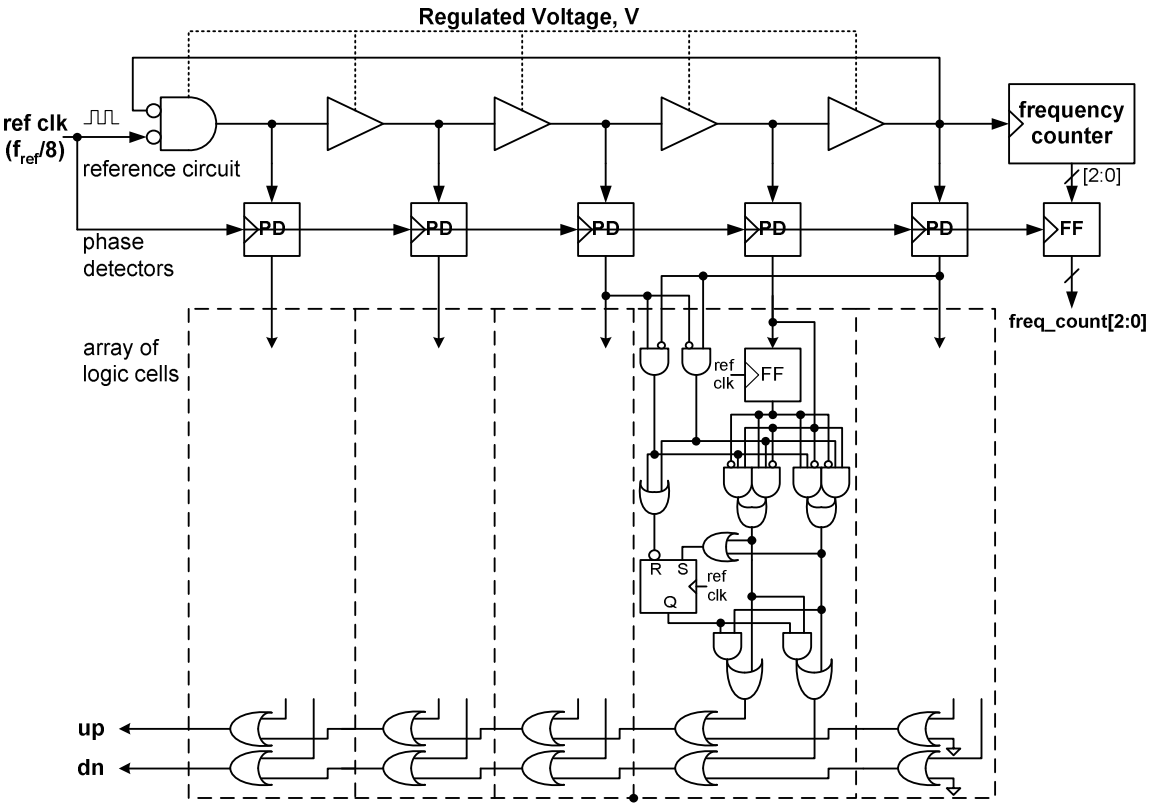


Figure 2.18: Sensor circuit implementation when the reference circuit is a ring oscillator

corresponding to the 80-th inverter (**lock**). The loop is considered locked when f dithers across the f_{ref}-boundary.

2.4.2 Sensor for a Reference Ring Oscillator

If the reference circuit is a ring oscillator, the sensor can be implemented as in Figure 2.18. In fact, in adaptive supply links, the reference circuit is a replica of the local PLL's VCO since this adaptive power-supply regulator also serves as the global frequency regulator for the local PLLs. The sensor for a reference ring oscillator is basically equivalent to a long delay line folded into a small ring and operates in a similar way with the sensor for a delay line described in the last subsection.

The ring oscillator in this example has five stages and operates off of the voltage V. Its frequency thus indicates the maximum frequency f of the system at voltage V and the

controller's goal is to regulate f to f_{ref}. For easy measurement of f, one of the stages is conditioned so that the oscillator starts in a known state when the reference clock falls. If the oscillator just returns to this state after one reference clock period, f and f_{ref} are matched.

Again for higher resolution in measuring f, the oscillator is run for 4 reference cycles and checked if it has made 4 complete oscillations. The oscillator starts running when the divide-by-8 reference clock falls and stops when it rises. A 3-bit counter with saturation counts the number of complete oscillations made during the running period. The polarity of the error $f - f_{ref}$ is then determined by whether the count has reached 4 (**error**). If the count is greater than 6, the frequency f is considered too high (**high**) and if less than 2, f is too low (**low**). The measurement range is therefore from $f_{ref}/2$ to $3/2 \cdot f_{ref}$. Also when the divided reference clock rises, the five phase detectors in Figure 2.18 sample the oscillator's final state, providing the finer resolution of $f_{ref}/40$ ($= |\Delta f|$).

Similar to the sensor based on a delay line, the array of logic cells detects the change in f. But the difference with the previous case is that the clock edge being detected can be either rising or falling, because the oscillator inverts itself every half cycle. When a cell detects a change, it first determines the polarity of the edge by examining the adjacent phase detector outputs. For example, if the preceding phase detector output is 1 and the next one's output is 0, the cell in between has detected a rising edge. Other than this minor difference, the array of logic cells work similarly with the delay line case and detects whether the frequency f has changed upward (**up**) or downward (**dn**).

2.4.3 Discontinuous Mode Operation

As the efficiency of the adaptive power-supply regulator is greatly improved by the controller power scaling, the efficiency is now limited by the loss of the buck converter. When the load current is low, actively switching the buck converter high and low will cycle the current back and forth through the inductor. This circulating current degrades the power efficiency significantly as most of the power is dissipated through the series resistance of the power transistors rather than being delivered to the load. The discontinuous mode operation [39, 40] avoids this circulating current by driving the buck converter high for a while

and letting it float until the voltage drops below the reference, instead of actively driving it low.

The digital sliding controller can support this discontinuous mode operation with a little modification. Once the controller reaches lock, the frequency f dithers around the reference f_{ref} and the control decision is based solely on the error polarity and the guaranteed minimum pull-down time $T_{u=0}$. If the pull-down transistor of the buck converter is disabled, the buck converter will be floating while f is above f_{ref} and start driving high again when f drops below f_{ref}. Thus, the buck converter can operate in the discontinuous mode simply by disabling the pull-down transistor and the efficiency at low loads can be greatly improved.[6] However, while the loop is in transient, the sliding control law requires that the buck converter switch in the normal continuous mode. The pull-down transistor is disabled only when the sensor indicates that the loop is in lock.

2.5 Measurement Results

This section discusses the measurement results from the prototype chip that was fabricated in National Semiconductor 0.25-μm CMOS process [30],[31]. The prototype chip, shown in Figure 2.19, uses a delay line as the reference circuit and the chip's overall characteristics are summarized in Table 2.1. The power transistors are integrated on the chip, and the inductor and the capacitor of the buck converter are external. Although the results presented here pertain to the delay line case, the controller with a ring oscillator as the reference circuit is expected to have similar performance, since the sensors have the same interfaces and perform the same function. In fact, the controller with a ring oscillator was fabricated on the adaptive supply link test chip, which required 40% less area and 30% less power than the delay line case.[7]

Figure 2.20 plots the power dissipation of the digital sliding controller and the power

[6] The decision on whether to operate the buck converter in continuous mode or in discontinuous mode is made by a manual input. However, automatic detection of the circulating current is possible [29].

[7] Actually, the measured power was 75% higher due to the change in the reference frequency. The controller with a delay line operates at 20-FO4 clock while the controller with a ring oscillator operates at 8-FO4 clock. If normalized for this frequency difference, the controller with a ring oscillator would require 30% lower power.

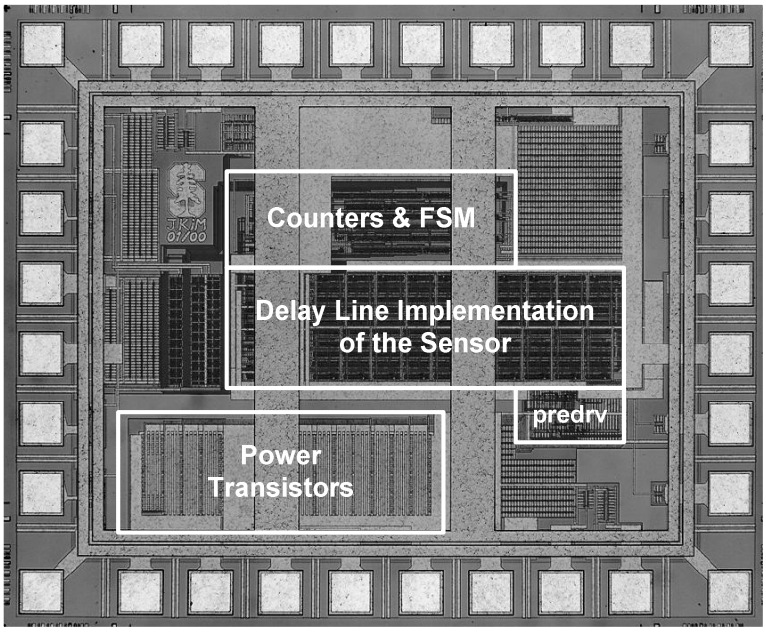


Figure 2.19: Digital sliding controller prototype chip micrograph

Table 2.1: Digital sliding controller prototype chip characteristics

Nominal supply voltage(Vdd)	2.5V
Regulated supply voltage range	1.1-2.3V
Total die area	$1.3 \times 1.1\mathrm{mm}^2$
Area occupied by the controller	$0.7 \times 0.5\mathrm{mm}^2$
On-chip power transistors	4.4mm (P), 2.2mm (N)
Off-chip components	15.2-μH (L), 21.6-μF (C)
Power dissipation of the controller	5.7mW @2.3V, 0.4mW @1.1V.
Switching frequency	460-860 kHz
Voltage ripple	$<$ 15mV
Settling time (99%)	$<$ 80-μs
Power efficiency	89-95%

efficiency of the adaptive power-supply regulator, measured at various reference frequencies (f_{ref}) and the equivalent voltages (V_{ref}). Since the controller operates at a variable frequency and voltage, its power scales as $V_{ref}^2 \cdot f_{ref}$, as shown in Figure 2.20(a). With this scaling, the controller power overhead stays as a constant fraction of the total power and the power efficiency is only limited by the buck converter loss itself. For testing purposes, we modulated the on-chip resistive load to emulate the load power scaling of a typical digital system, $V^2 f$.

When the load current level is high, actively switching the buck converter high and low by operating in the continuous mode is more efficient, as shown in Figure 2.20(b). The measured power efficiency ranges from 89 to 95% as the regulated voltage varies from 1.1 to 2.3V. The lower efficiency at 1.1V is mainly due to series loss of the power transistors. However, when the load current is low, continuous mode operation circulates a large amount of current back and forth across the inductor, while delivering small average current to the load. Thus, the series loss of the power transistors becomes dominant and the power efficiency degrades significantly, as shown in Figure 2.20(c), where the buck converter is driving no loads except the controller.[8] As discussed in Section 2.4.3, the circulating currents can be avoided by operating in the discontinuous mode. Figure 2.20(d) shows the improvements of the power efficiencies when the buck converter operates in the discontinuous mode. The load current level is the same as in Figure 2.20(c).

In Figure 2.20(d), the data point for 1.1V is missing because the controller could not properly turn the buck converter into the discontinuous mode operation. Below 1.1V, the low sampling rate ($f_{ref}/8 < 16\mathrm{MHz}$) causes a long loop delay which lets the frequency f drift to the next quantization boundary before being detected. Since the frequency f does not only dither across the reference point but also reaches the adjacent boundaries, the sensor does not assert the lock indication signal and therefore the pull-down transistor is never disabled. This problem can be mitigated either by lowering the resonant frequency of the buck converter or by increasing the sampling rate, e.g. from $f_{ref}/8$ to $f_{ref}/4$.

As mentioned in Section 2.2, the switching frequency of the sliding controller is not fixed externally, but varies as a function of V_{ref} (or equivalently f_{ref}) and the hysteresis

[8] In Figure 2.20(c) and (d), the power efficiencies are calculated as if the controller overhead is a part of the load power. This is to isolate the efficiency degradation due to the buck converter loss, since otherwise the power efficiency would be 0 (no load current).

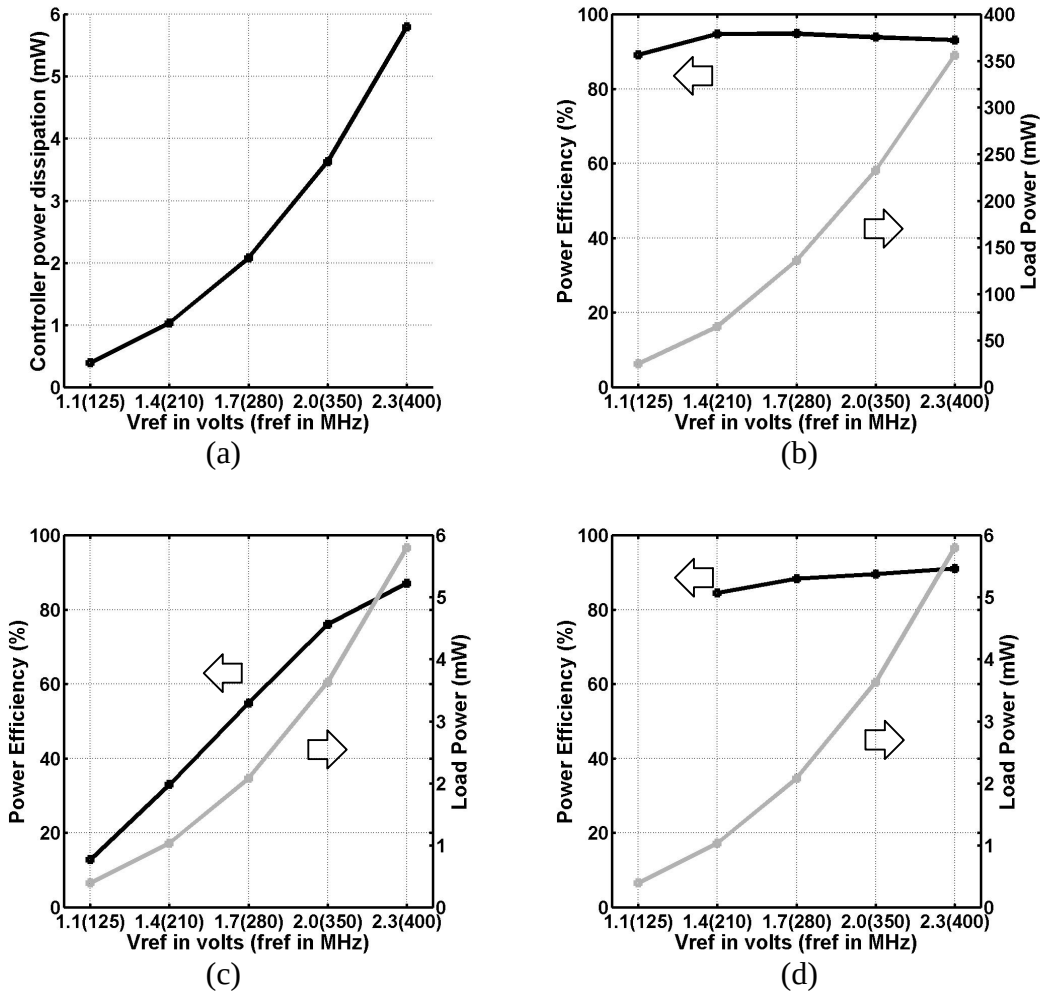


Figure 2.20: Power efficiencies of the digital sliding controller: (a) power dissipation of the controller, (b) power efficiencies when operating in continuous mode with high load currents, (c) in continuous mode with low load currents, (d) in discontinuous mode with low load currents

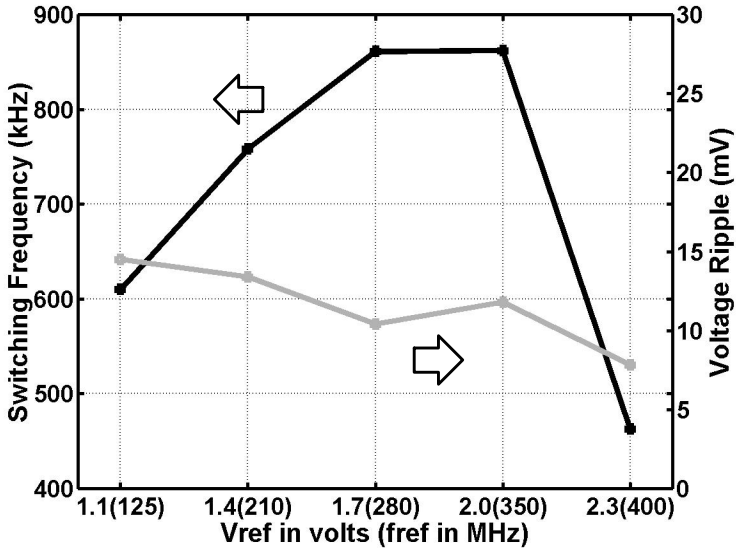
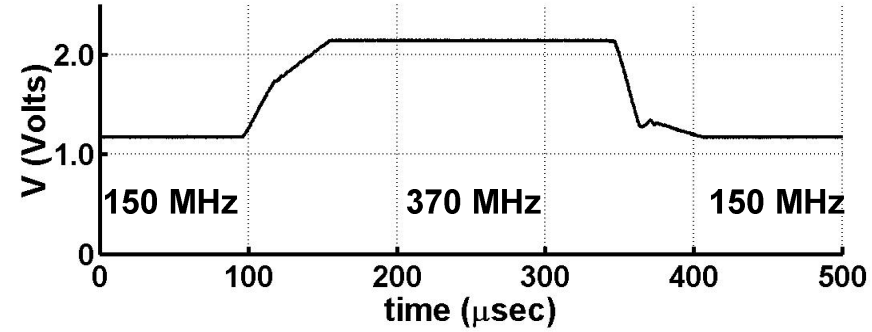


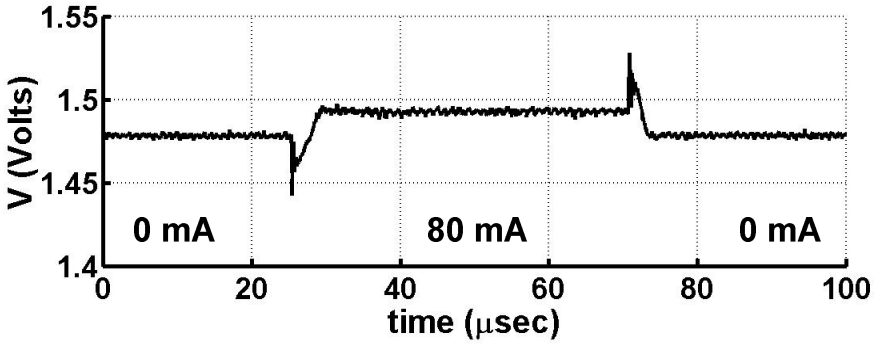
Figure 2.21: Switching frequency and voltage ripple

Δ (or equivalently N_Δ). Eq(2.1) predicts that the switching frequency should increase from 0 to $Vdd/2$ and then decrease above $Vdd/2$. The measurements verify that this is generally the case, as shown in Figure 2.21. However, the lower switching frequencies and the larger voltage ripples than predicted are observed at low voltages, where again the low sampling rates caused longer loop delays and effectively increased the hysteresis. According to Eq(2.2), the voltage ripple magnitude should vary as the inverse function of the switching frequency, but strong correlation has not been well observed because of noise from the off-chip environment. Even with these noise sources, the worst-case peak-to-peak voltage ripple was still less than 15-mV at all operating conditions.

The transient response of the digital sliding controller is shown in Figure 2.22(a) when there is a step change in the reference frequency. The plot shows that there are two transient phases. First, when the frequency is out of the measurement range of the sensor, the voltage is just driven hard toward the locking point. Once the voltage comes within the measurement range, the sliding control starts switching the buck converter and settles the voltage to the final value. The worst-case 99% settling time is less than 80-μs, which is faster than most switching regulators using linear control with comparable resonant frequencies of the buck converter.



(a)



(b)

Figure 2.22: Transient responses of the digital sliding controller. (a) For a step change in the reference frequency f_{ref}. (b) For a step change in the load current from 0 to 80mA

Figure 2.22(b) shows the load transient response. A step change in load current of 80-mA was made. The disturbed voltage returned to its steady-state value within 10-μs. A slight displacement of the dc voltage level was observed when the load current was varied, because the voltage was measured off the chip and the regulator tries to compensate the voltage drop between the on-chip and off-chip nodes.

2.6 Summary

Adaptive power-supply regulation minimizes the power dissipation by regulating the voltage to the minimum that can satisfy the desired performance. Dynamic regulation of the

supply voltage requires a feedback control loop that consists of a reference circuit to model the system performance versus voltage, a buck converter for efficient DC voltage generation, and a controller that stabilizes the loop.

This chapter presented a digital controller that uses sliding control for adaptive power-supply regulation. Sliding control needs the derivative of the controlled variable f and the digital controller's coarse resolution in measuring f makes it difficult to implement the original sliding control law directly. So we derived an alternative form of sliding control, which makes the digital implementation feasible. According to this new sliding control law, the sensor detects the frequency f of the reference circuit and its change Δf. The implementation of the sensor was discussed for both types of reference circuits: a delay line and a ring oscillator. The test chip demonstrated that the digital sliding controller can operate at variable frequency f and voltage V, and thus achieve high efficiencies of 89-95% over a wide operating range.

Chapter 3

Low-Voltage Parallelized Transmitter and Receiver

The previous chapter discussed adaptive power-supply regulation as a way of minimizing the power dissipation while operating at a specified clock frequency. This chapter discusses the transmitter and receiver circuits that operate off of this adaptive supply. To achieve low energy-per-bitrate ratios, the link transceivers must have high throughputs while operating at low supplies. This chapter presents parallel and low-voltage transceiver circuits to satisfy this goal.

Parallelism in transceivers obviates the need for high-frequency clocks to achieve high bitrates [2],[46]. For example, the multiplexing transmitter and demultiplexing receiver shown in Figure 3.1 can support high bitrates while each sub-transmitter and sub-receiver operate at a lower frequency, $bitrate/M$. Multiple transmitters connected in parallel convert low-frequency parallel data streams into a single high-frequency stream on the channel. Multiple phases of the lower-frequency clock control the on- and off-timing of each transmitter. Similarly, the parallel receivers convert the high-frequency data stream back to the low-frequency parallel data streams. Again, multiphase clocks evenly divide a clock period and set the receiving window of each receiver.

Parallelism lowers the clock frequency needed to support a certain bitrate and thus provides opportunities for saving power via adaptive power-supply regulation. As discussed in Chapter 1, lowering both the frequency and voltage can bring significant savings in

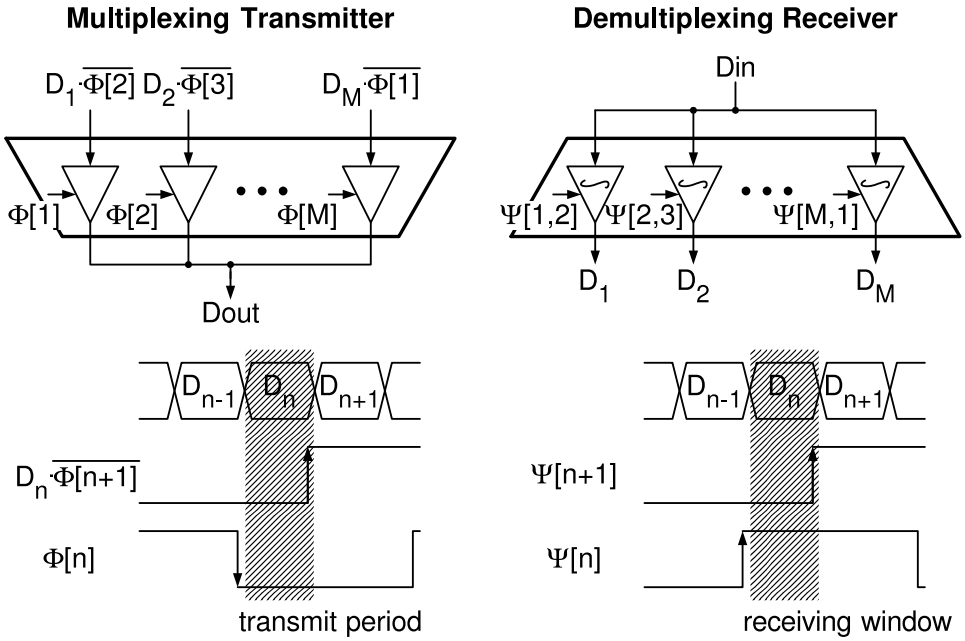


Figure 3.1: Parallelism in high-speed links: time-division multiplexing

power despite the penalty of increased area and switched capacitance [33]. The power reduction via parallelism is mainly due to the voltage being adaptively scaled down for the lower clock frequency. Without voltage scaling, parallelism does not lead to power savings because the reduction in power due to the lower frequency is canceled by the increased switched capacitance.

Therefore, to fully exploit the potential power saving of parallelism, the link circuits must be able to operate at low supply voltages. Unfortunately, most conventional transceiver circuits face difficulties as the supply voltage drops near the threshold voltage, Vth. The lower-bound on the supply voltage posed by these circuits limits the power savings achievable by adaptive power-supply regulation.

This chapter begins by reviewing previous circuit implementations of parallelized transmitters and receivers and identifying their limits on the supply voltage. Ways to extend these limits are then investigated. In most cases, the difficulty stems from the reduced voltage headroom as the supply V becomes comparable to Vth. Various techniques either to cancel Vth or to avoid Vth-related issues are presented. Although these techniques enable a supply voltage as low as $1.6 \cdot Vth$, they come with overheads in power and area. The

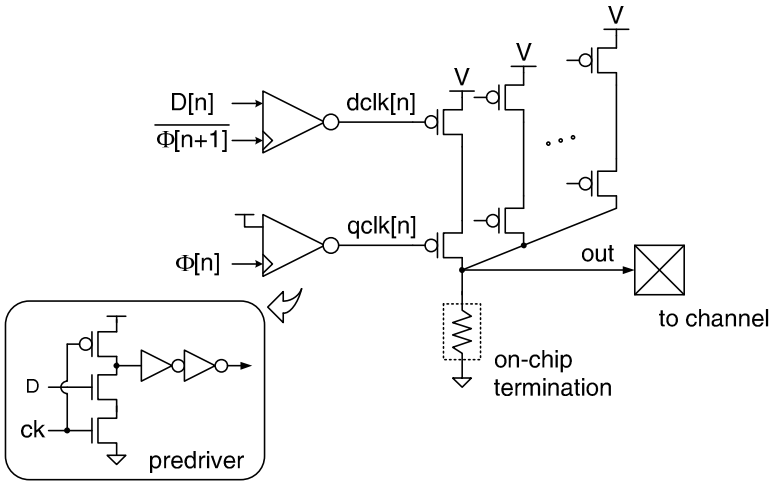


Figure 3.2: Output-multiplexing transmitter by Yang et al. [2]

overall effectiveness of these techniques is then analyzed.

Another challenge associated with the higher degree of parallelism is the stringent matching requirement among the parallel components. These parallel components must be identical and any mismatch may result in degradation of signal quality. For example, time spacings between the multiphase clocks must be precise to reduce uncertainty in signal timing. Also, the parallel transmitters must produce the same voltage levels to reduce uncertainty in signal swing. The issues related with precise generation of multiphase clocks and the impacts of transistor mismatch on low-voltage, parallel operation will be discussed in Chapter 4 and 5, respectively.

3.1 Low-Voltage Parallelized Transmitter

3.1.1 Multiplexing Transmitter

Shown in Figure 3.2 is the multiplexing transmitter proposed in [2]. Since it is difficult to propagate pulses that are shorter than two gate delays at on-chip high-impedance nodes, the transmitter multiplexes the data at its last stage driving the low-impedance off-chip node. The transmitter has multiple drivers connected in parallel, each of which consists of two pMOS transistors in series. Thus, each individual driver is active only when its inputs,

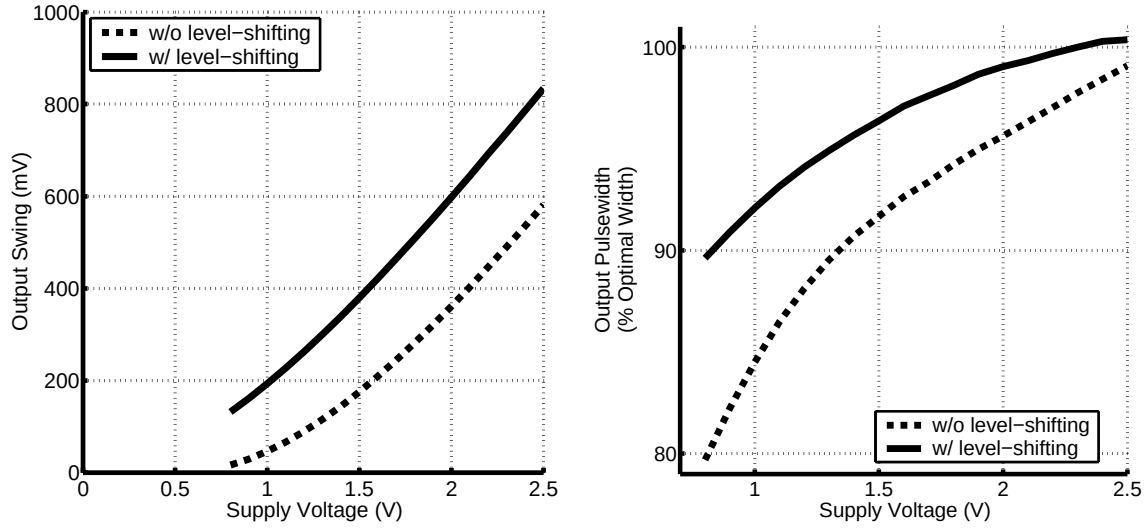


Figure 3.3: (a) Simulated output swing versus supply voltage, (b) simulated output pulse-width versus supply voltage

$dclk[n]$ and $qclk[n]$, are both low. Since these inputs are aligned with the clock phases that are one phase apart, $\Phi[n]$ and $\Phi[n+1]$, the active period of each driver becomes $1/M$ of the clock period, where M is the number of phases and the multiplexing rate. The combination of the clock phases connected to the drivers makes M active periods that cover the clock cycle. The predriver on the top enables or disables the input $dclk[n]$ depending on the data bit being transmitted. The purpose of the predriver driving $qclk[n]$ is to match the delays between the two input paths.

The transmitter in Figure 3.2 uses pMOS drivers instead of the nMOS drivers used in the original implementation [2]. It is because the supply V is adaptively adjusted to each chip's own conditions, and therefore this voltage cannot be the common reference across different chips. The signal is instead referenced to ground driven by the pMOS drivers. The use of pMOS drivers in fact has a substantial cost, and we will explore other alternatives in Section 3.1.4.

The factors that limit the minimum supply voltage of this multiplexing transmitter are related to the threshold voltage of the driver. First, the output signal swing drops rapidly as the supply voltage V approaches the threshold voltage Vth. Since the drain current of a

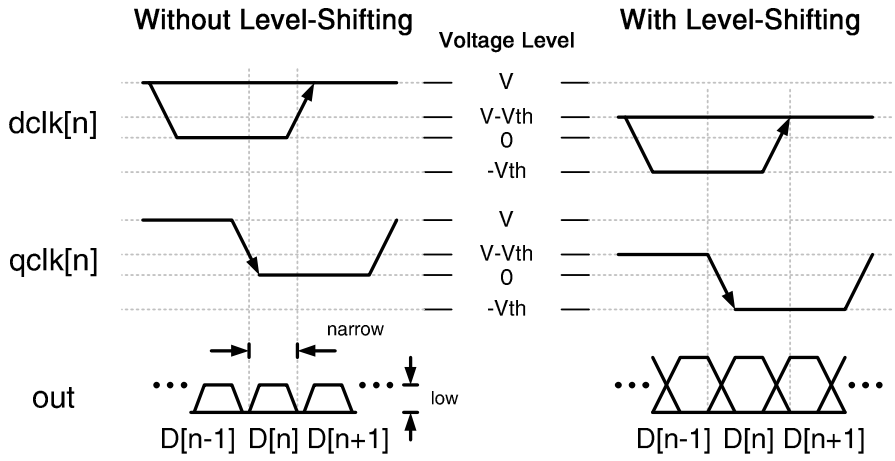


Figure 3.4: Voltage waveforms of the driver inputs and outputs: with and without level-shifting

transistor in saturation scales as $(V-Vth)^{\alpha}$ [47], the output swing will vary with the supply voltage as plotted in Figure 3.3(a). If the dominant noise on links is the on-chip switching noise whose magnitude is proportional to the supply V, it is desirable that the output swing also scale as V. However, with the current scaling as $(V - Vth)^{\alpha}$, an increased driver size would be necessary to support the desired swing at low supply V, which may reduce the power saving benefits of low-voltage operation.

Second, a lower supply voltage results in a narrower pulse-width at the output [32]. Figure 3.4 illustrates the waveforms of the driver inputs, $dclk[n]$ and $qclk[n]$, and the variation of the output pulse-width with the supply voltage. The driver is conductive only when both inputs are below $V - Vth$. Therefore, as the supply V decreases and Vth becomes a larger portion of the swing, the driver will turn on later but turn off sooner. As a result, the output pulse-width decreases by 20% as the supply voltage drops from 2.5V to 0.8V, as plotted in Figure 3.3(b).

An alternative implementation of multiplexing transmitter is proposed by Lee et al. [48]. In this transmitter, the data streams are multiplexed at the input of the driver instead of at the output of the driver. To sustain fast signal transitions at an on-chip high-impedance node, this input-multiplexing transmitter traded off the gain of the multiplexing stage to increase its bandwidth. Input-multiplexing is effective in reducing power dissipation in

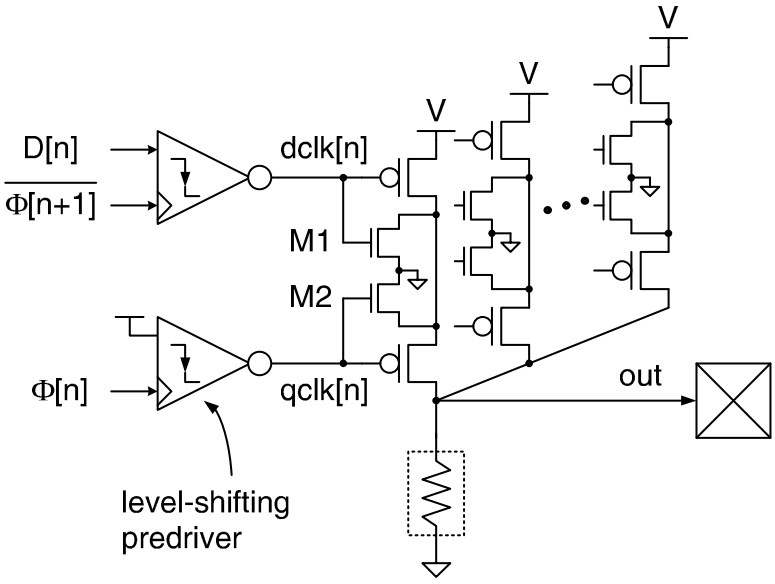


Figure 3.5: Output-multiplexing transmitter with level-shifting.

the high voltage range because the switched capacitive load is reduced compared to the output-multiplexing case. However, the input-multiplexing transmitter does not lend itself well to low-voltage operation since an initially high voltage swing is needed to enhance the bandwidth.

The next subsection describes a way of extending the low supply limit of the output-multiplexing transmitter, by canceling the threshold voltage of the pMOS driver and making it effectively zero.

3.1.2 Multiplexing Transmitter with Level-Shifting Predriver

Figure 3.5 shows the output-multiplexing transmitter with level-shifting predrivers. The predrivers shift the voltage levels of the driver inputs down by Vth so that the pMOS driver inputs swing between $-Vth$ and $V - Vth$. With level-shifting, the pMOS driver behaves as if its threshold voltage Vth is zero and the problems related to Vth are mitigated. For example, the output swing scales as V^{α} instead of $(V - Vth)^{\alpha}$ and thus the driver can produce a reasonable output swing at low supplies. Also, the output pulse-width is less sensitive to supply voltage variation because the turn-on and turn-off points of the driver

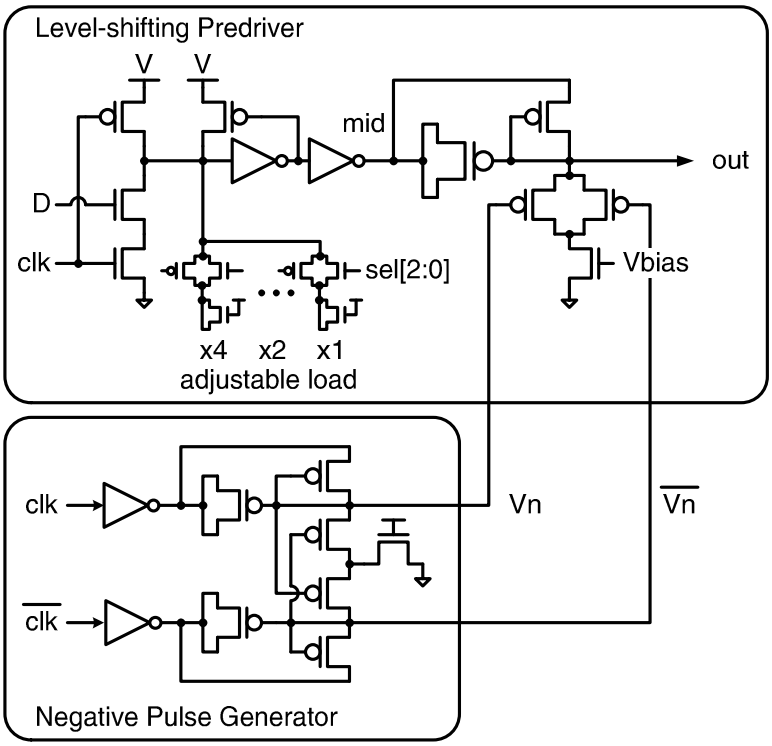


Figure 3.6: Level-shifting predriver and negative pulse generator.

are now independent of V. The plots in Figure 3.3 illustrate the effectiveness of level-shifting.

Level-shifting raises the gate voltage of the pMOS driver up to only $V - Vth$ and therefore the pMOS driver may leak current to the output even when it is supposed to be inactive. The transistors M1 and M2 in Figure 3.5 are added to shunt this leakage current to ground.

Figure 3.6 is the circuit detail of the level-shifting predriver. The level-shifting circuit consists of a boost capacitor and a biasing circuit. When the voltage on node mid is high at V, a diode-clamp and a small current source bias the voltage level on node out at $V - Vth$, storing Vth across the boost capacitor. When the signal on mid switches down to ground, the diode-clamp becomes inactive and the boost capacitor pushes the output voltage down to $-Vth$. To avoid voltage swing attenuation due to charge sharing, a large enough boost capacitor is needed. For example, an 80μm-wide and 0.64μm-long pMOS gate capacitor was used to drive each 16μm-wide pMOS driver.

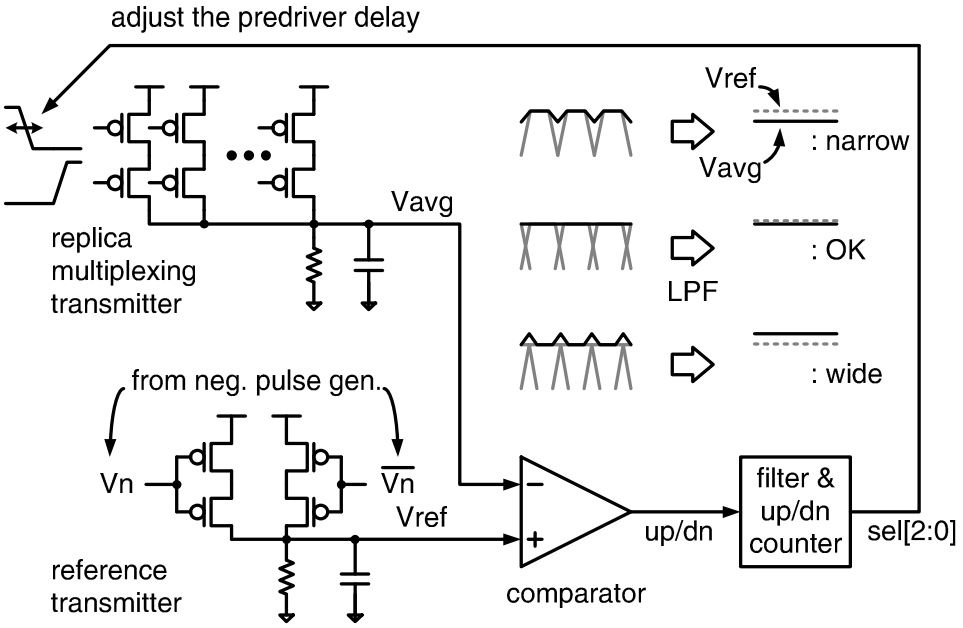
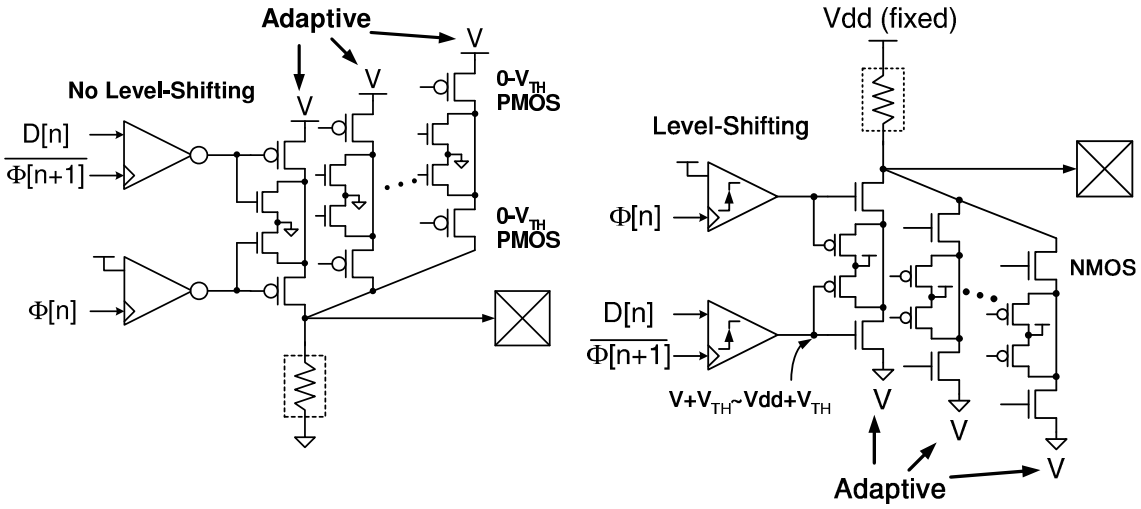


Figure 3.7: Transmitter output pulse-width adjustment loop.

The current source in the biasing circuit keeps the diode-connected pMOS slightly on and thus maintains the Vth-drop between nodes mid and out. However, the nMOS current source cannot connect directly to the node out since its voltage swings below ground and can forward-bias the p-n junction of the nMOS transistor. Two parallel pMOS transistors are inserted between the nMOS device and the node out, and the negative pulse generator in Figure 3.6 alternatively switches the gate voltages of these pMOS devices down to $-Vth$ to keep this path always conductive. The negative pulse generator operates in a similar manner to the level-shifting predriver and the voltage doubler in [49].

3.1.3 Output Pulse-Width Adjustment

In an output-multiplexing transmitter, in order to maximize the timing margin, it is important to control the pulse-width so that the falling edge of one pulse and the rising edge of the subsequent pulse coincide. Although the level-shifting predriver reduces the pulse-width variation, an additional control is required to ensure the optimal pulse-width. The feedback loop shown in Figure 3.7 adjusts the pulse-width by varying the predriver delays



(a) (b)

Figure 3.8: Alternative transmitter circuits: (a) zero-Vth pMOS driver, (b) nMOS driver with negative adaptive supply

via the digitally-adjustable loads in Figure 3.6. The feedback loop compares the reference voltage to the average output value of a replica multiplexing transmitter that transmits all 1's. If the pulses are too wide, the average is higher than the reference, and vice versa [50]. Although the reference voltage should be the output of a single driver that is constantly on, the input voltage cannot be held constantly at $-Vth$. Two parallel drivers with alternatively switching inputs are used to generate the reference instead.

3.1.4 Power and Area Overheads of the Level-shifting Transmitter

The level-shifting predriver stretches the minimum supply of the multiplexing transmitter down to 0.8V with Vth=0.55V, thus $1.4 \cdot Vth$. However, the circuits that enable low-voltage operation have overheads in terms of both power and area, which is disadvantageous for high-voltage operation. This subsection examines the overheads of the proposed multiplexing transmitter.

First, the level-shifting predriver dissipates additional power because it must switch the large boost capacitor. Level-shifting is done by storing Vth across the capacitor and driving

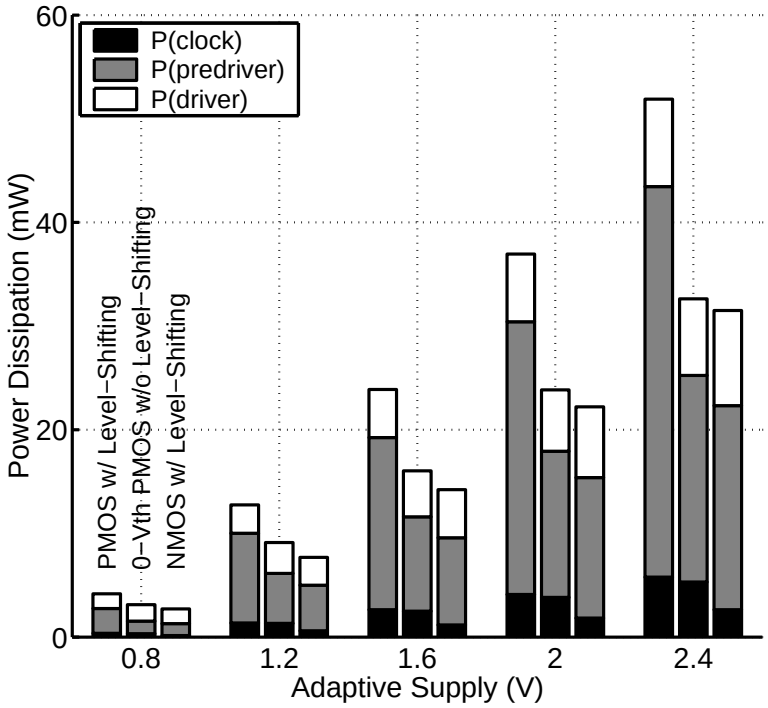


Figure 3.9: Power dissipation at various supply voltages. Power is normalized to the voltage swing of 200mV.

one of its terminals. To maintain the transfer gain close to unity, the capacitor must be large compared to the capacitance on the other terminal to the supply rails. Therefore, the boost capacitor occupies a large silicon area and has a large parasitic bottom-plate capacitance (nwell-to-substrate capacitance) that dissipates a fair amount of power when it is switched.

Second, a pMOS driver is less efficient in driving currents than an nMOS driver and therefore with the same signal swing, the pMOS driver occupies more area and presents more capacitance to the data and clock inputs. pMOS drivers are used in order to transmit signals referenced to ground because the adaptive Vdd cannot serve as a common reference across different chips.

To estimate the power overhead, the proposed transmitter is compared with a few alternatives that are shown in Figure 3.8. Figure 3.8(a) is a transmitter with zero-Vth pMOS drivers that excludes the overhead of level-shifting. Figure 3.8(b) is a level-shifting transmitter that uses more efficient nMOS drivers. In this case, the higher supply Vdd serves as

a common fixed reference and the lower supply Vss is adaptively adjusted.

Figure 3.9 compares the power dissipations of these transmitters at various supply voltages. Each power is normalized to a fixed output voltage swing of 200mV and the total power is broken down by the contributions of the drivers, the predrivers, and the last-stage clock buffers. Comparison with the zero-Vth driver suggests that 47% of the predriver power and 34% of the total transmitter power are consumed by level-shifting in the pMOS-based transmitter. Comparison with the nMOS driver suggests that the power penalty of using pMOS drivers is 47% of the predriver power and 54% of the clock power. Based on these results, a transmitter with a zero-Vth nMOS driver would dissipate only 22% of the predriver power and and 54% of the clock power of the pMOS level-shifting transmitter. The total power would be only 38%. A significant amount of power is being dissipated to perform level-shifting and to use pMOS drivers.

3.2 Low-Voltage Parallelized Receiver

3.2.1 Demultiplexing Receiver

At the receiving end, the high-frequency bit stream gets demultiplexed into low-frequency parallel streams by a set of parallel receivers, as shown in Figure 3.10. Each receiver branch consists of a front-end that preconditions the incoming signal and a comparator that determines whether the received bit is a 0 or a 1.

The front-end of the receiver uses an integrate-and-dump filter proposed by Sidiropoulos et al. [3]. The integration of the received signal over the bit period rejects high-frequency noise and emulates the maximum-likelihood receiver for non-return-to-zero (NRZ) pulses [51]. The integration period is defined by two clock phases, $\Psi[n]$ and $\Psi[n+1]$, as illustrated in Figure 3.11. After the integration period, the front-end holds the resulting signal while the comparator resolves, and gets reset before the next integration period begins. The parallel receivers rotate their activity such that at any given time, only one of them is integrating and the others are either holding or being reset. This timing is achieved by using multiphase clocks.

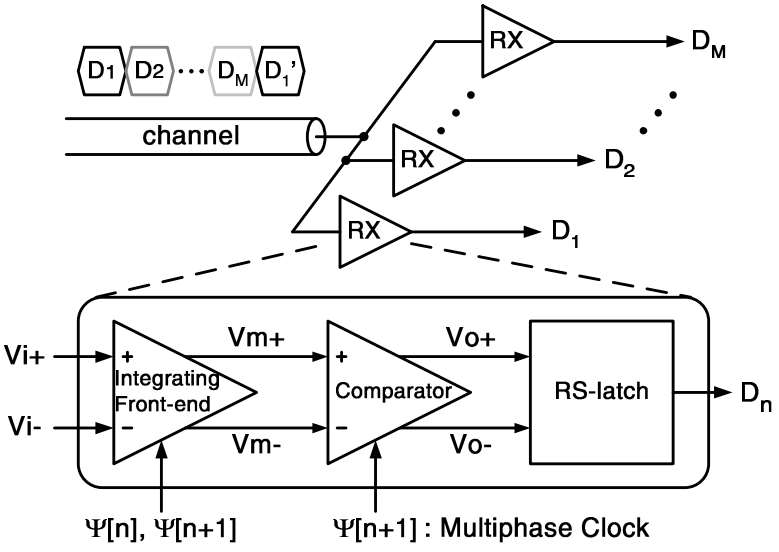


Figure 3.10: Demultiplexing receiver

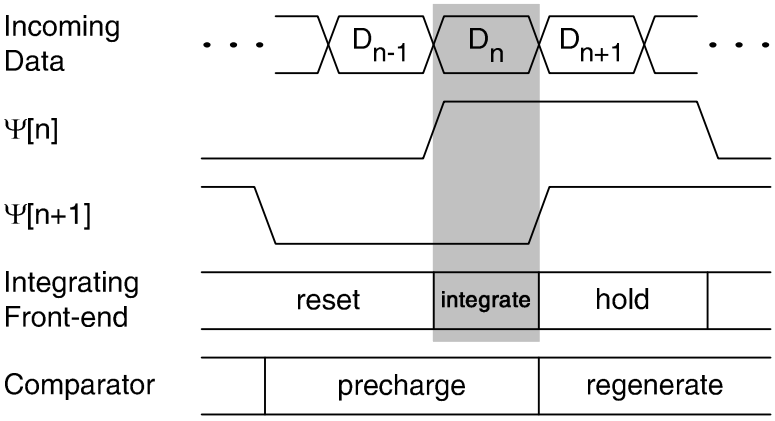


Figure 3.11: Integrating receiver timing diagram

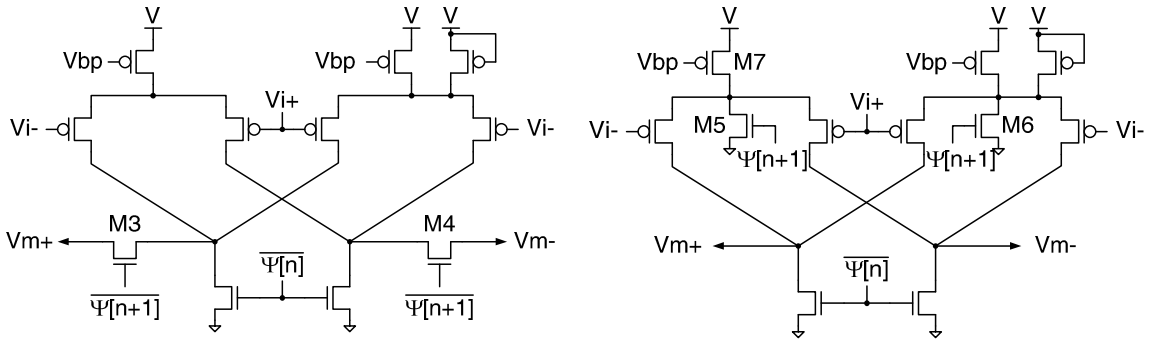


Figure 3.12: Current integrating receiver front-end: (a) previously published [3], (b) modified for low-voltage operation

The following subsections examine the current-integrating receiver previously implemented by Sidiropoulos et al. [3]. The factors that limit its minimum supply are the large voltage headroom required for the MOS sample-and-hold switches and the poor sensitivity of the comparator operating at low supplies and with low input levels. Techniques to extend these limits are then discussed.

3.2.2 Current-Integrating Receiver Front-End

Figure 3.12 is the circuit detail of the current-integrating receiver. Figure 3.12(a) shows the previous implementation in [3] with modifications that allow a multiplexing factor greater than 2. During reset, the output voltages, $Vm+$ and $Vm-$, are discharged to ground by the nMOS pull-down transistors. When $\Psi[n]$ rises, the integration period begins and the input differential pair steers the current onto the output nodes, where the integral of the current becomes the voltage. The complementary differential pair biased at a lower current level is to equalize the charge injection from the inputs [9]. When $\Psi[n+1]$ rises, the integration period ends and the nMOS switches M3 and M4 hold the output voltages for comparison.

The nMOS sample-and-hold switches, M3 and M4, set the limit of the minimum supply voltage. The integrating stage is biased so that the output voltages lie between 0 and Vth to keep the differential pairs in saturation. Therefore, for the switches M3 and M4 to be conductive, their gate voltages must be at least $2Vth$ plus some overdrive [52]. This

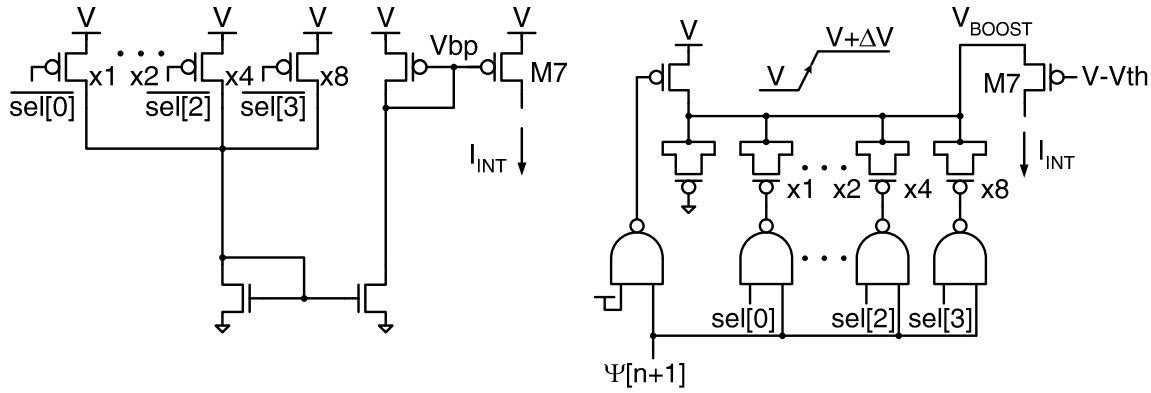


Figure 3.13: Biasing the integrating current via: (a) segmented current sources, (b) capacitive charge pumps

required gate voltage swing for the sample-and-hold switches limits the minimum supply of the integrating receiver.

Figure 3.12(b) is the proposed integrating front-end that performs sample-and-hold without these MOS switches. The nMOS transistors M5 and M6 hold the output voltages by discharging the tail nodes and thus disabling the current steering. This small modification extends the minimum supply down to $1 \cdot Vth$ plus gate overdrive. The tail nodes are charged up again during the reset period, so that the bias currents will be at their full levels when the integration begins.

Figure 3.13 shows two biasing schemes for the integrating front-end. The purpose of adjusting the bias is to keep the output voltages within the desired range (0 to Vth) and thus to keep the differential pairs saturated. Both biasing schemes provide digitally-programmable current levels, in order to obviate the need for distributing an analog, noise-sensitive bias voltage across the chip.

Figure 3.13(a) is a more conventional approach where the gate bias Vbp of the integrating stage current source M7 is adjusted via the segmented current sources and the current mirror. For the integrated output to have a constant voltage swing, the bias current must scale inverse-proportionally with the bit period and proportionally with the clock frequency. The source terminals of the segmented current sources are tied to the adaptive supply V,

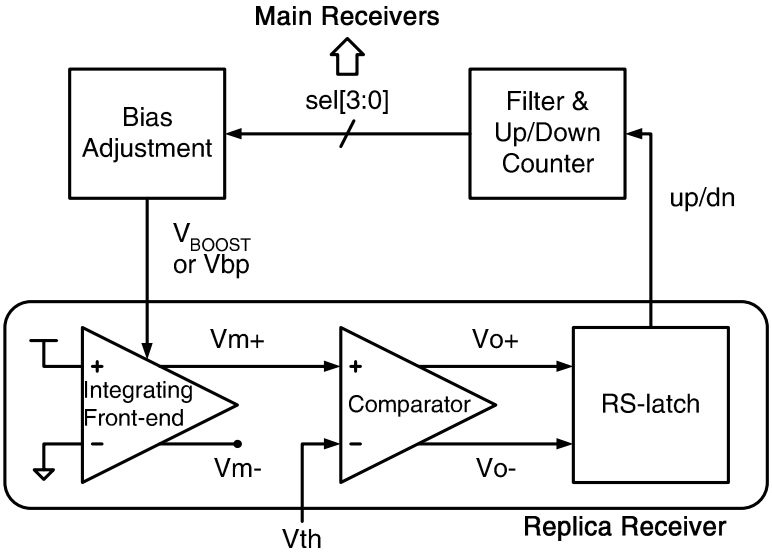


Figure 3.14: The feedback biasing loop for the integrating front-end

so their currents will scale with the adaptive supply and thus with the clock frequency as desired. Although the scaling via the adaptive supply is not perfect, it helps narrow the required adjustment range which must be wide enough to include the desired bias points at all operating conditions.

To stretch the voltage headroom further, the biasing scheme shown in Figure 3.13(b) instead boosts the source voltage of the current source while holding the gate voltage fixed at $V - Vth$. The source voltage V_{BOOST} is initially held at V while the integrating stage is off duty, i.e. $\Psi[n+1]$ is high. When $\Psi[n+1]$ falls, the selected NAND gates switch one side of the capacitors high and boosts the voltage V_{BOOST} from V to $V+\Delta V$. The raised amount ΔV is adjusted by selecting different combinations of NAND gates, therefore adjusting the bias current. Similar to the segmented current source, the bias current will scale with the adaptive supply and with the clock frequency, because ΔV is proportional to the NAND gate output swing, V.

This capacitive charge pump scheme achieves a minimum supply of 0.9V with Vth of 0.55V, whereas the segmented current source scheme achieves 1.1V. However, switching the capacitors every cycle increases the total receiver power by 8%. Unless the robust operation at very low supply is desired, the segmented current source scheme is the better

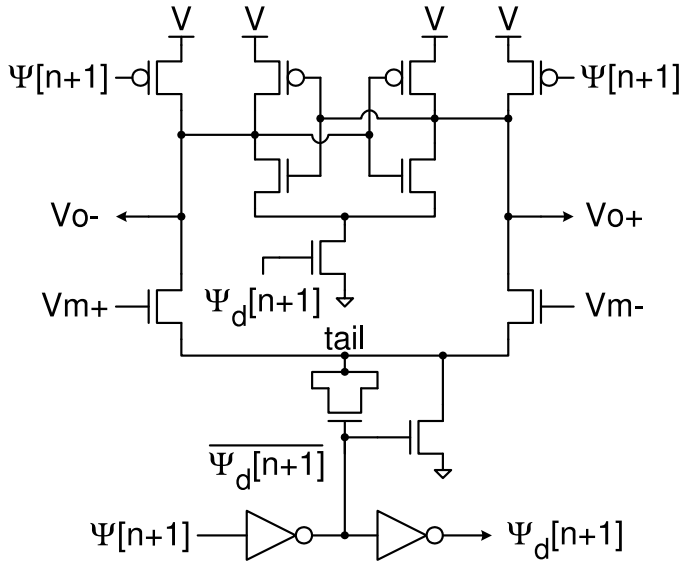


Figure 3.15: Low-voltage comparator with low common-mode input.

solution for low power.

Figure 3.14 shows the digital feedback biasing loop for the integrating front-end. The receiver is replicated for the feedback loop and its inputs are driven as strong 1. The comparator detects if the voltage swing of the front-end positive output is either above or below the desired level, Vth. Based on the comparator output, the up/down-counter adjusts the digital selection bits and varies the bias current level accordingly.

3.2.3 Low-Voltage Comparator

The role of the comparator is to recover full digital levels of the integrated output signals. Since these signals range from 0 to Vth at most, the comparator must be able to resolve signals with such low common-modes while operating at low voltages. Most comparators in the literature unfortunately fail this requirement. For example, the StrongArm latch [53] needs a minimum supply of at least $1.5Vth$ if the pMOS input pairs are used to accommodate the low input levels. The latches with nMOS input pairs cannot resolve signals below Vth because the nMOS input devices will not be conducting.

The proposed comparator in Figure 3.15 uses an nMOS input pair but with a charge-injector on its tail. When the clock $\Psi[n+1]$ is low, the output voltages $Vo+$ and $Vo-$

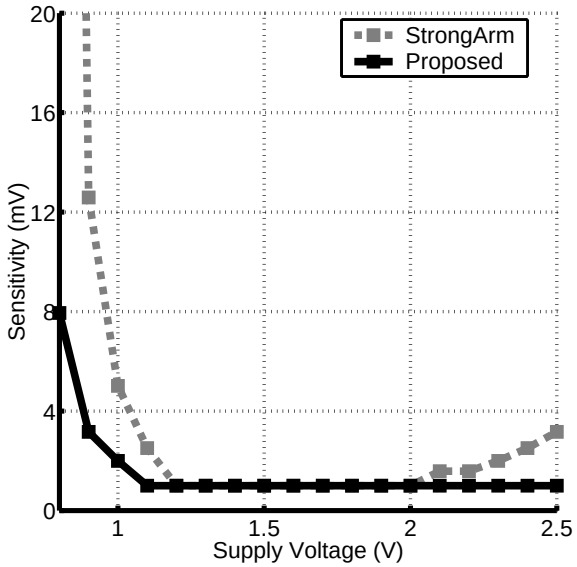


Figure 3.16: Simulated comparator sensitivity

are precharged high and the node $tail$ is discharged to ground. When the clock $\Psi[n+1]$ rises, the capacitor injects negative charge into the node $tail$. This charge is then steered by the input pair to selectively discharge the output voltages. After a small delay, the positive feedback formed by the cross-coupled inverters is activated and regenerates the initial voltage difference to a full level.

It is a concern whether this negative charge injection can cause a latch-up [54]. The p-n junction may be forward-biased momentarily, but it is not likely that significant minority carrier injection will occur since the voltage drop below ground is only about half of Vth or 300mV. Furthermore, the amount of charge injected is too small (50fC) to sustain a large current that can initiate a latch-up.

Figure 3.16 plots the sensitivity of the proposed comparator along with that of the StrongArm latch with a pMOS input pair. At 0.9V, the sensitivity of the proposed comparator is 3mV while that of the StrongArm latch degrades to 13mV. Also, the sensitivity of the StrongArm latch degrades again at high supplies because the low input voltage level causes the pMOS input pair to fall out of saturation. The proposed comparator shows good sensitivity over the wide operating range.

3.3 Summary

To further exploit the power saving benefits of adaptive power-supply regulation, this chapter presented the parallelized transmitter and receiver that can operate at very low supplies. Parallelism greatly improves the bitrate of a link while low-voltage operation saves a significant amount of power.

Several design techniques were presented to enable low-voltage operation of the transceiver. In the transmitter, the use of level-shifting mitigated the problems associated with the threshold voltage of the driver. In the receiver, the integrating front-end avoided using the sample-and-hold MOS switches that require large voltage headroom. The comparator used subthreshold charge-steering to accommodate low input common-modes and to operate at low supplies.

These low-voltage techniques extended the minimum supply limit of the transceiver down to $1.6Vth$, 0.9V with Vth of 0.55V. Although some of the techniques have overheads in power and area and thus have little benefit at high voltage range, they will gain more importance as CMOS processes scale and the supply voltage continues to decrease relative to the threshold voltage. Since the additional overhead is due mostly to the capacitive boosting used to generate new voltage levels, having more explicit supply voltages may prove more viable in such cases.

Chapter 4

Per-Pin Multiphase Clock Generation and Recovery

The previous chapter described a parallelized transmitter and receiver that can achieve high aggregate bitrates while operating at moderate clock frequencies. This parallelized transceiver operates in a time-division multiplexing way, where only one of the branches is active at any given time. The timing to select each transmitter and receiver branch in sequence is governed by multiple clocks with equal phase spacing. Non-uniform spacing between the clock phases can result in one of the bit periods being shorter than the others and effectively reduce the timing margins of the link. Therefore, for maximum timing margins, it is important to maintain low static phase offsets as well as low jitter. This chapter describes how to accurately generate multiphase clocks with low overhead in power and area.

Multiphase clocks can either be distributed from a central point or be generated at each pin locally. In both cases, precisely matching the delays of the multiple clock paths is important to maintain equal spacing between the clock phases. However, even with identical layout, the distribution paths are subject to random mismatches that can be either time-invariant (static offset) or time-varying (jitter). The longer the path delay, the larger the mismatch [55],[56],[57]. For this reason, distributing the centrally-generated multiphase clocks is usually impractical. Generating multiphase clocks locally at each pin can keep the distribution path short and thus the static offset and clock jitter low.

Local clock generation also makes per-pin timing adjustment easier. Recovering clocks from different sources and compensating skew between channels both require separate timing adjustment circuits for each pin [58]. If the multiple clock phases are distributed from a central source, the local timing circuit will need to shift all the phases equally by the desired amount to adjust timing. The complexity is high and matching the phase-shifts is difficult. A better approach is to directly generate the multiphase clocks that are synchronized to the data.

However, having phase-locked loops (PLLs) or delay-locked loops (DLLs) generate the multiphase clocks locally at each I/O pin can be costly in power and area. To address this problem, a dual-loop architecture for per-pin multiphase clock generation and recovery is presented. The global feedback loop in this dual-loop architecture is the adaptive power-supply regulator. The adaptive supply not only maximizes the energy efficiency, but also enables small silicon area and optimum scalable performance at all frequencies of operation.

The first section describes the dual-loop architecture for local multiphase clock generation and recovery. It discusses the benefits of adaptive supply and reviews the general principles of adaptive bandwidth PLL/DLL. The following sections then describe the circuit details of the per-pin clock generation and recovery loops. Both the PLL and DLL implementations are presented and their power, area, and jitter performance are compared.

4.1 High-Level Architecture

4.1.1 Dual-Loop Architecture

Figure 4.1 illustrates a dual-loop architecture for multiphase clock generation and recovery. The adaptive power-supply regulator presented in Chapter 2 serves as the global loop and the PLLs or DLLs that produce multiphase clocks for the multiplexing transceiver serve as the local loops. The global loop and the local loops are linked through the adaptive supply, which is dynamically adjusted for the operating frequency as well as the process and temperature conditions.

The role of the adaptive power-supply regulator in this dual-loop architecture is twofold.

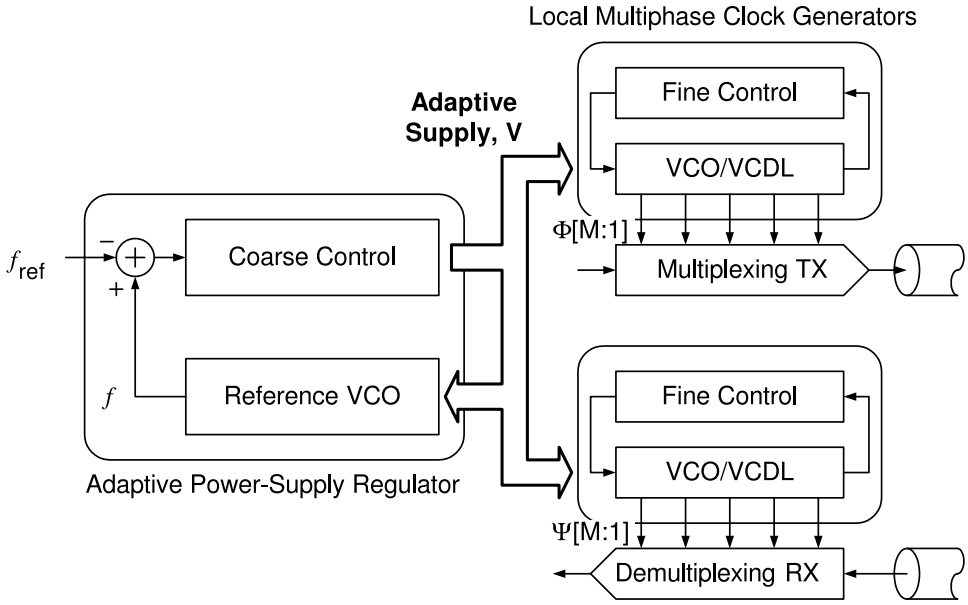


Figure 4.1: Dual-loop architecture for multiphase clock generation and recovery

First, the adaptive power-supply regulator maximizes the energy efficiency of the local PLLs and DLLs by lowering the supply voltage to the minimum that supports the operating frequency [30],[31]. Second, by doing so, the adaptive power-supply regulator effectively regulates the frequencies of the local voltage-controlled oscillators (VCOs) and the delays of the local voltage-controlled delay-lines (VCDLs) to be close to f_{ref} and $1/f_{ref}$, respectively. The adaptive power-supply regulator adjusts the supply voltage so that its reference VCO oscillates at the reference frequency f_{ref}. Then the local VCOs and VCDLs are expected to have frequencies close to f_{ref} since they are identical to the reference VCO and operate off of the same voltage level as the adaptive supply [59].

Exploiting the fact that the adaptive supply coarse-tunes the frequencies and delays of the local VCOs and VCDLs, the fine-tuning range of the local PLLs and DLLs can be narrow as long as it can compensate for the offsets due to on-chip mismatches. The worst-case mismatch analysis shows that $\pm$15% range would be wide enough for the fine frequency tuning.

This narrow tuning range leads to lower VCO/VCDL gain in the local PLLs and DLLs and thus to smaller loop capacitor area. The desired bandwidth of the PLL sets a certain value for $K_{VCO} \cdot I_{cp}/C$, where K_{VCO} is the VCO gain, I_{cp} is the charge pump current,

and C is the filter capacitance. Wide-range PLLs typically have large K_{VCO}'s and demand the filter capacitance of over 100pF, whose area can easily dominate the overall PLL area. Thanks to the narrow fine-tuning range and the low K_{VCO}, the local PLLs in this dual-loop architecture have the filter capacitance of only 2.5pF, saving significant area.

The dual-loop architecture enables both low VCO gain and wide frequency range, which are difficult to achieve in conventional PLLs [60]. Here the global loop is the existing adaptive power-supply regulator, incurring no additional overhead. The adaptive supply can also be used to scale the PLL and DLL bandwidth optimally with the operating frequency, which is described next.

4.1.2 Adaptive-Bandwidth PLL/DLLs

Adaptive-bandwidth PLLs and DLLs refer to a class of PLLs and DLLs in which the loop dynamics scale proportionally with the operating frequency. For example, the self-biased technique proposed by Maneatis in [61] achieves a fixed bandwidth to operating frequency ratio by varying the charge pump current and the feedforward zero properly with the VCO bias condition. The similar design principle has also been applied to the regulated-supply PLL and DLL, as demonstrated in [62].

An adaptively-scaled bandwidth maintains the optimal behavior of PLLs and DLLs at all operating frequencies and keeps it robust against process, temperature, and voltage variations. A PLL with a fixed bandwidth, on the other hand, is often forced to compromise its optimal performance in order to meet the stability requirement at all possible conditions. For example, a PLL with an operating range of 1 to 100MHz cannot have a fixed bandwidth higher than 0.1MHz, which may be too low to reject sufficient noise at 100MHz. An adaptive-bandwidth PLL would scale its bandwidth from 0.1 to 10MHz as the frequency varies from 10 to 100MHz, achieving the optimum loop dynamics over the whole range.

For scalable loop dynamics, PLLs and DLLs must satisfy the following requirements: the loop bandwidth ω_n must maintain a constant ratio with the operating frequency and the damping factor ζ must remain constant at all frequencies. In terms of the PLL adjustments on phase and frequency ($\Delta\phi$ and $\Delta\omega$, respectively) upon the detection of the phase error

ϕ_{err}, the adaptive-bandwidth requirement can be expressed equivalently as:

$$\begin{aligned} \Delta\phi &= C_\phi \cdot \phi_{err}, \\ \Delta\omega/\omega_{ref} &= C_\omega \cdot \phi_{err}, \end{aligned} \tag{4.1}$$

where ω_{ref} is the reference frequency in radians per second and C_ϕ and C_ω are constants. The phase and the relative frequency ($\omega_{out}/\omega_{ref}$) of the VCO are updated by the amounts that have fixed ratios with the phase error. Similarly, in adaptive-bandwidth DLLs, the delay of the VCDL is updated by the fixed portion of the delay error D_{err}, expressed as:

$$\Delta D = C_D \cdot D_{err}, \tag{4.2}$$

where ΔD is the change in VCDL delay and C_D is a constant. Appendix B discusses the design principles of adaptive-bandwidth PLLs and DLLs in detail and reviews a few different implementations.

The local PLLs and DLLs in this dual-loop architecture satisfy the relations in Eq(4.1) and Eq(4.2) by exploiting the adaptive supply as a global bias voltage. The adaptive supply properly varies the charge pump currents and the VCO control gains with the operating frequency and adapts to the process and temperature variations. The circuit implementation of the local PLLs and DLLs presented in the following sections will demonstrate how the adaptive bandwidth is realized.

4.2 Multiphase Clock Generation PLL/DLL

This section describes the local PLLs and DLLs that generate multiphase clocks for the multiplexing transmitter. Figure 4.2 shows the block diagrams of the clock generator PLLs and DLLs. The VCO or VCDL has two control inputs, coarse and fine, and produces the multiple clock phases by tapping equally-spaced outputs. The voltage regulator suppresses the noise on the VCO/VCDL supply and subsequently the jitter on the clocks. The phase detector measures the error between the reference phase and the output phase, and the charge pump adjusts the control voltage accordingly to cancel the error. For DLLs, the additional duty-cycle control on the input clock is necessary to ensure the precise 180°

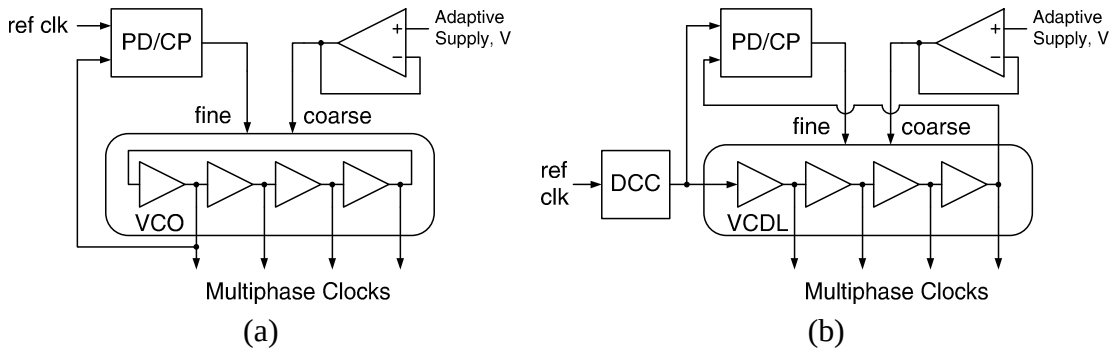


Figure 4.2: Local multiphase clock generators: (a) PLL and (b) DLL

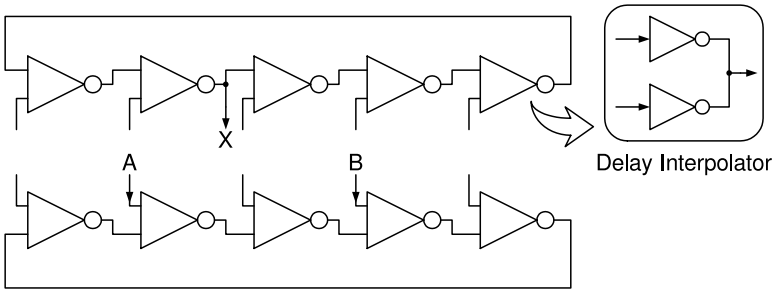
lock of aligning the rising edge of the input clock and the falling edge of the output clock [63].

The following subsections then visit each of these blocks shown in Figure 4.2 in detail. For the blocks common to both the PLL and DLL, the discussions focus mainly on the PLL.

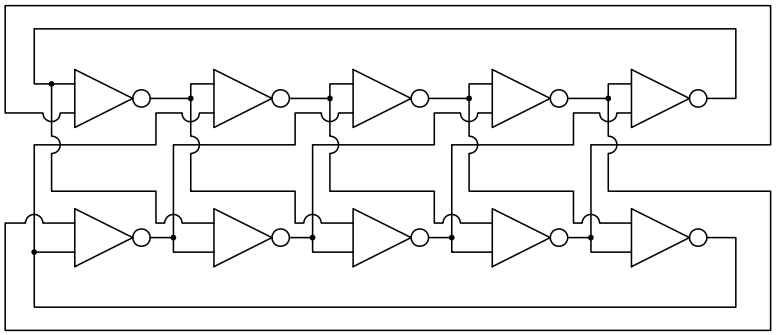
4.2.1 Coupled Voltage-Controlled Oscillator

The primary role of the local VCOs is to generate multiphase clocks. In addition, the identical VCO in the adaptive power-supply regulator has to indicate the maximum frequency at which the I/O circuits can operate. The maximum frequency of the I/O circuits is bounded by the minimum clock pulse-width that the clock buffers can propagate without attenuation [32],[10]. In order to have the VCO frequency track the clock buffer bandwidth, the delay elements of the VCO should be identical to the clock buffers, which, in most cases, are CMOS inverters. Unfortunately, a ring-oscillator made of single-ended inverters cannot provide both true and complementary clock outputs.

Instead, two single-ended ring-oscillators can be coupled to generate both true and complementary clocks. The theory of coupling multiple oscillators in general is described in [64]. Each buffer stage of the ring oscillator is basically a delay interpolator that has two inputs; one input from the previous stage on the same ring and the second input coupled from the other ring. The delay interpolator consists of two inverters with their outputs shorted together.



(a)



(b)

Figure 4.3: (a) Ring-oscillators with coupling inputs, (b) coupled VCO

There are in fact multiple ways of coupling two ring-oscillators to have 180° phase difference. The common way is to connect the output X from one ring to the input A of the other ring, as shown in Figure 4.3(a). This is equivalent to having two cross-coupled inverters between true and complementary outputs [65],[32]. Another way is to connect the output X to the input B [66],[67]. We use the latter scheme since it can support higher operating frequency thanks to the forward interpolation of phases. The overall VCO circuits are shown in Figure 4.3(b).

For DLLs, it is possible to build delay-lines with similar coupling. Delay-lines need inputs and outputs where the clock signal enters and leaves. Starting from the coupled VCO shown in Figure 4.3(b), one can break one pair of ring connections and use the resulted open ends as the inputs and outputs. It would be necessary to match the driving and loading conditions at these ends to those of the other buffers. The coupling connections, on the other hand, can still form a complete ring, which can help suppressing the reference clock

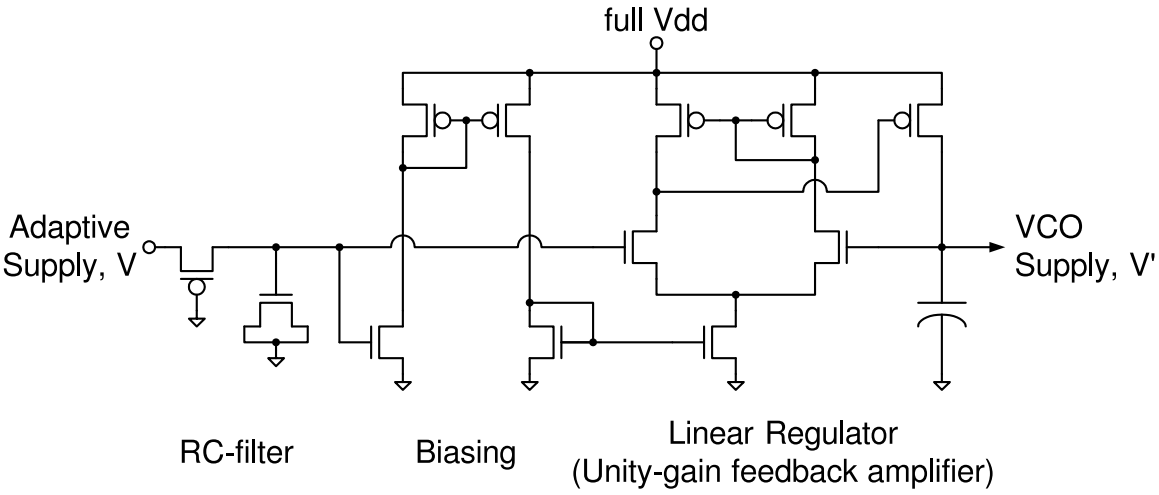


Figure 4.4: Combination of an RC filter and a linear regulator to filter noise on VCO supply

jitter. For better matching, the adaptive power-supply regulator can also be built with a reference delay-line, as described in Chapter 2.

4.2.2 Filtering Noise on the VCO Supply

To stabilize the VCO frequency and thus reduce clock jitter, it is important that the supply voltage of the VCO be kept constant. The adaptive supply taken directly from the global regulator can suffer from noise since the entire I/O circuits operate off of this supply.

A combination of a low-pass RC filter and a linear regulator shown in Figure 4.4 generates a quieter replica voltage that tracks the average value of the adaptive supply. The cut-off frequency of the RC filter scales proportionally with the clock frequency because the series resistance of the pMOS device varies with the adaptive supply and therefore with the clock frequency. This adaptive cut-off frequency enables effective filtering of the digital switching noise which has a spectrum that also scales with the clock frequency.

Voltage ripple from the switching regulator is not as serious a concern as the on-chip switching noise since the ripple has a small magnitude ($<$ 20mV) and low frequency ($<$ 1MHz). The low-frequency phase drift due to this voltage ripple will be corrected by the PLL.

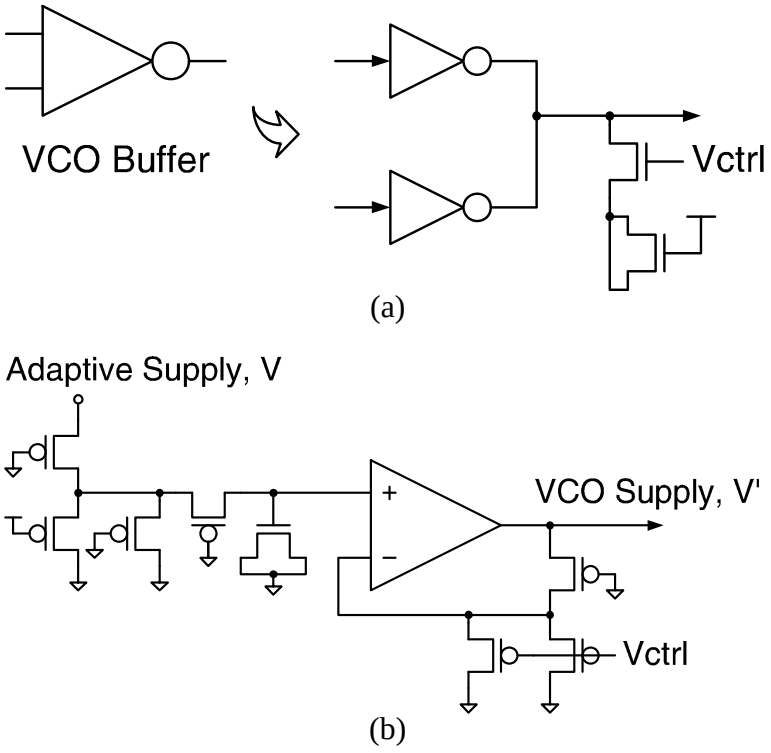


Figure 4.5: Fine frequency-tuning methods: (a) variable capacitive load, (b) variable offset in the VCO supply

The linear regulator is a differential amplifier driving a pMOS current source in a unity-gain feedback configuration. The linear regulator operates at the fixed nominal supply, so its power dissipation scales as Vf instead of V^2f, trading power for lower jitter. The bias current of the differential amplifier tracks that of the VCO, in order to reduce the power overhead of the amplifier at low supplies and to properly scale the output common-mode to reduce the voltage offset.

The VCO itself consists of inverters that generate noise while switching. The regulator in [62] added a load current that is 2-3 times larger than the VCO current to suppress the disturbance at the cost of increased power. The regulator in this design instead used a capacitive load of approximately 2pF at the regulator output to filter high frequency noise. However, this added capacitance may adversely affect the stability of the linear regulator, in which case compensation is required. The use of Miller capacitance across the second stage incurs the smallest area, but has poor supply noise rejection. Various compensation

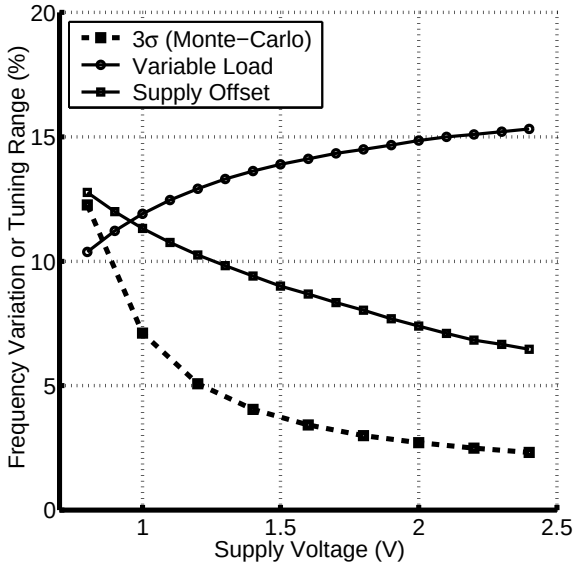


Figure 4.6: Ranges of two fine tuning methods, also with the expected frequency mismatches from the Monte-Carlo simulation

schemes are available and their trade-offs have been well studied [68].

4.2.3 Fine Frequency Tuning

Once the local VCO frequency is coarse-tuned by its supply voltage, another mechanism is required for the local PLLs to fine-tune the frequency. Figure 4.5 shows two implementations of fine frequency adjustment. Figure 4.5(a) uses a variable capacitive load at each buffer output to modulate the delay [69]. Figure 4.5(b) introduces an offset between the global adaptive supply and the local VCO supply, by adjusting the linear regulator closed-loop gain away from unity. Although both tuning methods can achieve the desired fine-tuning range, the use of supply offset is more advantageous because it presents less loading at the VCO buffer outputs.

The primary purpose of fine-tuning is to cancel the frequency offset between the global VCO and the local VCOs stemming from on-chip mismatches. Figure 4.6 plots the fine-tuning ranges of both methods at various supply voltages. The 3-σ variation in VCO frequency obtained from Monte-Carlo simulations is also shown on the plot [70],[71]. At high supplies, frequency offsets due to β-mismatch dominate and the variation is small (2.5%).

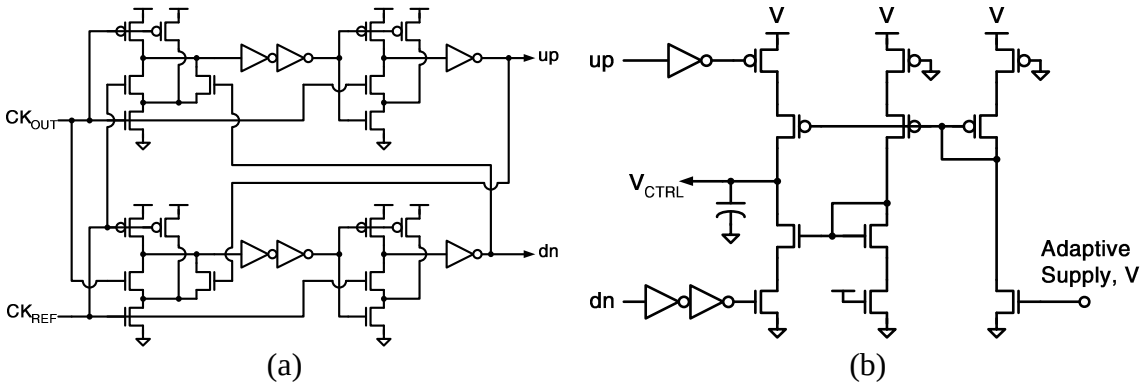


Figure 4.7: (a) Linear phase-only detector and (b) charge pump for PLL and DLL

However, at low supplies, the offsets due to Vth-mismatch dominate and the variation increases as $V - Vth$ decreases.

Figure 4.6 shows that as the supply voltage decreases, the percentage tuning range of the supply offset method increases while that of the variable load method decreases. The tuning range of the variable load method decreases because the dynamic range of the control voltage is reduced ($V - Vth$). The range of the supply offset method increases because the frequency-to-supply sensitivity of the VCO is higher at lower supplies. In this sense, the supply offset method is better. The variable load method requires a wide enough tuning range to cover the large frequency mismatch at the low voltage end, resulting in an overly wide tuning range at high voltages that may dilute the benefits of having a narrow fine-tuning range.

4.2.4 Phase Detector and Charge Pump

Most PLLs employ phase-frequency detectors, which extend the capture range by having extra state elements to detect cycle-slipping. For the local clock generation PLLs being described, however, the coarse frequency-tuning provided by the adaptive supply makes this extended capture range less useful. Linear phase-only detectors have wide enough lock-in range that can cover the required fine-tuning range of the VCO. Figure 4.7(a) shows the circuit details of the phase-only detector proposed in [62], which is simpler than most phase-frequency detectors [72],[73].

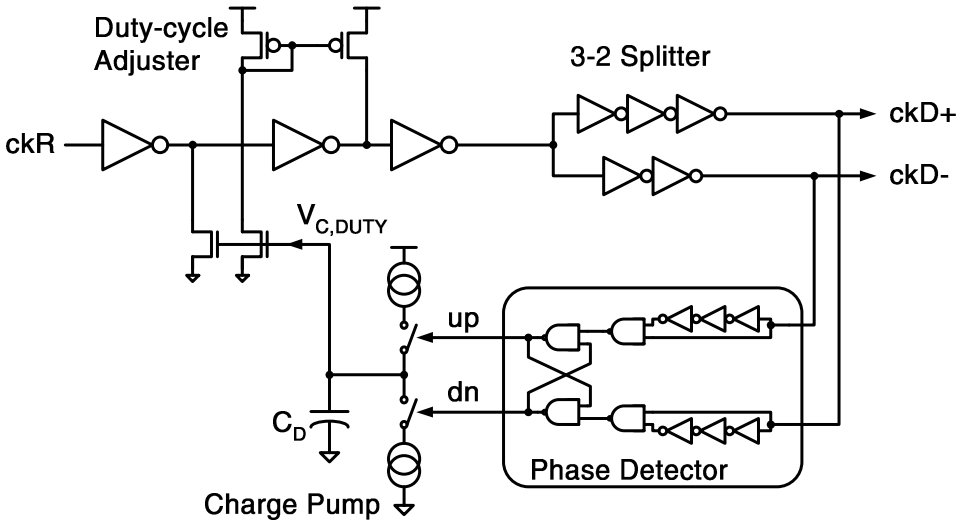


Figure 4.8: Duty-cycle correction on the DLL input clocks

Figure 4.7(b) shows a push-pull type charge pump similar to the one in [62]. The current level is biased with the adaptive supply V in order to scale the loop bandwidth proportionally with the operating frequency. The charge pump current I_{cp} scales as Vf_{ref} ($= (V - Vth)^{\alpha}$) [47],[20], while the VCO fine-frequency tuning gain (K_{VCO}) scales as f_{ref}/V. The VCO frequency changes by a constant fraction of the coarse-tuned frequency f_{ref} as V_{CTRL} sweeps from 0 to V, thus establishing K_{VCO}. As a result, the product of I_{cp} and K_{VCO} will scale as f_{ref}^2, satisfying the adaptive-bandwidth criteria for charge-pump PLLs in Eq(B.4). See Appendix B for the derivation of this criteria originating from Eq(4.1).

4.2.5 Duty-Cycle Corrector

The delay-line of the clock generation DLL tries to span 180 ° by aligning the output rising edge to the input falling edge. However, if the input clock has a duty-cycle different from 50%, the delay span will be different from 180 °. Figure 4.8 shows the duty-cycle corrector that ensures the 50% duty cycle of the DLL input clock. The duty-cycle is adjusted by varying the pull-up and pull-down strengths of the clock buffers.

The phase detector measures the distances between the rising edges of the true and complementary signals of the DLL input clock, **ckD+** and **ckD-**. The phase detector asserts

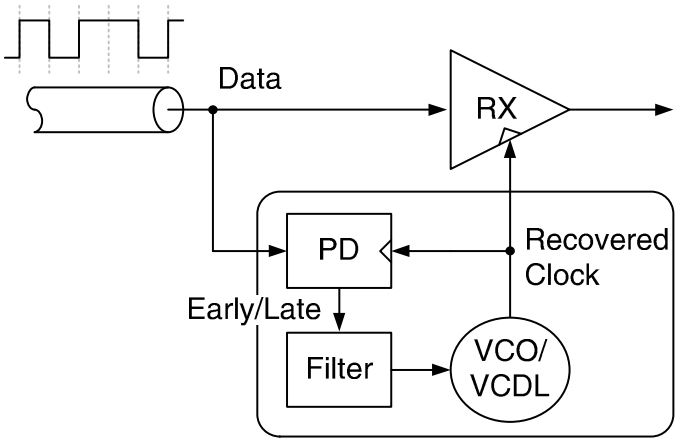


Figure 4.9: Recovering receiver timing from the data stream

dn during the period from the rising edge of **ckD+** and the rising edge of **ckD-**, and asserts **up** during the other period from the rising edge of **ckD-** to the rising edge of **ckD+**. The charge pump controlled by this phase detector adjusts the control voltage $V_{C,DUTY}$ and thus the duty-cycle of the clocks. The feedback loop will settle when these two periods are equally long, and equivalently, when the duty-cycle is 50%. The infinite DC gain of the charge pump keeps the remaining duty-cycle error smaller than that of the amplifier-based duty-cycle corrector [63].

4.3 Multiphase Clock Recovery PLL/DLL

As mentioned in the introduction, the limited pin resources and the increasing number of high-speed links on a chip are likely to prefer serial links that recover timing from the data stream, to parallel links that require separate clocks transmitted by the source. Figure 4.9 illustrates a timing recovery loop of a serial link. From the transitions embedded in the data stream, the loop determines whether its timing is early or late and adjusts it accordingly. This section describes per-pin clock recovery loops which also provide multiphase clocks for the demultiplexing receivers.

Shown in Figure 4.10 is the phase detector for a multiphase clock recovery loop. A duplicate set of data receivers sampling the data edge rather than the center serves as the phase detector. Depending on the data received before and after the edge, the output of

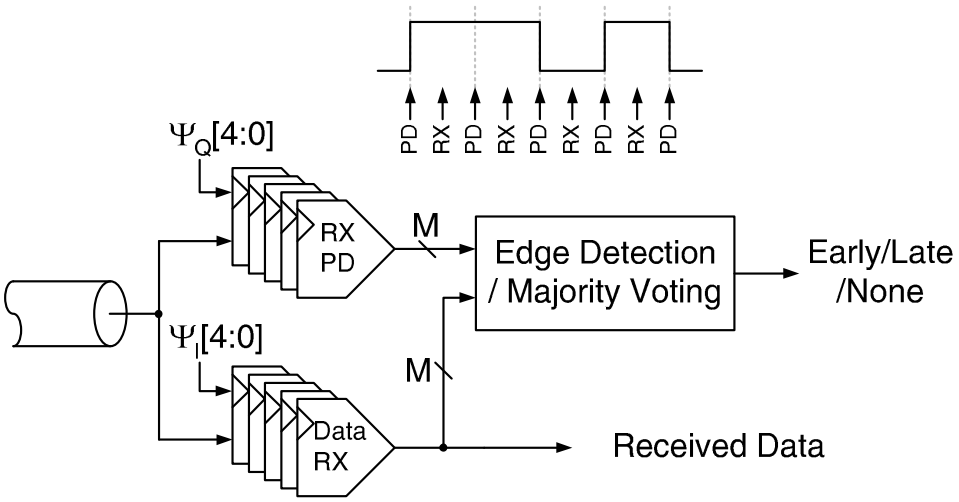


Figure 4.10: Phase detector for per-pin clock recovery loops

the phase detecting receiver can determine whether the recovered timing is early or late [74]. These phase detecting receivers need to be triggered off the quadrature clocks, i.e. the clocks shifted by one half bit period. With an odd number of phases spanning 360°, the quadrature clocks are simply the complementary version of the clocks. With an even number of phases, however, additional circuits such as phase interpolators may be needed to generate the intermediate phases [2],[75].

In this multiphase system, the outputs from the multiple phase detectors are aggregated to generate one output per cycle by using a majority voting circuit. The detailed issues with the multiphase system and the implementation of the majority voting circuit are discussed later in Section 4.3.1.

Using a replica receiver as the phase detector is ideal for achieving zero timing offsets. Most receivers have setup times that introduce offsets between the times when the clock fires and when the data is actually sampled. Therefore, using any circuit other than the receiver itself for phase detection risks having unmatched timing offsets.

However, this kind of phase detector can provide only the polarity of the timing error, but not the magnitude. In other words, the phase detector output can only be one of the followings: early (-1), late (+1), or no transition (0). The discrete outputs of the phase detector makes the clock recovery loop bangbang-controlled. Bangbang-controlled loops are inherently nonlinear, which distinguish themselves from the more conventional linear

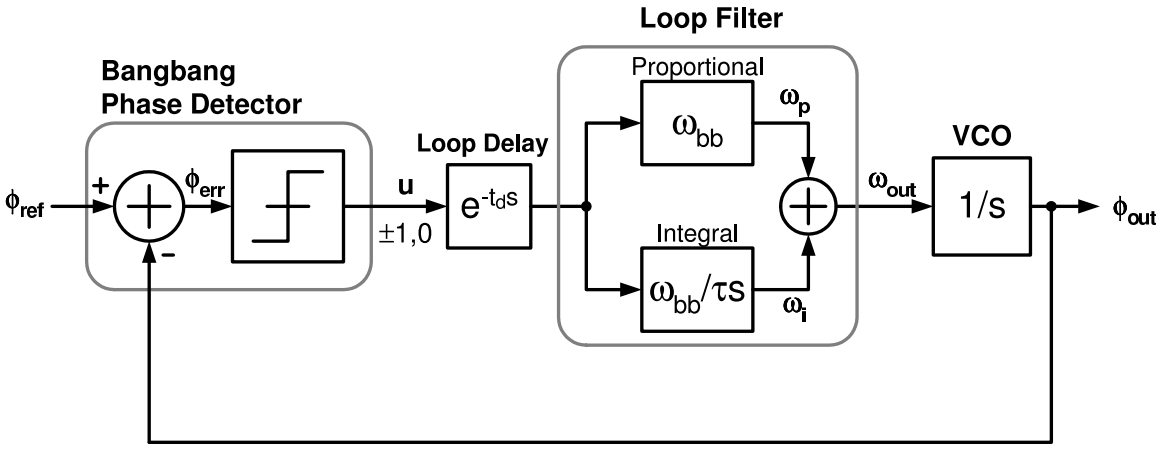


Figure 4.11: Block diagram of a bangbang-controlled clock recovery PLL

control loops.

The following subsections describe the multiphase clock recovery PLL and DLL, both of which are bangbang-controlled. Similar to the clock generator PLL and DLL, the clock recovery PLL and DLL are also the local loops that exploit the adaptive power-supply regulator as the global loop and use the same design principles to achieve adaptive bandwidth. The circuit details are presented and the comparison between the PLL and the DLL in terms of their power, area, and jitter performance are discussed.

4.3.1 Bangbang-Controlled Clock Recovery PLL

Figure 4.11 shows the block diagram of the multiphase clock recovery PLL. The loop dynamics of a bangbang-controlled PLL have been studied in [76] and [77], and are discussed in more detail in Appendix C.

The output of the bangbang phase detector controls the timing of the VCO through two paths: the proportional path and the integral path. The proportional path steps the VCO phase up or down by a fixed amount depending on the error polarity. The integral path, on the other hand, steps the frequency of the VCO.

For stability of the loop, the proportional gain must be sufficiently larger than the integral gain. This is equivalent to having large enough damping ratios in linear PLLs. However, an integral gain that is too low may slow down the frequency tracking of the PLL

and make the loop vulnerable to noise. The analysis in Appendix C suggests that the ratio $\tau/t_{d,eff}$ of 20 to 40 is a good compromising point between stability and tracking. The effective loop delay $t_{d,eff}$ is the actual loop delay plus one update period, taking into account that the discrete-time sampling action of the phase detector can effectively add delay.

Even when the loop is in lock, the bangbang action of the phase detector causes the VCO phase to dither. This dithering jitter has a peak-to-peak value of $2\omega_{bb} \cdot t_{d,eff}$, which is proportional to both the loop gain and the loop delay. Therefore, the magnitude of the loop gain is constrained by the desired timing accuracy of the application.

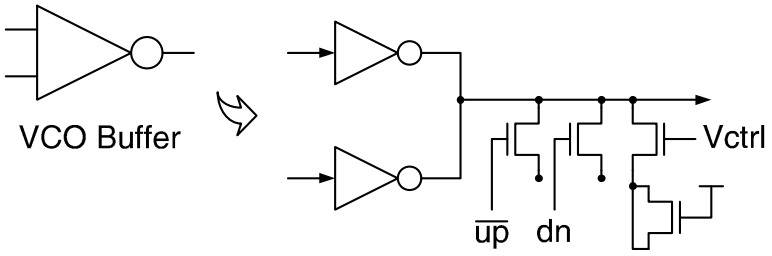
Proportional Control: Stepping the VCO Phase

The proportional control path of the bangbang PLL induces a change in VCO phase by making a momentary step in frequency (ω_{bb}) that lasts for one cycle (t_{update}). Figure 4.12 shows the implementation of this proportional control suitable for each of the fine-tuning methods described in Section 4.2.3. In the variable load method (Figure 4.12(a)), some amount of extra capacitive load is switched in or out depending on the bangbang phase detector outputs (**up** and **dn**). In the supply offset method (Figure 4.12(b)), a positive or negative current step is injected at the VCO supply, which builds a voltage step due to the finite output impedance of the linear regulator [62],[61].

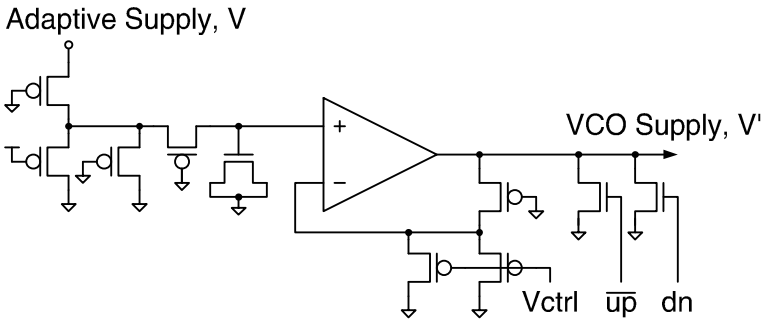
In both cases, the proportional frequency step ω_{bb} scales proportionally with the coarse-tuned VCO frequency, f_{ref}. The adaptive supply affects the series resistance of the switch, the magnitude of the current step, and the output impedance of the regulator. Since the resulting phase step is equal to ω_{bb} times the cycle time t_{update}, the phase step in radians stays constant independent of the operating frequency. Therefore, the clock recovery PLL meets the phase criteria for adaptive bandwidth in Eq(4.1), as well as the frequency criteria already discussed in Section 4.2.4.

When the loop is in lock, the VCO clock edge will dither by this phase step. Therefore, a small phase step is desired to keep the dithering jitter small. However, a small phase step will slow down the PLL tracking response. More importantly, the *lock-in range*, i.e. the range of initial frequency difference at which the PLL can lock without cycle-slipping, will become very narrow ($\pm\omega_{bb}$), with a small phase step.

The PLL can still acquire lock beyond the lock-in range. The range of frequency at



(a)



(b)

Figure 4.12: Proportional controls in: (a) variable load method, (b) supply offset method

which the PLL can lock after some cycle-slipping is called *pull-in range*. However, the pull-in range can be limited by the loop delay; according to the analysis in Appendix C, the pull-in range is $\pm 0.5\pi / t_{d,eff}$.

The clock recovery PLL in this work can acquire lock when the initial frequency of the VCO is within 7% of the reference. It is in fact narrower than the fine frequency tuning range, $\pm 15\%$. If the frequency difference is larger than 7%, the phase detector will not be able to pull the frequency to lock by itself. In this case, a frequency acquisition aid is necessary to guide the loop to lock.

Frequency Sweeping as an Acquisition Aid

A wide-range clock recovery PLL, in general, needs a sophisticated frequency acquisition aid because there are multiple frequencies at which the PLL may fall in lock, e.g. the harmonics. Fortunately, in the clock recovery PLL with the coarse frequency tuning, the

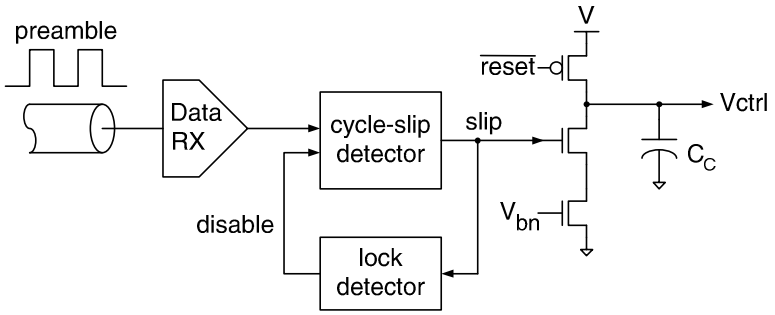


Figure 4.13: Frequency sweeping as the frequency acquisition aid

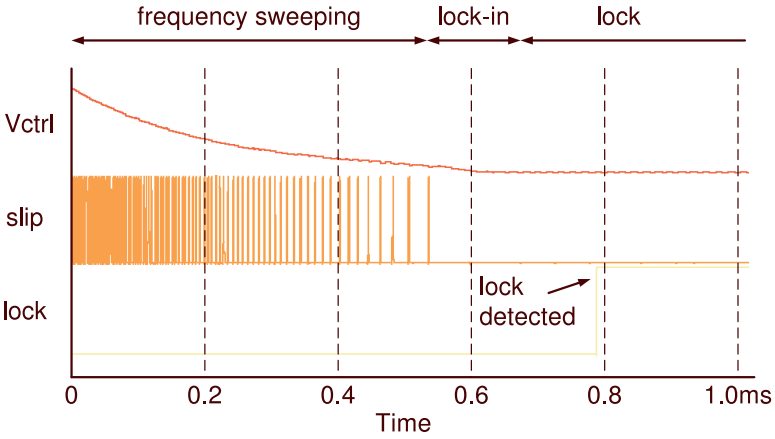


Figure 4.14: Transient response during locking

narrow fine-tuning range guarantees that there is only one possible lock state. Therefore, a simpler scheme such as frequency-sweeping can be used to aid the frequency acquisition [78].

Figure 4.13 illustrates implementation of the frequency sweeper that we used. During preamble mode, the transmitter side is assumed to send a full-transition signal 10101010. The control voltage V_{CTRL} is initialized to its maximum value V and therefore the VCO frequency also starts from its highest value. The frequency sweeper pulls charge out of the filter capacitor C_C and lowers V_{CTRL} whenever the loop slips a cycle. Cycle-slipping implies that there is a difference between the transmitted rate and the receiving rate, and is detected by comparing two consecutive bits received. If these two bits are not different, the cycle-slip detector asserts the signal *slip*. This way, the VCO frequency is swept downward until the frequency error is no longer detected by cycle-slipping. Once the VCO frequency

gets to within the lock-in range, the phase-tracking loop will have enough control gain to acquire lock. Figure 4.14 shows the transient during frequency sweeping and phase lock.

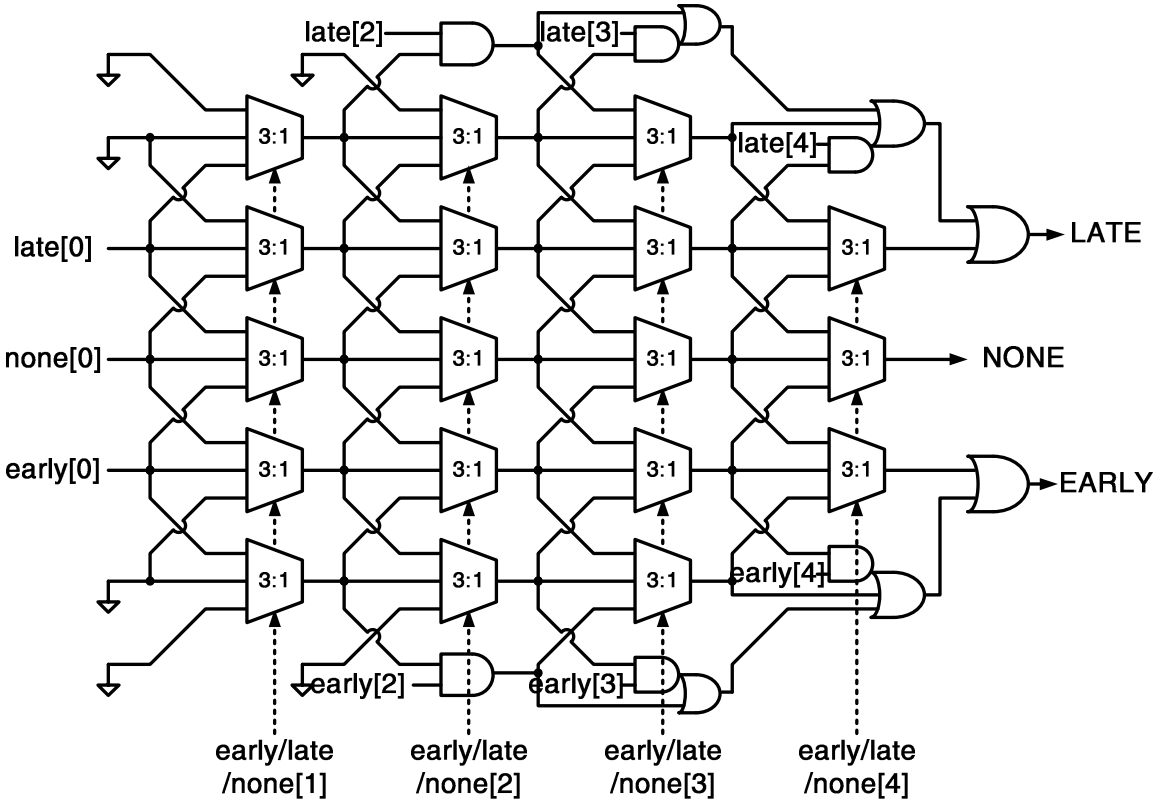
The frequency-sweeping current is biased so that it has a constant ratio with the main charge pump current. The sweeping current should be neither too large nor too small for the intended operation and the detailed analysis on its valid range is presented in Appendix C. The valid range is in fact quite wide and is a function of the loop gain ω_{bb} and the loop delay $t_{d,eff}$.

Decimating Multiple Phase Detector Outputs: Majority Voting

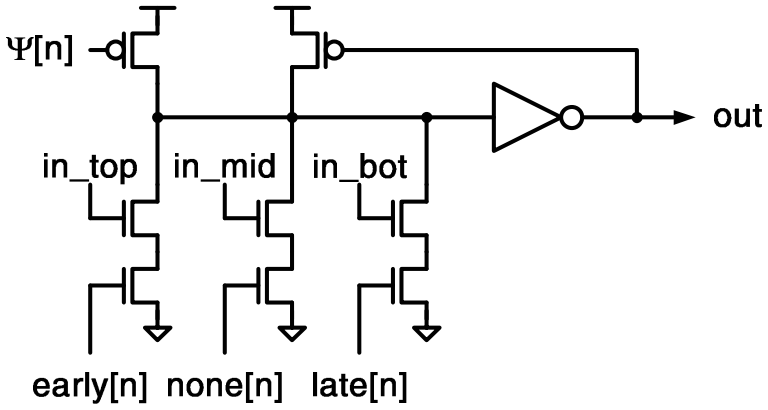
In the clock-recovery applications that use multiphase clocks, there can be more than one data transition per cycle where the timing decision can be made. At each of those transitions, a phase detector may be placed. In fact, placing too few phase detectors may create additional false locking states, e.g. at subharmonics of the reference frequency. Unless some coding can ensure the uniform distribution of transitions over different phases, it is generally necessary to place phase detectors at all phases.

In this case, there will be multiple phase detector outputs per cycle. These outputs can be applied individually to update the VCO timing or can be aggregated to a single output per cycle, e.g. by taking the majority vote among them. There are trade-offs between the two schemes which are discussed in Appendix C. Majority-voting has the disadvantage of increasing the loop delay, but has the advantage of keeping the control gain unchanged for the wide range of transition density.

Figure 4.15 shows the details of the majority-voting circuit. Basically, the circuit keeps a tally of up's and down's from the phase detectors as they resolve to their decisions at different phases. The skew-tolerant domino clocking [79] helps processing these different phase domain signals without the extra step of synchronization and thus reducing the overall loop delay. Short loop delays are critical in minimizing the dithering jitter and extending the pull-in range of a bangbang-controlled PLL.



(a)



(b)

Figure 4.15: Majority-voting for decimating multiple phase detector outputs: (a) circuit block diagram, (b) 3:1 multiplexer with skew-tolerant domino clocking

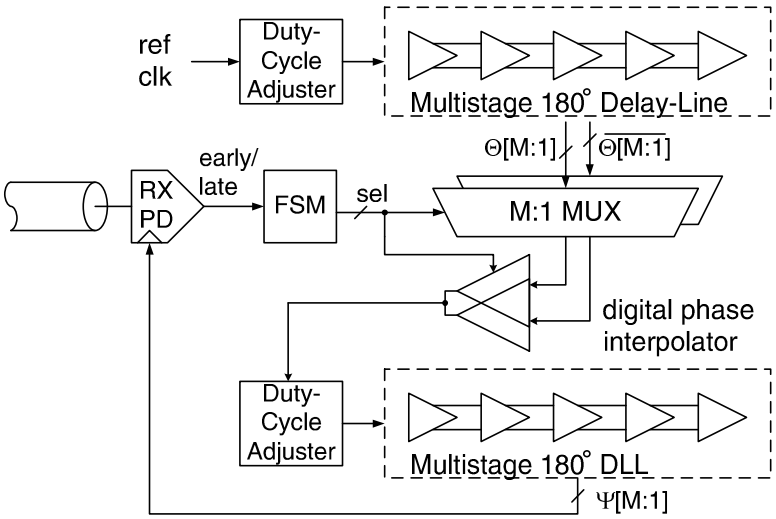


Figure 4.16: Triple-loop architecture of the clock recovery DLL

4.3.2 Bangbang-Controlled Clock Recovery DLL

A DLL can also perform bangbang-controlled clock recovery. The overall architecture of the multiphase clock recovery DLL is shown in Figure 4.16, which is based on the semi-digital dual-loop DLL proposed in [63]. The first loop creates multiple clock phases from the reference clock and the second digital loop selects two phases among them and interpolates them to generate the tracking phase. The finite-state machine adjusts the tracking phase by changing the phase selection and the interpolating ratio, depending on the output of the bangbang phase detector.

The second loop gives only one phase that tracks the optimal timing. Instead of having multiple phase selectors and interpolators to provide more phases [80], the DLL in Figure 4.16 has a third loop that generates multiple phases. Both the first and the third loop need 50% duty-cycle control on their input clocks to ensure the correct 180 ° lock.

The clock recovery DLL thus has a triple-loop architecture. Similar to the other PLLs and DLLs described above, the three loops in this DLL all serve as the local loops to the global adaptive power-supply regulation loop. For example, the bandwidth of the feedback loop is scaled proportionally with the clock frequency by biasing the charge pump current with the adaptive supply. The bandwidth of the interpolating buffers is also scaled with the

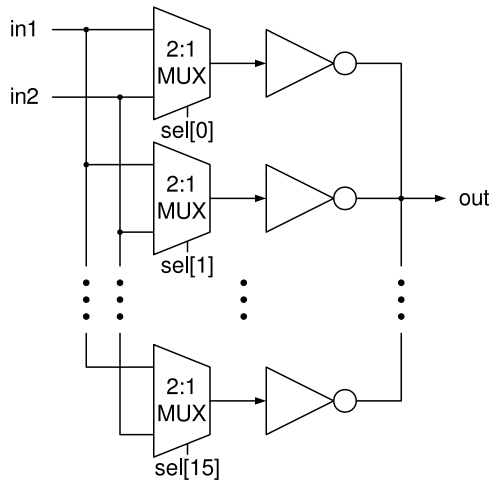


Figure 4.17: Digital phase interpolator

aid of the adaptive supply, as discussed next.

Digital Phase Interpolator

Figure 4.17 shows the digital phase interpolator used in the clock recovery DLL. The interpolator is basically a collection of inverters with their outputs tied together. The 2:1 multiplexers select the input signals for the inverters and thus the interpolation weight.

For the smooth transition of the interpolated output at all operating frequencies, the bandwidth of the interpolating stage must scale with the frequency [63]. The interpolating inverters operating off of the adaptive supply coincidentally meet this requirement.

4.3.3 Comparison between PLL- and DLL-based Clock Recovery

Table 4.1 summarizes the power, area, and jitter performance of the PLL-based and the DLL-based clock recovery circuits reported in [81], which generate five clock phases. As discussed in Section 4.3.2, the clock recovery DLL needs an additional delay-line to generate the multiphase clocks for the demultiplexing receiver. This additional circuitry adds 110% more power and 70% more area, compared to those of the PLL. Although the clock recovery DLL does not have VCOs that accumulate jitter, the long delay through the three

Table 4.1: Comparison between PLL- and DLL-based clock recovery

	PLL	DLL
Power dissipation		
@ 3.5Gb/s	16.6mW	34.4mW
@ 2.5Gb/s	7.9mW	16.8mW
@ 1.0Gb/s	2.2mW	4.6mW
Area occupation	0.16mm^2	0.27mm^2
Peak-to-peak jitter (700MHz, quiet supply)	36.7ps	37.8ps

cascaded loops and the duty-cycle correctors increases the DLL noise sensitivity. The measured peak-to-peak jitters of the PLL and the DLL were in fact comparable, suggesting that the PLL-based clock recovery is more favorable in multiphase clock applications.

4.4 Summary

This chapter presented a dual-loop architecture that reduces the power and area overhead of the per-pin multiphase clock generation and recovery circuits. The adaptive power-supply regulator serves as the global frequency regulation loop. The adaptive supply minimizes the power dissipation and reduces the area of the local PLLs and DLLs. The VCO frequency is coarse-tuned by the global loop, which eases the frequency acquisition of the PLL.

The PLLs and DLLs exploit the adaptive supply as a global bias that adjusts loop parameters to achieve adaptive bandwidth. Adaptive bandwidth PLLs and DLLs maintain optimal behaviors at all operating frequencies and process/temperature conditions. In case of clock recovery, the design issues of the bangbang-controlled PLL and DLL are discussed. The complexity of the DLL in recovering multiphase clocks makes the PLL-based clock recovery more favorable.

Chapter 5

Design Trade-Offs in Adaptive-Supply Serial Links

The adaptive-supply link presented in this dissertation provides a unique opportunity to explore the trade-offs among design parameters. For example, the efficient adaptive power-supply regulator and the low-voltage transceivers enable trading excess bitrate for reduced power. The parallelized transceivers and power- and area-efficient multiphase clock generators help achieve higher bitrate per pin, which can also be traded for lower power. However, high degrees of parallelism may degrade the signal quality due to mismatch between parallel devices.

This chapter explores the trade-offs among bitrate, power dissipation, degree of parallelism, and signal quality. As introduced in Chapter 1, the figure-of-merit to compare different designs is the energy-per-bitrate ratio. This metric measures the total energy for the link to transfer one bit normalized to the bitrate. A lower energy-per-bitrate ratio indicates a higher efficiency in terms of both speed and power.

First, the results from the experimental chip fabricated in 0.25μm CMOS technology are presented. Then, we extrapolate the measured data to evaluate designs with different degrees of parallelism and with different process technologies. The impacts of transistor mismatch on each link component are also discussed. As the CMOS process advances toward 0.05μm channel length, it is anticipated that transistor mismatch and limited channel capacity will depreciate the benefits of parallelism and low-voltage operation.

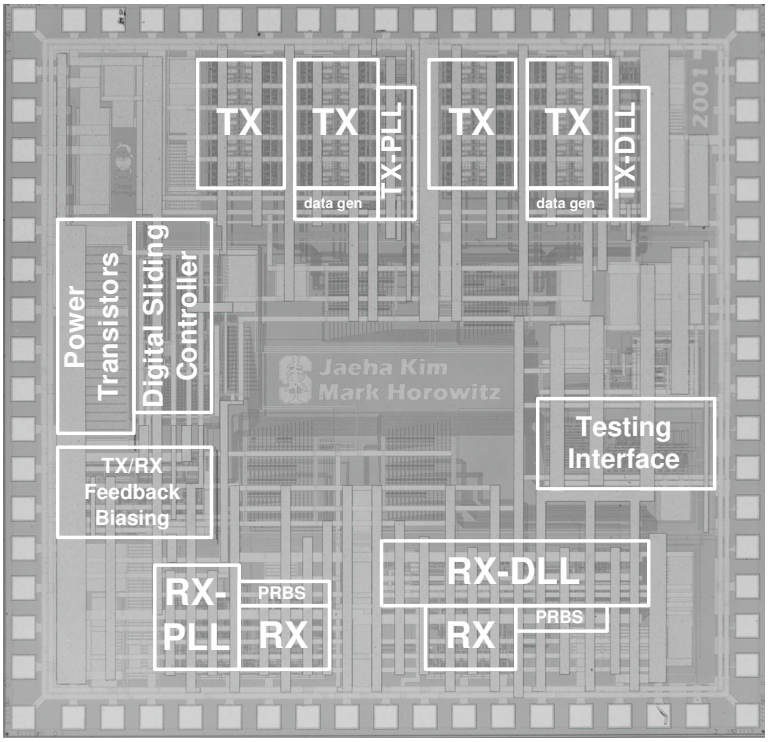


Figure 5.1: Die micrograph of the experimental adaptive-supply serial link chip

5.1 Experimental Chip Results

Figure 5.1 is the die micrograph of the adaptive-supply serial link test chip fabricated in 0.25-μm standard CMOS technology. This experimental chip contains two types of serial link: a PLL-based link and a DLL-based link. The 5:1 multiplexing transmitter and 1:5 demultiplexing receiver achieve a bitrate that is five times higher than the clock frequency. The chip also includes the adaptive power-supply regulator presented in Chapter 2 with on-chip power transistors. The adaptive power-supply regulator determines the optimal voltage level for the specified bitrate and supplies it to both types of serial links on the chip. The inductor and the capacitor for the buck converter are external to the chip.

The first experimental chip used the variable capacitive loads to perform local fine frequency tuning and successfully operated at 450Mb/s–3.5Gb/s over the regulated supply range of 0.9–2.5V [81]. However, this performance was lower than anticipated due to the VCO's wire loads that were not duly accounted for. The cross-coupling connection inside

Table 5.1: Adaptive-supply serial link chip characteristics

Technology	0.25μm nwell CMOS
Nominal supply voltage	2.5V
Threshold voltage	0.55V
Total die area	3.1$\times$2.9mm^2
Regulated supply range	0.9–2.5V
Clock frequency range	130–1000MHz
Bitrate range	0.65–5.0Gb/s
Total power per link	9.7–380mW
Bit error rate @ 5.0Gb/s	$< 10^{-15}$
Regulator efficiency	83–94%
Ripples on regulated supply	$<$ 20mV

the VCO adds a significant amount of capacitance at each output (80fF) and should not be overlooked. The second chip adjusted for these wire loads and operated at the initially expected rates: 650Mb/s–5.0Gb/s [82]. In addition, the second chip implemented the supply offset method for fine frequency tuning, which also helped reduce the VCO loads. Table 5.1 summarizes the characteristics of the revised chip.

Figure 5.2(a) plots the total power dissipation of each link versus the bitrate. The measured data include the power dissipated on the channel and it is assumed that the output swing can scale proportionally with the supply voltage. This assumption holds if the dominant noise for the link is from switching activity on the chip and the link's signal swing is sufficiently larger than the fixed offset components, e.g. receiver input-referred offset.

The power dissipation of transmitters, receivers, and clock buffers scale as $V^2 f$ because the dominant portion of their power is due to switching capacitive nodes at a frequency f and at a voltage swing of V. The power dissipated on the channel does not scale as $V^2 f$ but has a relatively small portion, which is generally true for links with small swings and high multiplexing rates.

The power of the voltage-controlled oscillators and delay-lines, on the other hand, scales at a slower rate, Vf, because they do not operate directly off of the buck-regulated supply, but off of the cleaner supply generated from the linear regulators in order to keep the clock jitter low. The linear regulator operates off of the fixed nominal supply, so its

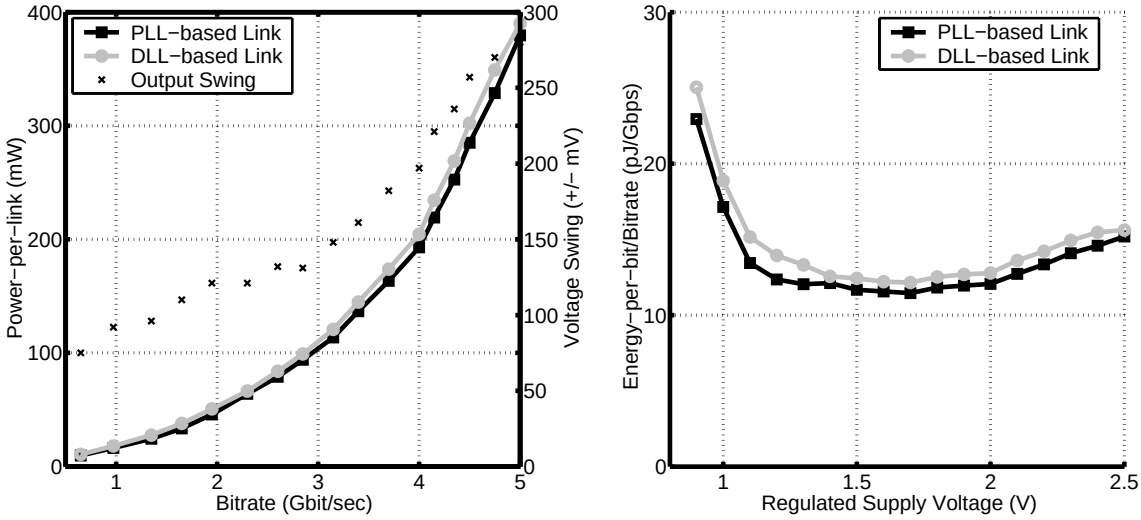


Figure 5.2: (a) Per-link power dissipation and output swing versus bitrate, (b) energy-per-bitrate metric versus regulated supply

power scales only due to the load current scaling ($I \propto Vf$).

Figure 5.3 plots the scaling trends of these two power components for comparison: the power drawn from the regulated supply and the power drawn from the fixed nominal supply. At a regulated supply of 2.5V, the power dissipation due to the linear regulators accounts for only 9% of the total and the overall power scaling is closer to the V^2f trends, as shown in Figure 5.2(a). However, as the supply voltage drops, the Vf-scaling portion increases to 18% in the PLL-based links and 22% in the DLL-based links. As a result, at low voltage range, the total power scales less aggressively than V^2f.

Figure 5.2(b) plots the energy-per-bitrate ratio at each operating point, the figure-of-merit to assess the link's efficiency in speed and power. For digital systems consisting of velocity-saturated MOS transistors, the analogous energy-delay product is expected to have its minimum at a supply voltage of $2Vth$ [20]. In the adaptive-supply links, the minimum energy-per-bitrate ratio was found at a higher voltage of 1.7V, about $3Vth$, because of the portion of the power that does not scale as V^2f. However, the curve is fairly flat and the energy-per-bitrate ratios close to the minimum can be achieved between 1.4V and 2.0V. Note that the optimal point for power-per-bitrate, the other common metric, is at the lowest possible supply and at the lowest possible bitrate, which is generally not the solution of

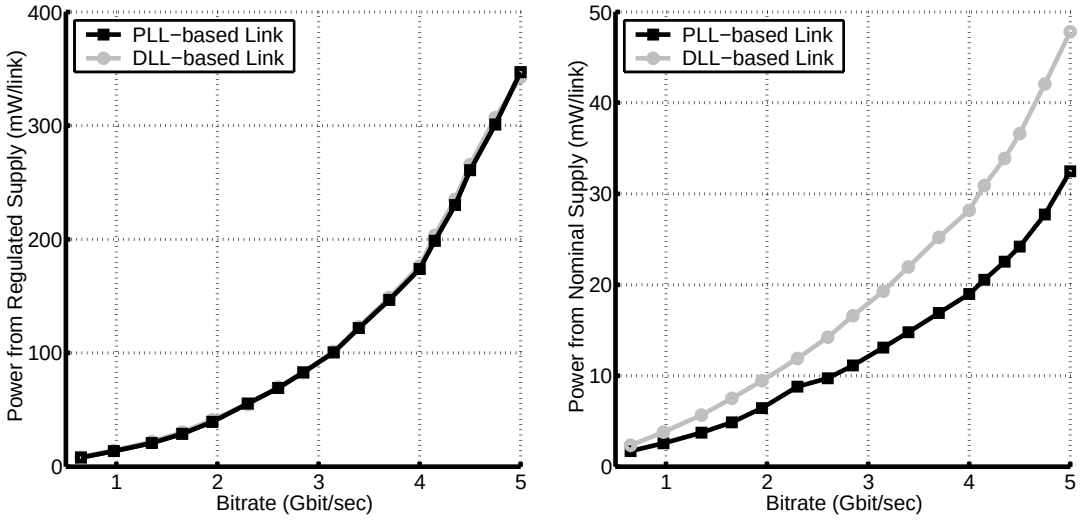


Figure 5.3: Scaling trends of two power components: the power drawn (a) from the regulated supply ($V^2 f$) and (b) from the fixed nominal supply ($V f$)

interest.

Fig. 5.4 shows the power dissipation of each component in the link operating at the optimum supply of 1.7V. The power dissipation is still dominated by switching the on-chip capacitive nodes, which justifies the use of adaptive power-supply regulation to keep the dynamic switching power to its minimum.

5.2 Impacts of Transistor Mismatch on Design Trade-Offs

Parallelized architectures such as the multiplexing transmitter and demultiplexing receiver allow achievement of bitrates that are much higher than the clock frequency. Parallelism is especially beneficial for adaptive-supply links since it can generate higher performance that can be turned into further power reduction. In other words, at a certain bitrate, a higher degree of multiplexing lowers the required clock frequency and supply voltage, and the reduction in frequency and voltage leads to large savings in power.

However, parallelized architectures demand good matching between the parallel, supposedly identical components. Transistors that are laid out exactly the same may still have

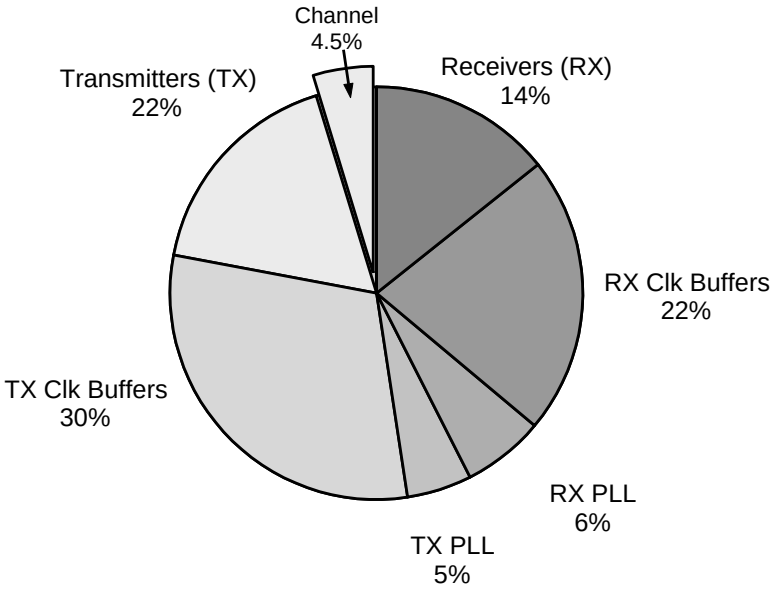
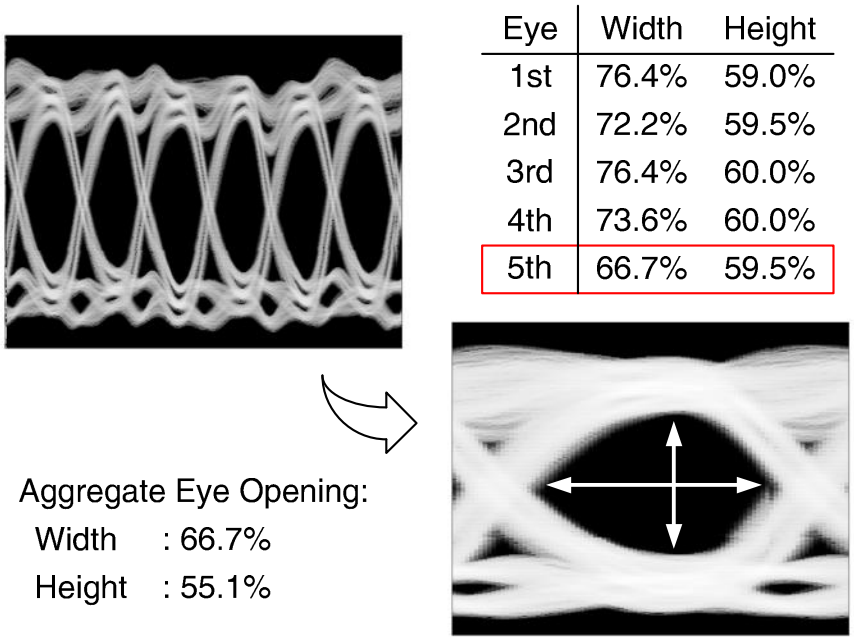


Figure 5.4: Breakdown of power dissipation of a PLL-based link operating at 1.7V



Eye	Width	Height
1st	76.4%	59.0%
2nd	72.2%	59.5%
3rd	76.4%	60.0%
4th	73.6%	60.0%
5th	66.7%	59.5%

Figure 5.5: Transmitter eye diagram of the adaptive-supply serial link operating at 3.0Gbps: the individual eyes at different phases and their folded, aggregate eye

random offsets and achieving good matching becomes more difficult when there are more transistors to be matched [83],[84]. Any mismatch between these parallel components may effectively appear as noise and degrade the quality of the signal. For example, non-uniform spacings between the clock phases can narrow the data eye and reduce the timing margin. Similarly, offsets between the parallel branches of the multiplexing transceiver can reduce the voltage margin. The measured transmitter eye in Figure 5.5 illustrates this reduction in eye opening due to timing and voltage mismatches. The eyes at different phases have slightly different openings and when they are folded into one, the aggregate eye has the worst opening of all.

Low-voltage operation aggravates the effects of transistor mismatch. For example, the same offsets in threshold voltage (V_{th}) and current factor (β) cause the larger mismatch in drain currents when the gate-to-source voltage (V_{GS}) is lower [85]. Since low voltage operation means low gate overdrives in most cases, the degradation in signal integrity may pose a limit on the power savings achievable by lowering the supply.

This section examines how transistor mismatch influences the design decisions for low-power and high-speed links. The effects of transistor mismatch on clock phase offsets, transmitter offsets, and receiver offsets are discussed and the overall impact on energy-per-bitrate scaling is estimated. Some key challenges to low-voltage, parallel operation are then identified.

5.2.1 Timing Margin Degradation due to Transistor Mismatch

The use of multiphase clocks can increase timing uncertainty due to the static offsets between different clock phases and the dynamic jitter on each phase. Figure 5.6 illustrates the narrowed timing window due to static phase offsets and dynamic jitter.

Non-uniform phase spacings cause the data transition edges not to overlap at the exact same time on the eye diagram. Peak-to-peak variation of each phase's offset from its ideal point determines the horizontal reduction of the data eye. This quantity can also be interpreted as the difference between the maximum and minimum integral nonlinearities (INL) of the phase steps. This peak-to-peak difference generally increases with the number of clock phases being used.

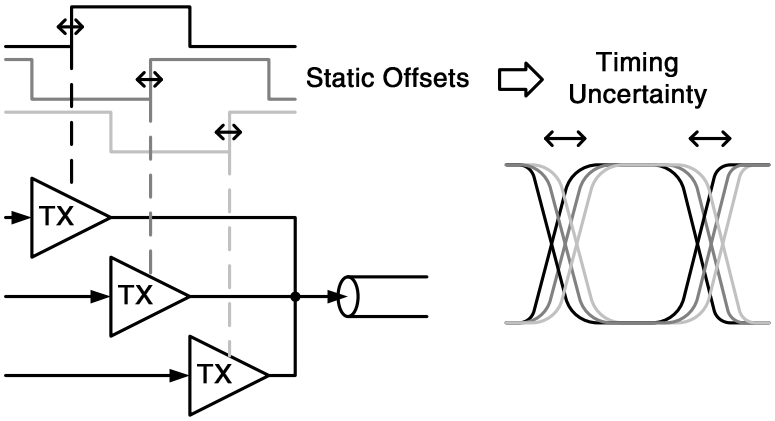


Figure 5.6: Timing margin degradation due to static phase offsets

In addition, for a certain number of stages in the oscillator, tapping more phases leads to less number of buffer delays making up the unit phase spacing. The spacing that consists of less number of buffer delays has larger percentage variation in time, because the delay increases linearly with the number of buffers but the variation in delay increases only with its square-root.

The combined effects of the increased number of phases and increased variation in the unit phase spacing make the degradation in timing margin vary approximately linear with the multiplexing rate M, as plotted in Figure 5.7(a). The plot is for the VCO used in the 0.25-μm experimental chip and the mismatch parameters of $A_{Vth} = 9.0$mV$\cdot\mu$m and $A_\beta = 2.5\%\cdot\mu$m are assumed [85]. The VCO runs at the on-chip clock period of 8 fanout-of-4 (FO4) inverter delays, which is about 1GHz at the nominal 2.5V in 0.25μm technology. Multiplexing rates of 7, 9, and those above 10 are omitted in the analysis because they require more than 6 stages in the VCO and thus require the VCO buffers with fanouts less than 2, which are impractical in terms of power and area.

On the other hand, dynamic jitter on the clock also increases the timing uncertainty of the link. The clock jitter depends primarily on the loop bandwidth of the PLL and the bandwidth of the noise rejection loop, and does not depend strongly on the number of clock phases being used. At a given clock frequency, the bit time varies inverse-linearly with the multiplexing rate M but the jitter on each clock phase remains relatively unchanged. As a result, the jitter per unit interval (UI) increases linearly with M. Figure 5.7(a) also plots the

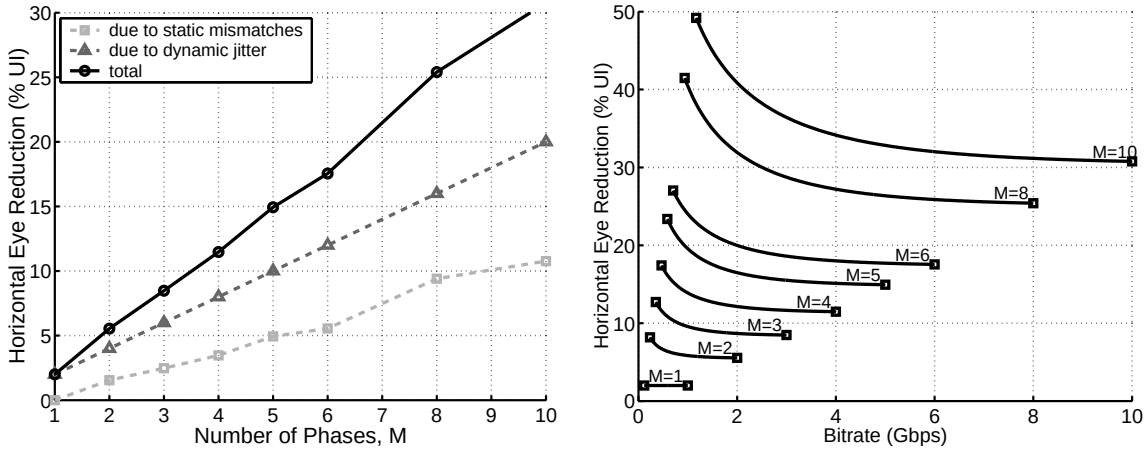


Figure 5.7: Timing margin degradation due to transistor mismatch and clock jitter

degradation in timing margin due to dynamic jitter. Here it is assumed that the PLL output clock has a peak-to-peak absolute jitter of 2% of the clock period.

Figure 5.7(b) shows the dependency of timing margin degradation on the multiplexing rate M as well as on the bitrate ($= M \cdot f$). The static phase offsets become worse at lower voltage and at lower clock frequency f, but the dynamic jitter does not necessarily increase in terms of percentage of the cycle time. However, static offsets can be reduced by upsizing the buffers of the VCO since matching improves with the square-root of the gate area. Thus, the jitter sets a harder bound on the timing accuracy than the static offsets and makes the use of M above 6 impractical. It should be noted that the timing uncertainties of both the transmitter PLL clocks and the receiver PLL clocks must be considered when estimating the overall timing margin of the link.

5.2.2 Voltage Margin Degradation due to Transistor Mismatch

Mismatch among the parallel transmitters and receivers can degrade the signal's voltage margins. The voltage margin will be limited by the transmitter branch with the smallest swing and by the receiver branch with the largest offset. Moreover, parallelized architectures increase the capacitive loading at their merging points, i.e. at both terminals of the channel. This increased loading can attenuate the signal's amplitude at high frequencies

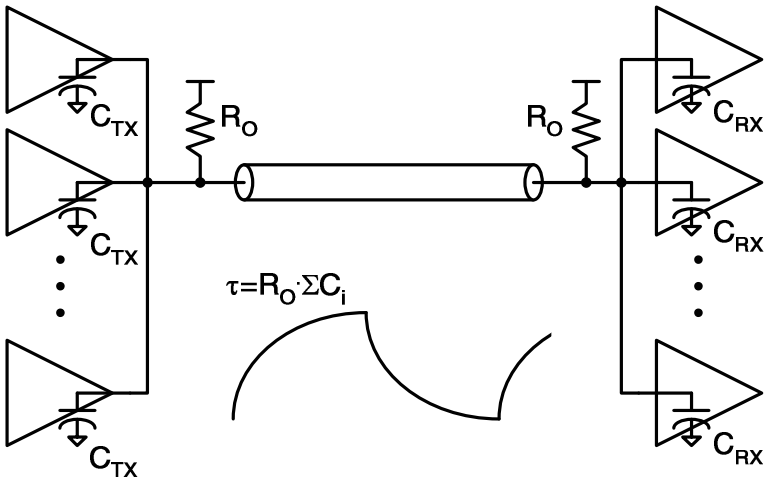


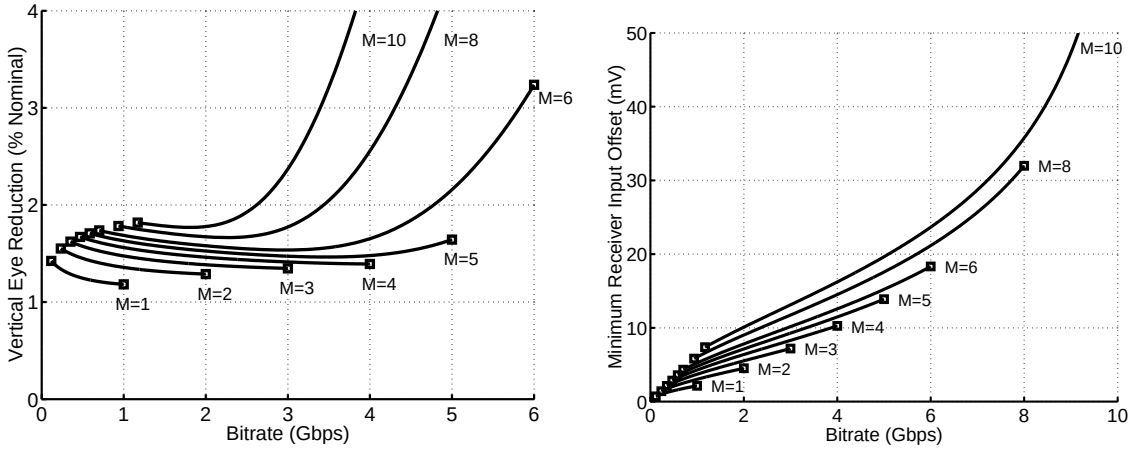
Figure 5.8: Signal attenuation due to the increased loading of transmitters and receivers

and narrow the eye opening.

Figure 5.8 illustrates how capacitive loading can limit the maximum frequency of the signal which travels into and out of the channel. The signalling rate is limited by the RC time constant, where R is the channel impedance and C is the total capacitance at the terminal [37],[10]. Simply downsizing the transmitter and receiver to keep the loading small would aggravate the mismatch issues and make it difficult to achieve higher bitrates by using more parallelism or more advanced process.

Figure 5.9(a) plots the percentage reduction of the transmitter eye. In this case, the size of each transmitter branch is fixed so that the transmitter has a constant swing at each supply voltage. As M increases, the total output loading increases proportionally, and the transmitter eye is reduced by the increased RC time constant as well as by the increased probability of mismatch. For a given M, the eye opening is generally worse at lower bitrates because Vth-mismatch causes larger offsets at lower supplies. However, for large values of M, the transmitter eye degrades also at high bitrates as the bitrate exceeds the limit imposed by the output loading.

Figure 5.9(b) plots the input receiver offset versus bitrate and M. For each bitrate and M, the receiver is sized as large as possible while meeting the constraint of $RC < 0.25\cdot$bit time. A fixed capacitance of 0.5pF is assumed at the receiver input to account for the loading of electrostatic discharge (ESD) protection circuits, pin and pad parasitics, etc.



(a) (b)

Figure 5.9: Voltage margin degradation due to transistor mismatch and increased loading: (a) transmitter eye reduction for a fixed-sized transmitter branch, (b) minimum receiver offset constrained by the total capacitive loading at its input

The plot illustrates that the receiver offsets are significantly increased at M above 6, due to the small receiver size constrained by the input capacitive loading.

5.2.3 Impacts of Transistor Mismatch on Energy-per-Bitrate Scaling

Figure 5.10 plots the scaling trend of the energy-per-bitrate metric for various multiplexing rates and for two different process technologies: 0.25μm and 0.05μm. The power dissipation of each design point is extrapolated from the experimental chip results and scaled properly to meet the specification of 15% timing and 10% voltage margin degradation. Although the degradation of signal quality due to transistor mismatch can be recovered by upsizing the devices, it comes at a price of increased power and area. Ideal equalization is assumed to set the analysis apart from the timing and voltage degradation due to intersymbol interference (ISI).

Figure 5.10 suggests that device mismatches shift the optimal supply for the lowest energy-per-bitrate to a higher voltage as a higher multiplexing rate M is used. It is because poor matching at low voltages depreciates the power savings. In a 0.05μm process, the transistor matching becomes so poor that the energy-per-bitrate ratio favors the highest supply at all ranges of M.

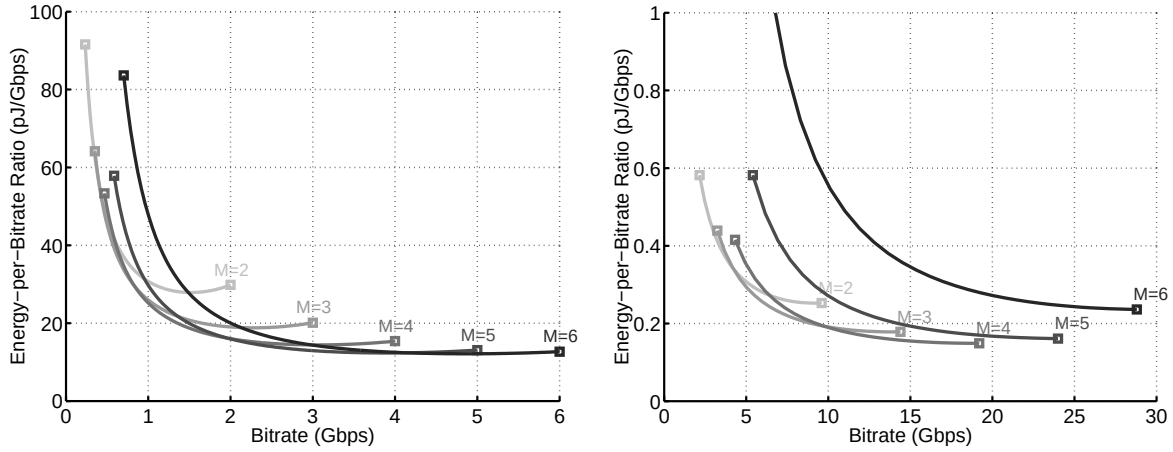


Figure 5.10: Energy-per-bitrate metric versus the multiplexing rate M and the bitrate: (a) in 0.25μm technology, (b) in 0.05μm technology

The increased effects of transistor mismatch with a higher M limits the power savings achievable by using a more parallel architecture. In a 0.25μm process, the optimal M is about 4 or 5 depending on the bitrate, but in a 0.05μm process, the optimal M decreases down to 3 or 4.

The energy-per-bitrate scaling in Figure 5.10 suggests that adjusting the device size only is not an effective way of addressing the mismatch issues and may significantly reduce the power saving benefits of the low-voltage, parallel operation. The next section lists some examples of more effective ways to alleviate the mismatch problems.

5.3 Related Works

Methods of reducing the static phase offsets of the clocks have already been proposed in literature. Offset cancellation can be done either statically during the calibration period [75] or dynamically using continuous adjustment [55]. However, reducing the overhead of the circuits that monitor the offset and adjust the phase spacings still remains a challenge. Some voltage offset-cancellation techniques in state-of-the-art A/D and D/A converters may have the potential of being applied in the time domain.

The issues of the mismatch between the parallel branches of the transmitter and the

bandwidth limitations imposed by the increased loading have also been addressed. Equalization [86] is now a common technique for links to overcome limited bandwidth due to parasitic loading as well as that of the channel itself. Recognizing the difference among the transmitter branches, the framework of applying different equalization coefficients and different amplitudes for each branch has been developed [87]. Moving the multiplexing point to an on-chip node is also an effective way of reducing the transmitter output loading [48].

Canceling receiver input offsets is a topic extensively studied in A/D converters. The high-frequency but low-resolution requirement of high-speed links favors the flash architectures with trimmable offsets [88],[48]. The ability to compensate the offsets allows the use of smaller-sized devices which significantly reduce the loading at the receiver input.

5.4 Summary

The experimental adaptive-supply serial link chip has proved that low-voltage, parallel operation of the link can greatly reduce power dissipation. At its optimal point, the adaptive-supply link achieved a bitrate of 3.1 Gbps with only 113mW of power. However, the low-voltage, parallel operation may become less viable as the effects of device mismatch deteriorate at lower voltage and with higher degree of parallelism. Fortunately, a number of options are available to preserve its effectiveness.

Chapter 6

Conclusions

The proposed adaptive-supply serial link has demonstrated that adaptive power-supply regulation and the use of parallelism are effective in reducing the power dissipation and increasing the performance of the link. Low power dissipation as well as high bitrate is critical to enabling the large-scale integration of the links and meeting the communication bandwidth requirement of the IC chip.

To maximize power savings, the delay of the on-chip circuit must be continuously monitored and the supply voltage must be regulated to the minimum required via a feedback control loop. Chapter 2 presented a digital sliding controller to stabilize such a feedback regulation loop that employs a buck converter for efficient voltage conversion. This sliding controller can operate at the variable supply and frequency so that its power scales proportionally with the load power. The fabricated test chip demonstrated a power efficiency of 89-95% while delivering 23-155mA of load current. Although analog sliding control is widely used in switching supplies for its fast transient and robust stability, an alternative form of the sliding control law is necessary to utilize the more accurate, noise-free estimate of the derivative function.

The adaptive power-supply regulator is essentially a frequency regulation loop that dynamically controls the on-chip circuit performance and therefore the predictability and scalability of the circuit speed can be exploited throughout the link circuit design. For example, the reference circuit for the adaptive power-supply regulator is an inverter-based VCO, which was replicated in the local PLLs so that their frequencies can be coarse-tuned

to the reference frequency by the adaptive supply. This relatively small initial frequency error allowed the PLL to trade VCO tuning range for smaller loop filter area and simpler frequency acquisition aids. In addition, the adaptive supply which tracks the operating frequency served as a bias voltage that adjusts the current of the charge pump and the proportional gain of the PLL to scale its loop bandwidth proportionally with the frequency. Similarly, the bandwidth of the delay interpolator and slew rate of the transmitter are also scaled simply by using inverters operating off of the adaptive supply. As a side benefit, many analog circuits with feedback biasing loops were replaced with inverter-based circuits operating off of the adaptive supply. Not only the circuits are simplified, but also their power becomes more likely to scale as V^2f, justifying the application of adaptive power-supply regulation in the first place. Note that the dynamic voltage scaling schemes with a few preset voltage levels [23] cannot fully benefit from this predictable and scalable circuit performance because the die-to-die and wafer-to-wafer variabilities still need to be accounted.

There still remain some analog circuits in the link that require more voltage headroom than the digital circuits and Chapter 3 presented circuit techniques to extend the minimum supply limits of the parallelized transmitters and receivers. For example, in the transmitter, the predriver performed level-shifting to mitigate the issues stemming from the threshold voltage of the driver. In the receiver, the integrating front-end avoided using sample-and-hold MOS switches which require large voltage headroom. The comparator used subthreshold charge-steering to accommodate low input common-modes while operating at low supplies. Among these techniques, the level-shifting predriver incurred the most power and area overhead. These overheads could have been avoided if a zero-Vth device was supported or if an nMOS driver was used and the lower supply Vss was adjusted instead of Vdd. Since a large portion of the additional power was dissipated in switching the capacitor that generates a shifted voltage, using an explicit supply can be another alternative. The necessity of low-voltage operation driven by the continued scaling of Vdd relative to Vth will eventually force link designers to face these challenges even if their links operate at the nominal supply.

The per-pin timing circuits also utilized the adaptive power-supply regulator to reduce the power and area overheads of locally generating multiphase clocks for the multiplexing

transmitter and receiver. A local timing circuit dedicated to each pin is necessary to recover the clock from a different source and to minimize the static phase offsets and clock jitter by keeping the clock distribution path short. Chapter 4 presented a dual-loop architecture that uses the adaptive power-supply regulator as the global frequency regulation loop. As mentioned above, the narrowed range of expected VCO frequency helped reduce the PLL area and simplified frequency acquisition. The adaptive supply serves as a global bias that optimally scales the loop dynamics with the operating frequency as well as minimizing the power. Although similar techniques have been previously demonstrated [59],[89], the global loop incurs little overhead for the adaptive-supply link because the adaptive power-supply regulator is reused. Chapter 4 also discussed the design issues of bangbang-controlled clock recovery and compared the implementations of the PLL-based and DLL-based clock recovery. It is found that the ease of adjusting multiphase clock timings makes the PLL-based clock recovery preferable to the DLL counterpart.

Mismatch between parallel circuits will limit further exploitation of low-voltage, parallel operation to achieve even lower energy-per-bitrate ratios. Unless these mismatches are properly compensated, the timing margin degradation due to static phase offsets and jitter, and the voltage margin degradation due to non-uniform transmitter output swing and receiver offsets will shift the optimal energy-per-bitrate point toward higher voltage and less parallel design. The analysis in Chapter 5 projected that the optimal multiplexing rate will decrease from 4-5 in a 0.25μm process to 3-4 in a 0.05μ process if only sizes are adjusted to reduce the mismatch. Fortunately, there are abundant possibilities to reduce mismatch both in time and voltage domains, keeping the approaches of adaptive-supply links promising.

Appendix A

Steady-State Analysis of Sliding Control

Here the equations Eq(2.1) for the voltage ripple ΔV_{pp} and the switching frequency f_{sw} are derived via the phase-portrait analysis. Figure A.1 shows the phase portrait of the sliding controller when the feedback loop is in lock. Hysteresis in the comparator results in two decision boundaries shifted by $+\Delta$ and $-\Delta$ for $u = Vdd$ and $u = 0$, respectively. Between these two lines the feedback loop forms a limit cycle. The larger the hysteresis, the longer the limit cycle. Once the limit cycle is formed, the voltage ripple ΔV_{pp} corresponds to the peak-to-peak variation in V and the buck converter's switching period $1/f_{sw}$ corresponds to the time period of the limit cycle. Therefore, ΔV_{pp} and f_{sw} are both functions of the hysteresis parameter Δ.

First, let's derive the equation of the buck converter dynamics which determines the phase-portraits in Figure A.1. In the linear network model of the buck converter shown in Figure A.2, L and C are the inductance and the capacitance of the resonant filter, respectively, R_o is the effective series resistance of the switch transistor, and R_L is the average load resistance. From the Kirchhoff's Current Law, the current flowing through the inductor, I, is expressed as:

$$I = C\frac{dV}{dt} + \frac{V}{R_L}. \tag{A.1}$$

And from the Kirchhoff's Voltage Law, the potential difference between u and V is:

$$u - V = I \cdot R_o + L\frac{dI}{dt}. \tag{A.2}$$

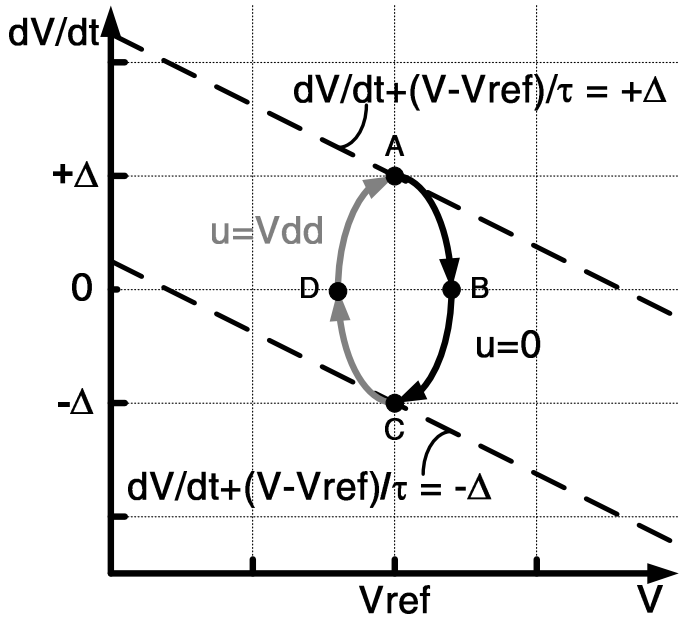


Figure A.1: Phase portraits of the steady-state limit-cycle

It is desirable for the voltage regulator to have small disturbance on its output V and high efficiency in supplying the voltage. A well-designed buck regulator achieves over 90% efficiency and has voltage ripple less than 5%. For high efficiencies, the energy loss due to the series resistance R_o is minimized and therefore we can ignore the term R_o for this analysis. Since the voltage ripple is small, we can assume that the voltage V is fairly constant at V_{ref}.

With these assumptions, the equations Eq(A.1) and Eq(A.2) can be approximated as:

$$\begin{aligned} I &\cong C\frac{dV}{dt} + \frac{V_{ref}}{R_L}, \\ u - V &\cong L\frac{dI}{dt} = LC\frac{d^2V}{dt^2} = \frac{1}{\omega_n^2} \cdot \frac{d^2V}{dt^2}, \end{aligned} \tag{A.3}$$

where ω_n is the resonant frequency of the buck converter, defined as $1/\sqrt{LC}$. The final expression

$$\frac{1}{\omega_n^2} \cdot \frac{d^2V}{dt^2} = u - V \tag{A.4}$$

is the dynamic equation of the buck converter.

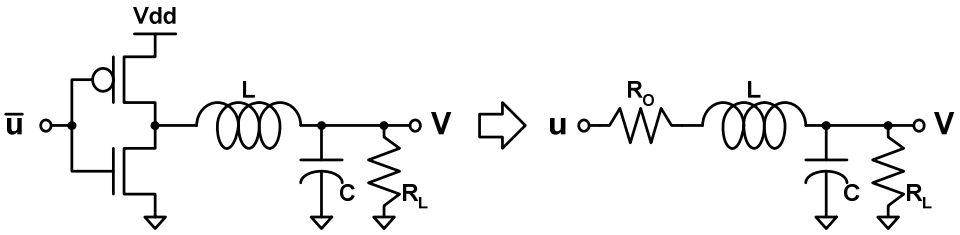


Figure A.2: Linearized model of the buck converter

From this equation, one can derive the geometric relationship between the two state variables V and dV/dt and find the equations for the phase-portrait curves shown in Figure A.1. Substituting $\frac{d^2V}{dt^2}$ with $\frac{d}{dV}(\frac{dV}{dt}) \cdot \frac{dV}{dt}$ and integrating both sides of Eq(A.4) results in the equation of an ellipse, expressed as:

$$\frac{1}{\omega_n^2}\left(\frac{dV}{dt}\right)^2 + (u - V)^2 = C, \tag{A.5}$$

where C is a constant determined by the initial condition. Applying the boundary conditions $dV/dt + (V - V_{ref})/\tau = +\Delta$ and $dV/dt + (V - V_{ref})/\tau = -\Delta$ gives the steady-state phase-portrait as a set of two partial ellipses that meet at the points A$(V_{ref}, +\Delta)$ and C$(V_{ref}, -\Delta)$:

$$\begin{array}{rll} \frac{1}{\omega_n^2}\left(\frac{dV}{dt}\right)^2 + V^2 & = \left(\frac{\Delta}{\omega_n}\right)^2 + V_{ref}^2 & \text{when } u = 0 \ (V \geq V_{ref}), \\ \frac{1}{\omega_n^2}\left(\frac{dV}{dt}\right)^2 + (Vdd - V)^2 & = \left(\frac{\Delta}{\omega_n}\right)^2 + (Vdd - V_{ref})^2 & \text{when } u = Vdd \ (V \leq V_{ref}). \end{array} \tag{A.6}$$

Here it is assumed that V_{ref} is not very close to either rail of the supplies, Vdd or 0. If V_{ref} is close to one of the supplies, u will seldom switch to the other supply. This is not an interesting case anyway since there is no point in converting one supply voltage to another with the same value.

From Eq(A.6), the peak-to-peak voltage ripple ΔV_{pp} can be found. The distance on the V-axis between the two points with $dV/dt = 0$ is:

$$\Delta V_{pp} = V_{max} - V_{min} = \sqrt{\left(\frac{\Delta}{\omega_n}\right)^2 + V_{ref}^2} + \sqrt{\left(\frac{\Delta}{\omega_n}\right)^2 + (Vdd - V_{ref})^2} - Vdd. \tag{A.7}$$

This equation indicates that ΔV_{pp} is minimum at $V_{ref} = Vdd/2$ and increases as V_{ref} gets closer to either Vdd or 0. As V_{ref} approaches Vdd or 0, the voltage ripple ΔV_{pp} reaches the upperbound of approximately Δ/ω_n if $\Delta/\omega_n \ll Vdd$. As the term Δ/ω_n appears frequently, we will define a new parameter $V_H \equiv \Delta/\omega_n$. V_H is the normalized hysteresis parameter Δ with respect to the resonant frequency ω_n. V_H has the dimension of voltage and is equal to the worst-case voltage ripple of the buck regulator with the design parameters of ω_n and Δ.

If V_{ref} is sufficiently far from either supply rails, or quantitatively, $V_H \ll V_{ref} \ll Vdd - V_H$, then Eq(A.7) can be approximated in a simpler form:

$$\Delta V_{pp} \cong \frac{V_H^2}{2} \cdot \frac{Vdd}{V_{ref} \cdot (Vdd - V_{ref})}. \tag{A.8}$$

using the formula $\sqrt{1+x} \cong 1 + \frac{1}{2}x$ for $x \ll 1$.

The peak-to-peak current ripple ΔI_{pp} can be derived directly from the phase-portrait in Figure A.1. Recall that the phase portrait is a set of two partial ellipses that meet at A$(V_{ref}, +\Delta)$ and C$(V_{ref}, -\Delta)$. The peak-to-peak variation of dV/dt is therefore 2Δ and since $I \cong C\frac{dV}{dt} + \frac{V_{ref}}{R_L}$, the current ripple ΔI_{pp} is simply:

$$\Delta I_{pp} = 2C\Delta. \tag{A.9}$$

Finally, the switching frequency f_{sw} is calculated from the time that the system state makes one round-trip along the limit cycle in Figure A.1. The elapsed time while $u = 0$, from the point A to C through B, is:

$$\begin{aligned} T_{u=0} &= \int_{A\to B\to C} \left(\frac{dI}{dt}\right)^{-1} \cdot dI \\ &= \int_{A\to B\to C} \left(C\frac{d^2V}{dt^2}\right)^{-1} \cdot dI \\ &\cong \int_{A\to B\to C} \left(-\frac{V_{ref}}{L}\right)^{-1} \cdot dI \\ &= \left(-\frac{L}{V_{ref}}\right) \cdot (-2C\Delta) \\ &= \frac{2\Delta}{\omega_n^2} \cdot \frac{1}{V_{ref}}. \end{aligned} \tag{A.10}$$

Similarly, the elapsed time while $u = Vdd$, from the point C to A through D, is:

$$\begin{aligned} T_{u=Vdd} &= \int_{C\to D\to A} \left(\frac{dI}{dt}\right)^{-1} \cdot dI \\ &= \frac{2\Delta}{\omega_n^2} \cdot \frac{1}{Vdd - V_{ref}}. \end{aligned} \tag{A.11}$$

Then the expression for the switching frequency f_{sw} is:

$$\begin{aligned} f_{sw} &= \frac{1}{T_{u=0} + T_{u=Vdd}} \\ &\cong \frac{1}{\frac{2\Delta}{\omega_n^2} \cdot \frac{1}{V_{ref}} + \frac{2\Delta}{\omega_n^2} \cdot \frac{1}{Vdd - V_{ref}}} \\ &= \frac{\omega_n^2}{2\Delta} \cdot \frac{V_{ref} \cdot (Vdd - V_{ref})}{Vdd} \\ &= \frac{\omega_n}{2V_H} \cdot \frac{V_{ref} \cdot (Vdd - V_{ref})}{Vdd}. \end{aligned} \tag{A.12}$$

Contrary to the voltage ripple case, the switching frequency is the highest at $Vdd/2$ and decreases as V_{ref} approaches to the supply rails, either Vdd or 0. It is interesting to note that there is a simple relationship between the switching frequency and the voltage ripple derived from Eq(A.8) and Eq(A.12):

$$f_{sw} \cdot \Delta V_{pp} \cong \frac{\Delta}{4}. \tag{A.13}$$

Appendix B

CMOS Adaptive-Bandwidth PLL/DLL

Adaptive-bandwidth phase-locked loops (PLL) or delay-locked loops (DLL) refer to a class of PLLs and DLLs that scale their loop dynamics proportionally with the reference frequency [61],[62]. For example, in a linear PLL [78], an adaptive-bandwidth PLL maintains a constant ratio between the loop bandwidth and the reference frequency and keeps the damping factor constant regardless of process, temperature, and voltage variations.

An adaptive bandwidth helps sustain the optimal performance of the PLL or DLL over all operating range and against process, temperature, and voltage variations. A phase-locked loop is inherently a sampled-data system and the loop bandwidth must be at least a decade below the reference frequency in order to avoid instability due to the sampling delay. For a PLL with a fixed bandwidth, its bandwidth must be set to a decade below the lowest possible operating frequency. Moreover, considering the variations in process, temperature, and voltage, one must spare enough margin to ensure stability at all operating points. As a result, the PLL achieves the desired performance only at one particular operating point, and at all other points, the PLL is suboptimal. On the other hand, the PLL that adapts its bandwidth for different operating points guarantees the optimal performance at all cases.

This appendix starts with a general model of the adaptive-bandwidth PLL/DLL and reviews its CMOS implementations published in literature. As CMOS process advances, however, the implementations of adaptive bandwidth will face some limitations in scaling the bandwidth properly. The key requirement for the adaptive bandwidth PLL/DLL will be discussed to help identify those design challenges.

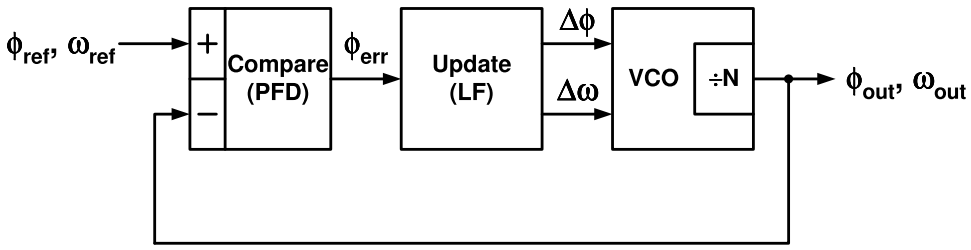


Figure B.1: A general PLL model

B.1 A General Model of the Adaptive-Bandwidth PLL/DLL

Figure B.1 shows an abstract view of a typical second-order PLL. A PLL is a feedback system that tries to match the phase ϕ_{out} and the frequency ω_{out} of the voltage-controlled oscillator (VCO) to those of the reference clock, ϕ_{ref} and ω_{ref}, respectively. In case of frequency multiplication, the outputs ϕ_{out} and ω_{out} are some fraction ($1/N$) of the VCO's direct phase ϕ_{vco} and frequency ω_{vco}. Considering ϕ_{out} and ω_{out} instead of ϕ_{vco} and ω_{vco} avoids the explicit notation of the multiplication factor N and thus keeps the analysis simpler.

A phase/frequency detector (PFD) detects the difference ϕ_{err} between the two phases, ϕ_{ref} and ϕ_{out}. Most phase detectors do this by comparing the positions of the clock edges, either rising or falling, but not both. Therefore, the comparison can be performed only once every cycle and the feedback loop is essentially a discrete-time system that acts upon the phase error ϕ_{err} sampled at the frequency of ω_{ref}.

Once the phase error is detected, a loop filter (LF) makes an appropriate action to reduce the error. In a conventional second-order PLL [78], the loop filter implements an integral control (a pole placed at DC) to suppress the gain-dependent static phase offset and adds a proportional control (a zero on the left-halfplane) to stabilize the resulting loop which has two poles. The sum of these control gains adjust the frequency of the VCO and align the phase.

An alternative but equivalent view of the loop filter is that it updates the phase and the frequency of the VCO by some amount, $\Delta\phi$ and $\Delta\omega$, respectively, every reference cycle period. The integral control can be viewed as the adjustment on the VCO frequency and the proportional control as the adjustment on the phase. The integrating functions provided

by the LF and the VCO imply that these changes $\Delta\phi$ and $\Delta\omega$ accrue on the phase and the frequency of the previous cycle, respectively. This way of interpreting the controls in PLLs provides some new insights in understanding the requirements for the adaptive-bandwidth PLLs as will be seen later.[1]

Given the model in Figure B.1, the adaptive-bandwidth criteria of the constant bandwidth-to-frequency ratio ω_n/ω_{ref} and the constant damping factor ζ can be expressed as the constant gain factors, C_ϕ and C_ω, in the equations below:

$$\begin{aligned} \Delta\phi &= C_\phi \cdot \phi_{err} \\ \Delta\omega/\omega_{ref} &= C_\omega \cdot \phi_{err} \end{aligned} \tag{B.1}$$

Upon the detection of the error ϕ_{err}, an adaptive-bandwidth PLL must adjust its output phase by an amount that is proportional to ϕ_{err} and its output frequency by an amount that is proportional to $\omega_{ref} \cdot \phi_{err}$. This new criteria can be generally applied to any types of PLL, digital or analog, and has more direct implications for implementation than the parameters ω_n and ζ.

The relationship between the constants C_ϕ and C_ω with the conventional parameters ω_n and ζ are:

$$\begin{aligned} C_\phi &= 4\pi \cdot \zeta \cdot \tfrac{\omega_n}{\omega_{ref}}, \\ C_\omega &= 2\pi \cdot \left(\tfrac{\omega_n}{\omega_{ref}}\right)^2. \end{aligned} \tag{B.2}$$

These relationships verify that the constant bandwidth-to-frequency and the constant damping factor requirements are indeed equivalent to the constant C_ϕ and C_ω. C_ϕ of 1.0 and C_ω of 0.05 corresponds to ω_n/ω_{ref} of 0.089 and ζ of 0.89.

Similarly, for the DLL model shown in Figure B.2, the requirement for the constant bandwidth-to-frequency ratio ω_n/ω_{ref} can be expressed in terms of the change in delay ΔD of each comparison cycle:

$$\Delta D = C_D \cdot D_{err}, \tag{B.3}$$

where C_D is $2\pi\frac{\omega_n}{\omega_{ref}}$. At each reference cycle, the output delay is adjusted by ΔD which is

[1]This discrete-time PLL model concerns only with the aggregate control gains over a full comparison cycle period and cannot predict the effects that are due to the different transients within the period, for example, frequency sidebands [90]. Analysis of those effects would require a model which could be nonlinear or time-varying [91].

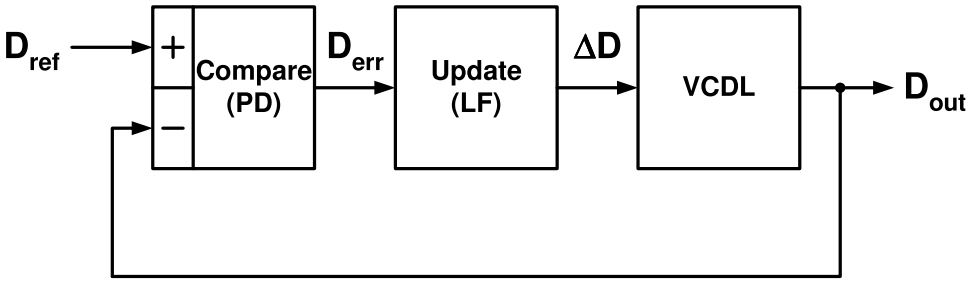


Figure B.2: A general DLL model

proportional only to the error D_{err}. C_D of 0.5 corresponds to ω_n/ω_{ref} of 0.08.

B.2 CMOS Implementations

B.2.1 Charge-Pump PLL/DLL

Currently the most popular loop filter being used in PLLs is a charge pump followed by a passive RC filter, as shown in Figure B.3 [92]. The phase-frequency detector generates pulses whose widths are proportional to the phase error and the charge pump dumps either positive or negative charge onto a capacitor (C) for the duration of those pulses. While inactive, the charge pump has infinite output impedance, which allows a robust realization of pure integration (the pole located at DC). Therefore, the loop gain at DC is ideally infinite and the gain-related static offsets are eliminated[2]. The resistance (R) in series with the capacitor implements the proportional control (the zero). During the charge transfer, the transient current develops a voltage across the resistor whose aggregate effect over a full period is proportional to the present phase error.

By applying Eq(B.1), we can find the adaptive-bandwidth criteria for a charge-pump PLL:

$$\begin{aligned} I_{cp} \cdot R \cdot K_{VCO}/N &= C_\phi \cdot \omega_{ref} \\ I_{cp}/C \cdot K_{VCO}/N &= C_\omega \cdot \omega_{ref}^2 \end{aligned} \tag{B.4}$$

where I_{cp} is the charge-pump current (A), R is the loop filter resistance (Ω), C is the loop

[2]The main causes of the remaining static offsets in charge-pump PLLs are the mismatch between the up and down currents of the charge pump, the mismatch within the phase-frequency detector, and the mismatch between the reference clock path and the feedback clock path.

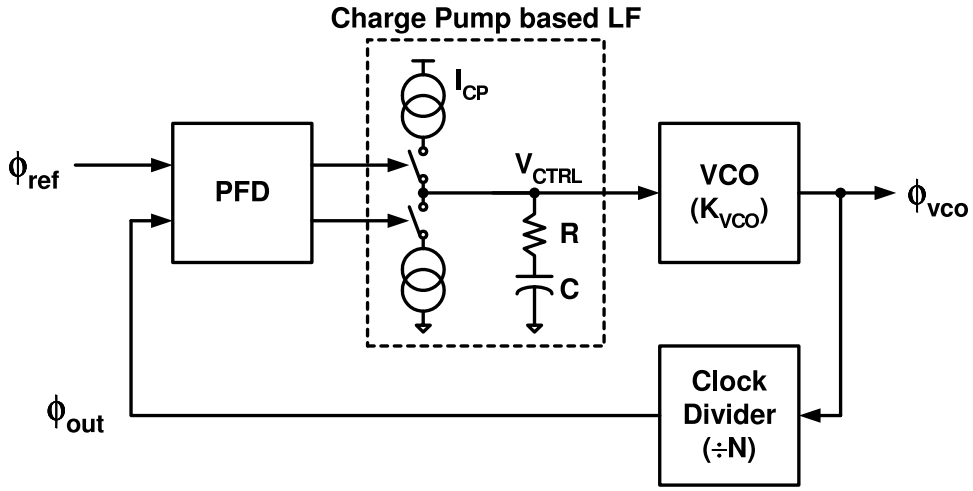


Figure B.3: A charge-pump PLL

filter capacitance (F) , K_{VCO} is the VCO gain (rad/s/V), and N is the dividing ratio of the clock divider.

If the capacitance C is fixed, the second criteria in Eq(B.4) implies that the charge-pump current I_{cp} must scale as below:

$$\begin{aligned} \frac{1}{I_{cp}} &\propto \frac{1}{\omega_{ref}^2} \cdot \frac{K_{VCO}}{N} \\ &= \frac{1}{\omega_{ref}^2} \cdot \frac{1}{N} \frac{\partial \omega_{vco}}{\partial V_{CTRL}} \\ &= \frac{1}{\omega_{out}^2} \cdot \frac{\partial \omega_{out}}{\partial V_{CTRL}} \\ &= -\frac{\partial}{\partial V_{CTRL}} \left(\frac{1}{\omega_{out}}\right) \\ &\propto -\frac{\partial T_{out}}{\partial V_{CTRL}}, \end{aligned} \tag{B.5}$$

where T_{out} is the output clock period which is equal to the reference clock period when locked. And the filter resistance R must scale as:

$$R \propto \frac{1}{\omega_{ref}} \propto T_{ref}. \tag{B.6}$$

Eq(B.5) and Eq(B.6) are therefore the key criteria for a charge-pump PLL to have adaptive bandwidth.

Similarly, in a charge-pump DLL case, the criteria in Eq(B.3) translates to:

$$I_{cp}/C \cdot K_{VCDL} = C_D, \tag{B.7}$$

where K_{VCDL} is the voltage-controlled delay line (VCDL) gain defined as $-\partial D_{out}/\partial V_{CTRL}$[3]. Assuming that the loop capacitance C is at a fixed value, we get a similar criteria to Eq(B.5) for keeping C_D constant:

$$\frac{1}{I_{cp}} \propto -\frac{\partial D_{out}}{\partial V_{CTRL}}. \tag{B.8}$$

The following subsections will then visit two CMOS PLL designs that satisfy these conditions. Both designs rely on the scaling of the VCO gain and the charge pump current with the control voltage V_{CTRL}. Eq(B.5) and Eq(B.6) will help understand why these designs work and what can be a challenge as we stretch these designs to future CMOS processes. Since the criteria for the adaptive-bandwidth DLL are almost identical to those for the adaptive-bandwidth PLL, the same design principles can be applied to the DLL cases as well.

B.2.2 Self-Biased PLLs with Symmetric-Load Buffers

Shown in Figure B.4 is the voltage-controlled oscillator circuit used in the self-biased PLL proposed in [61]. Each buffer is a differential stage with a so-called symmetric load, which is a parallel combination of the diode-connected pMOS device and a near-triode pMOS device. The I-V characteristic of the symmetric load is fairly linear, which provides good rejection of the dynamic supply noise. A replica-feedback biasing circuit dynamically controls the voltage swing and the bias current of the oscillator, against low-frequency supply variation and process/temperature variation. The feedback amplifier adjusts the bias current I_{BIAS} so that the voltage swing V_{SWING} matches the control voltage V_{CTRL}. The half-buffer replicas in the bias generator basically translate I_{BIAS} to V_{SWING} through the diode-connected pMOS devices. The voltages V_{CTRL} and V_{SWING} are referenced to the higher supply rail in Figure B.4.

[3]The negative sign is from the assumption that the delay D_{out} decreases as V_{CTRL} increases, being consistent with the VCO case where the frequency ω_{out} increases as V_{CTRL} increases.

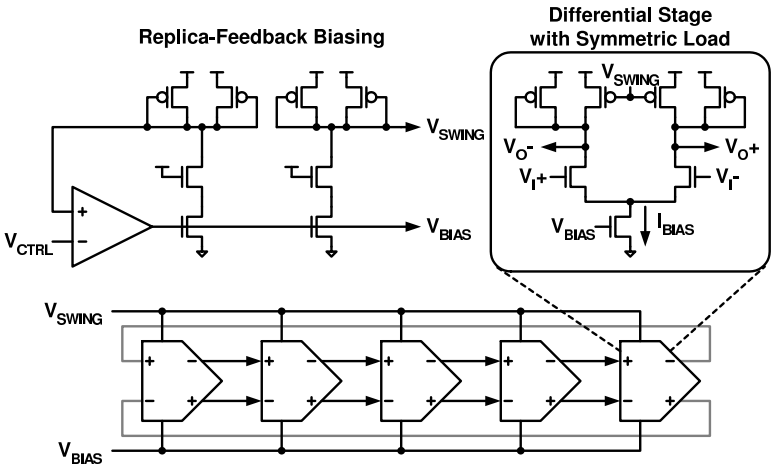


Figure B.4: The voltage-controlled oscillator of the self-biased PLL. Also shown are the differential buffer stage with the symmetric load and the replica-feedback biasing circuit

The self-biased PLL makes use of the fact that the cycle period of this ring oscillator, T_{vco}, can be expressed as:

$$T_{vco} \cong C_b \frac{V_{SWING}}{I_{BIAS}} = C_b \frac{V_{CTRL}}{I_{BIAS}}, \tag{B.9}$$

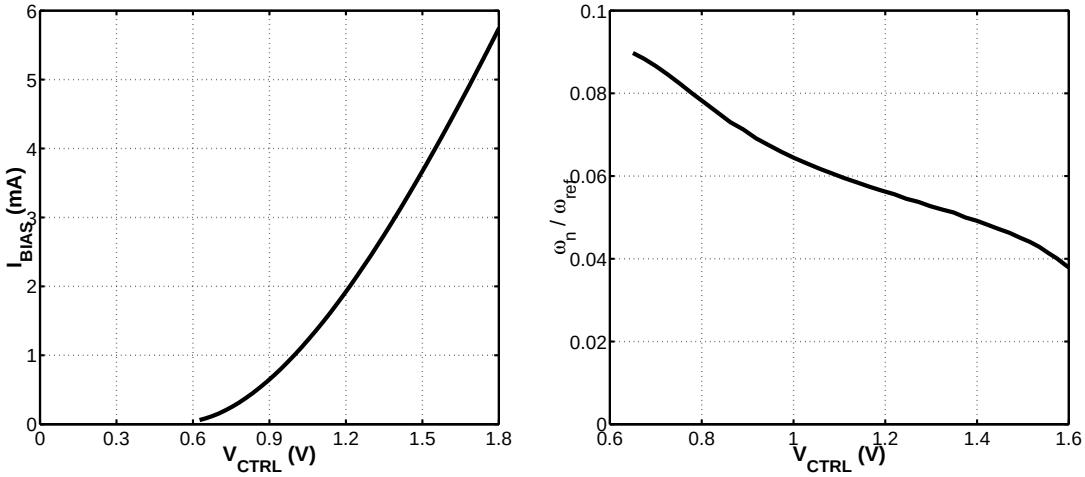
where C_b is the effective load capacitance seen by the buffer stages in the ring. To scale the bandwidth of this PLL with the reference frequency, the charge-pump current I_{cp} must satisfy the condition in Eq(B.5). Combining Eq(B.5) and Eq(B.9) yield the expression for I_{cp}:

$$\begin{aligned} \tfrac{1}{I_{cp}} &\propto -\tfrac{\partial T_{out}}{\partial V_{CTRL}} \\ &= -N \tfrac{\partial T_{vco}}{\partial V_{CTRL}} = -N \tfrac{\partial}{\partial V_{CTRL}} \left(C_b \tfrac{V_{CTRL}}{I_{BIAS}} \right) \\ &\propto \tfrac{N}{I_{BIAS}} \left(\tfrac{\partial I_{BIAS}/I_{BIAS}}{\partial V_{CTRL}/V_{CTRL}} - 1 \right). \end{aligned} \tag{B.10}$$

If the relative transconductance $\frac{\partial I_{BIAS}/I_{BIAS}}{\partial V_{CTRL}/V_{CTRL}}$ can be assumed constant, then the charge-pump current I_{cp} for adaptive bandwidth is simply:

$$I_{cp} \propto \frac{I_{BIAS}}{N}. \tag{B.11}$$

The self-biased PLL in [61] scales the charge-pump current I_{cp} proportionally with the VCO bias current I_{BIAS} by using a charge pump of which circuits are almost identical to



(a) (b)

Figure B.5: The case of the self-biased PLL with symmetric-load buffers: (a) I_{BIAS} versus V_{CTRL}, (b) ω_n/ω_{ref} versus V_{CTRL}

the differential buffer stage and applying the same bias voltage V_{BIAS} to set its current. By varying the device sizes of the charge pump, one can adjust the ratio between I_{cp} and I_{BIAS} and thus the bandwidth-to-frequency ratio, ω_n/ω_{ref}.

Although the relative transconductance is fairly constant at high values of V_{CTRL}, it increases as V_{CTRL} drops near to the threshold voltage of the pMOS transistor, $V_{th,p}$. Figure B.5 plots the bias current I_{BIAS} versus V_{CTRL} and the resulting bandwidth-to-frequency ratio ω_n/ω_{ref} for the self-biased PLL simulated in 0.25-μm CMOS technology with the nominal threshold voltage V_{th} of 0.55V. For low values of V_{CTRL}, the relative transconductance grows high and the charge-pump current I_{cp} scaling simply as I_{BIAS}/N is larger than the desired value from Eq(B.10). As a result, the bandwidth becomes higher than it is desired at low V_{CTRL}'s. On the other hand, for V_{CTRL}'s above 1.5V, the transistors fall out of the saturation region and the circuits do not operate as intended. As the CMOS process scales and as the nominal Vdd decreases more rapidly than V_{th} does, the valid V_{CTRL}-range will become narrower and pose a challenge to PLL designers.

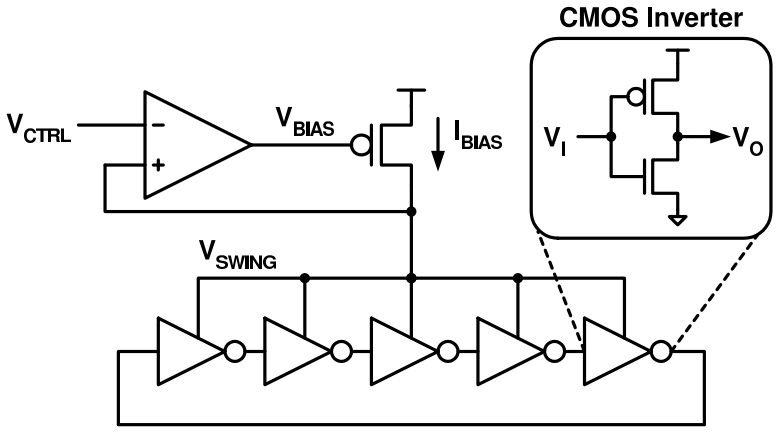
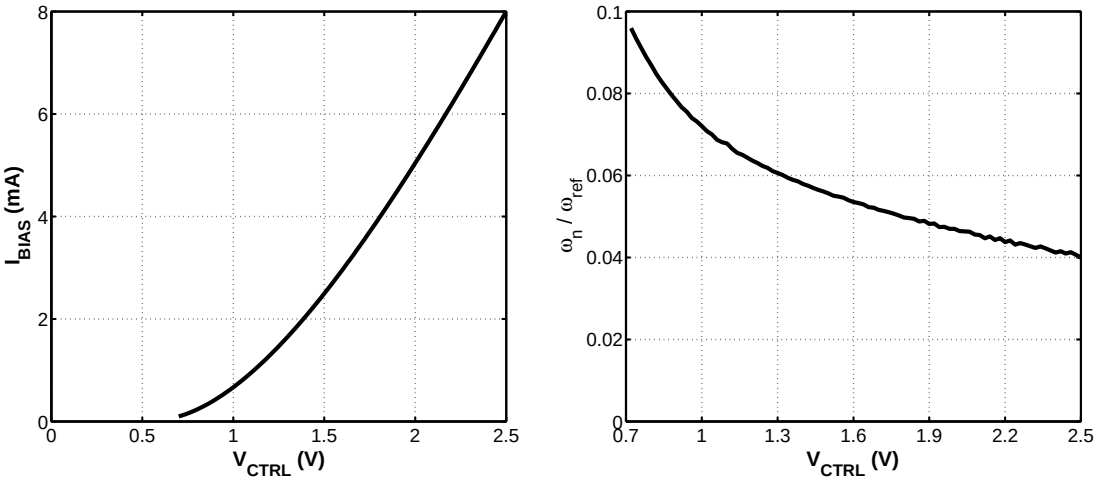


Figure B.6: The inverter-based VCO and its biasing circuit for the regulated-supply PLL

B.2.3 Regulated-Supply PLL/DLLs with CMOS Inverter Stages

The regulated-supply PLL described in [62] operates in similar principles to the self-biased PLL. Figure B.6 illustrates the voltage-controlled oscillator of the regulated-supply PLL and its supporting bias generator. The VCO is made of CMOS inverters and the bias generator is basically a linear voltage regulator that controls the VCO supply to adjust the frequency while suppressing unwanted noise on the VCO supply. Similar to the self-biased PLL case, the control voltage V_{CTRL} sets the desired voltage swing of the oscillator and the bias generator adjusts the bias current I_{BIAS} so that the voltage swing V_{SWING} matches to V_{CTRL}. Then the same equation Eq(B.9) also holds for this inverter-based VCO and the adaptive bandwidth can be achieved by scaling the charge-pump current I_{cp} proportionally with I_{BIAS}. Note that in the regulated-supply PLL in Figure B.6, the voltages V_{CTRL} and V_{SWING} are referenced to ground.

Figure B.7 plots I_{BIAS} versus V_{CTRL} and the normalized bandwidth ω_n/ω_{ref} of the regulated-supply PLL simulated in 0.25-μm CMOS process when I_{cp} scales proportionally with I_{BIAS}. The normalized bandwidth is fairly constant at high V_{CTRL} region, but it gradually increases as V_{CTRL} drops for the same reasons as in the self-biased PLL. The charge-pump current I_{cp} scaling as I_{BIAS}/N becomes higher than desired when the relative transconductance increases at low values of V_{CTRL}. To mitigate this problem, the designers of the regulated-supply PLLs have varied the channel length of the charge pump's current



(a) (b)

Figure B.7: The case of the regulated-supply PLL: (a) I_{BIAS} versus V_{CTRL}, (b) ω_n/ω_{ref} versus V_{CTRL}

source devices to find the I-V characteristics that can compensate the difference.[4]

[4]This section discussed how to scale I_{cp} for adaptive bandwidth, however it did not cover how to scale the loop resistance R as $1/\omega_{ref}$. An elegant way of doing this is to use the proportional control pulse of which amplitude is proportional to I_{cp} and the phase error and of which duration is proportional to the reference clock period. Maneatis et al. [93] describes this type of loop filter in the context of the self-biased PLLs.

Appendix C

Bangbang PLL Design for Clock Recovery

This appendix presents detailed analysis on bangbang-controlled phase-locked loops (PLLs). As mentioned in Chapter 4, a replica receiver is an ideal phase detector for zero timing offsets, but its binary outputs result in a bangbang-controlled, inherently-nonlinear feedback loop. Although its overall loop behavior is still very similar to that of the linear counterpart, a nonlinear PLL has unique properties that cannot be explained by linear control theory. For example, the locked state of a bangbang PLL is a limit cycle, as opposed to a single state. The tracking bandwidth of the bangbang PLL varies depending on the amplitude of the stimulus, while that of the linear PLL does not.

Figure C.1 shows an analytical model of a second-order bangbang PLL. The phase detector monitors the phase error, $\phi_{err} = \phi_{ref} - \phi_{out}$, but detects only its polarity (+1 or -1). The phase detector output is updated every reference clock cycle, $t_{update} = 2\pi/\omega_{ref}$.[1] Then this phase detector output u controls the output frequency ω_{out} and output phase ϕ_{out} through two paths: the proportional path and the integral path. The proportional path sets the frequency ω_p to either $+\omega_{bb}$ or $-\omega_{bb}$ based on the current value of u only. The integral path, on the other hand, sets the frequency ω_i to the time-integral of the past values of u

[1] In fact, it is more common that the phase detector updates every VCO cycle, but the non-uniform sampling interval makes the analysis very difficult. Fortunately, the effects due to this non-uniform sampling are usually secondary.

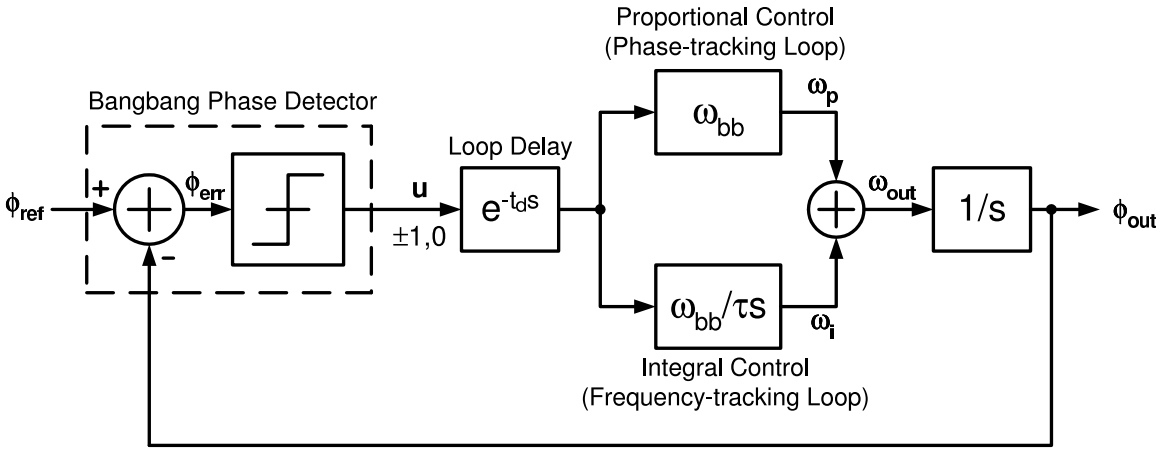


Figure C.1: Analytical model of a second-order bangbang PLL

scaled by ω_{bb}/τ. The output frequency ω_{out} is then the sum of two frequencies, ω_p and ω_i, and the time-integral of ω_{out} becomes the output phase ϕ_{out}.

For each phase error detected, the proportional path makes a phase step of $\omega_{bb} \cdot t_{update}$ and the integral path makes a frequency step of $\omega_{bb}/\tau \cdot t_{update}$, either up or down. In this sense, the proportional path of the PLL is also called the phase-tracking loop and the integral path is called the frequency-tracking loop [76].

In most PLLs, there is a time delay between when the PLL detects the error and when it reacts to it. Possible causes are the delay of the phase detector and the delay of the clock buffers. The transfer function $e^{-t_d s}$ in Figure C.1 models this loop delay of t_d seconds.

This appendix first analyzes the locked behavior of the bangbang PLL. Next, it discusses the amplitude-dependent tracking behavior of the PLL, followed by its out-of-lock behavior. Finally, in the case of a multiphase system, different approaches to handle multiple phase detector outputs are compared. Throughout the appendix, the effects of loop delay are extensively analyzed.

C.1 Locked Behavior of a Bangbang PLL

Unlike a linear PLL, the locked state of a bangbang PLL is a limit cycle. The phase and frequency of the VCO dither near the lock point instead of fully converging to a single state. In a strict sense, it means that a bangbang PLL is never stable. Therefore, here we

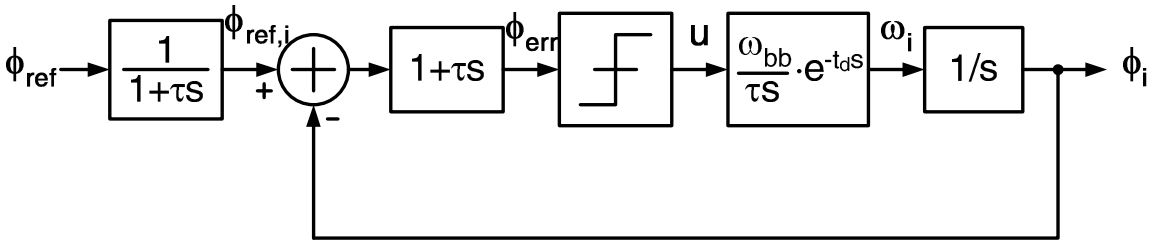


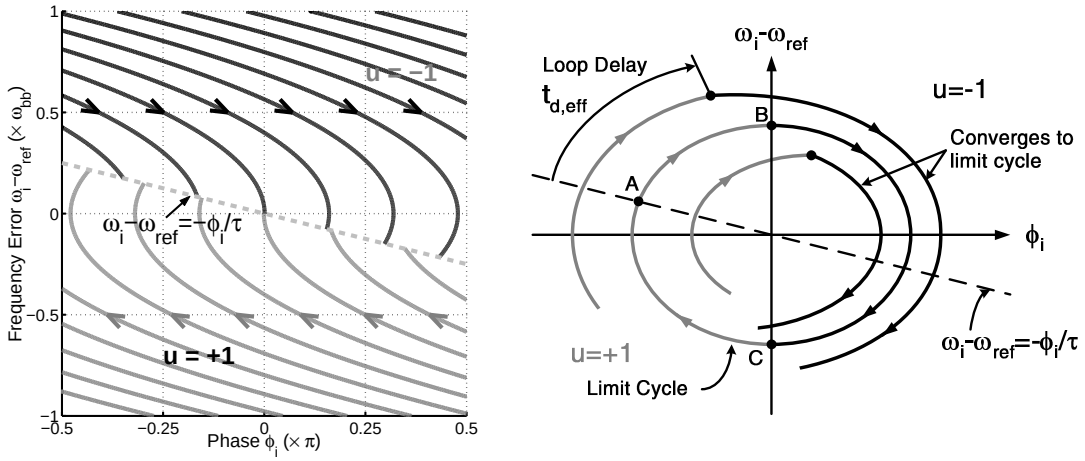
Figure C.2: An equivalent model of a second-order bangbang PLL

use a looser definition of stability, which implies the existence of a bounded limit cycle that the PLL converges to. Besides the existence, the phase-trajectory of the limit cycle is of concern because it determines the amount of clock jitter due to bangbang dithering. This section derives the condition for stability and the expression for the dithering jitter, with the aid of phase-portrait analysis introduced in Chapter 2.

Upon the detection of the phase error ϕ_{err} in Figure C.1, both the phase-tracking loop and the frequency-tracking loop respond to it and update the output phase ϕ_{out}. Therefore, in a locked state, each loop has its own contribution to the total dithering jitter. Our goal is to separate those individual contributions using phase-portrait analysis. Unfortunately, the model shown in Figure C.1 is difficult for hand analysis because the transfer function between u and the output variable ϕ_{out} is rather cumbersome (two poles and one zero).

The equivalent model shown in Figure C.2 is easier for performing hand analysis and separating the two contributions. The output variable is changed to the VCO output phase solely due to the integral control, $\phi_i(s) = \omega_i(s)/s$, and thus the transfer function between u and the output ϕ_i is simpler with only two poles at DC. The functions $\frac{1}{1+\tau s}$ and $1 + \tau s$ are inserted before the phase detector to keep equivalence with the previous model. The resulting model is in fact in the form of sliding control introduced in Chapter 2, and a similar nonlinear analysis can be applied to derive the limit cycle of the frequency-tracking loop, i.e. the locked behavior of the integral control variables ω_i and ϕ_i. Once the limit cycles of ω_i and ϕ_i are found, we can derive the limit cycles of the other variables such as ω_p and ϕ_{out} using the relations, $\omega_p(s) = \tau s \cdot \omega_i(s)$ and $\phi_{out}(s) = (1 + \tau s)\phi_i(s)$.

Based on this equivalent model, the phase portraits of the second-order bangbang PLL are drawn in Figure C.3(a). The state variables are ϕ_i and ω_i, which are chosen as the x-axis and y-axis, respectively. The phase detector makes decision on u based on the polarity



(a) (b)

Figure C.3: Limit cycle of the frequency-tracking loop: (a) phase portraits, (b) convergence to a limit cycle

of the phase error, ϕ_{err}, which is equal to $(1+\tau s)(\phi_{ref,i}-\phi_i) = -(\phi_i + \tau(\omega_i - \omega_{ref}))$, assuming an ideal reference clock for the sake of stability analysis. On the state-space, this decision criteria corresponds to a linear boundary, expressed as $\omega_i - \omega_{ref} = -\phi_i/\tau$. The phase detector output u is $+1$ if the state (ϕ_i, ω_i) is located below the boundary and -1 if it is above the boundary. Within each region divided by this boundary line, the state follows a parabolic trajectory, expressed as:

$$(\omega_i - \omega_{ref})^2 = \frac{2\omega_{bb}}{\tau} \cdot u(t-t_d)\phi_i + C_o, \tag{C.1}$$

where C_o is a constant determined by the initial condition [43]. Figure C.3(a) plots the phase portraits with various values of C_o.

Eq(C.1) has a term $u(t-t_d)$ in it, implying that the updated loop response will not take effect until the loop delay t_d has elapsed. In the meantime, the phase portrait of one region will persist even if it crosses to the other region, and this extended trajectory due to loop delay will eventually set the limit cycle, as illustrated in Figure C.3(b). When the initial trajectory is too small a parabola, the loop delay causes the state to cross the y-axis, $\phi_i = 0$, before the switched phase detector output u comes into effect. In this case, the trajectory diverges to a larger parabola. Similarly, when the initial trajectory is too large

a parabola, the switch happens before the state reaches the y-axis, leading to a smaller parabola. Therefore, the state converges to a trajectory of two symmetric parabolas with their ends just meeting at the y-axis. Ideally, the traversal time of this trajectory between the boundary-crossing point (point A) and the y-axis crossing point (point B) should be equal to the loop delay t_d.

In reality, however, the limit-cycle trajectories look more complicated than the one shown in Figure C.3(b) for the following reasons. First, the discrete sampling period of the PLL effectively increases the loop delay. For example, the phase detector cannot detect the change in phase error polarity until the next sampling instant. Thus, the loop delay can be increased by as much as one sampling period, t_{update}. Second, the PLL updates its frequency and phase in discrete steps. The desired lock point may not lie exactly on one of these discrete levels, and in this case, the PLL will dither between the two adjacent levels. The dithering pattern and frequency will differ depending on the relative position of the lock point with respect to the discrete grid of the PLL phase and frequency. Third, the loop delays that are not exact multiples of t_{update} increases the randomness of the locked behavior. A fractional loop delay causes the discrete levels of phase and frequency to move over time, and thus causes the dithering pattern and frequency to change over time. With all these effects combined, a bangbang PLL in a locked state can exhibit a complicated, aperiodic behavior that varies strongly with the initial conditions of the PLL and that is very difficult to analyze.

Although analyzing the limit-cycle trajectories directly seems difficult, we can find the bounds of the limit-cycle trajectories to characterize the maximum deviation in the state variables, e.g. the peak-to-peak clock jitter. In fact, it can be shown that the limit-cycle trajectories are bounded by the two symmetric parabolas discussed above for an ideal case except with the increased loop delay, $t_d + t_{update}$. As this quantity appears frequently, we call it the *effective loop delay* and define $t_{d,eff}$ as:

$$t_{d,eff} = t_d + t_{update}. \tag{C.2}$$

When the bangbang PLL is in its locked state, the state (ϕ_i, ω_i) always lie within the bounded region defined by the two parabolic curves. Since these bounds are tight, we

can derive the worst-case peak-to-peak variations in the state variables ϕ_i and ω_i, exploiting the geometry of this bounded region. The rest of this section derives the analytical expression for the dithering jitter as well as the condition for the bangbang PLL stability.

From symmetry, the expressions for these parabolic boundaries are:

$$\begin{array}{ll} (\omega_i - \omega_{ref})^2 & = \frac{2\omega_{bb}}{\tau} \cdot \left(\phi_i + \frac{\Delta\phi_{i,pp}}{2}\right) \quad \text{for } \phi_i \leq 0 \\ (\omega_i - \omega_{ref})^2 & = -\frac{2\omega_{bb}}{\tau} \cdot \left(\phi_i - \frac{\Delta\phi_{i,pp}}{2}\right) \quad \text{for } \phi_i \geq 0. \end{array} \tag{C.3}$$

where $\Delta\phi_{i,pp}$ is the peak-to-peak variation in the phase ϕ_i, the output variable of the frequency-tracking loop. To determine $\Delta\phi_{i,pp}$, we apply the condition that the traversal time from point A to point B is equal to $t_{d,eff}$. First, the frequency ω_A at which the parabolic curve and the decision line $\omega_i - \omega_{ref} = -\phi_i/\tau$ meet (point A) is:

$$\omega_A = \omega_{ref} + \omega_{bb}\left(\sqrt{1 + \frac{\Delta\phi_{i,pp}}{\omega_{bb}\tau}} - 1\right). \tag{C.4}$$

And, the frequency ω_B at which the parabola meets the y-axis (point B) is:

$$\omega_B = \omega_{ref} + \sqrt{\frac{\omega_{bb}\Delta\phi_{i,pp}}{\tau}}. \tag{C.5}$$

With the expressions of ω_A and ω_B, the traversal time from point A to point B can be calculated as:

$$\begin{array}{ll} t_{A\to B} & = \int_{A\to B}\left(\frac{d\omega_i}{dt}\right)^{-1} \cdot d\omega_i \\ & = \int_{A\to B}\left(\frac{\omega_{bb}}{\tau}\right)^{-1} \cdot d\omega_i \\ & = \frac{\tau}{\omega_{bb}}(\omega_B - \omega_A) \\ & = \tau\left(1 + \sqrt{\frac{\Delta\phi_{i,pp}}{\omega_{bb}\tau}} - \sqrt{1 + \frac{\Delta\phi_{i,pp}}{\omega_{bb}\tau}}\right), \end{array} \tag{C.6}$$

which should be equal to $t_{d,eff}$. Solving this equation yields the expression for the peak-to-peak variation in ϕ_i:

$$\Delta\phi_{i,pp} = \frac{\omega_{bb}\tau}{4} \cdot \left(\frac{2\kappa - 1}{\kappa(\kappa - 1)}\right)^2, \tag{C.7}$$

where κ is defined as $\tau/t_{d,eff}$. The solution for $\Delta\phi_{i,pp}$ exists only when $\kappa > 1$, thus the condition for stability.

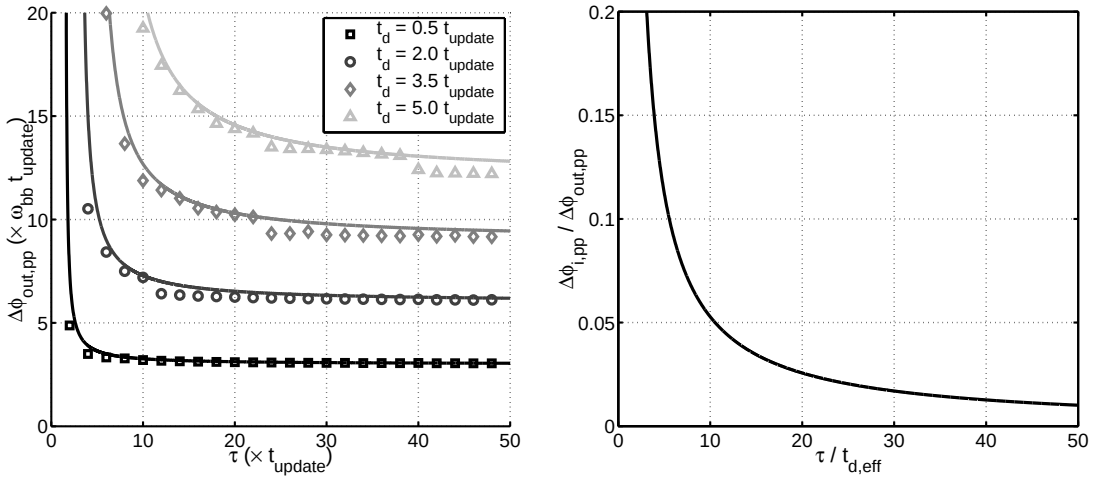


Figure C.4: Dithering jitter versus loop parameters: (a) total dithering jitter $\Delta\phi_{out,pp}$ versus τ with various loop delays, (b) ratio between two dithering jitters as a function of $\kappa = \tau/t_{d,eff}$

The geometric expression of the limit-cycle boundary also helps derive the total dithering jitter of the output phase ϕ_{out}. Since ϕ_{out} is equal to $(1+\tau s)\phi_i = \phi_i + \tau\omega_i$, the peak-to-peak variation in the output phase, $\Delta\phi_{out,pp}$, is equal to the peak-to-peak variation in $\phi_i + \tau\omega_i$ within the limit-cycle boundary, or equivalently, the peak-to-peak translation width of the decision line that meets the limit-cycle boundary. For κ greater than $1+1/\sqrt{2}$, the peaks occur at points B and C, and in this case, the worst-case peak-to-peak dithering jitter $\Delta\phi_{out,pp}$ is expressed as:

$$\begin{aligned}
\Delta\phi_{out,pp} &= (\phi_B + \tau\omega_B) - (\phi_C + \tau\omega_C) \\
&= \tau\cdot(\omega_B - \omega_C) \\
&= \tau\cdot\left(\sqrt{\tfrac{\omega_{bb}\Delta\phi_{i,pp}}{\tau}} - \left(-\sqrt{\tfrac{\omega_{bb}\Delta\phi_{i,pp}}{\tau}}\right)\right) \\
&= 2\sqrt{\omega_{bb}\tau\cdot\Delta\phi_{i,pp}} \\
&= \omega_{bb}\tau\tfrac{2\kappa-1}{\kappa(\kappa-1)} \\
&= \omega_{bb}t_{d,eff}\tfrac{2\kappa-1}{\kappa-1}.
\end{aligned} \tag{C.8}$$

The dithering clock jitter is mainly determined by the bangbang frequency step ω_{bb} and the effective loop delay $t_{d,eff}$. Thus, to reduce the dithering jitter, it is important to keep

the loop delay small and to constrain ω_{bb} accordingly. With κ increasing from 2 to infinity, $\Delta\phi_{out,pp}$ decreases from $3\omega_{bb}t_{d,eff}$ down to $2\omega_{bb}t_{d,eff}$. The factor $\kappa = \frac{\tau}{t_{d,eff}}$ determines the stability of the bangbang PLL and plays a similar role to the damping factor in linear PLLs. Small values of κ indicates that the proportional gain is low compared to the integral gain (underdamped) and the PLL will not converge to a locked state if κ is less than 1 (unstable). In case of a zero loop delay ($t_{d,eff} = t_{update}$), the factor κ reduces to the Walker's stability factor $\xi = 2\tau/t_{update}$, which he derived from the ratio between the proportional phase step ($\omega_{bb}t_{update}$) and the integral phase step ($\omega_{bb}t_{update}^2/2\tau$) [76]. The expressions for κ and $\Delta\phi_{out,pp}$ are more general equations including the loop delay effects.

To verify Eq(C.8), the analytical predictions are compared against the results from numerical simulations [94]. Figure C.4(a) plots the dithering jitter measured from the simulation as points and the predicted values as solid lines, as a function of τ. Since the bangbang PLL has a different behavior depending on the initial condition, we ran multiple simulations with the randomized initial values for ω_i and ϕ_{out}, and recorded the worst-case dithering jitter. The measured results match well with the analytical predictions, verifying that the expression in Eq(C.8) is indeed a tight bound on the dithering clock jitter.

Figure C.4(b) plots the ratio of $\Delta\phi_{i,pp}$ to $\Delta\phi_{out,pp}$, which is a function of κ only. The larger the factor κ, the smaller the dithering contribution of the integral-control loop and the lower the total dithering jitter. To reduce the jitter close to the minimum, $2\omega_{bb}t_{d,eff}$, values of κ more than 20 are appropriate. However, too large κ may slow down the frequency-tracking loop and degrade the noise response, as described in the next section.

C.2 Tracking Behavior of a Bangbang PLL

The previous section described the locked state of a bangbang PLL; this section describes how the PLL responds to a small disturbance on its locked state. For instance, the responses to a drifting reference phase, to noise on the output phase, and to noise that affects the output frequency. General tracking behavior of a bangbang PLL is first discussed and the maximum noise that the PLL can tolerate is derived with the notion of effective bandwidth.

When tracking the variation in the reference phase ϕ_{ref}, the bangbang PLL does not lose phase-lock if the phase drift of each cycle is less than the proportional phase step,

$\omega_{bb}t_{update}$, or equivalently, if the reference phase is drifting away with a relative frequency less than ω_{bb}. The phase-tracking loop can correct this small enough phase error in one step and the tracking phase error ϕ_{err} will be kept within a certain bound, $\Delta\phi_{out,pp}/2+\omega_{bb}t_{update}$. Figure C.5(a) shows the response of a bangbang PLL tracking the sinusoidal variation in the reference phase.

While the phase-tracking loop maintains phase-lock, the frequency-tracking loop adjusts the base frequency ω_i gradually toward the reference frequency ω_{ref}. For example, when the reference frequency makes a step change, initially the phase-tracking loop compensates for the phase drift and causes the average phase detector output u_{AVG} to be:

$$u_{AVG} = -\frac{\omega_i - \omega_{ref}}{\omega_{bb}}. \tag{C.9}$$

This nonzero average of the phase detector output will shift the base frequency ω_i toward the reference frequency and reduce the phase drift for which the phase-tracking loop has to compensate. Therefore, the difference between ω_i and ω_{ref} will decrease exponentially with a time constant τ, as expressed below:

$$\omega_i - \omega_{ref} = (\omega_i - \omega_{ref})|_{t=0} \cdot e^{-t/\tau} \tag{C.10}$$

When the reference phase drifts away at a faster rate than $\omega_{bb}t_{update}$ per cycle, however, the output phase of the bangbang PLL will not be able to move fast enough to keep the locked state. This is called *slope overlimiting* and illustrated in Figure C.5(b). For a sinusoidal variation in the reference phase, $\phi_{ref} = A_N cos\omega_N t$, where A_N is the amplitude of the disturbance in radians and ω_N is its frequency, the condition for a phase-tracking loop to experience slope overlimiting is:

$$\frac{d\phi_{ref}}{dt}|_{max} = A_N\omega_N > \omega_{bb}. \tag{C.11}$$

Interestingly, the tracking bandwidth, i.e. the maximum noise frequency ω_N that the bangbang PLL can track without slope overlimiting, depends on the noise amplitude A_N. This is contrary to linear PLLs which have tracking bandwidths that are independent of the

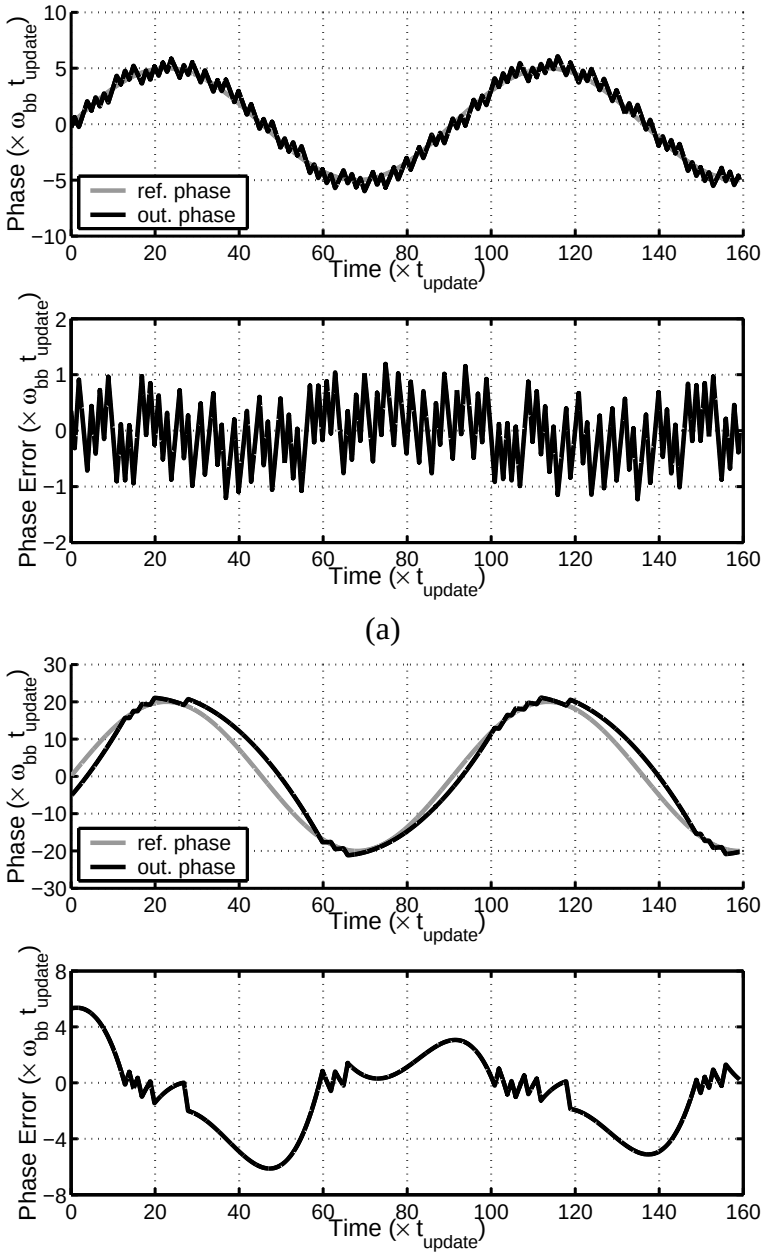
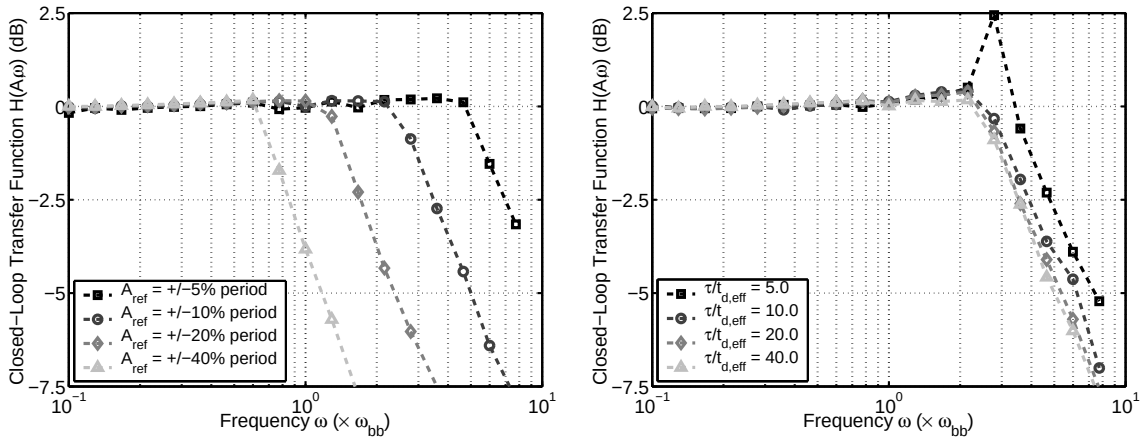


Figure C.5: Bangbang PLL tracking the sinusoidal variation of the reference phase : (a) without slope overlimiting, (b) with slope overlimiting



(a) (b)

Figure C.6: Responses of a bangbang PLL to the sinusoidal variation of the reference phase: (a) with different amplitudes, (b) with different stability factor $\kappa = \tau/t_{d,eff}$

noise amplitude. This unique amplitude-dependent property of the bangbang PLL tracking bandwidth is plotted in Figure C.6(a). The transfer gain of this nonlinear system is approximated as the output amplitude divided by the input noise amplitude at each noise frequency ω_N.[2] If the reference phase varies either with high frequency or with large amplitude, the PLL output phase cannot follow the change to its full extent and the transfer gain is reduced.

The frequency-tracking loop helps extending the tracking bandwidth, but too large integral gain ω_{bb}/τ may cause peaking in the transfer function, as illustrated in Figure C.6(b). When the phase-tracking loop experiences slope-overlimiting, the phase detector output can have a long string of either +1's or -1's, which accumulates the integral-controlled frequency ω_i in one direction. By the time the loop recovers from slope-overlimiting, the accumulated ω_i may become too far from the reference frequency ω_{ref}, especially when the integral gain is too large or the loop delay is too long. In this case, the phase-tracking loop will face another slope-overlimiting, this time because of the frequency error $|\omega_i - \omega_{ref}|$ being larger than ω_{bb}. In other words, the momentum gained by the frequency-tracking loop

[2]Unlike a linear system, the response of a nonlinear system to a sinusoidal input is not an equal-frequency sinusoid with a new amplitude. Therefore, the transfer gain at each frequency cannot be rigorously defined. The approximate transfer gain used here is to illustrate the amplitude-dependent noise response of a bangbang PLL.

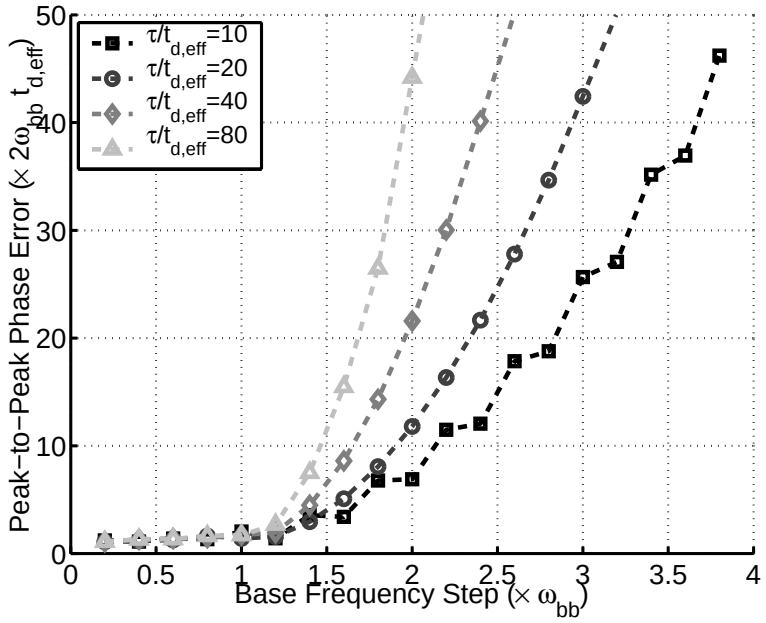


Figure C.7: Response of a bangbang PLL to a step variation in VCO frequency

will make the output phase overshoot and thus induce peaking in the transfer function. The peak appears just above the cut-off bandwidth where the slope-overlimiting lasts for the longest period of time, and it becomes larger for the longer loop delay and the higher integral gain. Therefore, the peaking in the transfer gain is a function of the ratio between the loop delay and the integral gain, $\kappa = \tau/t_{d,eff}$, which has a similar implication to the damping factor in linear PLLs. Figure C.6(b) suggests that κ values of 20 to 40 are reasonable for small peaking and wide bandwidth.

In addition to the reference phase noise, the PLL may also have noise on its output phase ϕ_{out}. The only difference in the PLL response is that the PLL will try to reject the output phase noise instead of tracking it. In this case, the tracking bandwidth in Eq(C.11) and Figure C.6 pertains to the maximum frequency of the output phase noise that the PLL can filter. Low-frequency noise can be compensated by the PLL and get filtered before its full amplitude propagates to the output. High-frequency noises, however, will not be suppressed by the PLL and will propagate to the output.

Another important noise source in a bangbang PLL is the noise on the frequency ω_i.

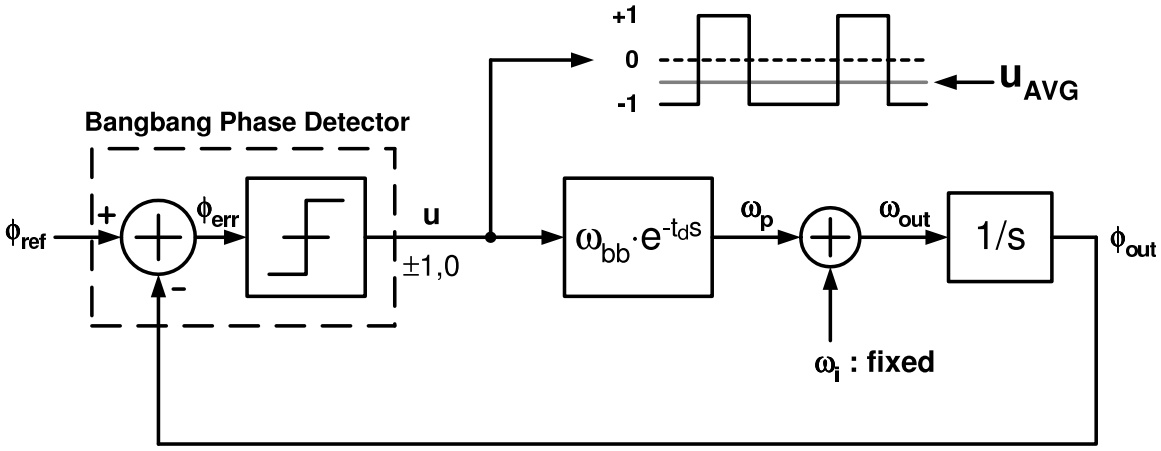


Figure C.8: Pull-in behavior of the bangbang PLL: the phase detector output has nonzero average value

For example, the supply noise may vary the VCO frequency. Figure C.7 shows the peak-to-peak phase error when different frequency steps are applied to ω_i. When the frequency step is less than ω_{bb}, the phase-tracking loop can reject the phase noise and the phase error is kept within the dithering jitter level. When the frequency step is larger than ω_{bb}, however, the phase-tracking loop will momentarily lose lock until the frequency-tracking loop brings ω_i close back to ω_{ref}. The duration of the lost phase-lock will be longer for lower integral gain and for longer loop delay. Therefore, the peak phase error is larger for the larger values of $\kappa = \tau / t_{d,eff}$. Although large κ is good for stability and for low dithering jitter, it can slow down the frequency-tracking response and makes the PLL more susceptible to frequency noise.

C.3 Out-of-Lock Behavior of a Bangbang PLL

We discussed that when the VCO base frequency ω_i is within $\pm\omega_{bb}$ from the reference frequency ω_{ref}, the phase-tracking loop will acquire lock without cycle-slipping. In other words, the lock-in range of the bangbang PLL is:

$$\text{lock-in range} : |\omega_i - \omega_{ref}| < \omega_{bb}. \tag{C.12}$$

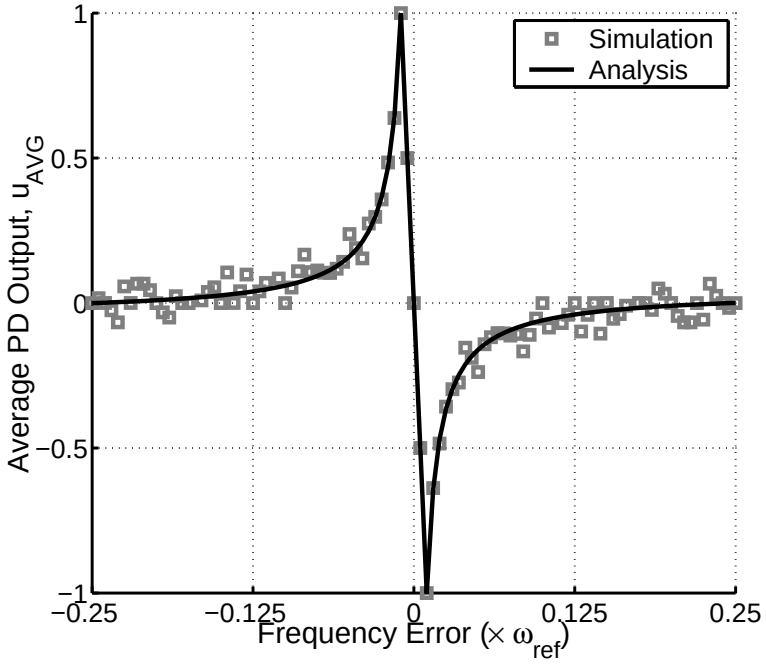


Figure C.9: Average phase detector output versus frequency error: a case with a zero loop delay

When ω_i is out of this range, the phase error accumulates at a faster rate than the phase-tracking loop can keep up. In this case, the phase error will keep increasing to the point where it crosses the cycle boundary (called *cycle-slipping*) and start increasing again. In the meantime, the phase-tracking loop will make efforts to acquire lock and the phase detector will have some nonzero average output. If this average phase detector output can pull the base frequency ω_i close enough to ω_{ref}, the phase-tracking loop will be able to acquire lock eventually. The frequency range that the PLL can gain lock after some cycle-slipping is called *pull-in range*.

The pull-in range can be found by examining the average phase detector output u_{AVG} for each frequency error $\omega_i - \omega_{ref}$. Figure C.8 illustrates that when ω_i is held constant at a frequency distant from ω_{ref}, the phase detector will have a periodic output as the PLL slips cycles. In case of a zero loop delay, the phase detector output u will be $+1$ exactly when the phase error ϕ_{err} is between 0 and $+\pi$, and -1 when ϕ_{err} is between $-\pi$ and 0. In the meantime, the output frequency ω_{out} will alternate between $\omega_i + \omega_{bb}$ and $\omega_i - \omega_{bb}$.

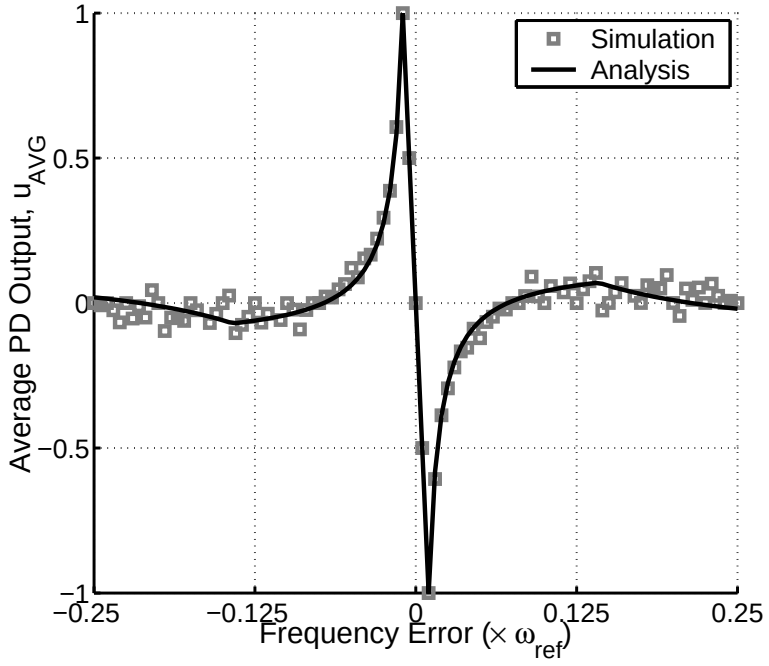


Figure C.10: Average phase detector output versus frequency error: a case with a nonzero loop delay ($t_d = 2.5 \cdot t_{update}$)

While ω_{out} is at one of the two frequencies that is closer to ω_{ref} than the other, the phase error will accumulate at a slower rate and thus the corresponding phase detector output will last for a longer period of time. As a result, the phase detector output will have a net average value that directs ω_i toward ω_{ref}. The plot of average phase detector output u_{AVG} versus frequency error $\omega_i - \omega_{ref}$ is shown in Figure C.9 for the case of a zero loop delay. The pull-in force u_{AVG} is highest near ω_{ref} but decreases as the frequency error increases because the difference in phase-drifting rates diminishes.

With a zero loop delay, u_{AVG} always gives the right polarity that pulls the loop toward lock. The pull-in range is as wide as $\pm\omega_{ref}/2$, beyond which the loop will be pulled toward the harmonics. However, for nonzero loop delays, the polarity of u_{AVG} may become opposite; the loop may be pushed away instead of being pulled in. Figure C.10 shows the plot of u_{AVG} versus frequency error when the loop delay t_d is equal to $2.5 \cdot t_{update}$. Due to the loop delay, the pull-in range is much narrower; only $\pm 7\%$ of the reference frequency in this case. Since no PLL has truly zero loop delay, understanding the effects of loop delay

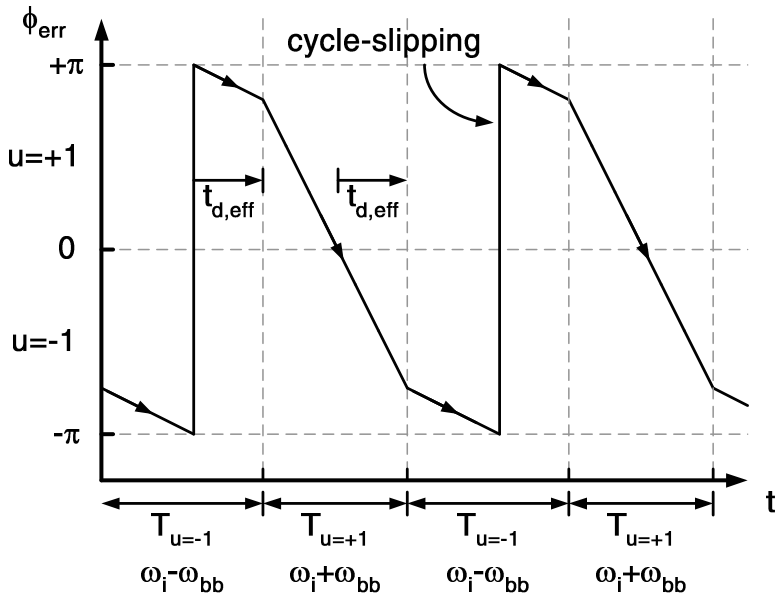


Figure C.11: Drifting of the phase error ϕ_{err} in case of nonzero loop delay

on the pull-in range is important. The rest of this section derives the expression for the pull-in range as a function of the loop delay.

Figure C.11 shows drifting of phase error versus time when the loop delay is non-zero and $\omega_i > \omega_{ref} + \omega_{bb}$. When the phase detector output u is $+1$, the PLL output frequency ω_{out} is $\omega_i + \omega_{bb}$ and the phase error ϕ_{err} will drift at a rate, $d\phi_{err}/dt = (\omega_i + \omega_{bb}) - \omega_{ref}$. Similarly, when u is -1, $d\phi_{err}/dt = (\omega_i - \omega_{bb}) - \omega_{ref}$. Due to the loop delay, the phase detector output u does not switch until the effective delay $t_{d,eff}$ elapses after ϕ_{err} crosses the decision boundary $\phi_{err} = 0$ or $\pm\pi$. The durations for which u is $+1$ and -1 are, respectively:

$$\begin{aligned} T_{u=+1} &= \tfrac{\pi+2\omega_{bb}t_{d,eff}}{\omega_i-\omega_{ref}+\omega_{bb}}, \\ T_{u=-1} &= \tfrac{\pi-2\omega_{bb}t_{d,eff}}{\omega_i-\omega_{ref}-\omega_{bb}}, \end{aligned} \tag{C.13}$$

for $\omega_{bb} < \omega_i - \omega_{ref} < \pi/t_{d,eff}$. Then the average phase detector output u_{AVG} is:

$$u_{AVG} = \frac{(\omega_i - \omega_{ref}) \cdot t_{d,eff} - \pi/2}{\pi/2 \cdot (\omega_i - \omega_{ref})/\omega_{bb} - \omega_{bb} t_{d,eff}} \quad \text{for } \omega_{bb} < \omega_i - \omega_{ref} < \pi/t_{d,eff}. \tag{C.14}$$

Eq(C.14) suggests that the polarity of u_{AVG} will be the opposite when the frequency error

reaches $\pi/2t_{d,eff}$. Therefore, the pull-in range of a bangbang PLL is:

$$\text{pull-in range} : |\omega_i - \omega_{ref}| < \frac{\pi}{2t_{d,eff}}. \tag{C.15}$$

The longer the loop delay, the narrower the pull-in range. When the frequency error $\omega_i - \omega_{ref}$ reaches beyond $\pi/t_{d,eff}$, the phase error ϕ_{err} will drift at a large enough rate that it crosses the next decision boundary before the loop delay elapses. In this case, the expression for u_{AVG} becomes:

$$u_{AVG} = \frac{3\pi/2 - (\omega_i - \omega_{ref}) \cdot t_{d,eff}}{\pi/2 \cdot (\omega_i - \omega_{ref})/\omega_{bb} + \omega_{bb} t_{d,eff}} \text{ when } \pi/t_{d,eff} < \omega_i - \omega_{ref} < 2\pi/t_{d,eff}. \tag{C.16}$$

In Figure C.10, the predicted trend from the analysis shows a good match with the numerical data points. The numerical data points show irregularities at large frequency errors where $T_{u=+1}$ and $T_{u=-1}$ are small multiples of t_{update} and the quantization noise becomes dominant.

C.4 A Frequency-Acquisition Aid: Frequency Sweeping

Section 4.3.1 described frequency-sweeping as a frequency-acquisition aid. The idea is to first initialize the frequency ω_i to the highest end and to step it down by some amount whenever the PLL detects cycle-slipping. When the PLL is out of the pull-in range, the frequency sweeper guides the PLL toward lock. Once it comes within the pull-in range, the control action of the PLL dominates and drives the loop toward lock. Therefore, the frequency step of the frequency sweeper ω_s is bounded on two sides; it must be large enough to overcome the opposite-sign pull-in force but also be small enough to guarantee that ω_i will always fall into the lock-in range of the PLL without skipping it.

The upperbound of the frequency step ω_s is $2\omega_{bb}$, since the lock-in range is $2\omega_{bb}$-wide. Once ω_i falls into the lock-in range, the phase-tracking loop will prevent cycle-slipping and will acquire lock while frequency sweeping is essentially disabled. If the frequency step is so large that ω_i skips the lock-in range, however, the PLL will have cycle-slipping again and this time the frequency sweeper will pull the PLL away from lock. The PLL may never

be able to return to the lock-in range if the pull-in force is not strong enough to counteract the frequency sweeper. It is because detecting cycle-slipping only cannot distinguish the polarity of the frequency error and thus the frequency is swept only in one direction.

The lowerbound, on the other hand, depends on the opposite-sign pull-in force u_{AVG}, which has a peak value of $\omega_{bb}t_{d,eff}/\pi$ at $|\omega_i - \omega_{ref}| = \pi/t_{d,eff}$. Without any frequency acquisition aid, the frequency ω_i will drift away from this worst-case point at the average rate of $2\omega_{bb}^2 t_{d,eff}/\omega_{ref}\tau$ per cycle. In the meantime, the cycle-slipping happens $\pi/\omega_{ref}t_{d,eff}$ times per cycle on average, implying that the minimum frequency step of $(2\omega_{bb}t_{d,eff})^2/2\pi\tau = 2\omega_{bb}/\kappa \cdot \Delta\phi_{p,pp}/2\pi$ is required for the frequency sweeper to overcome the opposite-sign pull-in force barrier.

Summarizing both constraints, the expression for the possible range of ω_s is:

$$\left(\frac{2}{\kappa} \cdot \frac{\Delta\phi_{p,pp}}{2\pi}\right) \cdot \omega_{bb} < \omega_s < 2\omega_{bb}. \tag{C.17}$$

For $\Delta\phi_{p,pp}$ of 2% of the unit interval (UI) and κ value of 40, the range becomes $0.001 \cdot \omega_{bb} < \omega_s < 2 \cdot \omega_{bb}$, which is fairly wide. In adaptive-bandwidth PLLs, the frequency step ω_s can have a fixed ratio to ω_{bb} for a wide operating range, by scaling the frequency sweeping current proportionally with the main charge pump current.

C.5 Multiphase System Issues

For clock-recovery applications that use multiple phases instead of high frequencies to distinguish finely-spaced timings, detecting timing error involves dealing with multiple phase detector outputs every clock period. For example, in clock recovery PLLs described in Chapter 4, total M phase-detecting receivers cover every possible data transitions within a cycle, giving at most M timing decisions per cycle, where M is the multiplexing rate.

It should be noted that covering only a subset of transitions, i.e. using less than M phase detectors, can create some additional false-locking states due to frequency aliasing. Figure C.12 illustrates the case when the phase detector detects the timing of only one out of five possible edges. The subharmonic frequencies such as 4/5 or 6/5 of the reference frequency become equally-probable lock states. Although it is still possible to use less than

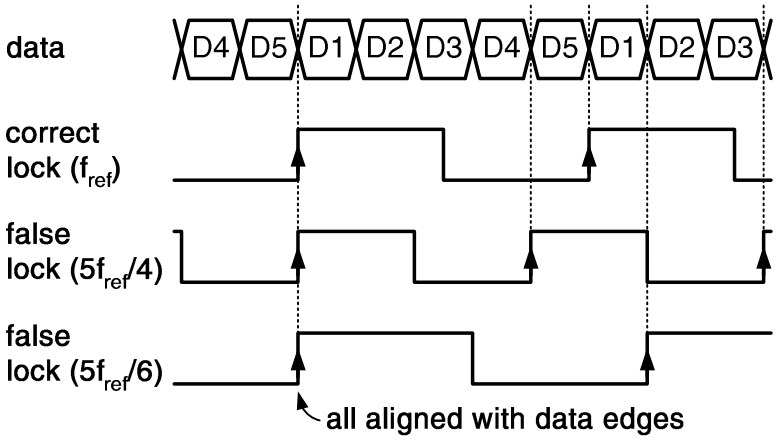


Figure C.12: False locking modes of a bangbang clock recovery. Cases for a 5-phase multiplexing link

M phase detectors and let the frequency acquisition aid prevent false lock, the full coverage of data transition points is generally desirable to fully utilize the timing information embedded in the data stream.

Given the multiple phase detector outputs per cycle, there are several ways to handle them to control the timing of the VCO. A straightforward way is to apply them all individually to the VCO. Since it is difficult to time-multiplex these phase detector outputs into a single high-frequency output stream, the PLL may need multiple charge pumps or multiple frequency-control inputs for the VCO. In this case, the effect of one phase detector output can span more than one phase and up to one cycle (M phases). Thus, the control periods of multiple phase detectors may overlap in time, effectively increasing the overall loop gain. Lowering individual control gain may be needed to keep the dithering jitter low.

For an equal total gain, the overlap in control periods in fact reduces the dithering jitter. When the VCO timing turns from late to early, the aggregate control gain u will gradually increase from -1 to $+1$ in small steps ($= 1/M$) as the phase detectors switch their outputs at different times. It is contrary to the case with no overlaps where the control gain u switch abruptly between -1 and $+1$. This gradual change in u lowers the peak-to-peak dithering jitter, at most by a factor of 2 when M is large and the loop delay is 0. However, the jitter-lowering benefit diminishes as the loop delay increases.

Another way to handle multiple phase detector outputs is to decimate them into one

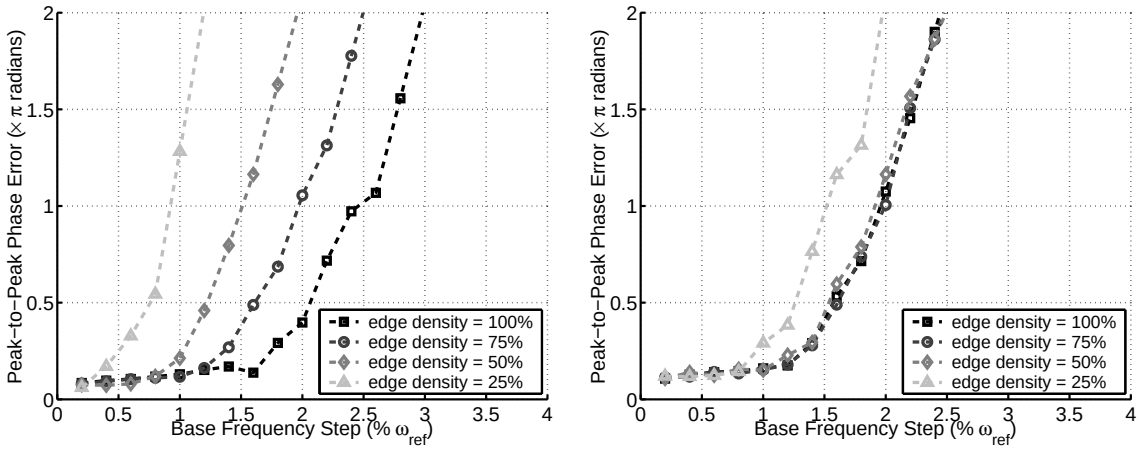


Figure C.13: ω_i-noise response of multiphase bangbang PLLs: (a) applying multiple PD outputs individually, (b) decimating them by the majority vote

aggregate output per cycle, e.g. by taking the majority vote, so that the PLL does not need multiple control ports for the VCO. However, majority-voting will effectively increase the loop delay, adversely affecting the dithering jitter. The majority-voting circuit needs to wait for all M decisions to be ready before making the final aggregate decision and thus the loop delay is longer for the decision that arrives earlier. Nonetheless, the benefit of decimating is that the overall loop gain becomes less sensitive to transition density. The loop gain will remain constant as long as there is at least one transition out of M, which can be ensured by proper coding. Constant loop dynamics enable easier performance estimation and design verification.

Figure C.13 compares the frequency noise responses of the multiphase bangbang PLLs for these two cases. The two phase detector implementations are normalized to the equal dithering jitter and a loop delay t_d of 1 cycle is assumed. As expected, the non-decimating phase detector case has a stronger dependency on the transition density than the decimating case. For high transition densities, the non-decimating case experiences less jitter and thus exhibits higher bandwidth. The overlap in control periods reduced the dithering jitter and allowed the peak loop gain to be higher for the specified jitter. However, for transition rates lower than 75%, the effective gain of the non-decimating phase detector drops below that of the decimating phase detector and the decimating case shows better noise response.

Bibliography

[1] D. Burger, J. R. Goodman, and A. Kagi. Memory Bandwidth Limitations of Future Microprocessors. In *Proceedings of 23rd Annual International Symposium on Computer Architecture*, pages 79–90. Association of Computing Machinery, August 1996.

[2] C.-K. Yang and M. Horowitz. A 0.8-μm CMOS 2.5Gb/s Oversampling Receiver and Transmitter for Serial Links. *IEEE Journal of Solid-State Circuits*, 31(12):2015–2023, December 1996.

[3] S. Sidiropoulos and M. Horowitz. A 700 Mbps/pin CMOS Signalling Interface Using Current Integrating Receivers. *IEEE Journal of Solid-State Circuits*, 32(5):681–690, May 1997.

[4] G. E. Moore. Cramming more components onto integrated circuits. *Electronics*, April 1965.

[5] W. Dally and J. Poulton. *Digital Systems Engineering*. New York: Cambridge University Press, 1998.

[6] B. Landman and R. L. Russo. On a Pin vs. Block Relationship for Partitioning of Logic Graphs. *IEEE Transactions on Computers*, C-20(12):1469–1479, December 1971.

[7] D. K. Jeong and et al. Design of PLL-Based Clock Generation Circuits. *IEEE Journal of Solid-State Circuits*, 22(4):255–261, April 1987.

[8] M. Horowitz and et al. PLL Design for a 500 MB/s Interface. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 160–161, February 1993.

[9] T. H. Lee and et al. A 2.5V CMOS Delay-locked Loop for 18 Mbit, 500 Megabyte/s DRAM. *IEEE Journal of Solid-State Circuits*, 29(12):1491–1496, December 1994.

[10] A. Chandrakasan, W. J. Bowhill, and F. Fox, editors. *Design of High-Performance Microprocessor Circuits*. New York: IEEE Press, 2001.

[11] E. Reese and et al. A Phase-Tolerant 3.8 GB/s Data-Communication Router for Multi-Processor Super Computer Backplane. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 296–297, February 1994.

[12] R. Nair and et al. A 28.5GB/s CMOS Non-Blocking Router for Terabits/s Connectivity between Multiple Processors and Peripheral I/O Nodes. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 224–225, February 2001.

[13] A. Jain and et al. A 1.2GHz Alpha Microprocessor with 44.8GB/s Chip Pin Bandwidth. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 240–241, February 2001.

[14] V. Pathak and et al. 40Gb/s ASIC Switch Design Using Low-Jitter Clock Recovery. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 226–227, February 2001.

[15] P. Landman and et al. A 62Gb/s Backplane Interconnect ASIC Based on 3.1Gb/s Serial-Link Technology. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 72–73, February 2002.

[16] Semiconductor Industry Association. The National Technology Roadmap for Semiconductors, Technology Needs. Technical report, November 1997.

[17] P. P. Gelsinger. Microprocessors for the New Millennium: Challenges, Opportunities, and New Frontiers. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 22–23, February 2001.

[18] M. A. Horowitz. Low-power Processor Design Using Self-clocking. *Presented at the Workshop on Low-Power Electronics*, August 1993.

[19] R. Gonzalez and M. A. Horowitz. Energy Dissipation in General Purpose Microprocessors. *IEEE Journal of Solid-State Circuits*, 31(9):1277–1284, September 1996.

[20] R. Gonzalez, B. M. Gordon, and M. A. Horowitz. Supply and Threshold Voltage Scaling for Low Power CMOS. *IEEE Journal of Solid-State Circuits*, 32(8):1210–1216, August 1997.

[21] P. Macken, M. Degrauwe, M. Van Paemel, and H. Oguey. A Voltage Reduction Technique for Digital Systems. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 238–239, February 1990.

[22] T. D. Burd, T. A. Pering, A. J. Stratakos, and R. W. Broderson. A Dynamic Voltage Scaled Microprocessor System. *IEEE Journal of Solid-State Circuits*, 35(11):1571–1580, November 2000.

[23] L. T. Clark and et al. A Scalable Performance 32b Microprocessor. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 230–231, February 2001.

[24] K. Nowka and et al. A 0.9V to 1.95V Dynamic Voltage-Scalable and Frequency-Scalable 32b PowerPC Processor. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 340–341, February 2002.

[25] V. Gutnik and A. Chandrakasan. Embedded Power Supply for Low-Power DSP. *IEEE Transactions on VLSI Systems*, 5:425–435, December 1997.

[26] L. Nielsen, C. Niessen, J. Sparso, and K. van Berkel. Low-Power Operation Using Self-Timed Circuits and Adaptive Scaling of Supply Voltage. *IEEE Transactions on VLSI Systems*, 2:391–397, December 1994.

[27] A. Dancy and A. Chandrakasan. Techniques for Aggressive Supply Voltage Scaling and Efficient Regulation. In *Proceedings of IEEE Custom Integrated Circuits Conference*, pages 579–586, 1997.

[28] W. Namgoong, M. Yu, and T. Meng. A High-Efficiency Variable-Voltage CMOS Dynamic DC-DC Switching Regulator. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 380–381, February 1997.

[29] G. Wei and M. A. Horowitz. A Fully Digital, Energy-Efficient, Adaptive Power-Supply Regulator. *IEEE Journal of Solid-State Circuits*, 34(4):520–529, April 1999.

[30] J. Kim and M. Horowitz. An Efficient Digital Sliding Controller for Adaptive Power-Supply Regulation. In *IEEE Symposium on VLSI Circuits, Digest of Technical Papers*, pages 133–136, June 2001.

[31] J. Kim and M. A. Horowitz. An Efficient Digital Sliding Controller for Adaptive Power-Supply Regulation. *IEEE Journal of Solid-State Circuits*, 37(5):639–647, May 2002.

[32] G.-Y. Wei, J. Kim, D. Liu, S. Sidiropoulos, and M. A. Horowitz. A Variable-Frequency Parallel I/O Interface with Adaptive Power-Supply Regulation. *IEEE Journal of Solid-State Circuits*, 35(11):1600–1610, November 2000.

[33] A. Chandrakasan, S. Sheng, and R. Broderson. Low-Power CMOS Digital Design. *IEEE Journal of Solid-State Circuits*, 27(4):473–484, April 1992.

[34] R. Gu and et al. 0.5-3.5Gb/s Low-Power Low-Jitter Serial Data CMOS Transceiver. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 352–353, February 1999.

[35] R. Farjad-Rad, C.-K. K. Yang, M. Horowitz, and T. H. Lee. A 0.3-μm CMOS 8-Gb/s 4-PAM Serial Link Transceiver. *IEEE Journal of Solid-State Circuits*, 35(5):757–764, May 2000.

[36] P. Kinget and M. Steyaert. Impact of Transistor Mismatch on the Speed Accuracy Power Trade-Off of Analog CMOS Circuits. In *Proceedings of the IEEE Custom Integrated Circuits Conference (CICC)*, pages 333–336, 1996.

[37] C.-K. K. Yang. *Design of High-Speed Serial Links in CMOS*. PhD thesis, Stanford University, Stanford, CA, 1998.

[38] N. Weste and K. Eshraghian. *Principle of CMOS VLSI Design: A Systems Perspective*. Addison-Wesley Publishing Company, second edition, 1993.

[39] D. Monticelli. Switching Power Supply Design. In *Proceedings of IEEE Symposium on Low Power Electronics*, page 64, October 1995.

[40] R. W. Erickson. *Fundamentals of Power Electronics*. New York: Chapman and Hall, 1997.

[41] F. Bilalovic, O. Music, and A. Sabanovic. Buck converter regulator operating in the sliding mode. In *Proceedings of Power Conversion International Conference (PCI)*, pages 331–340, April 1983.

[42] R. Venkataramanan, A. Sabanovic, and S. Cuk. Sliding mode control of DC-to-DC converters. In *Proceedings of IEEE Industrial Electronics Society Conference (IECON)*, pages 251–258, 1985.

[43] J.-J. E. Slotine and W. Li. *Applied Nonlinear Control*. New Jersey: Prentice Hall, 1991.

[44] C. L. Portmann and T. H. Y. Meng. Power-efficient Metastability Error Reduction in CMOS Flash A/D Converters. *IEEE Journal of Solid-State Circuits*, 31(8):1132–1140, August 1996.

[45] S. Borkar. Design Challenges of Technology Scaling. *IEEE Micro*, 19(4):22–23, July–August 1999.

[46] K. Lee and et al. A CMOS Serial Link for Fully Duplexed Data Communication. *IEEE Journal of Solid-State Circuits*, 30(4):353–364, April 1995.

[47] T. Sakurai and A. R. Newton. Alpha-power Law MOSFET Model and Its Application to CMOS Inverter Delay and Other Formulas. *IEEE Journal of Solid-State Circuits*, 25(4):584–593, April 1990.

[48] M.-J. Lee, W. Dally, and P. Chiang. Low-Power Area-Efficient High-Speed I/O Circuit Techniques. *IEEE Journal of Solid-State Circuits*, 35(11):1591–1599, November 2000.

[49] Y. Nakagome and et al. An Experimental 1.5-V 64-Mb DRAM. *IEEE Journal of Solid-State Circuits*, 26(4):465–472, April 1991.

[50] R. Farjad-Rad, C.-K. K. Yang, M. Horowitz, and T. H. Lee. A 0.4-μm CMOS 10-Gb/s 4-PAM Pre-emphasis Serial Link Transmitter. *IEEE Journal of Solid-State Circuits*, 34(5):580–585, May 1999.

[51] R. D. Gitlin, J. F. Hayes, and S. B. Weinstein. *Data Communications Principles*. Perseus Publishing, July 1997.

[52] J. Crols and M. Steyaert. Switched-Opamp: An Approach to Realize Full CMOS Switched-Capacitor Circuits at Very Low Power Supply Voltages. *IEEE Journal of Solid-State Circuits*, 29(8):936–942, August 1994.

[53] J. Montanaro and et al. A 160MHz, 32b, 0.5W CMOS RISC Microprocessor. *IEEE Journal of Solid-State Circuits*, 31(11):1703–1714, November 1996.

[54] S. M. Sze. *Physics of Semiconductor Devices*. New York: Wiley, second edition, 1981.

[55] L. Wu and Jr. W. C. Black. A Low-Jitter Skew-Calibrated Multiphase Clock Generator for Time-Interleaved Applications. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 396–397, February 2001.

[56] K. Yamaguchi and et al. 2.5GHz 4-phase Clock Generator with Scalable and No Feedback Loop Architecture. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 398–399, February 2001.

[57] T. Saeki and et al. A 1.3 Cycle Lock Time, Non-PLL/DLL Jitter Suppression Clock Multiplier Based on Direct Clock Cycle Interpolation for "Clock on Demand". In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 166–167, February 2000.

[58] E. Yeung and M. A. Horowitz. A 2.4Gb/s/pin Simultaneous Bidirectional Parallel Link with Per-pin Skew Compensation. *IEEE Journal of Solid-State Circuits*, 35(11):1619–1628, November 2000.

[59] R. J. Baumert and et al. A Monolithic 50-200 MHz CMOS Clock-Recovery and Retiming Circuit. In *Proceedings of the IEEE Custom Integrated Circuits Conference (CICC)*, pages 14.5.1–4, 1989.

[60] S. B. Anand and B. Razavi. A 2.75Gb/s CMOS Clock Recovery Circuit with Broad Capture Range. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 214–215, February 2001.

[61] J. G. Maneatis. Low-jitter Process-Independent DLL and PLL based on Self-Biased Techniques. *IEEE Journal of Solid-State Circuits*, 31(11):1723–1732, November 1996.

[62] S. Sidiropoulos, D. Liu, J. Kim, G. Wei, and M. Horowitz. Adaptive Bandwidth DLLs and PLLs using Regulated Supply CMOS Buffers. In *IEEE Symposium on VLSI Circuits, Digest of Technical Papers*, pages 124–127, June 2000.

[63] S. Sidiropoulos and M. Horowitz. A Semidigital Dual Delay-Locked Loop. *IEEE Journal of Solid-State Circuits*, 32(11):1683–1692, November 1997.

[64] J. G. Maneatis and M. A. Horowitz. Precise Delay Generation Using Coupled Oscillators. *IEEE Journal of Solid-State Circuits*, 28(12):1273–1282, December 1993.

[65] B. W. Garlepp and et al. A Portable Digital DLL for High-Speed CMOS Interface Circuits. *IEEE Journal of Solid-State Circuits*, 34(5):632–644, May 1999.

[66] S. J. Lee, B. Kim, and K. Lee. A Novel High-speed Ring Oscillator for Multiphase Clock Generation Using Negative Skewed Delay Scheme. *IEEE Journal of Solid-State Circuits*, 32(2):289–291, February 1997.

[67] L. Sun and T. A. Kwasniewski. A 1.25-GHz 0.35-μm Monolithic CMOS PLL Based on a Multiphase Ring Oscillator. *IEEE Journal of Solid-State Circuits*, 36(6):910–916, June 2001.

[68] K. Itoh. *VLSI Memory Chip Design*. New York: Springer-Verlag, April 2001.

[69] M. G. Johnson and E. L. Hudson. A Variable Delay Line PLL for CPU-Coprocessor Synchronization. *IEEE Journal of Solid-State Circuits*, 23(5):1218–1223, October 1988.

[70] Q. Zhang and et al. SPICE Modeling and Quick Estimation of MOSFET Mismatch Based on BSIM3 Model and Parametric Tests. *IEEE Journal of Solid-State Circuits*, 36(10):1592–1595, October 2001.

[71] M. J. M. Pelgrom, H. P. Tuinhout, and M. Vertregt. Transistor Matching in Analog CMOS Applications. In *IEEE International Electron Devices Meeting (IEDM), Technical Digest*, pages 915–918, December 1998.

[72] H. Ebenhoech. Make IC Digital Frequency Comparators. *Electron. Design*, 15(14):62–64, July 1967.

[73] H. Notani and et al. A 622-MHz CMOS PLL with Precharge-Type Phase-Frequency Detector. In *IEEE Symposium on VLSI Circuits, Digest of Technical Papers*, pages 129–130, June 1994.

[74] J. D. H. Alexander. Clock Recovery from Random Binary Signals. *Electronics Letters*, 11:541–542, October 1975.

[75] D. Weinlader, R. Ho, C.-K. K. Yang, and M. Horowitz. An Eight Channel 36GSample/s CMOS Timing Analyzer. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 170–171, February 2000.

[76] R. C. Walker and et al. A Two-Chip 1.5GBd Serial Link Interface. *IEEE Journal of Solid-State Circuits*, 27(12):1805–1811, December 1992.

[77] Y. M. Greshishchev and et al. A Fully Integrated SiGe Receiver IC for 10-Gb/s Data Rate. *IEEE Journal of Solid-State Circuits*, 35(12):1949–1957, December 2000.

[78] F. M. Gardner. *Phaselock Techniques, 2nd Edition*. New York: John Wiley & Sons, 1979.

[79] D. Harris and M. A. Horowitz. Skew-Tolerant Domino Circuits. *IEEE Journal of Solid-State Circuits*, 32(11):1702–1711, November 1997.

[80] M.-J. E. Lee and et al. An 84-mW 4-Gb/s Clock and Data Recovery Circuit for Serial Link Applications. In *IEEE Symposium on VLSI Circuits, Digest of Technical Papers*, pages 149–152, June 2001.

[81] J. Kim and M. A. Horowitz. Adaptive-Supply Serial Links with Sub-1V Operation and Per-Pin Clock Recovery. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 268–269, February 2002.

[82] J. Kim and M. A. Horowitz. Adaptive-Supply Serial Links with Sub-1V Operation and Per-Pin Clock Recovery. *IEEE Journal of Solid-State Circuits*, 37(11):1403–1413, November 2002.

[83] K. R. Lakshmikumar and et al. Characterisation and Modeling of Mismatch in MOS Transistors for Precision Analog Design. *IEEE Journal of Solid-State Circuits*, 21(12):1057–1066, December 1986.

[84] T. Mizuno and et al. Experimental Study of Threshold Voltage Fluctuation Due to Statistical Variation of Channel Dopant Number in MOSFET's. *IEEE Transactions on Electron Devices*, 41(11):2216–2221, November 1994.

[85] M. Pelgrom, A. Duinmaijer, and A. Welbers. Matching Properties of MOS Transistors. *IEEE Journal of Solid-State Circuits*, 24(5):1433–1440, October 1989.

[86] W. Dally and J. Poulton. Transmitter Equalization for 4Gb/s Signalling. In *Proceedings of Hot Interconnects IV*, pages 29–39, August 1996.

[87] V. Stojanovic, G. Ginis, and M. Horowitz. Transmit Pre-emphasis for High-Speed Time-Division-Multiplexed Serial-Link Transceiver. In *IEEE International Conference on Communications*, pages 1934–1939, April 2002.

[88] W. Ellersick, C.-K. Yang, V. Stojanovic, S. Modjtahedi, and M. Horowitz. A Serial Link Based on 8 GSa/s A/D and D/A Converters in 0.25μm CMOS. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 58–59, February 2001.

[89] B. Lau and et al. A 2.6 Gb/s Multi-Purpose Chip to Chip Interface. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pages 162–163, February 1998.

[90] T.-C. Lee and B. Razavi. Stabilization Technique for Phase-locked Frequency Synthesizers. In *IEEE Symposium on VLSI Circuits, Digest of Technical Papers*, pages 39–42, June 2001.

[91] A. Hajimiri, S. Limotyrakis, and T. H. Lee. Jitter and Phase Noise in Ring Oscillators. *IEEE Journal of Solid-State Circuits*, 34(6):790–804, June 1999.

[92] F. M. Gardner. Charge-pump Phase-locked Loops. *IEEE Transactions on Communications*, COM-28:1849–1858, November 1980.

[93] J. G. Maneatis and et al. Self-Biased High-Bandwidth Low-Jitter 1-to-4096 Multiplier Clock Generator PLL. In *IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, February 2003.

[94] M. H. Perrot. Fast and Accurate Behavioral Simulation of Fractional-N Frequency Synthesizers and Other PLL/DLL Circuits. In *Proceedings of 39th Design Automation Conference (DAC)*, pages 498–503, 2002.