



Ph.D. Dissertation

Ring-Oscillator-Based Frequency Synthesizers for High-Speed Serial Links

고속 시리얼 링크를 위한 고리 발진기를 기반으로 하는 주파수 합성기

by

Hyojun Kim

August 2022

Department of Electrical and Computer Engineering College of Engineering Seoul National University Ph.D. Dissertation

Ring-Oscillator-Based Frequency Synthesizers for High-Speed Serial Links

고속 시리얼 링크를 위한 고리 발진기를 기반으로 하는 주파수 합성기

by

Hyojun Kim

August 2022

Department of Electrical and Computer Engineering College of Engineering Seoul National University

Ring-Oscillator-Based Frequency Synthesizers for High-Speed Serial Links

고속 시리얼 링크를 위한 고리 발진기를 기반으로 하는 주파수 합성기

지도교수 정 덕 균 이 논문을 공학박사 학위논문으로 제출함

2022년 8월

서울대학교 대학원

전기·정보공학부

김효준

김효준의 공학박사 학위 논문을 인준함

2022년 6월

위 위	권 장	김재하	(인)
부위	원장	정덕균	(인)
위	원	전동석	(인)
위	원	최우석	(인)
위	원	한재덕	(인)

Abstract

In this dissertation, major concerns in the clocking of modern serial links are discussed. As sub-rate, multi-standard architectures are becoming predominant, the conventional clocking methodology seems to necessitate innovation in terms of low-cost implementation. Frequency synthesis with active, inductor-less oscillators replacing LC counterparts are reviewed, and solutions for two major drawbacks are proposed. Each solution is verified by prototype chip design, giving a possibility that the inductor-less oscillator may become a proper candidate for future high-speed serial links.

To mitigate the high flicker noise of a high-frequency ring oscillator (RO), a reference multiplication technique that effectively extends the bandwidth of the following all-digital phase-locked loop (ADPLL) is proposed. The technique avoids any jitter accumulation, generating a clean mid-frequency clock, overall achieving high jitter performance in conjunction with the ADPLL. Timing constraint for the proper reference multiplication is first analyzed to determine the calibration points that may correct the existent phase errors. The weight for each calibration point is updated by the proposed *a priori* probability-based least-mean-square (LMS) algorithm. To minimize the time required for the calibration, each gain for the weight update is adaptively varied by deducing *a posteriori* which error source dominates the others. The prototype chip is fabricated in a 40-nm CMOS technology, and its measurement results verify the low-jitter, high-frequency clock generation with fast calibration settling. The presented work achieves an rms jitter of 177/223 fs at 8/16-GHz output, consuming 12.1/17-mW power.

As the second embodiment, an RO-based ADPLL with an analog technique that addresses the high supply sensitivity of the RO is presented. Unlike prior arts, the circuit for the proposed technique does not extort the RO voltage headroom, allowing high-frequency oscillation. Further, the performance given from the technique is robust over process, voltage, and temperature (PVT) variations, avoiding the use of additional calibration hardware. Lastly, a comprehensive analysis of phase noise contribution is conducted for the overall ADPLL, followed by circuit optimizations, to retain the low-jitter output. Implemented in a 40-nm CMOS technology, the frequency synthesizer achieves an rms jitter of 289 fs at 8 GHz output without any injected supply noise. Under a 20-mV_{rms} white supply noise, the ADPLL suppresses supply-noise-induced jitter by -23.8 dB.

keywords: frequency synthesizer, phase noise, jitter, all-digital phase-locked loop (ADPLL), ring oscillator (RO), multi-phase clock, digitally controlled resistor (DCR), reference multiplication, supply noise **student number**: 2017-28301

Contents

Al	ostrac	et		i
Co	onten	ts		iii
Li	st of [Fables		v
Li	st of l	Figures		vi
1	Intr	oductio	n	1
	1.1	Motiva	ation	3
		1.1.1	Clocking in High-Speed Serial Links	4
		1.1.2	Multi-Phase, High-Frequency Clock Conversion	8
	1.2	Disser	tation Objectives	10
2	RO-	Based I	High-Frequency Synthesis	12
	2.1	Phase-	Locked Loop Fundamentals	12
	2.2	Towar	d All-Digital Regime	15
	2.3	RO De	esign Challenges	21
		2.3.1	Oscillator Phase Noise	21
		2.3.2	Challenge 1: High Flicker Noise	23
		2.3.3	Challenge 2: High Supply Noise Sensitivity	26

CONTENTS

3	Filte	ering RO Noise	28
	3.1	Introduction	28
	3.2	Proposed Reference Octupler	34
		3.2.1 Delay Constraint	34
		3.2.2 Phase Error Calibration	38
		3.2.3 Circuit Implementation	51
	3.3	IL-ADPLL Implementation	55
	3.4	Measurement Results	59
	3.5	Summary	63
4	RO	Supply Noise Compensation	69
	4.1	Introduction	69
	4.2	Proposed Analog Closed Loop for Supply Noise Compensation	72
		4.2.1 Circuit Implementation	73
		4.2.2 Frequency-Domain Analysis	76
		4.2.3 Circuit Optimization	81
	4.3	ADPLL Implementation	87
	4.4	Measurement Results	90
	4.5	Summary	98
5	Con	clusions	99
A	Note	es on the 8×REF	102
B	Note	es on the ACSC	105
Ał	bstract (In Korean) 119		

List of Tables

3.1	Comparison with Prior High- N RO-Based Frequency Synthesizers	67
4.1	Comparison with Prior RO-Based Frequency Synthesizers with Sup-	
	ply Noise Compensation	97

List of Figures

1.1	Data rates of state-of-the-art SerDes (transceivers) versus technology	
	node	2
1.2	Depiction of a generic SerDes	3
1.3	Abstract diagrams of TX architectures	5
1.4	Quadrature phase generation using injection locking	9
2.1	Schematic of a generic PLL.	13
2.2	Third-order analog LF preceded by a charge pump	14
2.3	Schematic of the first-order DSM	17
2.4	Phase noise profile of an oscillator.	22
2.5	Schematic of an inverter-based single-ended RO	24
2.6	Performance of state-of-the-art RO-based frequency synthesizers	25
3.1	Subharmonic injection locking implemented with an FTL	29
3.2	Subharmonic injection locking implemented with a PLL	30
3.3	MDLL for bandwidth extension.	31
3.4	DLL preceded by an edge combiner multiplying the reference	32
3.5	Cascade of an ADPLL and an edge generator with calibration achiev-	
	ing high N and low jitter	32
3.6	Block diagram of a generic frequency octupler (8×REF)	34

LIST OF FIGURES

3.7	Timing diagram of the $8 \times REF$ voltage nodes considering all associ-	
	ated delays.	35
3.8	Modulation points constituting the domain of a bijective function for ε .	40
3.9	LMS system for adaptively cancelling the periodic jitter arising from ε .	41
3.10	Probability-based LMS system for adaptively cancelling ε with the aid	
	of the BBPD output.	42
3.11	Calculation results of (3.12) before calibration settlement. (\blacklozenge : X_0 , \blacktriangledown :	
	$X_1, \blacktriangle: X_2, \blacksquare: X_3, \textcircled{\bullet}: X_4) \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots$	44
3.12	PDF of ε_i after calibration settlement	45
3.13	Calculation results of (3.12) after calibration settlement. (\blacklozenge : X_0 , \blacktriangledown :	
	$X_1, \blacktriangle: X_2, \blacksquare: X_3, \textcircled{\bullet}: X_4) \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots$	46
3.14	PDF of δ'_i	47
3.15	(a) Overall architecture and (b) operation flow chart of the proposed	
	8×REF calibration	49
3.16	Simple example of a two-point calibration with the proposed VSS-	
	LMS algorithm.	51
3.17	Calibration point to which each W_{ε_i} should be applied	52
3.18	Modified architecture of the 8×REF with the F-DL and G_{ϕ} · unit	52
3.19	CML stage for the C-DL.	53
3.20	Circuit implementation of the C-DCC	53
3.21	Circuit implementation of the F-DL	54
3.22	Final analog path of the implemented 8×REF	54
3.23	Timing diagrams depicting the effect of injection to the $8 \times \text{REF}$ cali-	
	bration in presence of ε_2 .	56
3.24	Overall architecture of the proposed IL-ADPLL	57
3.25	Chip photomicrograph.	59
3.26	Measured phase noise plot at the 8×REF output	60

3.27	Measured phase noise plot at 8-GHz output.	61
3.28	Measured spectra at 8-GHz output	62
3.29	Measured phase noise plot at 16-GHz output.	63
3.30	Measured spectra at 16-GHz output.	64
3.31	Transient behaviors of $W_{arepsilon}$	65
3.32	FoM comparisons with prior state-of-the-art works	68
4.1	Typical LDO integrated to mitigate the supply sensitivity of an RO-	
	based DCO.	69
4.2	Depiction of prior arts with RO supply noise compensation	71
4.3	Conventional RO-based ADPLL with a DCR	72
4.4	Thévenin equivalent circuit of the DCR with respect to the RO under	
	a one-bit transition in $D_{\rm FTW}.~\alpha$ is determined by $D_{\rm FTW}$ at which the	
	transition takes place.	73
4.5	Overall block diagram of the ADPLL with the proposed ACSC	74
4.6	Circuit implementation of the passive filters	75
4.7	Signal flow diagrams depicting (a) $H(s)$ and (b) $H_{\rm R}(s)$ each given (4.3).	77
4.8	Comparison between frequency responses of $O(s)$ with a single-pole	
	EA and a two-pole EA.	79
4.9	(a) Comparison between $ ilde{H}(s) $ and $ \hat{H}(s) $. (b) $H_{ m mid}$ over $k_{ m s,DCR}$	79
4.10	Magnitudes of the noise transfer functions	82
4.11	PSD mask used for the optimization of the ACSC parameters	83
4.12	Frequency response of the ACSC in conjunction with the ADPLL	
	where $f_{\rm p,H}$ and $f_{\rm p,EA}$ coincide with $f_{\rm z,PLL}$ and $f_{\rm BW,PLL}$, respectively.	83
4.13	Consideration for determining the device parameters of (a) the HPF	
	and (b) the EA. The resulting $W_{\rm EA}$ satisfies $f_{\rm BW,PLL} \leq f_{\rm p,EA}$	84
4.14	Simulated phase noise of the free-running 8-GHz clock.	85

LIST OF FIGURES

4.15	Circuit implementation of the DCR and the R-DCR	86
4.16	Simulated discrepancy between $k_{s,DCR}$ and $\hat{k}_{s,DCR}$ over D_{FTW}^{I}	87
4.17	Simulated $ H(s) $ in the typical corner. The cut-off frequency at a few	
	gigahertz is attributed to parasitic capacitance at $V_{\text{DD,I}}$	87
4.18	Simulated statistics of H_{mid} in eight different PVT corners	89
4.19	Chip photomicrograph.	90
4.20	Measured phase noise without any injected supply noise (when the	
	ACSC is enabled).	91
4.21	Measured phase noise under a 1-MHz sinusoidal supply noise	92
4.22	Transient waveforms of $V_{\rm DD}$ and $V_{\rm DD,I}$ that are observed through an	
	on-chip unity gain buffer	92
4.23	Measured (a) $\Delta Spur$ and (b) $\sigma_{\rm jitter}$ under sinusoidal supply noise	93
4.24	Measured phase noise under a 20-mV $_{rms}$ white supply noise. \hdots	94
4.25	Measured peak-to-peak time interval error (TIE) under a 20-m $\!V_{rms}$	
	white supply noise.	94
4.26	(a) σ_{jitter} under white supply noise and (b) the resulting FoM _{SNC} . With-	
	out the ACSC, the phase lock fails when $V_{\rm rms} > 30$ mV	95
4.27	(a) $\Delta Spur$ and (b) FoM _{SNC} versus static supply variation	95
4.28	Pie chart for power consumption breakdown.	96
A.1	Modified Fig. 3.10 with the finite ADPLL bandwidth accounted for.	102
A.2	Calculation results of (3.12) given $d = \begin{bmatrix} -1 & -1 & -1 & -1 & 1 & 1 & 1 \end{bmatrix}^{T}$ after	
	calibration settlement with the finite ADPLL bandwidth accounted for.	
	$(\blacklozenge: X_0, \forall: X_1, \blacktriangle: X_2, \blacksquare: X_3, \spadesuit: X_4)$	103
	(,,,,,,,, .	
B.1	Simplified representation of the passive filters	106

Chapter 1

Introduction

Driven by the incessant development of modern communication technologies, computer networks nowadays connect billions of electronic devices all over the globe [1], making our lives rich and evolving. In this era of internet-of-things (IoT), the foremost demand imposed on a data center is the constant advance of data access capacity; speeding up its chip-to-chip/module wired communication bandwidth is the key for successfully managing the massive data traffic. Up to this date, the aggressive scaling of modern integrated circuit (IC) technology, enabling very large-scale integration (VLSI), has been the main contributor to meeting this requirement. Nonetheless, it has been after all the designers' knowledge and practice that drew the best of the given technology into an IC, fostering a stand-alone field called high-speed¹ link, or SerDes (Serializer/Deserializer), design.

Over the past decades, the desire for low-cost IC has mainly resulted in complementary metal-oxide-semiconductor (CMOS) predominating over other technologies, with its device feature size having been shrunk continuously, the latest one reaching a few nanometers [3]. The transit frequency of a short-channel transistor, which benchmarks the intrinsic speed limit of a technology, tends to be inversely proportional to

¹Despite the absence of an exact criterion, 'high-speed' commonly refers to data rate > Gb/s. Note that, since 2014, the International Solid-State Circuits Conference (ISSCC) has been using the term 'ultra high-speed' with the introduction of a 60-Gb/s link [2].



Figure 1.1: Data rates of state-of-the-art SerDes (transceivers) versus technology node.

the gate length [4], meaning that the scaling also advances the uppermost bandwidth of data communication ICs. This relation is also evident from the trends of published state-of-the-art SerDes² data rates and technology feature sizes, as plotted in Fig. 1.1.

Fig. 1.2 shows the architecture of a generic SerDes. At the transmitter (TX), a parallel data bus from the preceding digital layer is serialized by a high-frequency clock synthesized on-chip from an external low-frequency crystal reference (XO). The data travels through an interconnecting channel, whose ac characteristic varies upon applications, and then arrives at the receiver (RX). The RX equalizes the input and conducts clock and data recovery (CDR), the recovered data of which is processed by the following digital layer. An important interpretation made on such a system is that the horizontal (timing) characteristics of its critical voltages essentially rely on the quality of the given clock, which is in general represented by a metric *jitter*.

This dissertation explores a low-cost method for realizing the clock domain of high-speed serial links that suffices modern industry standards. Two major drawbacks of a high-frequency, *inductor-less* oscillator are investigated, and a solution for each is developed and verified via prototype design and experiments. The first embodiment is

²From presentations at the ISSCC from 2014 to 2022.



Figure 1.2: Depiction of a generic SerDes.

a phase-locked loop (PLL) that fully exploits the digital design paradigm to overcome the poor quality of the oscillator and the variability to environmental factors. The second embodiment is a PLL that outputs stable clock in presence of power supply noise, validating its feasibility in power-hungry VLSI systems. These two techniques together demonstrate a possibility that an inductor-less oscillator may become a candidate for future low-cost SerDes implementations.

1.1 Motivation

Several classifications and standards of wireline communication have been defined according to the characteristics of the target interface, e.g., physical configuration and channel response, as they primarily determine the SerDes architecture. While Common Electrical I/O (CEI) standards published by Optical Internetworking Forum (OIF) are adopted in a wide range of applications [5], Peripheral Component Interconnect Express (PCIe) is the most general in use for medium-reach (MR), chip-to-chip interfaces [6]. Despite the diversity, the data-rate trends of such standards over the last two decades are well in line to each other, the growth being about two-fold over four

years [7]. TX/RX signal jitter and eye mask are of common specification items in all standards, and the requirements thereof directly scale with the unit interval (UI), which is equal to the inverse of the data rate. Power consumption is also closely related to the data rate and is always desired to be minimized, especially when it comes to a data center, in which a myriad of ICs are concentrated.

To curtail the cost for intellectual property (IP) development in advanced technology nodes, recent developers are paying large efforts to supporting multi-standards and backward compatibility, the latter obligated in some standards, e.g., PCIe and DisplayPort. The most critical challenge for achieving this is to enable a single-chip SerDes to operate with a wide range of data rate and give proper functionalities for diverse protocols at the same time. However, this complicates the overall architecture and brings additional design obstacles, particularly at the maximum data rate, due to some unavoidable hardware added to critical nodes.

Conclusively, today's prospective SerDes designers' task sums up to 'maximizing data rate and range' and 'minimizing hardware overhead and power'. The following sections will focus on examining challenges and opportunities that lie in the clocking of modern SerDes.

1.1.1 Clocking in High-Speed Serial Links

Clocking in a SerDes classically refers to the generation of clock signals with the frequency and timing margin compliant with specified data rate and TX/RX architecture. However, two common properties of modern SerDes have altered this definition to a broader extent: *sub-rate* and *multi-lane* architectures. These give rise to additional hardware elements, the design of which can be as complex as that of frequency synthesis itself. In this subsection, strategies for clocking in high-speed TX and RX respectively will be reviewed in brief.



Figure 1.3: Abstract diagrams of TX architectures.

Fig. 1.3(a) is the abstract diagram of a full-rate TX, whose output data stream is synchronized by a clock with the frequency equal to the data rate (or baud rate in the strict sense). A group of properly integer-divided clocks are there for aggregating the incoming parallel data bus through multiplexing. The final re-timing removes out the jitter associated with the previous latches/flip-flops and multiplexers, but this comes at the expense of high power dissipation largely due to data/clock signals operating at the very limit of the implemented technology. A handful alternative for overcoming this shortcoming is a half-rate architecture [8], as depicted in Fig. 1.3(b). A multiplexer driven by a half-rate clock outputs the final data stream, omitting the use of a fullrate clock throughout the overall system. This relaxes not only the design constraints on clock generation and fan-out buffering but also the timing window of critical data paths. In result, the required power consumption of the overall TX can be significantly reduced. However, since the driving multiplexer exploits both the rising and falling edges of the clock, serious distortion on the output eye may exist if the clock duty-cycle is not treated carefully. In order to accommodate even tougher data rate or to achieve higher power efficiency, this sub-rate clocking may be expanded further; quarter-rate clocking [9] once again halves the maximum required frequency of the system but, this time, requires an aligned four-phase (or quadrature) clock —ushering in the multiphase clocking regime.

The emergence of sub-rate clocking is also observed in the chronicle of RX architectures. As in Fig. 1.2, an RX, in its essential, comprises a series of equalizers that compensate for the voltage distortion present in the incoming data, followed by a decision circuit and a CDR engine. Its objective is to recover the transmitted data with the least possible bit-error rate (BER) by maximizing both the horizontal and vertical margin of the decision eye. The decision feedback equalizer (DFE) therein ideally eliminates the voltage tails of previous data symbols, namely post-cursor inter-symbol interference (ISI), thereby acquiring adequate decision eye margin. This, however, requires a group of high-frequency flip-flops for data and clock paths, consuming a significant portion of the RX power. Thus, we may surmise that its sub-rate counterparts [10], [11] may alleviate this concern in a similar sense with the aforementioned evolution of TX architectures. Nevertheless, a half/quarter-rate RX does not necessarily imply the use of two/four-phase clock. Many robust, versatile CDR circuits exploit both the data and edge samples [12], or even three samples per data [13], doubling/tripling the required clock phases.

Despite the popularity of sub-rate clocking, a TX beyond quarter-rate scheme, e.g., 1/8-rate, is very seldom reported in both literature and industry, perhaps because the generation and distribution of multi-phase clock entail a certain level of hardware overhead, which will be further discussed in the following section. On the other hand, the required number of clock phases in RX seems to be growing even today, being fueled by the recent development of analog-to-digital converter (ADC)-based receivers. To overcome the limited channel bandwidth at high data rates, multi-level modulation, such as pulse amplitude modulation 4-level (PAM4), is being more favored over the traditional non-return-to-zero (NRZ) transmission. However, the reduced eye margin of such a format inevitably necessitates the implementation of forward error correction (FEC) to relax the requirement on its raw BER [14]. This, in line with the technology scaling, resulted in the migration of equalization³ into one digital signal processor (DSP), with the data samplers replaced by an ADC. Eventually, this architecture has become the mainstream of the modern RX thanks to its good robustness, flexibility, and equalization linearity. As one may easily guess, the key ingredient for this transformation is a robust, high-speed ADC. Due to the digital-friendly nature, the successive approximation register (SAR) ADC has been the primary choice over the flash or pipeline ADC [15]. However, the sequential operation makes it hard to operate at high sampling rate, limiting its speed at ~ 1 GS/s even with the latest technology node [16]. Therefore, designers decided to run multiple ADC slices in parallel, i.e., in a time-interleaved (TI) manner, necessitating the sub-rate clocking. In 2019, Ali et al. [17] presented a 56-Gb/s link with an 8×4 TI SAR ADC; its first-rank track-and-holds (T/Hs) use quadrature clock. In 2020, a 112-Gb/s link [18] used a 6×6 scheme, and, in 2022, Segal *et al.* [19] achieved 224 Gb/s with a 16×4 topology. These state-of-the-art works evidence that TI has been one of the major foundations for the recent data rate expansion. Note that, in comparison with a sub-rate analog RX⁴, an ADC-based RX needs additional sophistication at clock, e.g., duty-cycle modulation, but still shares the same requirement that the clock phases should be de-skewed.

A common method to further overcoming limited per-pin bandwidth is to simply implementing several SerDes lanes in a single chip. This, obviously, comes at the proportional multiplication of area and power consumption. Upon mitigating this cost, the major concerns designers must go through lie in the clocking architecture. Clock generation (or synthesis) is achieved by an oscillator, which, by its very basis, can be classified into LC-based and RO-based one, the former predominating the latter in many (perhaps most) SerDes due to its superior jitter performance. However, the inductor therein, which is generally drawn in the top, thick metal layer for achieving

³A continuous-time linear equalizer (CTLE) or some other analog circuits, such as the inverter-based filter proposed by Zheng *et al.* [20], may still precede the ADC for better performance.

⁴An analog RX refers to one with an analog DFE, distinguishing it from ADC-based RX.

low noise, consumes a significantly large area, making designers to hesitate simply copy-and-pasting SerDes lanes. Further, multiple LC tanks placed in a single chip (within a distance of several milimeters) interfere with each other, perturbing the oscillation frequency as well as the noise profile in a quite unpredictable manner [21]. A good alternative for multi-lane clocking is to adopt only a single (global) LC oscillator and distribute the generated clock to each lane by buffering. However, with this topology, the distribution power now surfaces to the design consideration since the high-frequency clock signals should travel through a distance of a few hundred micrometers to a few millimeters, which results in very high load capacitance. Note that, in a very stringent condition, additional inductors for shunt/series-peaked clock buffering may be required to satisfy the jitter specification [22]. This global clock generation, which generally does not rely on the incoming data of any lane, also obligates the RX to perform phase interpolator (PI)-based CDR [23]. To simultaneously track the frequency offset between the global clock and the incoming data, the PI output should be as accurate as we intend, making its linearity the main design point. To do so, the PI should be fed by as many interpolation basis as possible, once again emphasizing the importance of the multi-phase clocking.

The above discussions highlight what are the key demands laid on the clocking of a modern —multi-standard, multi-lane, and low-cost —SerDes. Among them, circuit methodologies for multi-phase clock generation will be briefly studied in the following subsection. Note that the investigation on frequency synthesis will be made in the next chapter.

1.1.2 Multi-Phase, High-Frequency Clock Conversion

Upon starting from a single-phase (typically differential) clock, e.g., output of an LC oscillator, one may use a polyphase filter to generate a quadrature clock. A passive [24] and an active [25] filter respectively consume large area and power, and they



(a) Cross-coupled LC oscillators.



(b) LC-based I/Q divider.



(c) RO-based I/Q divider.

Figure 1.4: Quadrature phase generation using injection locking.

commonly suffer from inaccurate output phase and vulnerability to process, voltage, and temperature (PVT) variations. Alternatively, injection locking can be exploited to generate a quadrature clock. The most straightforward method is to place two identical resonators that are coupled to each other [26], as in Fig. 1.4(a). This, however, comes at the cost of not only larger area but also higher power consumption due to its tradeoff between the phase accuracy and noise. In addition, in multi-lane SerDes, since the distribution buffers should convey multiple signals, the power consumption would indeed increase further. Moreover, unavoidable process mismatches in buffers and routing metals as well as crosstalk from some near signals would result in phase offsets in the destinated clocks, requiring additional calibration at each lane.

The second topology using injection is to implement a divider circuit [27], as shown in Fig. 1.4(b). A clock with twice the target frequency is injected into two LC dividers, outputting a quadrature clock whose accuracy is not traded with the phase noise. However, despite the form factor being not as large as an LC oscillator, it requires a set of inductors, still being problematic in multi-lane realization. Instead, as depicted in Fig. 1.4(c), injection can be applied to a two-stage, resistor-loaded ring oscillator (RO) [28], allowing very compact design and a wide locking range. Here, despite the inferior phase noise, the RO does not contribute much to the divider output jitter by virtue of the noise suppression nature of the injection locking.

The common issue in the above injection topologies is that they need a twicefrequency oscillator, counteracting the advantage of sub-rate clocking. Moreover, those without using an RO are limited only to quadrature generation⁵, being not suitable for modern ADC-based RXs. A delay-locked loop (DLL) [29] is a good candidate for overcoming this; the input clock frequency need not be higher, and clock phases as many as required can be generated if the circuit bandwidth is met. Further, a DLL avoids the jitter accumulation that is present in an oscillator, showing low output noise and low power consumption. Nevertheless, it should be noted that the finite mismatch between delay stages must be removed out for the proper usage. Another alternative is to sub-harmonically inject an RO by a clean input clock [30]. Although it alleviates the burden for high-frequency clock synthesis, the capability of injection for suppressing the high RO noise is reduced. However, [18] has proven that this scheme readily satisfies the jitter specification of SerDes with the data rate of 56 Gbaud/s.

1.2 Dissertation Objectives

With these advances thus far, the well-accepted clocking architecture for today's SerDes can be dissected into the following order: 1) LC-based synthesis of single-phase, sub-rate clock, 2) distribution to each lane, 3) a DLL/RO for PI multi-phase

⁵A series of dividers may expand the output phase number but severely complicates the implementation.

input generation.⁶ Then, is it possible to simplify this whole process into just one step, thereby achieving even lower cost? With the RO seeming to be a vital clue, the rest of this dissertation will find an answer to this question.

The basics of frequency synthesis including the fundamental circuit blocks will be disccused in Chapter 2. The push for digital-friendly implementation of frequency synthesizers will be explained along with its pros and cons. Further, the two main reasons why RO-based frequency synthesis has lost its dominance in the high-speed era will be elaborated.

A solution for the first flaw of an RO-based frequency synthesizer—the inferior phase noise—is proposed in Chapter 3. Therein, Section 1 studies prior state-of-theart solutions and discusses what challenges are yet to be addressed. Using a reference octupler with a probability-based phase error calibration, Section 2 proposes a highbandwidth RO noise suppression technique. Section 3 explains the building blocks of the rest of the presented frequency synthesizer. The measurement results of a prototype chip implemented in a 40-nm CMOS technology are reported in Section 4, followed by final remarks on the work in Section 5.

Chapter 4 gives a solution for the second flaw—the susceptibility to supply noise. Therein, Section 1 explores prior solutions and examines the pros and cons for each. Section 2 introduces a novel analog technique for supply noise compensation in a highfrequency RO and then delves into the circuit design and optimization. The building blocks for the rest of the frequency synthesizer is explained in Section 3. The measurement results of a prototype chip implemented in a 40-nm CMOS technology are reported in Section 4, followed by a summary of the work in Section 5.

Chapter 5 summarizes this dissertation and finally discusses limitations of the proposed works and possible future research directions.

⁶Although a TX does not need to rotate the driving clock phase, a PI is commonly implemented to optimize the timing window of critical paths.

Chapter 2

RO-Based High-Frequency Synthesis

The output clock of every existent frequency synthesizer stems from a (preferably) clean reference signal that is provided either on-chip or off-chip. With its form factor and manufacture cost being suitable for a board-level integration , a (quartz) XO gives the highest frequency accuracy and stability and thus has become the prerequisite for most system designs. In a system operating at a few GHz or higher, since the manufacture-available XO frequency ranges only up to a few hundred MHz, this essentially rules out the frequency modulation using the direct digital synthesis (DDS), the maximum achievable output frequency of which is the half the reference frequency. Consequently, its usage has been completely oppressed by the frequency synthesis through a PLL over the last two decades, and therefore—'PLL' is now used as the fair equivalent to 'frequency synthesizer'. The first section of this chapter will give the fundamental consideration and analysis of PLL design.

2.1 Phase-Locked Loop Fundamentals

Fig. 2.1 shows the block diagram of a PLL in its simplest form. The oscillator clock is first divided and then is fed to a phase detector (PD), or often phase-frequency detector (PFD), that compares its phase/frequency with the reference clock. A loop



Figure 2.1: Schematic of a generic PLL.

filter (LF) processes the resulting information and accordingly modulates the oscillator frequency. These components together constitute a negative feedback loop, continuously generating clock with the desired frequency. In a steady state, with a properly designed loop, the output frequency is maintained to be $f_{out} = N f_{ref}$ where N and $f_{\rm ref}$ respectively denote the integer division ratio and the reference clock frequency. It then can be inferred that the modulation resolution is equal to f_{ref} . It is possible that the division ratio be a fractional number, i.e., $N + \alpha$, so that the PLL achieves a very fine resolution, which is obligated in most wireless applications to meet the stringent channel spacing requirement. In comparison, the data rates of SerDes standards tend to be much more discrete to each other, not essentially needing this factional-N PLL. Further, as well as the hardware being more complex than an integer-N counterpart, a fractional-N PLL inevitably introduces additional noise and spurious tones, which are never desired for meeting the stringent timing specification of a modern SerDes. However, the rising demand for multi-standard IP desires fractional-N PLLs to save the cost for multiple XOs. It should also be mentioned that, since a SerDes, being a highly synchronous system, radiates high electromagnetic energy that can interfere nearby electronics, it often requires a spread spectrum clocking (SSC), which uses a fractional-N PLL. Nevertheless, the scope of this dissertation includes only integer-N PLLs.



Figure 2.2: Third-order analog LF preceded by a charge pump.

The LF of a PLL is classically realized by a combination of resistor(s) and capacitor(s), as depicted in Fig. 2.2. A charge pump that precedes the LF generates UP/DN current pulse whose width is, in ideal, proportional to the phase difference between the reference and the divider output. Here, the resistor plays a key role to stabilizing the PLL; it contributes a real left-half-plane zero in the loop transmission, providing phase margin to some extent. An important remark to make is that the mismatch between UP and DN currents gives rise to periodic, undesired ripple at the control voltage, therefore contributing spur at f_{ref} . To suppress (not completely remove) this, an additional capacitor with a large value is added in shunt with the filter, arriving at the well-accepted (third order) Type-II PLL topology. The main design concern thereof is that its overall jitter performance trades with the power consumption of the charge pump and the area for the capacitors, which are often implemented off-chip if too large, e.g. over a few hundred picofarads.¹ Capacitors may be implemented using MOS capacitors, taking much less area, but would suffer from the inferior linearity and leakage current. Most importantly, the passive elements along with the charge pump cannot take the full advantage of technology scaling, and the cost for design and verification might hamper its portability to other technologies.

¹Such a large capacitance is required when a designer intends to achieve low PLL bandwidth so as to fully reduce the spurs.

2.2 Toward All-Digital Regime

As mentioned, the technology scaling, whose benefits stand out in digital ICs, has raised some skeptical visions to the above analog PLL. Starting from the work in 1960 by Westlake [31], which reported a method of connecting a loop filter to a digitally controlled oscillator (DCO), extensive efforts on replacing the conventional analog PLL components to digital counterparts have been made. In 1968, Gupta [32] gave the first theoretical analysis of implementing a digital loop filter, and, in 1978, D'Andrea [33] reported the effect of replacing a PD with a binary quantizer. However, never did a fully implemented digital PLL that satisfies the actual industry specifications disclose to the world in these early days. The vital reason, perhaps, was that no one came up with an idea to realize a DCO with *low-jitter* that is comparable with the existing analog ones. This resulted in researchers leaving the digital transformation behind and, instead, sticking to further advancing the performance and addressing the issues of analog PLLs.

Then, in 2004, Staszewski [34] demonstrated the first industry-applicable, fully integrated all-digital PLL (ADPLL) that operates in a digitally synchronous fixed-point phase domain. In this work, the number of rising transitions of the DCO output and the reference clock are counted and then compared by a synchronous arithmetic PD. The result is then filtered by a digital loop filter (DLF). Here, to avoid metastability issues, the comparison was performed in the same clock domain. The synchronous operation is achieved by oversampling the reference clock by the DCO output. An important aspect of this work is that, due to the edge counting method, phase quantization resolution is not acceptable for low-jitter operations. Therefore, Staszewski chose to correct the quantization error by means of a circuit named time-to-digital converter (TDC). The TDC measures the delay difference between the reference and the divided clock with the resolution of a single inverter delay, which is in the order of a few tens of

picoseconds with the implemented technology. In this way, the arithmetic PD and the TDC together replace the conventional analog PFD. A more crucial aspect of this PLL is that the DCO is preceded by a high-frequency delta-sigma modulator ($\Delta\Sigma M$) that actually solved the existing bottleneck of an ADPLL; The DSM frequently dithers the digital modulation word, thereby effectively suppressing the finite quantization noise of the DCO.

Quantization Noise

To gain an insight onto how a $\Delta\Sigma M$ works, first assume that the generic DCO frequency resolution is Δf_{DCO} . Then, its deviation from the exact frequency that the modulation word intends will be an uniformly distributed random variable, the magnitude of which is within $\Delta f_{\text{DCO}}/2$, readily giving its variance as $(\Delta f_{\text{DCO}})^2/12$. It is then converted to the phase quantity through an integration by the DCO represented by 1/f, followed by zero-order hold operation by the PD (TDC in [34]). Since the PD propagates the resulting phase error for each reference cycle, its noise is uniformly distributed from dc to $f_{\text{ref}}/2$, thus giving its single-sided power spectral density, or phase noise, as [35]

$$\mathcal{L}(f) = \frac{1}{12} \left(\frac{\Delta f_{\rm DCO}}{f}\right)^2 \frac{1}{f_{\rm ref}} \left(\operatorname{sinc} \frac{f}{f_{\rm ref}}\right)^2.$$
(2.1)

Recalling that this noise is added at the DCO output, it is high-pass filtered by the PLL loop, suppressing the +20-dB/dec upconversion region within the PLL bandwidth. Nevertheless, it is extremely difficult to realize Δf_{DCO} that actually gives a reasonable phase noise for real-application usage; considering the required frequency tuning range and the complexity for layout/routing for high-bit tuning word, a few megahertz is the realizable compromise for GHz operations. Then, by implementing a $\Delta \Sigma M$ between the control word and the DCO, a quite different phase noise is expected. The



Figure 2.3: Schematic of the first-order DSM.

modulator, whose operation frequency, $f_{\Delta\Sigma}$, is much higher than f_{ref} , passes noiseshaped fractional bits of the DLF output to the integer bits, making the effective DCO resolution to be $\Delta f_{\text{DCO},\text{eff}} = \Delta f_{\text{DCO}}/2^{N_{\text{f}}}$ where N_{f} is the fractional wordlength. With an adequate choice of N_{f} , substituting f_{DCO} by $f_{\text{DCO},\text{eff}}$ in (2.1) results in a greatly reduced phase noise contribution.²

Despite the advantage, a $\Delta\Sigma M$ introduces its own quantization noise, or dithering noise, to the DCO output. A first-order digital $\Delta\Sigma M$, as illustrated in Fig. 2.3, dithers its output, Y, according to the incoming bit stream, X, i.e., its function can be expressed as $Y = X + (1 - z^{-1}) \cdot Q$ where Q denotes its quantization error. Thus, the continuous-time transfer function from Q to Y is easily derived by $i2e^{-i\pi/f_{\Delta\Sigma}} \cdot$ $\sin(\pi f/f_{\Delta\Sigma})$, inferring a high-pass noise shaping within the frequency range of interest. However, depending on X, the output spectrum may not be completely random, entailing some spurious tones. In order to fully randomize the pattern, a higher-order $\Delta\Sigma M$, typically realized by cascading multiple first-order $\Delta\Sigma Ms^3$, can be used at the expense of higher random noise level. Now referring to the derivation of (2.1) and assuming an n^{th} -order $\Delta\Sigma M$, it is straightforward to write its contribution to the DCO

 $^{^{2}}N_{\rm f}$ directly scales with the computation burden of the $\Delta\Sigma M$, trading with the area and power taken by the digital domain.

³This topology is commonly referred to as a multistage noise-shaping structure (MASH).

output phase noise as

$$\mathcal{L}(f) = \frac{1}{12} \left(\frac{\Delta f_{\text{DCO}}}{f}\right)^2 \frac{1}{f_{\Delta\Sigma}} \left(2\sin\frac{\pi f}{f_{\Delta\Sigma}}\right)^{2n}.$$
(2.2)

It should be noted that as the resulting control word is the one that actually connects to the DCO, dithering its frequency, $\Delta f_{\text{DCO,eff}}$ should not replace Δf_{DCO} here. This thus emphasizes that the overall advantage of adopting a $\Delta \Sigma M$ only comes with a sufficiently high $f_{\Delta \Sigma}$.

There is another interesting defect of a $\Delta\Sigma M$ that was raised by Madoglio [36]. He claimed and verified that the dithering process induces not only the high-frequency random noise but also some in-band noise degradation. The $\Delta\Sigma M$ quantization noise accumulates over each reference period, being averaged by the DCO, and then is decimated by a TDC. This accumulate-and-dump operation, as in the decimation stage of a $\Delta\Sigma$ ADC, inherits noise folding that is different from the abovementioned DCO quantization noise and is added in the PLL loop with the same manner as the TDC noise⁴ is, i.e., it propagates to the DCO, being low-pass filtered. Its (un-filtered) phase noise *floor* at the DCO output is given by

$$\mathcal{L} = 2^{2n} \left(\frac{2\pi}{12}\right)^2 \left(\frac{\Delta f_{\text{DCO}}}{f_{\text{ref}}}\right)^2 \int_0^\pi \sin^{2(n-1)}(x) \cdot \sin^2(\frac{f_{\Delta\Sigma}}{f_{\text{ref}}}x) \, dx, \qquad (2.3)$$

which, despite being cumbersome, clearly indicates that n should be low for achieving better performance.

Nonlinear Limit Cycle

Be all told, are we done listing all noise elements introduced from the transformation into the digital realm? The answer is yet 'no'; The PD in a an ADPLL operates

⁴Of course, a TDC has its own device noise and quantization noise.

as a hard limiter, i.e, in a nonlinear way, incurring further performance non-idealities that can never be recognized under linear point-of-view. In [37], Dalt documented rigorous mathematical inductions on the nonlinear dynamics of how a one-bit phase quantizer, namely bang-bang phase detector (BBPD), affects the performance of an ADPLL. The timing jitter (with respect to the reference) of the DCO output normalized to its quantization step, τ , without loss of any generality, is expressed as the following sequence:

$$\tau_{k+1} = \tau_k + x_0 - R\psi_{k-D} - \operatorname{sgn}(\tau_{k-D})$$
(2.4)

where R and D respectively represent the ratio of integral to the proportional gain of the DLF and the total reference delay cycles of the DLF. Here, k corresponds to the kth cycle of the divided DCO clock. If a first-order loop is assumed, i.e., if R = 0 (which is typically impractical), then the peak-to-peak and root-mean-square jitter induced solely by the nonlinear behavior is given by $\tau_{pp} = 1 + 2D$ and $\sigma_{\tau} = (1 + D)/\sqrt{3}$, respectively. In fact, the jitter is not random but is rather deterministic in that τ repeats its trajectory every 2(1 + 2D) cycles, resulting in a spurious tone at the frequency equal to $f_{ref}/(2(1+2D))$. Then, in a second-order loop (R = 0), periodic trajectory is also observed at the normalized difference between the instantaneous and free-running DCO periods, ψ , which is written as

$$\psi_{k+1} = \psi_k + \operatorname{sgn}(\tau_{k+1}).$$
 (2.5)

Thus, τ and ψ together form an orbit in a steady-state condition. However, in order to stabilize the orbit, some conditions on τ and ψ should be satisfied. The constraint follows that the initial point of ψ , ψ_0 , should be within a certain interval depending on D and R. Since the number of possible orbit radius is indefinite, except for the case when $R \ge 2/(2D - 1)$, in which there is no ψ_0 giving a stable orbit, we cannot exactly predict which orbit will be the actual one. Nevertheless, we may still estimate the peak-to-peak jitter with the maximum orbit radius, which is the worst and, at the same time, the most probable case, and it is derived by

$$\tau_{\rm pp} = 2(1+D) + (1+D)R + (1+D)^3 R^2 + O(R^3), \tag{2.6}$$

indicating that both D and R are desired to be minimized.

These result also seems to be in analogy with the linear characteristic of a fundamental digital control loop in that the overall digital latency should be minimized so as to acquire enough phase margin. Particularly for an ADPLL, Bergmans [38] set up a rule-of-thumb for D that it should be at least five times smaller than the inverse of the normalized natural frequency. Stepping further from this, an innovation can be realized by making the proportional path to bypass the DLF but to directly modulate the DCO. The primary function of a DLF is to digitally integrate, i.e., lowpass filtering, the PFD output, and never was its duty to provide the proportional path. Performing the above nonlinear analysis with such a topology, we arrive at a somewhat different conclusion regarding the limit cycle. Staring from replacing the last term in (2.4) by sgn(τ_k), we may derive the condition for a stable trajectory of τ for M > D + 1. If $M \leq D + 1$, the ψ_{k-D} term will partially negate the τ trajectory direction at the plane-transition moments ($\tau = 0$), continuously decreasing the orbit radius. Eventually, the trajectory will converge to the origin and then maintain around it in a chaotic manner. It is therefore inferred that the additional nonlinearityinduced jitter is no more existent. However, in practice, since there is at least certain propagation delay from the PFD output to the DCO, it is inevitable that an ADPLL suffers from the limit cycle phenomenon to some degree.

In strive for optimization regarding the limit cycle, Marucci [39] conducted further analysis on the phenomenon and concluded that the minimum achievable jitter of an ADPLL is drawn when the limit-cycle-induced jitter and the intrinsic DCO noise coincide. Jang [40] then observed that, at this condition, the autocorrelation of the BBPD output with (1 + 2D)-lag becomes near zero and so implemented a simple digital adaptation logic that optimizes the DLF gains to track the minimum output jitter.

2.3 RO Design Challenges

Given the knowledge thus far, we now realize that the overall jitter resulting from the migration onto digital domain is still dictated by the analog noise performance. In view of this, this section will review the phase noise of an typical oscillator and discuss two major challenges of replacing an LC oscillator by an RO.

2.3.1 Oscillator Phase Noise

An ideal oscillator generates a perfect single-tone signal with frequency f_0 , i.e., its output waveform is written as $v(t) = A \cdot \cos(2\pi f_0 t + \phi_0)$ where A is the amplitude and ϕ_0 is an arbitrary phase constant. However, in reality, various sources, both internal and external, perturb the signal, giving rise to time-varying fluctuations in both amplitude and phase. Among them, the phase purity is the one that matters in clocking systems, and, although the amplitude variation may be translated into phase noise, it is hardly a concern since a limiter circuit well quenches its magnitude. Then, for a small amount of time-varying random phase quantity $\phi_n(t)$, the oscillation signal becomes

$$v(t) \simeq A \left\{ \cos(2\pi f_0 t) - \phi_n(t) \cdot \sin(2\pi f_0 t) \right\},$$
(2.7)

indicating that, in frequency domain, the spectrum of $\phi_n(t)$ is added near f_0 . Such characteristic leads to the definition of *phase noise* of a clock signal: the single-sideband power by phase fluctuation with respect to the carrier power. This quantity is com-



Figure 2.4: Phase noise profile of an oscillator.

monly represented by $\mathcal{L}(f)$ in the unit of dBc/Hz where f equals the offset frequency from f_0 . Here, for the types of signals under consideration, the noise power at both upper- and lower-side bands are coherent to each other, i.e., equal in intensity.

 $\mathcal{L}(f)$ of all CMOS-realized oscillators share a common profile in that it consists of three different slope $-1/f^3$, $1/f^2$, and $1/f^0$ —regions, as shown in Fig. 2.4. The two former regions arise from the $1/f^2$ upconversion of flicker (or colored) and white noise, respectively, by the oscillator's inherit phase accumulation, or integration, and the last from residual noise sources outside the oscillator.⁵ Therefore, noise of an oscillator, with respective coefficients, takes the general form

$$\mathcal{L}(f) = \frac{a_3}{f^3} + \frac{a_2}{f^2} + a_0.$$
(2.8)

The frequency-domain profile of random signal is well characterized by power spectral density, so it readily makes sense of using $\mathcal{L}(f)$ for phase noise measurement. Then, what about deterministic noise in the oscillator? A single-tone noise is indeed modulated onto f_0 , but its power exists only at a certain, single frequency, and thus cannot be partitioned into 1-Hz bandwidth. Therefore, such a noise, being called as a *spur*, is simply characterized by its power relative to the carrier power in the unit of dBc. It is worth noting that spurs in an ADPLL come from various sources, e.g., quantization nonlinearity as mentioned and supply pulling from nearby circuits, and

⁵The aforementioned amplitude fluctuation in the oscillator may take some portion of this region.

should be treated carefully so as to avoid additional timing errors.

2.3.2 Challenge 1: High Flicker Noise

Given the fact that phase noise refers to the amount of additive sideband near f_0 , the phase purity is usually represented by the quality factor defined by $Q = f_0/\Delta f$ where Δf is the -3-dB bandwidth of the output spectrum. A theoretical Q can be calculated by the ratio of the oscillator's stored energy to dissipated energy per cycle, but for an RO, such value cannot be obtained since it does not have an energy-storing element, i.e., an inductor. Nevertheless, its phase noise can be still estimated by using its feedback loop transfer function or impulse sensitivity function (ISF). In particular, using the latter method, Hajimiri [41] made some useful implications on the RO thermal noise characteristics. For an RO comprising CMOS delay cells, e.g., simple inverter chains, its thermal phase noise is derived to be

$$\mathcal{L}(f) \simeq \frac{8}{3} \cdot \frac{kT}{P} \cdot \frac{\gamma V_{\rm DD}}{V_{\rm ov}} \cdot \frac{f_0^2}{f}, \qquad (2.9)$$

where V_{ov} is the gate overdrive voltage at the middle of clock transitions, and P is the dissipated power. This indicates that, given the same P and f_0 , the phase noise does not rely on the number of stages. This give us a delightful highlight that the expansion of clock phases for RO-based sub-rate clocking would not further degrade the clock quality. In contrast, if a differential current-mode logic (CML) stage is used as the delay cell, then the RO phase noise becomes proportional to the number of stages. For this reason, the ROs presented in this dissertation adopt the CMOS stage to achieve the lowest possible jitter.

Being dominant in low-frequency region, flicker noise, which originates from fluctuation of the gate surface potential, makes the RO slowly wander around its frequency, the resulting jitter growing with the square of elapsed time. The empirical


Figure 2.5: Schematic of an inverter-based single-ended RO.

model for flicker noise [42] indicates that its magnitude is inversely proportional to the transistor dimension. Then, how does it relate to the phase noise of an CMOS RO? Abidi answered this question in [43], highlighting that, as one may have expected, large device dimension results in lower flicker noise to phase noise upconversion, i.e., lower a_3 in (2.8), at the expense of more power consumption. Further, it is proven that high number of stages is preferred. Due to this straightforward guideline and the high-pass filtering nature given by a PLL, flicker noise had not been treated with its importance beyond that of thermal noise. However, when it comes to a high-frequency regime, we face a different situation. Somewhat unlike LC resonators, the oscillation frequency of an RO is determined by the actual propagation time of its delay cells, meaning that the maximum value highly relies on the transit frequency of the given technology. Thus, given a certain number of stages for multi-phase generation, the channel length should often be minimized so as to meet the desired frequency. In other words, increasing the device dimension for enhancing jitter performance is restrained. To clarify this point, suppose an RO with inverter delay cells with width W and length L connecting no additional load elements, as shown in Fig. 2.5. Here, the load capacitance, including those for metal routing, is directly proportional to WL while the driving strength is determined by W/L. Thus, the achieved oscillation frequency decreases quadratically with L while being independent to W. Of course, we may still



Figure 2.6: Performance of state-of-the-art RO-based frequency synthesizers.

enlarge W for lowering the flicker noise but only to some point allowed by power budget left over after the high-frequency realization. In result, the flicker noise corner, which is the frequency point where flicker noise and thermal noise cross at, is placed at a very high frequency, e.g., over 1 MHz with L < 65 nm [44]—making a significant portion of the flicker noise no more a wandering component. Due to this reason, As evident from Fig. 2.6, RO-based frequency synthesizers with the output frequency over 8 GHz and multiplication factor over 30, achieving rms jitter less than 300 fs, are yet to be disclosed in literature, having been forcing designers to inevitably adopt LC-based clocking for high-speed interfaces.

2.3.3 Challenge 2: High Supply Noise Sensitivity

Another major challenge in the design of an RO comes from the severe degradation of jitter performance by supply noise. In a typical SerDes, a very high number of circuit elements, clocked at high-frequency clock, are integrated together, consuming intensive power from the supply, thereby inducing significant static/dynamic IR drop to the on-chip supply voltage. What is even worse is that since all circuit blocks are coupled together via power grid and substrate, dynamic power drawn by each block tends to interfere neighbors, resulting in additional performance deterioration of noise-sensitive analog circuits, especially oscillators. An LC oscillator determines its frequency mainly according to the supply-independent passive elements and therefore entails immunity to supply variation to some extent. In a CML RO, its differential stage well suppresses common-mode variations, giving acceptable supply noise sensitivity. In contrast, due to the fact that an inverter delay is a direct function of supply voltage, a CMOS RO is the most prone to supply noise among all feasible topologies.

Then, how exactly does supply noise appear at the output clock? Even for a DCO, supply voltage modulates f_{out} as if it is the control voltage of a VCO, therefore translating its noise to the output spectrum. In this sense, assuming supply noise with sufficiently small magnitude, the noise transfer function from its single-sided spectrum, $S_{v_{DD}}(f)$, to $\mathcal{L}(f)$ is simply given by $(K_{v_{DD}}/f)^2$ where $K_{v_{DD}}$ is the modulation gain in V/Hz and the 1/f term accounts for the oscillator integration, i.e.,

$$\mathcal{L}(f) = \left(\frac{K_{v_{\text{DD}}}}{2f}\right)^2 \cdot S_{v_{\text{DD}}}(f).$$
(2.10)

However, as mentioned, $K_{v_{DD}}$ of a CMOS RO is typically very high, and with high magnitude of supply noise itself, nonlinear effects should be taken into account in its frequency modulation mechanism. Suppose that an RO is modulated by a single-tone supply noise given by $v_n(t) = a_n \cdot \cos(2\pi f_n t)$. Then, using Bessel functions, the phase

deviation resulting from the integration over time gives the oscillator output waveform (assuming sinusoidal signal as in (2.7)) as

$$v(t) = A \sum_{k=-\infty}^{\infty} J_k(\beta_n) \cdot \cos\left\{2\pi (f_0 + kf_n)t\right\}$$
(2.11)

where β_n equals $K_{v_{DD}}a_n/f_n$. This indicates that such a noise introduces spur not only at its frequency but also at its harmonics.⁶ Thus, considering that the harmonics will be less attenuated by the high-pass filtering loop, the effect of supply noise to a highly sensitive RO is greater than one may have expected.

⁶For small β_n , we may consider only $J_1(\beta_n) \simeq \beta_n/2$, and the following noise transfer function becomes equivalent to $K_{v_{\rm DD}}/f$.

Chapter 3

Filtering RO Noise

3.1 Introduction

As described, the inferior device noise is the fundamental limitation of using an RO for SerDes clock synthesis. The most effective (and perhaps the only) method to lowering the thermal noise of a CMOS delay cell with a given supply voltage is to increase its power dissipation, as explain in (2.8). However, considering that the dynamic power for charging capacitance is proportional to its operation frequency, the feasibility of this solution is very restricted in high-frequency synthesis. Instead, given the noise-filtering nature of a PLL, one can effectively suppress the noise existent in the RO while nearly not trading with the power consumption; at the cost of higher input noise contribution, one can maximize the bandwidth of the PLL to minimize the RO noise delivered to the output spectrum. However, recalling that a PLL is of a negative feedback system, there exists a maximum bandwidth that ensures the stability of the loop, which is in general represented by the phase margin. E.g., Gardner [45] set up a rule-of-thumb for a third-order Type II analog PLL that its bandwidth should be less than one-tenth of f_{ref} to ensure its stability. This then gives us a straightforward solution to extending the maximum achievable bandwidth of a PLL: use a high- f_{ref} reference clock. However, both the manufacture cost and power dissipation of an



Figure 3.1: Subharmonic injection locking implemented with an FTL.

XO, which is used in general board-level integration as mentioned earlier, increase with its generation frequency. It is also worth mentioning again that the resolution of the achievable PLL output frequency, despite its importance not being significant in a SerDes, trades with f_{ref} . Therefore, for low-cost implementation, there have been extensive attempts to deriving a PLL architecture that breaks the bandwidth limit given a low-frequency XO. One feasible topology is to subharmonically inject the reference clock edge to the RO delay cell(s) [46], as depicted in Fig. 3.1. The injection realigns the output phase error resulted from the RO noise accumulated during each reference period. By doing so, some portion of the output spectrum that is originally allocated to the RO noise is replaced by the reference noise. In order to achieve this function, the free-running oscillation frequency, $f_{\rm fr}$, should be within a certain range (lock-in range, f_L), which is determined by the physical injection strength, with respect to Nf_{ref} . Further, even if this condition is met, any amount of frequency offset would result in the degradation of phase noise suppression and unwanted spurious tones at the harmonics of f_{ref} , i.e., we desire that f_{fr} exactly equals Nf_{ref} . This necessitates a simultaneously operating frequency-tracking loop (FTL), as PVT variations should be taken into account for CMOS designs. The major limitation of such a scheme is that it cannot sufficiently suppress the flicker noise component of high-frequency ROs. Roughly speaking, injection alone acts as a single-pole filter, i.e., it attenuates the RO



Figure 3.2: Subharmonic injection locking implemented with a PLL.

noise by only 20 dB/dec for the offset frequency up to f_L [47]. Therefore, the resulting spectrum may not be much superior as compared with a mere high-bandwidth, second-order PLL, where RO noise is suppressed by 40 dB/dec for the offset frequency up to the PLL zero.

To mitigate this, one may apply injection to an RO that is already being modulated through a PLL [48], as shown in Fig. 3.2. In this topology, as the two independent loop paths adjust the output phase to their respective reference phases, they may conflict with each other, giving rise to undesired periodic jitter. Thus, the phase mismatch between them should be simultaneously eliminated by a calibration loop. Despite the additional hardware, the injection pushes the overall noise-suppressing bandwidth to a higher value than the one given solely by the PLL [49]. Further, the bandwidth extension does not trade off with PD noise contribution since its transfer function, which is different from that of reference noise, is not affected by the injection path, overall achieving better performance than an FTL-based injection.

A multiplying DLL (MDLL) [50], as illustrated in Fig. 3.3, is also a promising architecture for flicker noise suppression. The muliplexer (MUX) *physically* replaces the output clock edge by the reference edge, periodically removing out the accumulated jitter of the PLL-locked RO. This phase correction mechanism can be viewed as an extreme case of injection locking; it achieves a very high noise-suppression



Figure 3.3: MDLL for bandwidth extension.

bandwidth, e.g., $\sim f_{\rm ref}/4$ in [51], but suffers from large periodic jitter due to the phase offset between the PD and MUX paths. Therefore, as in an injection-locked PLL, the MDLL must be in concurrent with a dedicated calibration loop. The critical drawback of an MDLL lies in the select logic for the MUX operation. The select logic should completely switch the MUX output before the target clock edge arrives, otherwise, the MUX output would experience an extra delay and therefore result in periodic distortion of the output clock. Thus, although an MDLL stands as a state-of-the-art architecture for <3-GHz output in terms of jitter-power efficiency, it lacks its feasibility for highfrequency clock synthesis.

In pursuit for a further breakthrough, recent researchers have sought methods to synthesize a (on-chip) clean mid-frequency clock to be the reference clock of a high-frequency PLL, thereby further widening the bandwidth given a low-cost, lowfrequency XO. The key to realizing this is to properly interpolate the given reference edges with low noise insertion. One primitive example is a DLL with an edge combiner [52], as illustrated in Fig. 3.4. In ideal, assuming the DLL is implemented with M delay stages, each phase spacing between subsequent stages become $T_{\rm ref}/M$ under settlement. Then, if a proper edge combiner follows the generated edges, it may



Figure 3.4: DLL preceded by an edge combiner multiplying the reference.



Figure 3.5: Cascade of an ADPLL and an edge generator with calibration achieving high N and low jitter.

generate a clock with the frequency $M f_{ref}$. In the same manner with the previously discussed architectures, this requires a sophisticated calibration engine since, in reality, mismatches among the delay stages, offsets present in the DLL, and PVT variations disarrange the interpolated phases. Nevertheless, the virtue of such a topology is that it does not accumulate thermal and flicker noise, avoiding their f^2 -up-conversion and thus giving very low-noise output as compared with an oscillator [53]. Coombs [54] first applied this idea to synthesizing a low-jitter, high-frequency RO-based clock by doubling the reference clock with a delay-line-based duty-cycle calibration. Advancing from this, [55] succeeded in quadrupling the reference without noise accumulation, achieving state-of-the-art jitter-power performance with $f_{out} = 4.8$ GHz and N = 54. However, in order to compensate for the large PVT variations present in the reference multiplier and possible input duty-cycle error in the XO, the three-point calibration for the phase interpolation requires a excessively long settling time in the order of a few milliseconds at worst, which is not acceptable in industry-applicable standards. In [57], another viable method for reference multiplication is presented. However, the long calibration time remains unsolved due to the required low noise contribution of its calibration PLL.

In strive for proceeding one step ahead of these, this chapter proposes a high-frequency, high-*N* RO-based injection-locked ADPLL (IL-ADPLL) implemented with a reference octupling technique [56]. In particular, it aims for giving a low-cost solution for clocking in PCIe Gen 5/6 and possibly other SerDes standards with similar data rates. The rest of this chapter is organized as follows. Section 2 describes the proposed reference octupler and its calibration method, and the implementation of the overall IL-ADPLL is elaborated in Section 3. Section 4 presents experimental results of a prototype chip implemented in a 40-nm CMOS technology, and finally, Section 5 concludes this chapter with a summary and comparisons.



Figure 3.6: Block diagram of a generic frequency octupler (8×REF).

3.2 Proposed Reference Octupler

Consider a generic frequency octupler (8×REF) which consists of three consecutive frequency doublers, each with a delay element and an exclusive-OR (XOR) gate, as depicted in Fig. 3.6. One may easily come up with its timing requirement for successfully octupling the input clock: The high-level window of the input reference (1×ref), $\Delta T_{1\times H}$, should be $T_{ref}/2$, giving 50% duty cycle, and the delay times in the first and second doublers be $T_{ref}/4$ and $T_{ref}/8$, respectively. This then implies that the 8xREF requires a three-point calibration circuitry. However, in real CMOS implementation, both the delay element and XOR gate experience input-dependent propagation delay variations, resulting in a more complex timing constraint.

3.2.1 Delay Constraint

To elaborate on this point, it is instructive to examine the exact timing behaviors of all voltage nodes in the 8×REF. For the sake of simple illustration, let us assume that they all are of fast-switching, amplitude-limited signals, as shown in Fig. 3.7. Here, the rise-to-rise time of each delay element is explicitly drawn and is denoted by ΔT_R with the subscript preceded by "2×" or "4×", indicating the first, or second doubler. Its difference between the fall-to-fall delay is represented by Δt_{RF} with a corresponding prefix. As for each XOR gate, there are four possible transition states, the propagation delays of which all differ from each other due to the unavoidable unsymmetry in its circuit configuration and device mismatches. Therefore, we represent the XOR delay



Figure 3.7: Timing diagram of the $8 \times \text{REF}$ voltage nodes considering all associated delays.

as Δt_{ij} where $i \in \{0, 1\}$ and $j \in \{0, 1\}$ indicate the input state before transition with additional subscript "2×", "4×", or "8×". Then, letting t = 0 the rising moment at the input reference, the absolute timings at the 8×REF output edges are given by

$$\begin{cases} t_{0} = \Delta t_{2\times00} + \Delta t_{4\times00} + \Delta t_{8\times00} \\ t_{1} = \Delta t_{2\times00} + \Delta T_{4\times R} + \Delta t_{4\times10} + \Delta t_{8\times11} \\ t_{2} = \Delta T_{2\times R} + \Delta t_{2\times10} + \Delta t_{4\times11} + \Delta t_{8\times00} \\ t_{3} = \Delta T_{2\times R} + \Delta t_{2\times10} + \Delta T_{4\times R} + \Delta t_{4\times RF} + \Delta t_{4\times01} + \Delta t_{8\times11} \\ t_{4} = \Delta T_{1\times H} + \Delta t_{2\times11} + \Delta t_{4\times00} + \Delta t_{8\times00} \\ t_{5} = \Delta T_{1\times H} + \Delta t_{2\times11} + \Delta T_{4\times R} + \Delta t_{4\times10} + \Delta t_{8\times11} \\ t_{6} = \Delta T_{1\times H} + \Delta T_{2\times R} + \Delta 2_{4\times RF} + \Delta t_{2\times01} + \Delta t_{4\times11} + \Delta t_{8\times00} \\ t_{7} = \Delta T_{1\times H} + \Delta T_{2\times R} + \Delta t_{2\times RF} + \Delta t_{2\times01} + \Delta T_{4\times R} + \Delta t_{4\times RF} + \Delta t_{4\times01} + \Delta t_{8\times11} \end{cases}$$
(3.1)

To gain some insights, we shift the observation time by t_0 , rewriting the above relation as

$$\begin{aligned} t'_0 &= 0 \\ t'_1 &= (t_1 - t_0) \\ t'_2 &= (t_2 - t_0) \\ t'_3 &= (t_2 - t_0) + (t_1 - t_0) + s_{4\times} \\ t'_4 &= (t_4 - t_0) \\ t'_5 &= (t_4 - t_0) + (t_1 - t_0) \\ t'_6 &= (t_4 - t_0) + (t_2 - t_0) + s_{2\times} \\ t'_7 &= (t_4 - t_0) + (t_2 - t_0) + (t_1 - t_0) + s_{4\times} + s_{2\times} \end{aligned}$$
(3.2)

where

$$s_{2\times} = \Delta t_{2\times 00} + \Delta t_{2\times 01} - \Delta t_{2\times 10} - \Delta t_{2\times 11} + \Delta t_{2\times RF}$$
(3.3)

and

$$s_{4\times} = \Delta t_{4\times00} + \Delta t_{4\times01} - \Delta t_{4\times10} - \Delta t_{4\times11} + \Delta t_{4\times\text{RF}}$$
(3.4)

respectively integrate the overall propagation mismatches in the first and second doubler, i.e., if all propagations are matched, then they will become zero. It is interesting to observe that the last doubler does not impact the whole timing performance despite the existence of delay mismatches; the timing information of the $8 \times \text{REF}$ is completed at the second doubler output, and the last doubler is there just for reversing the given falling edge. Provided that each edge spacing at the output should exactly be $T_{\text{ref}}/8$, it is straightforward to declare the actual timing constraint as

$$\begin{cases} t_4 - t_0 = T_{\text{ref}}/2 \\ t_2 - t_0 = T_{\text{ref}}/4 \\ t_1 - t_0 = T_{\text{ref}}/8 \\ s_{4\times} = 0 \\ s_{2\times} = 0 \end{cases}$$
(3.5)

—suggesting that the 8×REF, at minimum, requires a five-point calibration. This result does make sense in that if the XOR delays are negligible, then $t_1 - t_0$, $t_2 - t_0$, $t_4 - t_0$ become $T_{4\times R}$, $T_{2\times R}$, and $\Delta T_{1\times H}$, respectively, as expected. Likewise, if the reference is to be quadrupled, then one may easily find out that it requires a three-point calibration.

3.2.2 Phase Error Calibration

Defining the five phase error sources that are to be removed as

$$\begin{cases} \varepsilon_0 & \stackrel{\Delta}{=} (2\pi/T_{\rm ref})(t_4 - t_0) \\ \varepsilon_1 & \stackrel{\Delta}{=} (2\pi/T_{\rm ref})(t_2 - t_0) \\ \varepsilon_2 & \stackrel{\Delta}{=} (2\pi/T_{\rm ref})(t_1 - t_0) , \\ \varepsilon_3 & \stackrel{\Delta}{=} (2\pi/T_{\rm ref})s_{4\times} \\ \varepsilon_4 & \stackrel{\Delta}{=} (2\pi/T_{\rm ref})s_{2\times} \end{cases}$$
(3.6)

we may express the periodic phase error sequence of the 8×REF output edges from their ideal positions as a vector on the basis of $\boldsymbol{\varepsilon} = \{\varepsilon_0, ..., \varepsilon_4\}^{\mathrm{T}}$, i.e., from (3.2),

$$\boldsymbol{\phi} = \begin{bmatrix} \phi_{0} \\ \phi_{1} \\ \phi_{2} \\ \phi_{3} \\ \phi_{4} \\ \phi_{5} \\ \phi_{6} \\ \phi_{7} \end{bmatrix} \stackrel{\Delta}{=} \begin{bmatrix} (2\pi/T_{\text{rerf}})t'_{0} \\ (2\pi/T_{\text{rerf}})t'_{2} \\ (2\pi/T_{\text{rerf}})t'_{2} \\ (2\pi/T_{\text{rerf}})t'_{3} \\ (2\pi/T_{\text{rerf}})t'_{4} \\ (2\pi/T_{\text{rerf}})t'_{5} \\ (2\pi/T_{\text{rerf}})t'_{6} \\ (2\pi/T_{\text{rerf}})t'_{6} \\ (2\pi/T_{\text{rerf}})t'_{7} \end{bmatrix} = \boldsymbol{G}_{\boldsymbol{\phi}} \cdot \boldsymbol{\varepsilon} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \cdot \boldsymbol{\varepsilon}.$$
(3.7)

Note here that ϕ_0 is zero for being the reference edge. Due to PVT variations, we cannot assure in the design stage that ϕ of a fabricated chip be **0**. E.g., accounting for only global process corner, the expected deviation without any treatment is about $\pm 20\%$ for a typical CMOS technology. It is important to note that ε_3 and ε_4 should be treated as much as the others for the precise mid-phase generation despite their magnitudes generally being minor, otherwise, they will be the limiting factor of the

performance of the subsequent PLL. Then, how do we let a dedicated system detect ε ? The PD operation of the following PLL provides an answer to this. Suppose that the time constant of the PLL is sufficiently smaller than the correlation time present in the 8×REF output edge, the lowest of which is T_{ref} given ϕ .¹ Then, neglecting all other noise sources, the reference noise transfer function would result in the oscillator output phase (at the PD input in the strict sense), ϕ_{oclk} , locking to the dc level, or the time-average, of the 8xREF output phase, i.e., the mean of ϕ

$$\overline{\phi} \stackrel{\Delta}{=} \sum_{i=0}^{7} \phi_i = \begin{bmatrix} 1/2 & 1/2 & 1/2 & 1/4 & 1/4 \end{bmatrix} \cdot \varepsilon, \tag{3.8}$$

giving the time difference at the PD input for each ϕ_i as

$$\boldsymbol{\delta} \triangleq \begin{bmatrix} \overline{\phi} - \phi_0 \\ \overline{\phi} - \phi_1 \\ \overline{\phi} - \phi_2 \\ \overline{\phi} - \phi_3 \\ \overline{\phi} - \phi_4 \\ \overline{\phi} - \phi_5 \\ \overline{\phi} - \phi_6 \\ \overline{\phi} - \phi_7 \end{bmatrix} = \boldsymbol{G}_{\boldsymbol{\delta}} \cdot \boldsymbol{\varepsilon} = \frac{1}{4} \begin{bmatrix} 2 & 2 & 2 & 1 & 1 \\ 2 & 2 & -2 & 1 & 1 \\ 2 & -2 & 2 & 1 & 1 \\ 2 & -2 & -2 & -3 & 1 \\ -2 & 2 & 2 & 1 & 1 \\ -2 & 2 & -2 & 1 & 1 \\ -2 & -2 & 2 & 1 & -3 \\ -2 & -2 & -2 & -3 & -3 \end{bmatrix} \cdot \boldsymbol{\varepsilon}.$$
(3.9)

Stemming from the above analysis, a low-cost ε calibration method that uses *a priori*, or inductive, statistics of BBPD outputs is proposed. Further, the required settling time of the calibration is minimized by deducing ε *a posteriori* by also exploiting the BBPD statistics so as to meet the time-out constraint of SerDes standards mentioned earlier.

¹Such an assumption is not valid in most cases in that the PLL only provides a finite magnitude of the reference noise suppression. Further elaboration is made in Appendix A.



Figure 3.8: Modulation points constituting the domain of a bijective function for ε .

LMS Adaptation

 $(3.1)\sim(3.6)$ indicate that each element of ε , ε_i , can be calibrated respectively as follows (see Fig. 3.8): [ε_0 by modulating $\Delta T_{1\times H}$], [ε_1 by $\Delta T_{2\times R}$], [ε_2 by $\Delta T_{4\times R}$], [ε_3 by $\Delta t_{4\times 01}$], and [ε_4 by $\Delta t_{2\times 01}$]. Note that ε is a bijective function of the given calibration delays, meaning the modulations do not interfere with each other. Standing as a simple, powerful means of error cancellation, the least-mean-square (LMS) algorithm [58] gives us a good starting point for developing the desired calibration engine. Assuming ε is not time-varying, we may express the instantaneous phase error of the 8×REF output arising from each ε_i , $\phi_{\varepsilon_i}(n)$ where the index n is given from the sampling period of $T_{ref}/8$, as a periodic sequence given by the *i*-th column of G_{ϕ} . Then, as drawn in Fig. 3.9, an LMS system that cancels such a signal can be comprised with a proper input signal $u_{\varepsilon_i}(n)$ and an adaptive filter. The estimation of $\phi_{\varepsilon_i}(n)$, $\hat{\phi}_{\varepsilon_i}(n)$, is computed first, and the residual error resulting from the subtraction



Figure 3.9: LMS system for adaptively cancelling the periodic jitter arising from ε .

adjusts the tap weight for the cancellation. Adopting a single-tap filter, which gives the simplest form of LMS algorithm and justifies the use of scalar representations for the signals, the system can be described by means of equation

$$W_{\varepsilon_i}(n+1) = W_{\varepsilon_i}(n) + \mu_i u_{\varepsilon_i}(n) \cdot (\phi_{\varepsilon_i}(n) - \phi_{\varepsilon_i}(n)), \quad i = 0, 1, \dots, 4 \quad (3.10)$$

where the parameter μ_i is the step size for each update. For the intended operation thereof, $\phi_{\varepsilon_i}(n)$ as well as $\hat{\phi}_{\varepsilon_i}(n)$ should be a linearly scaled version of $u_{\varepsilon_i}(n)$ by the essentials of the traditional LMS algorithm. As an example, suppose that only ε_0 is a non-zero value in ε , and a BBPD-based PLL follows the 8xREF with the assumption of the sufficiently low bandwidth being still given. Then, despite the nonlinear nature of the BBPD, (3.9) indicates that the signum output of the BBPD is actually a linear function of $\hat{\phi}_{\varepsilon_i}(n)$ (minus $\overline{\phi}$). It is thus straightforward to set $u_{\varepsilon_0}(n)$ as a periodic sequence of the 0-th column of G_{δ} . However, this does not lead to a corollary that $u_{\varepsilon_i}(n)$ can be readily given by the *i*-th column of G_{δ} for a general ε ; the column vectors for ε_3 and ε_4 are not orthogonal to the others—meaning that the weight adaptations utilizing the BBPD output interfere with the others, possibly failing the whole calibration.



Figure 3.10: Probability-based LMS system for adaptively cancelling ε with the aid of the BBPD output.

A Priori Probability-based Weight Generation

To achieve a properly functioning low-cost ε calibration, we first model the 8×REF under calibration implemented to a BBPD-based ADPLL as illustrated in Fig. 3.10, with the index k given from the sampling period of T_{ref} . The calibration engine generates a time-varying estimate of ε , and then the resulting difference is passed through G_{δ} . The actual input of the BBPD, $\delta'(k)$, equals $\delta(k)$ minus the *stationary* noise inherent in the ADPLL, $\varphi(k)$. d(k), the 8-bit signum output (+1/-1) of the BBPD, suggests the use of the sign-sign variant of LMS algorithm [59], which updates its weights based on the *signed* value of the instantaneous gradient estimation. However, this still requires a proper reference signal for each ε_i , which is complex to obtain as described. We therefore propose using the estimated *a priori* probability that the gradient is positive or negative² given d(k), which is written as

$$X_i(k)|_{\boldsymbol{d}(k)} = \Pr(\varepsilon_i < 0|_{\boldsymbol{d}(k)}) - \Pr(\varepsilon_i > 0|_{\boldsymbol{d}(k)}).$$
(3.11)

²Equivalently, it represents the probability that ε_i is positive or negative.

Then, using Bayes' theorem, it is calculated by

$$X_{i}(k)|_{\boldsymbol{d}(k)} = \frac{\Pr(\boldsymbol{d} = \boldsymbol{d}(k)|_{\varepsilon_{i} < 0}) \cdot \Pr(\varepsilon_{i} < 0)}{\Pr(\boldsymbol{d} = \boldsymbol{d}(k))} - \frac{\Pr(\boldsymbol{d} = \boldsymbol{d}(k)|_{\varepsilon_{i} > 0}) \cdot \Pr(\varepsilon_{i} > 0)}{\Pr(\boldsymbol{d} = \boldsymbol{d}(k))}.$$
(3.12)

In order to obtain each probability, we require the probability density functions (PDFs) of the given prior random variables. $\varphi(k)$ may sufficiently be modeled as a Gaussian random vector such that each element follows $\mathcal{N} \sim (0, \sigma_{\text{iitter}})$, i.i.d. Then, how do we decide the PDF of ε_i ? Given no information with respect to the calibration circuitry, it is quite not clear to characterize the randomness of ε_i before the calibration is settled since it could be any value within the range given by possible PVT variations. Such a random variable is so-called an *uninformative prior* in Bayesian statistics, having been intriguing mathematicians into some sophisticated discussions on how to set its PDF.³ Among many approaches, we use the bounded (symmetric) uniform distribution, which have been suggested as a simple, proper choice by Carlin and Louis [60]; $\varepsilon_i \sim \mathcal{U}(-A_i, A_i)$. Here, we set $A_i = A$ for all *i* to simplify the calculation at some loss of generality.⁴ Fig. 3.11 plots the calculation results of (3.12) given a set of representative d for each ϵ_i . With $d = \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 \end{bmatrix}^T$, if $A \gg \sigma_{\text{iitter}}^5$, i.e., if the deterministic jitter is dominant, then $X_0 \rightarrow 1$ as one may intuitively expect. However, an interesting point to make here is that X_1 and X_4 also converge to some non-negligible values. Such somewhat odd results can be also found under the rest of the given *d*.

In ideal, when the calibration is well performed, each ε_i would settle to zero. However, as will be elaborated later, the designed $8 \times \text{REF}$ calibrates ε in a digital manner, implying that each W_{ε_i} dithers around its nominal point, resulting in each ε_i

³Elaboration on this is out of the scope of this dissertation.

⁴This does not sacrifice the given symmetry and the principle of maximum entropy.

⁵This assumption sufficiently holds when designing a high-performance ADPLL that gives σ_{jitter} of few hundred femtoseconds or less.



Figure 3.11: Calculation results of (3.12) before calibration settlement. (\blacklozenge : X_0 , \blacktriangledown : X_1 , \blacktriangle : X_2 , \blacksquare : X_3 , \blacklozenge : X_4)



Figure 3.12: PDF of ε_i after calibration settlement.

dithering around zero with a quantization step denoted by Δt_W , which is assumed to be equal for all ε_i . Thus, under settlement, ε_i is no more of uninformative prior, and its PDF can be readily modeled as a two-point discrete uniform distribution, as shown in Fig. 3.12. Accounting for this, Fig. 3.13 recalculates the probabilities, giving different results from Fig. 3.11. As will be shown in the next subsection, the circuitry for fine calibration adopts a digitally controlled delay line, the resolution of which could be as small as about one-hundred femtoseconds with the modern CMOS technology [61]. Therefore, the assumption $\Delta t_W \gg \sigma_{\text{jitter}}$ is not valid here; Δt_W is expected to be comparable or less than σ_{jitter} . Thus, it is more reasonable that X_i herein be obtained by a simple fractional value at some point where $0.5 < \Delta t_W / \sigma_{\text{jitter}} < 1$, e.g., $\mathbf{X}(k) = [0.8 \ 0 \ 0 \ 0.5]^{\text{T}}$ given $\mathbf{d}(k) = [-1 \ -1 \ -1 \ 1 \ 1 \ 1 \ 1]^{\text{T}}$.

Using the above calculations, we then prepare two different sets of look-up-tables (LUTs) that output X(k) according to d(k), one for before calibration (LUT-B) and the other for after calibration (LUT-A). Therefore, the calibration engine may update the weight vector as

$$\boldsymbol{W}_{\boldsymbol{\varepsilon}}(k+1) = \boldsymbol{W}_{\boldsymbol{\varepsilon}}(k) + \boldsymbol{\mu} \circ \boldsymbol{X}(k)|_{\boldsymbol{d}(k)}$$
(3.13)

where μ is the step-size vector, and \circ notates the element-wise multiplication (hadamard product).



Figure 3.13: Calculation results of (3.12) after calibration settlement. (\blacklozenge : X_0 , \blacktriangledown : X_1 , \blacktriangle : X_2 , \blacksquare : X_3 , \blacklozenge : X_4)



Figure 3.14: PDF of δ'_i .

A Posteriori Calibration Acceleration

There exists an optimum μ , μ_{\min} , that gives the minimum dithering bits of the settled W_{ε} (one-bit dithering as assumed for calculating the LUT-A). However, with this, the calibration might suffer from long settling time as mentioned earlier. To address this, we accelerate the calibration by employing a probability-based step size adaptation. In order to achieve this, the *magnitude* of each epsilon should be identified by some means of circuit realization. Since δ holds information on ε as explained, we may deduce it from δ a *posteriori*. Here, since the characteristics of only five variables are to be sought, we only need to observe five elements of δ , e.g., $\delta_0, \ldots, \delta_4$, which can be represented by

$$\tilde{\boldsymbol{\delta}} \stackrel{\Delta}{=} \begin{bmatrix} \delta_0 & \delta_1 & \delta_2 & \delta_3 & \delta_4 \end{bmatrix}^{\mathrm{T}} = \tilde{\boldsymbol{G}}_{\boldsymbol{\delta}} \cdot \boldsymbol{\varepsilon}$$
(3.14)

where the generation matrix \tilde{G}_{δ} is easily obtained from (3.9). By accumulating each element of $\tilde{\delta}$ for a certain period, we obtain its *digital* representation, $\tilde{D} = [D_0 D_1 D_2 D_3 D_4]^{\text{T}}$. Since each δ'_i follows $\mathcal{N}(\delta_i, \sigma_{\text{jitter}})$, its cumulative distribution function (CDF) is estimated by D_i , i.e.,

CHAPTER 3. FILTERING RO NOISE

$$D_i = D_{\max} \cdot \operatorname{erf}(\delta_i / \sigma_{\text{jitter}}) \tag{3.15}$$

where D_{max} is the maximum counting number, e.g., 255 with 8-bit counts, as illustrated in Fig. 3.14. Although it seems straightforward to derive δ_i by multiplying the inverse of erf(), which is of a nonlinear function, to D_i , this method is not feasible in a low-cost digital hardware due to its high computation complexity. However, since erf() is a monotone-increasing function that is symmetric about the origin, D_i itself may be used to identify the relative magnitude of each ϵ_i ; if \tilde{D} is multiplied by the inverse of \tilde{G}_{δ} , then, E—the digital representation of ε with some distortion in magnitudes induced by the accumulation nonliearity—is computed, i.e.,

$$\boldsymbol{E} = \tilde{\boldsymbol{G}}_{\boldsymbol{\delta}}^{-1} \cdot \tilde{\boldsymbol{D}} = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \\ 1 & 0 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 1 & -1 & 0 \\ -1 & 1 & 1 & 1 & 2 \end{bmatrix} \cdot \tilde{\boldsymbol{D}}.$$
 (3.16)

To mention again, E indicates which ε_i dominates the others, rather than estimating the magnitudes. Then, with the aid of the variable-step-size (VSS) algorithm [62], μ is adaptively varied such that

$$\boldsymbol{\mu}(m+1) = \alpha \boldsymbol{\mu}(m) + \gamma \boldsymbol{E}(m)^{\circ 2} \tag{3.17}$$

where $0 < \alpha < 1$ (for stability), $0 < \gamma$, and the index *m* is given from the update period determined by the counter bit number.



Figure 3.15: (a) Overall architecture and (b) operation flow chart of the proposed $8 \times \text{REF}$ calibration.

Overall Probability-based VSS-LMS Calibration

Fig. 3.15 illustrates the overall architecture and operation flow chart of the proposed probability-based VSS-LMS algorithm. Upon power-up, where it is very probable that the magnitude of ε_i is large, the accurate computation of **E** is not required, and thus, the accumulations first operate up to 6-bit counts. If any of $|D_i|$ exceeds a certain threshold, $|D_{\text{th,6bit}}|$, then μ is updated promptly, thereby further accelerating the calibration. If not, the counters continue the accumulations up to 8-bit, giving higher accuracy of the E deduction. Once all $|D_i|$ after 8-bit counts are within a certain threshold, $|D_{\text{th,8bit}}|$, then μ is set to μ_{\min} , with the VSS algorithm temporarily deactivated, and W_{ε} updated by means of the LUT-A. Thereafter, as each D_i holds information on all ε_i , only one of D_i is collected in background, monitoring any sudden disturbance. Here, D_0 is the proper choice for some reasons. First, since PVT variations tend to modulate each delay with the same polarity, it is advantageous to observe one that sums the delay variations with the same polarity. Further, the magnitude of delay variation is generally proportional to the absolute delay value, i.e., ε_0 to ε_2 are especially prone to PVT variations. Overall, disabling all other digital operations alleviates power overhead of the calibration under settled $W_{arepsilon}$ or slowmoving supply/temperature environments.

For better comprehension, Fig. 3.16 visualizes a simple example of the calibration behavior. Assume that only two phase error sources, ε_1 and ε_2 , exist in the 8×REF. Then, in line with the previous explanation, only two δ_i , e.g., δ_0 and δ_1 , are sufficient to deducing E_1 and E_2 . Specifically,

$$\begin{bmatrix} \delta_0 \\ \delta_1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \end{bmatrix}$$
(3.18)



Figure 3.16: Simple example of a two-point calibration with the proposed VSS-LMS algorithm.

along with the stationary noise components are sensed by the BBPD, followed by accumulations, giving D_0 and D_1 . Thus, the resulting

$$\begin{bmatrix} E_1 \\ E_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} D_0 \\ D_1 \end{bmatrix}$$
(3.19)

adaptively varies μ_1 and μ_2 such that, initially given small ε_1 and large ε_2 , the deduction sufficiently identifies that ε_2 is much larger than ε_1 , first calibrating ε_2 with a very fast speed while correcting ε_1 slowly. Then, as the calibration goes on, the overall adaptation speed slows down until both of them are well settled to dither around the origin.

3.2.3 Circuit Implementation

This subsection describes the circuit-level implementation of the 8×REF. As depicted in Fig. 3.17, each ε_i can be calibrated as follows: ε_0 by a duty-cycle corrector (DCC) at the input reference, ε_1 by the delay line 1 (DL1) at the first delay element, ε_2 by the DL2 at the second delay element, ε_3 by the DL3 that modulates the second



Figure 3.17: Calibration point to which each W_{ε_i} should be applied.



Figure 3.18: Modified architecture of the 8×REF with the F-DL and G_{ϕ} unit.

falling edge of the second XOR, and ε_4 by the DL4 that modulates the second falling edge of the first XOR. Two major concerns arise when realizing the required hardware elements. First, the DCC should facilitate large tuning range to cover large duty-cycle error of the XO output, and so do the DL1 and DL2 to cover large delay errors induced by PVT variations. Secondly, it is not straightforward to properly realize the DL3 and DL4 without significantly increasing the hardware complexity. To address these, each of W_{ε_0} to W_{ε_2} is first separated into a coarse word and a fine word. Then, each coarse word is fed to the coarse DCC (C-DCC), coarse DL1 (C-DL1), and coarse DL2 (C-DL2), respectively, and the fine words, along with W_{ε_3} and W_{ε_4} , are converted into



Figure 3.19: CML stage for the C-DL.



Figure 3.20: Circuit implementation of the C-DCC.

the fine-weight vector for ϕ

$$\boldsymbol{W}_{\boldsymbol{\phi}}^{\mathrm{f}} \stackrel{\Delta}{=} \left[W_{\phi_0}^{\mathrm{f}} \dots W_{\phi_7}^{\mathrm{f}} \right]^{\mathrm{T}} = \boldsymbol{G}_{\boldsymbol{\phi}} \cdot \boldsymbol{W}_{\boldsymbol{\varepsilon}}.$$
(3.20)

Next, the DL3 and DL4 are eliminated, and instead, $W_{\phi_0}^{f} \dots W_{\phi_7}^{f}$ are periodically applied to the fine DL (F-DL) that follows the original 8×REF via a digital multiplexer with a 3-bit counter, as illustrated in Fig. 3.18.



Figure 3.21: Circuit implementation of the F-DL.



Figure 3.22: Final analog path of the implemented 8×REF.

To alleviate the supply sensitivity (both static and dynamic) of the large delays of the C-DL1 and C-DL2, the differential CML stage, as in Fig. 3.19, is chosen over a single-ended CMOS stage. Here, to realize the large delays without loss of signal swings, two stages are cascaded. To accommodate the differential topology, the C-DCC is also implemented in a CML manner as in Fig. 3.20, where the difference in the output capacitance seen at the rising and falling edges is varied by digitally controlling C_{D+} . Here, the resistive CMOS switches that are controlled by the input signal let the output node experience the intended capacitance difference. In comparison with the C-DLs, the F-DL does not require large absolute delay but should be as linear as possible to accommodate the linear transformation of W_{ε} to W_{ϕ} described by (3.20). The CMOS-based F-DL, as in Fig. 3.21, acts as an adequate solution for achieving this; By adopting the word segmentation, the F-DL achieves high resolution, wide tuning range, and high linearity. The final analog path of the 8×REF implemented in the prototype is presented in Fig. 3.22. The reference signal from the external XO is first single-to-differential converted prior the C-DCC. The XORs are also implemented in a CML manner, overall completing the differential topology for mitigating supply sensitivity. The last XOR is followed by a CML-to-CMOS converter, whose output is then fed to the F-DL.

3.3 IL-ADPLL Implementation

This section will explain the design procedure of the IL-ADPLL cascaded to the $8 \times \text{REF}$. Since the injection is concurrent with the $8 \times \text{REF}$ calibration, it is instructive to examine how it interacts with *d* in the time domain. As an example, assume that only



(a) d without injection.



(c) d with pulse-gated injection in presence of path mismatch.

Figure 3.23: Timing diagrams depicting the effect of injection to the $8 \times \text{REF}$ calibration in presence of ε_2 .



Figure 3.24: Overall architecture of the proposed IL-ADPLL.

 ε_2 is of a positive value, and the remaining ε_i is zero. Then, without injection, d would alternate between 1 and -1 periodically as described earlier, and thus W_{ε_2} would settle so that ε_i is eliminated, as depicted in Fig. 3.23(a). Next, if the 8×REF is injected to the RO with a certain positive delay mismatch from the lock point given by the PLL, then the transient behavior of ϕ_{oclk} would be altered as in Fig. 3.23(b), resulting in significant periodic jitter. However, we may figure out that d is given as the same as before regardless of the injection strength. To eliminate the injection delay mismatch, we adopt the pulse-gating method proposed by [55]. This once again alters the behavior of ϕ_{oclk} , and this time, it also may change the periodic profile of d, resulting in the alternation and possibly deterioration of the calibration performance. Further, it should be noted that gating the injection pulse every N_{PG} cycles induces an additional spur at the frequency of $8f_{ref}/N_{PG}$. To address this, the 4-bit sequence of the BBPD output after each gating is used to generate a randomized gating period, $N'_{\rm PG} \sim \mathcal{U}(4, 19)$, giving the average of 11.5 and thereby maintaining the intended calibration function not hampered by the pulse gating. Then, we now arrive at the overall architecture of the IL-ADPLL, as illustrated in Fig. 3.24. Here, note that the analog path of the 8×REF is drawn in its single-ended representation. Along with the fast $8 \times \text{REF}$ calibration, a dead-zone frequency detector (DZ-FD) is used for fast lock of the IL-ADPLL. The pulser comprises a respective delay element and an XOR gate, and the pulse gate by the NAND logic driven by N'_{PG} . Lastly, a two-stage pseudo-differential CMOS RO [63] is used with the injection realized by the pulsed signal shorting the differential oscillation nodes [64].

3.4 Measurement Results

The prototype chip is implemented in a 40-nm CMOS technology, occupying a total active area of 0.065 mm² as shown in Fig. 3.25. The measured phase noise of the calibrated 8×REF output is plotted in Fig. 3.26. Shown in Fig. 3.27 is the phase noise plot at 8-GHz output with the doubled, quadrupled, and the octupled reference, each calibrated by the presented adaptation engine. By the reference multiplications, the noise suppression bandwidth for the RO is widened, respectively achieving integrated rms jitter (10 KHz ~ 90 MHz) of 418 fs, 267 fs, and 177 fs with 100-MHz reference. With the calibration enabled, we observed that the spur at the reference frequency is -68 dBc and the spur at the octupled reference frequency is -61.2 dBc, as shown in Fig. 3.28(a). When W_{ε} is fixed after settlement, i.e., when ε is of finite values without dithering, it is observed that the reference spur and its harmonics become significant,



Figure 3.25: Chip photomicrograph.


Figure 3.26: Measured phase noise plot at the $8 \times \text{REF}$ output.

as shown in 3.28(b). Taking advantage of the ease of realizing multi-rate output of an RO, the DCO may also output a 16-GHz clock. With the doubled, quadrupled, and the octupled reference, it achieves integrated rms jitter of 534 fs, 324 fs, and 223 fs, respectively, as shown in Fig. 3.29. Its output spectrum indicates that the spur at the reference frequency is -61.4 dBc and the spur at the octupled reference frequency is -55.8 dBc, as presented in Fig. 3.30(a). The reference spur and its harmonics also grow larger when the calibration is turned off, as shown in Fig. 3.30(b).

Plotted in Fig. 3.31(a) is the transient behavior of W_{ε} at a startup in presence of large duty-cycle error of the XO. Without the presented VSS algorithm and with μ_{min} , the calibration settles after about 960 us after initiation. On the other hand, with the VSS enabled, the calibration only requires about 70 us to settle, corresponding to more than 13-fold reduction. If the static supply voltage of the 8×REF is suddenly increased by 10%, then without the VSS, the calibration requires about 205 us to settle



Figure 3.27: Measured phase noise plot at 8-GHz output.

again. Meanwhile, with the VSS enabled, it only requires about 10us, as shown in Fig. 3.31(b). In addition, Fig. 3.31(c) indicates that if the static supply voltages of both the $8 \times \text{REF}$ and the DCO are increased by 10%, then without the VSS, the re-locking of the PLL and the calibration together require 215 us for settlement, while, with the VSS enabled, 25 us is just sufficient.

The total power consumption of the presented IL-ADPLL is 12.1 mW and 17 mW at 8-GHz and 16-GHz operations, respectively, with the overall digital hardware (after settlement) consuming 2.2 mW, and the analog path of the $8 \times \text{REF}$ consuming 2.3 mW. Table 3.1 summarizes the performance of this work and compares it with prior RO-based frequency synthesizers with high *N*. The presented IL-ADPLL outputs 8/16-GHz clock with the multiplication factor of 80/160, achieving the jitter-power FoM (FoM₁) of -244.2/-240.7 dB. Comparing with prior works, this work achieves the state-of-the-art the jitter-power-*N* FoM (FoM₂) at both 8-GHz and 16-GHz outputs.



(a) With the $8 \times \text{REF}$ calibration after settlement.



(b) Without the $8 \times \text{REF}$ calibration after settlement.

Figure 3.28: Measured spectra at 8-GHz output.



Figure 3.29: Measured phase noise plot at 16-GHz output.

3.5 Summary

In strive for further extending the PLL bandwidth and thereby sufficiently filtering the high flicker noise of a high-frequency RO, we designed the $8 \times \text{REF}$ that avoids jitter accumulation and then cascaded it to an ADPLL. Timing analysis revealed that the $8 \times \text{REF}$ requires a five-point calibration to correct the phase error present therein. Stemming from the LMS adaptation, which is a powerful steepest gradient search algorithm, a probability-based calibration engine is developed. Given the theory that there exists no set of basis signal for the five phase error sources that are orthogonal to each other and that the statistics of the BBPD output repeat every reference cycle, the weight for each calibration point is updated by pre-defined LUTs that observe the incoming 8-bit BBPD output sequence. To account for the known information on the behavior of the phase errors under calibration after settlement, two sets of LUTs are



(a) With the $8 \times \text{REF}$ calibration after settlement.



(b) Without the $8 \times \text{REF}$ calibration after settlement.

Figure 3.30: Measured spectra at 16-GHz output.



(c) After +10% variations in $V_{\text{DD,DCO}}$ and $V_{\text{DD,8}\times\text{REF}}$.

Figure 3.31: Transient behaviors of W_{ε} .

implemented. To accelerate the calibration under large PVT variations, we adopted the VSS algorithm into the calibration engine, letting the gain of each weight generation continuously varied throughout the adaptation. The analog path of the 8×REF is designed to be insensitive to supply variation, and the difficulty on realizing the circuits for calibrating phase errors that are induced by propagation mismatches is solved by implementing a CMOS-based DL with a fine resolution. Measurement results of a prototype chip demonstrated the effectiveness of the 8×REF on the flicker noise suppression and that of the calibration acceleration algorithm. Overall, it achieved state-of-the-art rms jitter of 177/223 fs at 8/16-GHz output, consuming 12.1/17-mW power.

		r			·				·	·	· · · ·	·		
This Work	40	ILCM	8×	16 [13.9 to 18.3]	160	223	[10k to 90M]	-61.4	-55.8	17	0.065	-240.7	-262.7	Probability-based VSS-LMS
				8 [6.4 to 9.9]	80	177	[10k to 90M]	-68	-61.2	12.1		-244.2	-263.2	
VLSI'2017 [23]	65	DPLL	None	7.68	64	373	[1k to 100M]	-65	N/A	6.48	0.075	-240.5	-258.5	ı
ISSCC'2016 [19]	65	ILCM	8×	5.184 [3 to 5.2]	96	354	[10k to 100M]	-62	-44	8.2	0.15	-240	-260	Trim+LMS
ISSCC'2014 [20]	65	ILCM	4×	4.752 [2.6 to 5.2]	88	421	[10k to 100M]	-53	-55	6.5	0.16	-239.4	-258.8	LMS
ISSCC'2014 [17]	65	Analog PLL	None	3.008	64	357	[1k to 80M]	-71	N/A	4.6	0.047	-242.3	-260.4	1
JSSC'2011 [18]	45	Analog PLL	None	2.4 [8 to 3]	106	970	[1k to 200M]	-65	N/A	4	0.015	-234.1	-254.5	1
	Technology (nm)	Architecture	Reference Multiplication	f ^{out} [Tuning Range (GHz)]	Multiplication Factor, N	Int. rms Jitter, σ_{jitter} (fs)	[Int. Range (Hz)]	Ref. Spur (dBc)	Multiplied Ref. Spur (dBc)	Total Power Consumption (mW)	Area (mm ²)	FoM_1^* (dB)	FoM_2^{**} (dB)	Ref. Multiplication Calibration Method

y Synthesizers.	
requency	
RO-Based F	
[igh-N])
Prior H	
n with	
ompariso	
1:0	
e 3.	
Tabl	

 $*FoM_{\rm I} = 10\log\left\{\left(\frac{\sigma_{\rm inter}}{1{\rm s}}\right)^2 \left(\frac{P_{\rm ower}}{1{\rm nW}}\right)\right\}. \\ **FoM_2 = FoM_{\rm I} + 10\log(1/N).$



Figure 3.32: FoM comparisons with prior state-of-the-art works.

Chapter 4

RO Supply Noise Compensation

4.1 Introduction

Another major challenge in the design of an RO-based frequency synthesizer is to mitigate the degradation of jitter performance by supply noise. One common solution to mitigating the supply sensitivity of an RO-based DCO is to employ a lowdropout (LDO) regulator [65]-[67], as depicted in Fig. 4.1. An LDO gives a high suppression gain if the bandwidth of its amplifier is sufficiently large [65] but requires additional voltage headroom over a DCO, significantly lowering the maximum tunable



Figure 4.1: Typical LDO integrated to mitigate the supply sensitivity of an RO-based DCO.

 f_{out} . More importantly, for a low-jitter generation, the control voltage of the LDO, V_{ctrl} , should give very low noise, increasing the overall hardware complexity.

Depicted in Fig. 4.2 are prior arts with RO supply noise compensation. In [68], an additional current source is implemented in the DCO to cancel the supply-induced variation of its frequency-controlling current. However, being configured as an open loop, its path gain suffers from process, voltage, and temperature (PVT) variations significantly. [69] proposes a background calibration scheme that continuously adjusts its path gain by leveraging the digital nature of the ADPLL: A periodic digital signal is injected into the DCO, and the integral word of the DLF therefrom is observed, giving information on the path gain to the calibration circuit. However, to minimize the noise contribution of the test signal, its frequency should be set far below the PLL bandwidth, requiring an excessive settling time for the calibration. Another digital calibration scheme proposed in [70] also suffers from a long calibration time which overwhelms the lock time of a typical PLL since its finite-state machine (FSM) operates with a frequency of several hundred hertz to acquire the digital profile of supply noise. Moreover, in presence of a single-tone supply noise whose frequency is above half the reference frequency of the ADPLL, its information cannot be successfully transferred into the DLF, nullifying the calibration. Instead of using an additional active device, [71] searches for the optimum f_{out} that gives the minimum sensitivity



(a) [68]



Figure 4.2: Depiction of prior arts with RO supply noise compensation.

to the supply. However, since this f_{out} should coincide with the specified f_{out} , an additional foreground calibration that compensates for PVT variations is required. In addition, the series resistors of several hundred ohms in the delay cells preclude high-frequency oscillation of the RO. Note that, in [69]-[71], the additional hardware for the calibration, which is in the form of a resistor or a current source, is placed over the DCO just as an LDO does.

In this chapter, an RO-based ADPLL with a PVT-robust analog circuit that compensates for supply-induced noise in the DCO without degrading its voltage headroom [72] is proposed. Furthermore, noise contribution of the additional hardware for the technique is minimized so as not to degrade the phase noise of the output clock.



Figure 4.3: Conventional RO-based ADPLL with a DCR.

The prototype ADPLL is fabricated in a 40-nm CMOS technology, achieving a jitterpower figure-of-merit (FoM) of -241 dB without supply noise and a jitter reduction of -23.8 dB in presence of a 20-mV_{rms} white noise on the supply.

The rest of this chapter is organized as follows. Section 2 overviews the operation principle of the proposed technique, followed thorough analysis in frequency domain and circuit optimization considering trade-offs between device noise and performance. Section 4 presents experimental results of the prototype chip, and finally, Section 5 concludes this chapter by summarizing the key contributions of the presented work.

4.2 Proposed Analog Closed Loop for Supply Noise Compensation

Fig. 4.3 depicts a conventional RO-based ADPLL whose f_{out} is modulated through a digitally controlled resistor (DCR) [73] by the frequency-tuning word, D_{FTW} . The internal supply voltage of the RO is given by

$$V_{\rm DD,I} = \frac{R_{\rm RO}}{R_{\rm DCR} + R_{\rm RO}} \cdot V_{\rm DD,I,Th}$$
(4.1)



Figure 4.4: Thévenin equivalent circuit of the DCR with respect to the RO under a onebit transition in D_{FTW} . α is determined by D_{FTW} at which the transition takes place.

where R_{DCR} and R_{RO} denote the resistance of the DCR and the RO, respectively, and $V_{\text{DD,I,Th}}$ the Thévenin voltage of the DCR seen from the RO, which is equal to V_{DD} at a fixed D_{FTW} . Then, a negative feedback system that suppresses supply-induced noise at $V_{\text{DD,I}}$ can be realized by providing a proper compensation component to $V_{\text{DD,I,Th}}$. However, with this configuration, the feedback system must overcome a crucial flaw. Since the DCR consists of switched PMOSs, the transient result of a transition in D_{FTW} can be approximated as a current step, I_{FTW} , applied to an internal voltage in the DCR, V_{FTW} , as shown in Fig. 4.4. This current source in parallel with the DCR is Thévenin equivalent to a voltage source in series with the DCR. Therefore, without a remedy, the feedback system would equally reject the response of the DCR modulation, neutralizing the ADPLL function. The proposed analog closed loop for supply noise compensation (ACSC) avoids this problem by utilizing a replica-based circuit implementation.

4.2.1 Circuit Implementation

The block diagram of the ADPLL with the ACSC is shown in Fig. 4.5. While the DCR is fed by the integral word from the DLF, D_{FTW}^{I} , precise phase alignment of the output clock is achieved through the proportional DCR (P-DCR) by the TDC output,



Figure 4.5: Overall block diagram of the ADPLL with the proposed ACSC.



Figure 4.6: Circuit implementation of the passive filters.

 $D_{\text{FTW}}^{\text{p}}$. In the ACSC, $V_{\text{DD,I}}$ passes through a low-pass filter (LPF1), and then it is fed to the negative input of an error amplifier (EA), $V_{\text{err-}}$. Meanwhile, the replica internal supply voltage, $V_{\text{R-DD,I}}$, passes through a high-pass filter (HPF) in which the DC and the low-frequency component of the output are replaced by those of $V_{\text{DD,I}}$ through another low-pass filter (LPF2), and then it is fed to the positive input of the EA, $V_{\text{err+}}$. The EA is followed by a V-to-I converter (VIC), and then the resulting current flows through the DCR, forming a negative feedback system that takes V_{DD} as its input and $V_{\text{DD,I,Th}}$ as its output. With the passive filters and the EA, a replica DCR (R-DCR) and a replica VIC (R-VIC) constitute a replica loop path that takes $V_{\text{R-DD,I}}$ as its output. Note that, unlike the DCR, the R-DCR is fed by the PVT-tracking word of $D_{\text{FTW}}^{\text{I}}$, $D_{\text{FTW}}^{\text{row}}$. Then, under certain constraints, which will be discussed later, $V_{\text{R-DD,I}}$ replicates the supply-induced $V_{\text{DD,I,Th}}$ but leaves out the transient effect of the DCR modulation when the ADPLL is in lock. Consequently, the supply-induced noise at both $V_{\text{DD,I}}$ and $V_{\text{R-DD,I}}$ are suppressed by the ACSC while the ADPLL operates transparently. The circuit level diagrams of the passive filters are shown in Fig. 4.6. To enhance the linearity of the EA over a wide dynamic range of supply noise, $V_{DD,I}$ is first down converted by a source follower (SF). Following the SF are two RC filters in parallel, one acting as the LPF1, and the other as the HPF and the LPF2 at the same time.

4.2.2 Frequency-Domain Analysis

If the output impedance of the SF is sufficiently smaller than R_L , then the LPF1 and the HPF are approximated as first-order filters with the bandwidth of $w_{p,L} = 1/(R_LC_L)$ and $w_{p,H} = 1/(R_HC_H)$, respectively, and their transfer functions represented as $H_{LPF1}(s)$ and $H_{HPF}(s)$, respectively. Likewise, the LPF2 is equivalent to a first-order filter with the bandwidth of $w_{p,H}$ and its transfer function represented as $H_{LPF2}(s)$. Therefore, in small-signal framework, the control voltage of the VICs is expressed as

$$v_{\text{ctrl}}(s) = A_{\text{EA}}(s) \cdot (v_{\text{err+}}(s) - v_{\text{err-}}(s))$$
$$= A_{\text{EA}}(s) \cdot \{H_{\text{HPF}}(s) \cdot v_{\text{R-DD,I}}(s)$$
$$+ (H_{\text{LPF2}}(s) - H_{\text{LPF1}}(s)) \cdot v_{\text{DD,I,Th}}(s)\}$$
(4.2)

where $A_{\text{EA}}(s)$ represents the gain of the EA. Then, denoting the gain of the VIC and the R-VIC as G_m and $G_{m,R}$, respectively, the transfer functions from v_{DD} to $v_{\text{DD,I,Th}}$, H(s), and from v_{DD} to $v_{\text{R-DD,I}}$, $H_{\text{R}}(s)$, are obtain by (B.6) and (B.7) (see Appendix B), respectively. It is noted here that the resistance of the P-DCR, $R_{\text{P-DCR}}$, should also be taken into account. However, since the P-DCR corrects only the minor phase error of the output clock, $R_{\text{P-DCR}}$ is sufficiently larger than R_{DCR} , thereby being ignored throughout the derivations. When the ADPLL is in lock, $D_{\text{FTW}}^{\text{P}}$ as well as the frequency-locking word of $D_{\text{FTW}}^{\text{I}}$, $D_{\text{FTW}}^{\text{col}}$, dithers around its nominal point, and its transient effect at $v_{\text{DD,I,Th}}$ can be also portrayed using Fig. 4.4 with $R_{\text{P-DCR}}$ and a respective α . On the contrary, when in lock, the resistance of the R-DCR, $R_{\text{R-DCR}}$,



Figure 4.7: Signal flow diagrams depicting (a) H(s) and (b) $H_{\rm R}(s)$ each given (4.3).

is set by a fixed $D_{\text{FTW}}^{\text{row}}$, being impervious to i_{FTW} . Nevertheless, since i_{FTW} , being translated into $v_{\text{DD,I,Th}}$, controls the EA through the two LPFs, $v_{\text{R-DD,I}}$ is still affected by i_{FTW} , with its transfer function, $Z_{\text{R}}(s)$, derived by (B.12). This, in turn, disturbs i_{FTW} from properly modulating $v_{\text{DD,I,Th}}$, and its transfer function, Z(s), is given by (B.13). Therefore, to prevent this, $v_{\text{DD,I,Th}}$ should be canceled out at the EA input, i.e., from (4.2),

$$w_{\rm p,L} = w_{\rm p,H}.\tag{4.3}$$

Then, regardless of other parameters, i_{FTW} is no more transmitted into $v_{\text{R-DD,I}}$, i.e., $Z_{\text{R}}(s)$ is now zero. In result, Z(s) returns to the one obtained without the ACSC, which is equal to αR_{DCR} . Now that supply noise is sensed only through $v_{\text{R-DD,I}}$ given this constraint, with $k_{\text{s,DCR}}$ and $k_{\text{s,}G_m}$ representing $R_{\text{R-DCR}}/R_{\text{DCR}}$ and $G_m/G_{m,\text{R}}$, respectively, we have

$$H(s)|_{(4.3)} = 1 - \frac{O(s)}{1 + \frac{k_{s,\text{DCR}}}{k_{s,G_m}} \cdot O(s)}$$
(4.4)

$$H_{\mathbf{R}}(s)|_{(4.3)} = \frac{1}{1 + \frac{k_{s,\text{DCR}}}{k_{s,G_m}} \cdot O(s)}$$
(4.5)

where

$$O(s) = R_{\text{DCR}}G_m \cdot A_{\text{EA}}(s) \cdot H_{\text{HPF}}(s).$$
(4.6)

Whereas (4.5) is clearly of a closed-loop system, interpreting (4.4) is non-trivial as illustrated in Fig. 4.7.

If the voltage across the DCR and the R-DCR by the compensation current of the VIC and the R-VIC, respectively, are given as the same, i.e., if

$$k_{\mathrm{s,DCR}} = k_{\mathrm{s},G_m},\tag{4.7}$$

then (4.4) and (4.5) converge to the same closed-loop function

$$\tilde{H}(s) \stackrel{\Delta}{=} \frac{1}{1+O(s)}.\tag{4.8}$$

It should be noted that, in order for H(s) and $H_{R}(s)$ to just replicate each other, (4.3) needs not be satisfied.

O(s) is in the form of a band-pass amplifier, and $\tilde{H}(s)$, therefore, of a band-reject filter. Fig. 4.8 exemplifies and compares the frequency responses of O(s) given with a single-pole EA and a two-pole EA, the latter, which corresponds to a cascode (or a cascaded) EA, offering higher DC gain, $A_{0,\text{EA}}$. Here, the lower band frequency is denoted as w_{LB} , and the upper band frequency as w_{UB} . Although a two-pole EA gives lower w_{LB} , w_{UB} is hardly increased owing to the extra pole. In addition, with a single-pole EA, a phase margin over 90° can always be achieved regardless of the loop parameters. Furthermore, as will be elaborated later, there exists a trade-off between $A_{0,\text{EA}}$ and the total noise contribution of the ACSC to the output clock. Therefore, a single-pole EA is chosen, and the resulting mid-band rejection gain, H_{mid} , is given by

$$\tilde{H}_{\text{mid}} \stackrel{\Delta}{=} \tilde{H}\left(s = iw_{\text{GM}}\right) = \frac{1}{1 + R_{\text{DCR}}G_m A_{0,\text{EA}}} \tag{4.9}$$

where w_{GM} denotes the geometric mean of $w_{p,H}$ and the pole frequency of the EA, $w_{p,EA}$. Now reckoning (4.4) as an open-loop system, if its second term is equal to



Figure 4.8: Comparison between frequency responses of O(s) with a single-pole EA and a two-pole EA.



Figure 4.9: (a) Comparison between $|\tilde{H}(s)|$ and $|\hat{H}(s)|$. (b) H_{mid} over $k_{\text{s,DCR}}$.

unity, then the complete cancellation of supply noise can be achieved. This can be satisfied if and only if $s = iw_{\text{GM}}$ and

$$k_{s,\text{DCR}} = k_{s,\text{DCR}}$$
$$\stackrel{\Delta}{=} k_{s,G_m} \left\{ 1 - \frac{1}{R_{\text{DCR}}G_m A_{0,\text{EA}}} \left(1 + \frac{w_{\text{p,H}}}{w_{\text{p,EA}}} \right) \right\}, \tag{4.10}$$

implying that, with finite $R_{\text{DCR}}G_mA_{0,\text{EA}}$ and $w_{\text{p,EA}}/w_{\text{p,H}}$, $k_{\text{s,DCR}}$ should be slightly smaller than k_{s,G_m} (refer to Appendix B). Fig. 4.9(a) plots the magnitude of H(s)given (4.3) and (4.10), $\hat{H}(s)$, indicating a vertical asymptote, i.e., an infinite gain, at $w = w_{\rm GM}$ in accordance with the derivation. Moreover, it achieves high gain near $w_{\rm GM}$ even with moderate values of $A_{0,EA}$ and G_m . Thus, for the remainder of this chapter, (4.3) and (4.10) together are referred to as the golden condition. Nevertheless, it is yet premature to conclude that the performance of the ACSC is defined as it is since the golden condition is derived from the open-loop point of view. However, by virtue of the replica-based configuration, PVT variations that obstruct the satisfaction of the golden condition take place in a very limited manner: The dependences of R_{DCR} and G_m over temperature are equally reflected to their replicas, and only the local process variation is inflicted to the replica pairs. It is also worth noting that k_{s,G_m} is hardly varied by the supply, and so is $k_{s,DCR}$ at a given D_{FTW}^{I} . Therefore, as illustrated in Fig. 4.9(b), we conclude that the ACSC is of a closed-loop system with its minimum guaranteed H_{mid} dictated by the mismatch that comes from these restrained PVT variations and additional systematic inaccuracy of the circuit implementation, which will be described later.

The EA itself is also affected by the supply, intervening in the operation of the ACSC. We may express the direct impact of supply noise on v_{ctrl} as a separate path with a gain $A_{v_{\text{DD}}}(s)$, giving

$$v'_{\text{ctrl}}(s) = v_{\text{ctrl}}(s) + A_{v_{\text{DD}}}(s) \cdot v_{\text{DD}}(s).$$
 (4.11)

With this, $\tilde{H}(s)$ is modified to

$$\tilde{H}'(s) = (1 - R_{\text{DCR}}G_m \cdot A_{v_{\text{DD}}}(s)) \cdot \tilde{H}(s).$$
(4.12)

Although the additional path multiplied to $\tilde{H}(s)$ seems to be capable of an independent

supply noise cancellation, meeting the exact required values of its parameters regresses the system back to a replica-less open-loop configuration. Notwithstanding, its gain should not be greater than unity so as not to degenerate \tilde{H}_{mid} . The EA used in the ACSC easily meets this requirement thanks to the differential topology, giving an additional rejection gain of about 2.2 dB at the frequencies from the DC to $w_{p,EA}$. It is also worth noting that, in static large-signal perspective, since the ADPLL itself adjusts $V_{DD,I}$ not to vary over the supply, the supply dependence of the input commonmode voltage of the EA and its subsequent impact on V_{ctrl} are further alleviated.

At this point, it is worth leaving some comments as for the SF. While the gain of the SF to $V_{\text{DD,I}}$ does not effect the transfer functions of the ACSC, an additional SF to $V_{\text{R-DD,I}}$ slightly degenerates the gain of the ACSC. Although, it is previously assumed that the gain of the SF is unity, the identical equations can be obtained even if it is not. On the other hand, assume that an additional SF to $V_{\text{R,DD-I}}$, with its gain denoted as $A_{\text{R,SF}}$, is implemented to the ACSC, then $A_{\text{SF,R}}$ does appear in the resulting transfer functions. For example, H(s) with $A_{\text{R,SF}}$ accounted is given by $1/(1 + R_{\text{DCR}}G_mA_{\text{R,SF}} \cdot A_{\text{EA}}(s) \cdot H_{\text{HPF}}(s))$, indicating a degradation of H_{mid} as compared with (4.9). Quantitatively, this would result in 1.5-dB to 2-dB loss of H_{mid} .

4.2.3 Circuit Optimization

A careful consideration on the noise contributions of the ACSC components should be made to ensure the low-jitter operation of the ADPLL. For the sake of simplicity, the following noise analysis assumes that H(s) is given by $\tilde{H}(s)$. As well as the compensation current of the VIC, its noise current is converted to $v_{\text{DD,I,Th}}$ through the transfer function

$$Z_{\text{NTF},1}(s) = R_{\text{DCR}}.\tag{4.13}$$



Figure 4.10: Magnitudes of the noise transfer functions.

The transfer function from the noise current of the R-DCR and the R-VIC to $v_{DD,I,Th}$ is derived as

$$Z_{\text{NTF},2}(s) = k_{\text{s,DCR}} R_{\text{DCR}}^2 G_m \cdot A_{\text{EA}}(s) \cdot H_{\text{HPF}}(s) \cdot \tilde{H}(s).$$
(4.14)

The noise voltage of $R_{\rm L}$, that of $R_{\rm H}$, and the input-referred noise voltage of the EA all undergo different transfer functions to $v_{\rm DD,I,Th}$, which are derived as

$$H_{\text{NTF},3}(s) = R_{\text{DCR}}G_m \cdot A_{\text{EA}}(s) \cdot H_{\text{LPF1}}(s) \cdot H(s), \qquad (4.15)$$

$$H_{\text{NTF},4}(s) = -H_{\text{NTF},3}(s),$$
 (4.16)

and

$$H_{\text{NTF},5}(s) = -R_{\text{DCR}}G_m \cdot A_{\text{EA}}(s) \cdot H(s), \qquad (4.17)$$



Figure 4.11: PSD mask used for the optimization of the ACSC parameters.



Figure 4.12: Frequency response of the ACSC in conjunction with the ADPLL where $f_{p,H}$ and $f_{p,EA}$ coincide with $f_{z,PLL}$ and $f_{BW,PLL}$, respectively.

respectively. Fig. 4.10 visualizes the noise transfer functions in frequency domain. It is worth mentioning that the PSDs at $v_{DD,I,Th}$ by the VIC, the R-VIC, and the EA take into account their flicker noise (1/*f*) contributions and are denoted as $S_{v_n}^{VIC}(f)$, $S_{v_n}^{R-VIC}(f)$, and $S_{v_n}^{EA}(f)$, respectively. Optimizing the ACSC parameters can be carried out by several approaches. Among them, as illustrated in Fig. 4.11, we introduce one intuitive, straightforward method where each phase noise contribution is aimed to be held below a specific mask, $\mathcal{L}_M(f)$. The target phase noise profile of the output clock, $\mathcal{L}_T(f)$, is



Figure 4.13: Consideration for determining the device parameters of (a) the HPF and (b) the EA. The resulting W_{EA} satisfies $f_{\text{BW,PLL}} \leq f_{\text{p,EA}}$.

first drawn roughly, and, with a noise margin $\Delta \mathcal{L}$, $\mathcal{L}_{M}(f)$ is set to $\mathcal{L}_{T}(f) - \Delta \mathcal{L}$. We then determine the maximum permitted PSD at $v_{DD,I,Th}$ by each ACSC component as

$$S_{v_n}^{\mathbf{M}}(f) = \frac{1}{|1 - G_{\text{PLL}}(f)|^2} \cdot \left(\frac{2f}{K_{v_{\text{DD}}}}\right)^2 \cdot \mathcal{L}_{\mathbf{M}}(f).$$
(4.18)

If $f_{p,H}$ and $f_{p,EA}$ coincide with the open-loop zero frequency of the ADPLL, $f_{z,PLL}$, and the ADPLL bandwidth, $f_{BW,PLL}$, respectively, then, as illustrated in Fig. 4.12, the ACSC flattens the overall supply sensitivity of the DCO at the frequency band from f_{LB} to f_{UB} with the suppression gain given by \tilde{H}_{mid} . Therefore, as a rule-of-thumb, the ACSC desires that $f_{p,H}$ be smaller than $f_{z,PLL}$ and $f_{p,EA}$ be larger than $f_{BW,PLL}$. Then, given a $f_{p,H}$, high R_{H} results in increased level of its PSD at $v_{DD,I,Th}$, $S_{v_n}^{R_H}(f)$, whereas high C_H leads to a larger chip area. Implied by Fig. 4.13(a) is that $S_{v_n}^{R_H}(f)$ always meets its requirement regardless of \tilde{H}_{mid} if R_H is small enough to push its -20-dB roll-off region below that of $S_{v_n}^{M}(f)$. In the presented work, R_H (= R_L) and C_H (= C_L) are given by 4 k Ω and 30 pF, respectively.

As for optimizing the EA, its device channel length, L_{EA} , is first considered. For the EA to harness $S_{v_n}^{\text{M}}(f)$ as efficiently as possible, L_{EA} that makes its flicker noise



Figure 4.14: Simulated phase noise of the free-running 8-GHz clock.

corner, $f_{1/f,EA}$, less than or equal to $f_{z,PLL}$ is chosen, also determining $A_{0,EA}$. Next, the device width of the EA, W_{EA} , is adjusted so that $S_{v_n}^{EA}(f_{LB})$ becomes equal to $S_{v_n}^{M}(f_{LB})$ as shown in Fig. 4.13(b). The required W_{EA} depends on G_m given that R_{DCR} and $A_{0,EA}$ are determined. Note that, assuming G_m is not excessively high, both $S_{v_n}^{VIC}(f)$ and $S_{v_n}^{R-VIC}(f)$ with moderate k_{s,G_m} are negligible to $S_{v_n}^{EA}(f)$, and thus the requirement on them can be easily met. Therefore, at this point, the decision on G_m is up to the trade-off between the supply rejection performance and the power consumption of the VICs as well as that of the EA required to meet its constraint. Note that, in this design, the resulting minimum required $f_{p,EA}$ is about 30 MHz, which is larger than $f_{BW,PLL}$. For a low-jitter PLL operation, lowering the device noise should be prioritized to the flattened suppression band condition despite the higher power consumption. Conclusively, $f_{p,EA}$ is given by 35 MHz, resulting in the EA power consumption of 0.82 mW (8.6% of the total).



Figure 4.15: Circuit implementation of the DCR and the R-DCR.

Under the golden condition, high k_{s,G_m} and subsequently high $k_{s,DCR}$ effectively reduce the power consumption of the R-VIC and the area of the R-DCR, respectively. However, they also aggravate the process variations of $G_{m,R}$ and R_{R-DCR} that deviate the ACSC from the golden condition. Thus, in this work, k_{s,G_m} is chosen as about 10, giving an adequate compromise between the trade-off. Fig. 4.14 plots the simulation result on the overall phase noise of the free-running clock along with the contribution of each ACSC component. At the offset frequency of 1 MHz and 10 MHz, the ACSC increases the overall phase noise by 1.6 dBc/Hz and 0.2 dBc/Hz, respectively.



Figure 4.16: Simulated discrepancy between $k_{s,DCR}$ and $\hat{k}_{s,DCR}$ over D_{FTW}^{I} .



Figure 4.17: Simulated |H(s)| in the typical corner. The cut-off frequency at a few gigahertz is attributed to parasitic capacitance at $V_{\text{DD,I}}$.

4.3 ADPLL Implementation

In the ADPLL, a 7-bit vernier-delay-line-based TDC [74] and a 4-stage pseudodifferential CMOS RO [63] are used. $f_{BW,PLL}$ is set high enough to sufficiently suppress the phase noise of the high-frequency RO, which has a high flicker noise corner as shown in Fig. 4.14, as well as that of the ACSC. The proportional path, as mentioned earlier, bypasses the digital loop latency and thereby prevents the phase margin of the ADPLL from being encroached by the high $f_{\text{BW,PLL}}$ [75].

The schematic of the DCR along with the R-DCR is shown in Fig. 4.15. The DCR consists of 32 row PMOS resistors each with 32 column PMOS switches, giving a 10-bit tuning step. It should be noted that, without a further treatment, the static current of the VIC increases the tuning range of the ADPLL. The minimum tunable frequency of the DCO, which is given at the maximum tunable R_{DCR} , is lowered by the additional voltage drop by the VIC, which can be quantified by $I_{\text{VIC}}(R_{\text{DCR}}||R_{\text{RO}})$, where $I_{\rm VIC}$ denotes the nominal current of the VIC. On the other hand, the decrease of the maximum tunable frequency of the DCO is less significant owing to the smaller R_{DCR} . As a consequence, the tuning range actually becomes higher, by about 0.6 GHz, as compared with that given without the ACSC. However, since the word length of D_{FTW} is fixed to 10bit, the increased tuning range results in higher K_{DCO} . This should be avoided so as to lower the in-band noise contribution of the DCO quantization noise and dithering noise. Therefore, both the maximum tunable frequency and the minimum tunable frequency is restored to what they were without the ACSC by reducing R_{DCR} at the expense of 16% increase of the DCR area and 7.4% decrease of $G_{\rm mid}$ of the ACSC. The resulting tuning range of the presented ADPLL is from 6.2 GHz to 9.9 GHz in the typical corner, being sufficiently large to give the 8-GHz oscillation under any PVT corners.

As opposed to the DCR, the R-DCR incorporates only a single column in each row. The PMOS resistors and switches thereof are sized so that, by $D_{\text{FTW}}^{\text{row}}$, $R_{\text{R-DCR}}$ is calibrated to roughly track $\hat{k}_{s,\text{DCR}}R_{\text{DCR}}^{-1}$ over PVT variations, resulting in the systematic

¹In the strict sense, $R_{\text{R-DCR}}$ should track $\hat{k}_{\text{s,DCR}}(R_{\text{DCR}}||R_{\text{P-DCR}})$ as stated earlier. Otherwise, H_{mid} would be degenerated especially in a case where $R_{\text{P-DCR}}$ is not sufficiently larger than R_{DCR} .



*Corners refer to {NMOS, PMOS, ${}^{\dagger}V_{\text{DD}}$, ${}^{\dagger\dagger}T$ } states. *F: +10% V_{DD} , S: -10% V_{DD} . †† F: -40°C, S: 85°C.

Figure 4.18: Simulated statistics of H_{mid} in eight different PVT corners.

inaccuracy mentioned earlier. Fig. 4.16 plots the the resulting discrepancy between $k_{s,DCR}$ and $\hat{k}_{s,DCR}$ throughout the overall D_{FTW}^{I} . The worst discrepancy due to the systematic inaccuracy is shown as about -2.1%. A 20k Monte Carlo simulation reveals that the peak-to-peak discrepancy by local process variation is about 3.8% in the slow corner. Shown in Fig. 4.17 is the simulated frequency response of H(s) in the typical corner. Here, offset variation of the EA is also taken into account. Note that the non-zero DC gain is attributed to the supply sensitivity of the EA described earlier. The simulated statistics of H_{mid} in eight different PVT corners are provided in Fig. 4.18.

The lock time is indeed a very important factor in that it highlights one of the advantages of this work. The simulated lock time of the presented ADPLL with D_{FTW} starting from its mid-code under the SSSS corner is 18.8 us while that under the FFFF corner is 17.3 us. These values are acquired from Verilog simulations where the proportional (β) and the integral (α) gains are set to give the lowest output jitter. In [69], it can be inferred from the last paragraph of the Appendix that the required calibration time is a few hundred microseconds; From Fig. 19.5.5 of [70], the calibration time is over a one second; In [71], since the operation principle of its digital



Figure 4.19: Chip photomicrograph.

calibration is similar to the FSM of [70], it can be inferred that its calibration time is comparable to that of [70], being larger than a few milliseconds at its minimum. Their claimed performance of supply noise rejection is achieved only after this long calibration (and even after a post-fabrication foreground calibration in [71]), and we believe that this is why they lack their performance under large random noise although appreciable single tone noise rejections are achieved therein. On the other hand, the characteristic of the supply noise transfer in the ACSC is already defined in the AC domain without requiring further calibration nor additional lock time thanks to the closed-loop configuration.

4.4 Measurement Results

The ADPLL with the ACSC is fabricated in a 40-nm CMOS technology, occupying a total active area of 0.055 mm² as shown in Fig. 4.19. The supply of the DCO is ACcoupled to a noise source provided by a function generator (Tektronix AFG 3102C)



Figure 4.20: Measured phase noise without any injected supply noise (when the ACSC is enabled).

via an off-chip capacitor. A digital oscilloscope (Tektronix DPO 4054) observes the actual transient profile of the on-chip V_{DD} along with $V_{DD,I}$ that is brought off-chip through a unity gain buffer located near the DCO. The evaluations are conducted with the nominal V_{DD} and f_{out} set to 1.1 V and 8 GHz, respectively. Fig. 4.20 shows the measured phase noise of the ADPLL without any injected supply noise, and the integrated rms jitter (from 10 kHz to 100 MHz), σ_{jitter} , is measured as 289 fs and 275 fs with the ACSC enabled and disabled, respectively.

When a 20-mV_{pp}, 1-MHz sinusoidal supply noise is injected, the ACSC reduces σ_{jitter} from 1.35 ps to 0.38 ps as shown in Fig. 4.21. Furthermore, the reduction in the supply-induced spur by the ACSC, $\Delta Spur$, at the 1-MHz offset frequency is measured as -21.5 dB. Note that the harmonic spurs, which are induced by the nonlinearity of the DCR and the P-DCR, are also suppressed. Shown in Fig. 4.22 are the observed waveforms of V_{DD} and $V_{\text{DD,I}}$, the former given by a 50-mV_{pp}, 1-MHz sinusoidal signal.



Figure 4.21: Measured phase noise under a 1-MHz sinusoidal supply noise.



Figure 4.22: Transient waveforms of V_{DD} and $V_{DD,I}$ that are observed through an onchip unity gain buffer.



Figure 4.23: Measured (a) $\Delta Spur$ and (b) σ_{jitter} under sinusoidal supply noise.

Here, $D_{\text{FTW}}^{\text{I}}$ and $D_{\text{FTW}}^{\text{p}}$ are fixed to give the nominal f_{out} of 8 GHz. The ACSC reduces the amplitude of $V_{\text{DD,I}}$ by -20.4 dB, giving an excellent agreement with the $\Delta Spur$. Note that, with the ACSC disabled, $V_{\text{DD,I}}$ is merely the voltage-divided result of V_{DD} as mentioned earlier. Fig. 4.23(a) plots $\Delta Spur$ at the frequency of the injected sinusoidal supply noise, $f_{n,\text{sin}}$, which is swept from 20 kHz to 90 MHz. The dashed line is the simulated minimum guaranteed profile of $\Delta Spur$. The worst peak-to-peak variation in the $\Delta Spur$ of four measured chips is 1.5 dB at $f_{n,\text{sin}} = 40$ MHz. Fig. 4.23(b) plots the change in σ_{jitter} versus $f_{n,\text{sin}}$.

Fig. 4.24 plots the phase noise under a strong (20-mV_{rms}) white supply noise, showing that σ_{jitter} is reduced from 8.67 ps to 0.63 ps with the ACSC. In addition, the measured peak-to-peak absolute jitter is reduced from 79.1 ps to 6.8 ps as shown in Fig. 4.25. Fig. 4.26(a) plots σ_{jitter} versus the strength of the injected white supply noise ranging from 1 mV_{rms} to 60 mV_{rms}. Without the ACSC, the σ_{jitter} grows exponentially until the noise strength becomes so large that the ADPLL falls into an out-of-lock state. With the ACSC, the σ_{jitter} also grows exponentially but with significantly lower values. To strictly quantify the performance of supply noise compensation, we employ



Figure 4.24: Measured phase noise under a 20-mV_{rms} white supply noise.



Figure 4.25: Measured peak-to-peak time interval error (TIE) under a 20-mV_{rms} white supply noise.

an FoM that is defined by

$$\operatorname{FoM}_{\operatorname{SNC}}\left(\operatorname{dB}\right) = 10\log\left(\frac{\hat{\sigma}_{j,\operatorname{noisy}}^2 - \sigma_{j,\operatorname{clean}}^2}{\sigma_{j,\operatorname{noisy}}^2 - \sigma_{j,\operatorname{clean}}^2}\right)$$
(4.19)



Figure 4.26: (a) σ_{jitter} under white supply noise and (b) the resulting FoM_{SNC}. Without the ACSC, the phase lock fails when $V_{\text{rms}} > 30$ mV.



Figure 4.27: (a) $\Delta Spur$ and (b) FoM_{SNC} versus static supply variation.

where $\hat{\sigma}_{j,noisy}$ and $\sigma_{j,noisy}$ denote σ_{jitter} under white supply noise with and without supply noise compensation, respectively, and $\sigma_{j,clean}$ is σ_{jitter} without any injected supply noise. It is worth noting that the ACSC under the desired operation condition only rejects supply noise in a specific frequency band that is vulnerable to supply noise. For this reason, the DC sensitivity of the DCO with the ACSC is merely the voltage-divided


Figure 4.28: Pie chart for power consumption breakdown.

result by the DCR and the RO as mentioned earlier. In [71], the supply sensitivity accounts the settled frequency hop due to a certain change in the DC level of V_{DD} , without considering the AC effect of the overall ADPLL. This metric is an important FoM for free-running clock sources such as those for low-speed readout systems. However, for precision-timing PLLs, what matters the most is the susceptibility of the overall output jitter by supply noise, which can be strictly quantified by FoM_{SNC}. The DC supply insensitivity would indeed help a PLL to continuously maintain its lock under supply hops without requiring additional settling time. However, if the calibration circuit itself requires a settling time for the given supply hops, which, in [69]-[71], is significantly high as compared with the lock-time of a typical PLL, this declaration is completely paralyzed. Fig. 4.26(b) plots FoM_{SNC} of the this work, giving -23.8 dB at $V_{\rm rms} = 20$ mV, with the peak-to-peak variation of the four measured chips of 0.9 dB.

Shown in Fig. 4.27 are $\Delta Spur$ and FoM_{SNC} versus the DC value of V_{DD} , which is varied by $\pm 10\%$ from its nominal value, indicating no significant performance deviation. The total power consumption is 9.48 mW, with the DCO and the ACSC consuming 67% and 17.6%, respectively, as shown in Fig. 4.28. Table 4.1 summarizes

ion
nsat
npe
Cor
se
[0]
Y N
lqqu
S
with
LS I
Ize
esi
nth
Syı
S.
enc
nb
Fre
[Ŋ
ase
-B
S
or]
, Li
Ηų
wit
'n
ris(
paı
шc
Ŭ
÷
e 4
abl

This Work	40	160	8	8	1.1	9.48	1.67, $(17.6%)$	-116.5	-67.7	0.289	-241	-258	0.055	8.7	0.63	-23.8	-14.7, 0.4	-27.3, 4	-21, 40	Closed Loop
VLSI'2017 [76]	65	50	3.2	10	1	2.73	0.45, $(16.5%)$	-84	N/A	7.5	-218.1	-236.2	0.047	14.1	7.8	-14.9	-32, 0.5	-27.5, 2.5	-25.2, 20	Regulated Charge Pump
ISSCC'2016 [70]	40	200	3.2	N/A	1.1	2.92	N/A	-105	N/A	3.54	-224.4	-236.4	0.022	N/A	N/A	N/A	-11 [‡] , 0.01	-7.6 [‡] , 1	$^{-2.6^{\ddagger}},$	Open Loop w/ Background Calibration
ISSCC'2014 [71]	40	26	2.418	4	1.1	6.4	N/A	N/A	-75	N/A	-221.6	-241.3	0.013	N/A	3.29	N/A	-33†, 5	-20†, 20	-5†, 80	Open Loop w/ Background Calibration ^{††}
ISSCC'2014 [68]	20	25	1.6	8	0.9	3.1	N/A	-86	N/A	5.83	-219.8	-237.8	0.012	N/A	5.89	N/A	N/A	N/A	N/A	Open Loop
JSSC'2011 [69]	130	625	2.5	9	1	3.1	0.24, (7.7%)	N/A	N/A	4.6	-221.8	-227.8	0.08	8	4.8	-13.6	N/A	N/A	N/A	Open Loop w/ Background Calibration
	Technology (nm)	fref (MHz)	f_{out} (GHz)	Number of Output Phase	$V_{\rm DD}$ (V)	Total Power Consumption (mW)	Power Consumption of Compensation Circuitry (mW), (% w.r.t. Total Power)	Phase Noise @ 1 MHz w/o Supply Noise (dBc/Hz)	Reference Spur (dBc)	$\sigma_{\rm j, clean}$ (ps)	FoM_1^* (dB)	FoM_2^{**} (dB)	Area (mm ²)	$20-mV_{rms}$ $\sigma_{j,noisy}$ (ps)	White Supply $\hat{\sigma}_{j,\text{noisy}}$ (ps)	Noise FoM _{SNC} (dB)	$\Delta Spur$ under a Sinusoidal Supply Noise (dB), $f_{n, \sin}$ (MHz)			Supply Noise Compensation Configuration

 $*FoM_{1} = 10 \log \left\{ \left(\frac{\sigma_{\rm locum}}{1s} \right)^{2} \left(\frac{Power}{1wW} \right)^{2} \right\} . \\ **FoM_{2} = FoM_{1} + 10 \log \left(\frac{f_{\rm ref}}{f_{\rm out}} \right).$

[†]Reduction in the spur by the background calibration compared with the manually configured operation point that gives its worst pushing factor. ^{††}Preceded by several foreground calibrations. [‡]Average value given from five chips.

the performance of the presented work and compares it with prior state-of-the-art PLLs with supply noise compensation schemes. The power consumption of this work is higher than the others due to the realization of the 8-phase RO, whose f_{out} is the highest, and the additional active devices in the ACSC. However, $\sigma_{j,clean}$ of this work is below one tenth of the others, achieving the best FoM₁, which is -241 dB. It should be noted that even if, in [69] and [70], the calibration point is well found, large supply noise may simultaneously change the bias condition and thus the path gain. This perturbation cannot be recovered promptly due to the slow update rate of the calibration. On the other hand, the ACSC is robust over PVT variations, giving the best performance under large white supply noise.

4.5 Summary

An RO-based ADPLL implemented with an analog circuit technique that suppresses supply-induced noise in the DCO is presented. The technique does not extort the voltage headroom of the DCO, nor does it add any circuit components in the delay cell of the RO, thereby allowing high-frequency oscillation. In addition, its performance is hardly disrupted by PVT variations thanks to the replica-based implementation. Furthermore, a comprehensive noise analysis reveals that it does not sacrifice the low-jitter operation of the ADPLL. Measured results show that it achieves an rms jitter of 289 fs at 8 GHz output without any injected supply noise and a supplynoise-induced jitter rejection of -23.8 dB, both being the best as compared with prior designs.

Chapter 5

Conclusions

Concerns in the clocking of modern SerDes along with its brief history and trend are described first in this dissertation. As sub-rate, multi-standard architectures are becoming predominant, the conventional clocking scheme—LC-based clock synthesis followed by multi-phase conversion—seems to face some new challenges in terms of the low-cost implementation. In pursuit of an innovation, challenges and opportunities that exist in the frequency synthesis using a high-frequency, inductor-less oscillator are reviewed, followed by the demonstration of embodiments that addresses the two major flaws.

As the first demonstration, we designed the $8 \times \text{REF}$ to extend the bandwidth of the following ADPLL and thereby sufficiently filtering the high flicker noise of a highfrequency RO. By avoiding jitter accumulation, $8 \times \text{REF}$ outputs a clean mid-frequency clock, overall achieving high jitter performance when cascaded to an ADPLL. Delay constraint in the $8 \times \text{REF}$ design is first analyzed, revealing that a five-point calibration is required to accurately correct the phase error. Given the theory that the statistics of the BBPD output repeat every reference cycle, the sequence of which depends on the values of the error sources, the weight for each calibration point is updated by pre-defined LUTs that are given from *a priori* probability calculations on the basis of the 8-bit BBPD output sequence. To minimize the settling time at the startup or after sudden environmental disturbance, the VSS algorithm is adopted into the calibration engine, identifying which error source is being dominant and then adjusts the gain of each weight generation continuously throughout the adaptation. The supply sensitivity of the large-delay paths of the 8×REF is minimized by adopting differential topology, and the circuit for calibrating phase errors that are induced by propagation mismatches is realized by implementing a CMOS-based DL. The prototype chip is fabricated in a 40-nm CMOS technology, and the measurement results thereof validated the effectiveness of flicker noise filtering and that of the VSS algorithm.

Despite the state-of-the-art jitter-power-N performance, the proposed IL-ADPLL entails some remaining concerns that are to be addressed in the future. First, the methodology of the probability-based weight generation lacks rigorous mathematical theory that enlightens the transfer function from phase error sources to the output jitter, making it difficult to exactly predict the resulting performance metrics. Second, although the analog path of the $8 \times \text{REF}$ avoids $1/f^2$ -upconvserion of its noise components, its thermal noise floor dominates the phase noise of the overall output clock of the IL-ADPLL due to the required large delays; while the presented work, with a 100-MHz reference, showed a remarkable performance improvement compared to prior arts, we may not sufficiently conclude that the proposed technique would always be beneficial regardless of specifications, e.g., mid-frequency generation with a lowerfrequency reference would require larger delays, resulting in higher thermal noise floor. Last but not least, the achieved performance is still inferior to the state-of-the-art LC-based frequency synthesizers, meaning that, in the ultra-high speed regime, it is insufficient that an RO be considered as a proper candidate over an LC counterpart. Notwithstanding, the presented work readily satisfies our motivation, the jitter performance for PCIe 5/6, and, we expect that it also meets the jitter requirements for <56-Gbaud/s SerDes in a liberal sense.

As the second demonstration, we presented an ADPLL that is implemented with a high-gain analog closed loop for RO supply noise compensation. While prior works including LDOs necessitate additional voltage headroom over the RO, the compensation circuit for the proposed technique is implemented in a parallel manner, allowing high-frequency oscillation. Further, the technique forms a null-filter under certain constraints, which rely on the difference in the values of analog pairs rather than the absolute values thereof. Therefore, the intended performance is robust over PVT variations, avoiding the use of additional calibration hardware. Moreover, a comprehensive analysis of noise contribution of the circuits for the supply noise compensation is conducted for the ADPLL to retain its low-jitter output. Implemented in a 40-nm CMOS technology, the ADPLL achieves an rms jitter of 289 fs at 8 GHz without any injected supply noise. Under a 20-mV_{rms} white supply noise, the ADPLL gives an rms jitter of 8.7 ps and 0.63 ps at 8 GHz when the ACSC is disabled and enabled, respectively. With the overall power consumption being 9.48 mW, the achieved performance is stateof-the-art among RO-based frequency synthesizers with supply noise compensation techniques.

Despite the effectiveness of the supply noise compensation, the proposed technique has two drawbacks. First, it requires large capacitors in order to widen the noise rejection band while not much affecting the output phase noise. Although the overall area is less than general LC-based frequency synthesizers, such passive elements counteract the benefits of technology scaling. Second, as the minimum guaranteed noise rejection gain depends on the local process variations, it may get degenerated with smaller technology nodes, possibly requiring a sophisticated calibration engine for the optimum performance.

Chapter A

Notes on the 8×REF

Weight Generation Accounting for the PLL Bandwidth

Indicated by (3.7) is that $\phi_{\varepsilon_0}(n)$, $\phi_{\varepsilon_1}(n)$, $\phi_{\varepsilon_2}(n)$ are 50%-duty-cycle, rectangular pulse signals with the fundamental frequencies of f_{ref} , $2f_{\text{ref}}$, and $4f_{\text{ref}}$, respectively; $\phi_{\varepsilon_3}(n)$ and $\phi_{\varepsilon_4}(n)$ are also rectangular pulse signals with the fundamental frequencies of $2f_{\text{ref}}$ and f_{ref} , respectively, but their duty cycle is not 50%, giving higher portion of their harmonics at the frequency domain. Therefore, the finite PLL bandwidth would suppress and distort each $\phi_{\varepsilon_i}(n)$ with different profiles, invalidating the use of the simple, linear generation matrix for predicting the BBPD output. To account for this, Fig. 3.10 is modified to Fig. A.1 where the time-invariant matrix H_{PLL} associates the loop transfer function of the ADPLL, whose closed-form expression is very cumber-



Figure A.1: Modified Fig. 3.10 with the finite ADPLL bandwidth accounted for.

some, hardly giving us an intuition. It is worth mentioning that, since $H_{PLL} \cdot \delta(k)$ is deterministic, the resulting $\phi_{8\times ref}(n)$ is of a cyclostationary noise. With a proper modeling of the transfer function of the designed ADPLL, whose bandwidth is given as about 100 MHz with the calibrated $8\times REF$, we obtain

$$\boldsymbol{\delta} - \boldsymbol{H}_{\text{PLL}} \cdot \boldsymbol{\delta} = \begin{bmatrix} 0.85 & 0.68 & 0.61 & 0.41 & 0.66 \\ 0.51 & 0.43 & -0.61 & 0.26 & 0.38 \\ 0.34 & -0.68 & 0.61 & 0.16 & 0.24 \\ 0.19 & -0.43 & -0.61 & -0.88 & 0.16 \\ -0.85 & 0.68 & 0.61 & 0.41 & 0.11 \\ -0.51 & 0.43 & -0.61 & 0.26 & 0.09 \\ -0.34 & -0.68 & 0.61 & 0.16 & -0.95 \\ -0.19 & -0.43 & -0.61 & -0.88 & -0.58 \end{bmatrix} \cdot \boldsymbol{\epsilon}. \quad (A.1)$$



Figure A.2: Calculation results of (3.12) given $d = \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 \end{bmatrix}^T$ after calibration settlement with the finite ADPLL bandwidth accounted for. (\blacklozenge : X_0 , \blacktriangledown : X_1 , \blacktriangle : X_2 , \blacksquare : X_3 , \blacklozenge : X_4)

settle with the LUT that does not consider the finite ADPLL bandwidth. Nevertheless, no solid clue that the LUT-A and LUT-B implemented in the presented prototype respectively give the best quantization noise and settling time is provided. Further, it should be noted that it is not straightforward to theoretically prove which LUTs are the optimal ones.

Chapter B

Notes on the ACSC

General Transfer Functions of the ACSC

Assuming that the gain of the SF is unity, Fig. 4.6 can be further simplified to Fig. B.1 as provided below. Then, the transfer function from $v_{DD,I}$ to v_{err} is simply given by

$$H_{\rm LPF1}(s) \stackrel{\Delta}{=} \frac{v_{\rm err}(s)}{v_{\rm DD,I}(s)} = \frac{1}{1 + sR_{\rm L}C_{\rm L}} = \frac{1}{1 + s/w_{\rm p,L}},\tag{B.1}$$

corresponding to a first-order low-pass filter. The equation for v_{err+} on the basis of $v_{DD,I}$ and $v_{R-DD,I}$ can be obtained by the current equation at v_{err+}

$$\frac{v_{\rm err+}(s) - v_{\rm R-DD,I}(s)}{1/(sC_{\rm H})} + \frac{v_{\rm err+}(s) - v_{\rm DD,I}(s)}{R_{\rm H}} = 0,$$
 (B.2)

which can be rearranged to

$$v_{\rm err+}(s) = \frac{1/(sC_{\rm H})}{R_{\rm H} + 1/(sC_{\rm H})} \cdot v_{\rm DD,I}(s) + \frac{R_{\rm H}}{R_{\rm H} + 1/(sC_{\rm H})} \cdot v_{\rm R-DD,I}(s)$$

= $H_{\rm LPF2}(s) \cdot v_{\rm DD,I}(s) + H_{\rm HPF}(s) \cdot v_{\rm R-DD,I}(s),$ (B.3)



Figure B.1: Simplified representation of the passive filters.

giving (4.2). If the impedance of the HPF seen from $V_{\text{R-DD,I}}$ is sufficiently large, the current equations at $v_{\text{DD,I,Th}}$ and $v_{\text{R-DD,I}}$ are

$$\frac{v_{\text{DD,I,Th}}(s) - v_{\text{DD}}(s)}{R_{\text{DCR}}} + G_m \cdot v_{\text{ctrl}}(s) = 0$$
(B.4)

and

$$\frac{v_{\text{R-DD,I}}(s) - v_{\text{DD}}(s)}{R_{\text{R-DCR}}} + G_{m,\text{R}} \cdot v_{\text{ctrl}}(s) = 0,$$
(B.5)

respectively, giving

$$H(s) = \frac{1 + (R_{\text{R-DCR}}G_{m,\text{R}} - R_{\text{DCR}}G_m) \cdot A_{\text{EA}}(s) \cdot H_{\text{HPF}}(s)}{1 + T(s)}$$
(B.6)

$$H_{\rm R}(s) = \frac{1 + (R_{\rm R-DCR}G_{m,\rm R} - R_{\rm DCR}G_m) \cdot A_{\rm EA}(s) \cdot (H_{\rm LPF1}(s) - H_{\rm LPF2}(s))}{1 + T(s)}$$
(B.7)

where

$$T(s) = A_{\text{EA}}(s) \cdot \left\{ R_{\text{DCR}}G_m \cdot \left(H_{\text{LPF2}}(s) - H_{\text{LPF1}}(s) \right) + R_{\text{R-DCR}}G_{m,\text{R}} \cdot H_{\text{HPF}}(s) \right\}.$$
(B.8)

Taking into account i_{FTW} instead of v_{DD} , the current equations at $v_{\text{DD,I,Th}}$, v_{FTW} , and $v_{\text{R-DD,I}}$ are

$$\frac{v_{\text{DD,I,Th}}(s) - v_{\text{FTW}}(s)}{(1-\alpha)R_{\text{DCR}}} + G_m \cdot v_{\text{ctrl}}(s) = 0, \tag{B.9}$$

APPENDIX B. NOTES ON THE ACSC

$$\frac{v_{\text{FTW}}(s)}{\alpha R_{\text{DCR}}} + \frac{v_{\text{FTW}}(s) - v_{\text{DD,I,Th}}(s)}{(1-\alpha)R_{\text{DCR}}} - i_{\text{FTW}}(s) = 0, \qquad (B.10)$$

and

$$\frac{v_{\text{R-DD,I}}(s)}{R_{\text{R-DCR}}} + G_{m,\text{R}} \cdot v_{\text{ctrl}}(s) = 0, \qquad (B.11)$$

respectively, giving

$$Z(s) = \frac{\alpha R_{\text{DCR}} (1 + R_{\text{R-DCR}} G_{m,\text{R}} \cdot A_{\text{EA}}(s) \cdot H_{\text{HPF}}(s))}{1 + T(s)}$$
(B.12)

$$Z_{\mathbf{R}}(s) = \frac{\alpha R_{\mathbf{DCR}} R_{\mathbf{R}-\mathbf{DCR}} G_{m,\mathbf{R}} \cdot A_{\mathbf{EA}}(s) \cdot (H_{\mathbf{LPF1}}(s) - H_{\mathbf{LPF2}}(s))}{1 + T(s)}.$$
 (B.13)

Proof of the Golden Condition

With the transfer function of the single-pole EA written by

$$A_{\rm EA}(s) = \frac{A_{0,\rm EA}}{1 + \frac{s}{w_{\rm p,\rm EA}}},$$
 (B.14)

(4.4) is equal to zero when

$$\left\{ \frac{w^2 w_{p,\text{EA}}(w_{p,\text{EA}} + w_{p,\text{H}})}{(w^2 + w_{p,\text{EA}}^2)(w^2 + w_{p,\text{H}}^2)} - \frac{w w_{p,\text{EA}}(w^2 - w_{p,\text{EA}}w_{p,\text{H}})}{(w^2 + w_{p,\text{EA}}^2)(w^2 + w_{p,\text{H}}^2)}i\right\} \\
\cdot \left(1 - \frac{k_{s,\text{DCR}}}{k_{s,G_m}}\right) \cdot R_{\text{DCR}}G_m A_{0,\text{EA}} = 1.$$
(B.15)

The imaginary part of its left-hand side becomes zero if $w = w_{\text{GM}}$, the real part yielding (4.10).

Bibliography

- A. Steegen, "Technology Innovation in an IoT Era," in *Proc. IEEE Int. Symp.* VLSI Circuits Dig. Tech. Papers, Jun. 2015, pp. C170-C171.
- [2] P. -C. Chiang, H. -W. Hung, H. -Y. Chu, G. -S. Chen, and J. Lee, "60Gb/s NRZ and PAM4 Transmitters for 400GbE in 65nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 42-43.
- [3] R. Xie *et al.*, "A 7nm FinFET Technology Featuring EUV Patterning and Dual Strained High Mobility Channels," in *Proc. IEEE Int. Electron Devices Meeting* (*IEDM*) Dig. Tech. Papers, 2016, pp. 2.7.1-2.7.4.
- [4] C. Enz and Y. Cheng, "MOS Transistor Modeling for RF IC design," in *IEEE J. Solid-State Circuits*, vol. 35, no. 2, pp. 186-201, Feb. 2000.
- [5] Optical Internetworking Forum (OIF) [Online].Available: https://www.oiforum.com/technical-work/current-work
- [6] K. Chang, G. Zhang, and C. Borrelli, "Evolution of Wireline Transceiver Standards: Various, Most-Used Standards for the Bandwidth Demand," in *IEEE Solid-State Circuits Magazine*, vol. 7, no. 4, pp. 47-52, Fall 2015.

- [7] D. C. Daly, L. C. Fujino, and K. C. Smith, "Through the Looking Glass-2020 Edition: Trends in Solid-State Circuits From ISSCC," in *IEEE Solid-State Circuits Magazine*, vol. 12, no. 1, pp. 8-24, Winter 2020.
- [8] C. Menolfi *et al.*, "A 16Gb/s Source-Series Terminated Transmitter in 65nm CMOS SOI," in *Proc. IEEE Int. Electron Devices Meeting (IEDM) Dig. Tech. Papers*, 2007, pp. 446-614.
- [9] J. Kim et al., "3.5 A 16-to-40Gb/s Quarter-Rate NRZ/PAM4 Dual-Mode Transmitter in 14nm CMOS," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2015, pp. 1-3.
- [10] T. Beukema *et al.*, "A 6.4-Gb/s CMOS SerDes Core with Feed-Forward and Decision-Feedback Equalization," *IEEE J. Solid-State Circuits*, vol. 40, no. 12, pp. 2633-2645, Dec. 2005.
- [11] K. J. Wong, A. Rylyakov, and C. K. Yang, "A 5-mW 6-Gb/s Quarter-Rate Sampling Receiver With a 2-Tap DFE Using Soft Decisions," *IEEE J. Solid-State Circuits*, vol. 42, no. 4, pp. 881-888, April 2007.
- [12] J. D. H. Alexander, "Clock Recovery from Random Binary Signals," *IET Electron. Lett.*, vol. 11, no. 22, pp. 541-542, 1975.
- [13] S.-H. Lee *et al.*, "A 5-Gb/s 0.25-/spl mu/m CMOS Jitter-Tolerant Variable-Interval Oversampling Clock/Data Recovery Circuit," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1822-1830, Dec. 2002.
- [14] "CEI-56G-LR-PAM4 Long Reach Implementation Agreement Draft Text", in Optical Internetworking Forum Contribution OIF, vol. 380, no. 03, June 2016.

- [15] B. Murmann, "The Successive Approximation Register ADC: A Versatile Building Block for Ultra-Low-Power to Ultra-High-Speed Applications," in *IEEE Communications Magazine*, vol. 54, no. 4, pp. 78-83, April 2016.
- [16] L. Kull *et al.*, "A 24–72-GS/s 8-b Time-Interleaved SAR ADC With 2.0–3.3-pJ/Conversion and >30 dB SNDR at Nyquist in 14-nm CMOS FinFET," in *IEEE J. Solid-State Circuits*, vol. 53, no. 12, pp. 3508-3516, Dec. 2018.
- [17] T. Ali et al., "A 180mW 56Gb/s DSP-Based Transceiver for High Density IOs in Data Center Switches in 7nm FinFET Technology," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2019, pp. 118-120.
- [18] J. Im *et al.*, "A 112Gb/s PAM-4 Long-Reach Wireline Transceiver Using a 36-Way Time-Interleaved SAR-ADC and Inverter-Based RX Analog Front-End in 7nm FinFET," *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 116-117.
- [19] Y. Segal et al., "A 1.41pJ/b 224Gb/s PAM-4 SerDes Receiver with 31dB Loss Compensation," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2022, pp. 114-115.
- [20] K. Zheng, Y. Frans, K. Chang, and B. Murmann, "A 56 Gb/s 6 mW 300 um² Inverter-Based CTLE for Short-Reach PAM2 Applications in 16 nm CMOS," in *in Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, 2018.
- [21] A. Poon, A. Chang, H. Samavati, and S. S. Wong, "Reduction of Inductive Crosstalk Using Quadrupole Inductors," in *IEEE J. Solid-State Circuits*, vol. 44, no. 6, pp. 1756-1764, June 2009.

- [22] J. Kim et al., "A 224Gb/s DAC-Based PAM-4 Transmitter with 8-Tap FFE in 10nm CMOS," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2021, pp. 126-128.
- [23] B. Abiri, R. Shivnaraine, A. Sheikholeslami, H. Tamura, and M. Kibune, "A 1to-6Gb/s Phase-Interpolator-Based Burst-Mode CDR in 65nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2011, pp. 154-156.
- [24] P. Andreani, S. Mattisson, and B. Essink, "A CMOS gm-C Polyphase Filter with High Image Band Rejection," in *Proc. Eur. Solid State Circuits Conf. (ESSCIRC)*, Sep. 2000.
- [25] M. Kaltiokallio and J. Ryynnen, "A 1 to 5GHz Adjustable Active Polyphase Filter for LO Quadrature Generation," in *Proc. IEEE RFIC Symp.*, May 2011, pp. 1-4.
- [26] A. Mirzaei, M. E. Heidari, R. Bagheri, S. Chehrazi, and A. A. Abidi, "The Quadrature LC Oscillator: A Complete Portrait Based on Injection Locking," in *IEEE J. Solid-State Circuits*, vol. 42, no. 9, pp. 1916-1932, Sept. 2007.
- [27] A. Mazzanti, P. Uggetti, and F. Svelto, "Analysis and Design of Injection-Locked LC dividers for Quadrature Generation," in *IEEE J. Solid-State Circuits*, vol. 39, no. 9, pp. 1425-1433, Sept. 2004.
- [28] A. Bonfanti, A. Tedesco, C. Samori, and A. L. Lacaita, "A 15-GHz Broad-Band Divide-by-2 Frequency Divider in 0.13-um CMOS for Quadrature Generation," in *IEEE Microwave and Wireless Components Lett.*, vol. 15, no. 11, pp. 724-726, Nov. 2005.

- [29] M. G. Johnson and E. L. Hudson, "A Variable Delay Line PLL for CPU-Coprocessor Synchronization," in *IEEE J. Solid-State Circuits*, vol. 23, no. 5, pp. 1218-1223, Oct. 1988.
- [30] S. Chen et al., "A 4-to-16GHz Inverter-Based Injection-Locked Quadrature Clock Generator with Phase Interpolators for Multi-Standard I/Os in 7nm Fin-FET," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2018, pp. 390-392.
- [31] P. Westlake, "Digital Phase Control Techniques," in *IRE Trans. Comm. Sys.*, vol. 8, no. 4, pp. 237-246, Dec. 1960.
- [32] S. Gupta, "On Optimum Digital Phase-Locked Loops," in *IEEE Trans. Comm. Tech.*, vol. 16, no. 2, pp. 340-344, April 1968.
- [33] N. D'Andrea and F. Russo, "A Binary Quantized Digital Phase Locked Loop: A Graphical Analysis," in *IEEE Trans. Communications*, vol. 26, no. 9, pp. 1355-1364, Sept. 1978.
- [34] R. B. Staszewski, C.-M. Hung, K. Maggio, J. Wallberg, D. Leipold, and P. T. Balsara, "All-Digital Phase-Domain TX Frequency Synthesizer for Bluetooth Radios in 0.13/spl mu/m CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf.* (*ISSCC*) *Dig. Tech. Papers*, Feb. 2004, pp. 272-527.
- [35] R. B. Staszewski, C.-M. Hung, N. Barton, M.-C. Lee, and D. Leipold, "A Digitally Controlled Oscillator in a 90 nm Digital CMOS Process for Mobile Phones," in *IEEE J. Solid-State Circuits*, vol. 40, no. 11, pp. 2203-2211, Nov. 2005.

- [36] P. Madoglio, M. Zanuso, S. Levantino, C. Samori, and A. L. Lacaita, "Quantization Effects in All-Digital Phase-Locked Loops," in *IEEE Trans. Circuits Syst. II*, vol. 54, no. 12, pp. 1120-1124, Dec. 2007.
- [37] N. D. Dalt, "A Design-Oriented Study of the Nonlinear Dynamics of Digital Bang-Bang PLLs," in *IEEE Trans. Circuits Syst. I*, vol. 52, no. 1, pp. 21-31, Jan. 2005.
- [38] J. W. M. Bergmans, "Effect of loop delay on stability of discrete-time PLL," in *IEEE Trans. Circuits Syst. I*, vol. 42, no. 4, pp. 229-231, April 1995.
- [39] G. Marucci, S. Levantino, P. Maffezzoni, and C. Samori, "Exploiting Stochastic Resonance to Enhance the Performance of Digital Bang-Bang PLLs," in *IEEE Trans. Circuits Syst. II*, vol. 60, no. 10, pp. 632-636, Oct. 2013.
- [40] S. Jang, S. Kim, S. Chu, G. Jeong, Y. Kim, and D. Jeong, "An Optimum Loop Gain Tracking All-Digital PLL Using Autocorrelation of Bang–Bang Phase-Frequency Detection," in *IEEE Trans. Circuits Syst. II*, vol. 62, no. 9, pp. 836-840, Sept. 2015.
- [41] A. Hajimiri, S. Limotyrakis, and T. H. Lee, "Jitter and phase noise in ring oscillators," in *IEEE J. Solid-State Circuits*, vol. 34, no. 6, pp. 790-804, June 1999.
- [42] K. K. Hung, P. K. Ko, C. Hu, and Y. C. Cheng, "A Unified Model for the Flicker Noise in Metal-Oxide-Semiconductor Field-Effect Transistors," in *IEEE Trans. Electron Devices*, vol. 37, no. 3, pp. 654-665, March 1990.
- [43] A. A. Abidi, "Phase Noise and Jitter in CMOS Ring Oscillators," in *IEEE J. Solid-State Circuits*, vol. 41, no. 8, pp. 1803-1816, Aug. 2006.

- [44] Y. Lee et al., "A -240dB-FoMjitter and -115dBc/Hz PN @ 100kHz, 7.7GHz Ring-DCOBased Digital PLL Using P/I-Gain Co-Optimization and Sequence-Rearranged Optimally Spaced TDC for Flicker-Noise Reduction," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, pp. 266-267, Feb. 2020.
- [45] F. M. Gardner, "Phaselock techniques," John Wiley & Sons, 2005.
- [46] Sheng Ye, L. Jansson, and I. Galton, "A Multiple-Crystal Interface PLL with VCO Realignment to Reduce Phase Noise," in *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1795-1803, Dec. 2002.
- [47] J. Lee and H. Wang, "Study of Subharmonically Injection-Locked PLLs," in *IEEE J. Solid-State Circuits*, vol. 44, no. 5, pp. 1539-1553, May 2009.
- [48] Y. Huang and S. Liu, "2.4-GHz Subharmonically Injection-Locked PLL With Self-Calibrated Injection Timing," in *IEEE J. Solid-State Circuits*, vol. 48, no. 2, pp. 417-428, Feb. 2013.
- [49] J. Chien *et al.*, "2.8 A pulse-position-modulation phase-noise-reduction technique for a 2-to-16GHz injection-locked ring oscillator in 20nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 52-53.
- [50] R. Farjad-Rad *et al.*, "A low-power multiplying DLL for low-jitter multigigahertz clock generation in highly integrated digital chips," in *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1804-1812, Dec. 2002.
- [51] A. Elshazly, R. Inti, B. Young, and P. K. Hanumolu, "Clock Multiplication Techniques Using Digital Multiplying Delay-Locked Loops," in *IEEE J. Solid-State Circuits*, vol. 48, no. 6, pp. 1416-1428, June 2013.

- [52] G. Chien and P. R. Gray, "A 900-MHz local oscillator using a DLL-based frequency multiplier technique for PCS applications," in *IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 1996-1999, Dec. 2000.
- [53] M.-J. E. Lee *et al.*, "Jitter transfer characteristics of delay-locked loops theories and design techniques," in *IEEE J. Solid-State Circuits*, vol. 38, no. 4, pp. 614-621, April 2003.
- [54] D. Coombs, A. Elkholy, R. K. Nandwana, A. Elmallah, and P. K. Hanumolu, "A 2.5-to-5.75GHz 5mW 0.3ps_{rms}-Jitter Cascaded Ring-Based Digital Injection-Locked Clock Multiplier in 65nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2017, pp. 152-153.
- [55] K. M. Megawer, A. Elkholy, M. G. Ahmed, A. Elmallah, and P. Kumar Hanumolu, "Design of Crystal-Oscillator Frequency Quadrupler for Low-Jitter Clock Multipliers," in *IEEE J. Solid-State Circuits*, vol. 54, no. 1, pp. 65-74, Jan. 2019.
- [56] H. Kim, H.-S. Oh, W. Jung, Y. Song, J. Oh, and D.-K. Jeong, "A 100MHz-Reference, 8GHz/16GHz, 177fsrms/223fsrms RO-Based IL-ADPLL Incorporating Reference Octupler with Probability-Based Fast Phase-Error Calibration," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2022, pp. 382-383.
- [57] F. Song, Y. Zhao, B. Wu, L. Tang, L. Lin, and B. Razavi, "A Fractional-N Synthesizer with 110fs_{rms} Jitter and a Reference Quadrupler for Wideband 802.11ax," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2019, pp. 264-266.
- [58] B. Widrow and M. E. Hoff, "Adaptive switching circuits," *Stanford Univ Ca Stanford Electronics Labs*, Jun. 1960.

- [59] R. W. Lucky, "Techniques for adaptive equalization of digital communication systems." in *Bell System Technical Journal*, vol. 45, no. 2, pp. 255-286, 1966.
- [60] B. P. Carlin and T. A. Louis, "Bayesian methods for data analysis," *CRC Press*, 2008.
- [61] A. Santiccioli *et al.*, "A 66-fs-rms Jitter 12.8-to-15.2-GHz Fractional-N Bang–Bang PLL With Digital Frequency-Error Recovery for Fast Locking," in *IEEE J. Solid-State Circuits*, vol. 55, no. 12, pp. 3349-3361, Dec. 2020.
- [62] R. H. Kwong and E. W. Johnston, "A Variable Step Size LMS Algorithm," in *IEEE Trans. Signal Processing*, vol. 40, no. 7, pp. 1633-1642, July 1992.
- [63] W. Bae *et al.*, "A 7.6 mW, 414 fs RMS-Jitter 10 GHz Phase-Locked Loop for a 40 Gb/s Serial Link Transmitter Based on a Two-Stage Ring Oscillator in 65 nm CMOS," in *IEEE J. Solid-State Circuits*, vol. 51, no. 10, pp. 2357–2367, Oct. 2016.
- [64] C. Liang and K. Hsiao, "An injection-locked ring PLL with self-aligned injection window," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2011, pp. 90-92.
- [65] A. Arakali, S. Gondi, and P. K. Hanumolu, "Low-Power Supply-Regulation Techniques for Ring Oscillators in Phase-Locked Loops Using a Split-Tuned Architecture," in *IEEE J. Solid-State Circuits*, vol. 44, no. 8, pp. 2169–2181, Aug. 2009.
- [66] E. Alon *et al.*, "Replica compensated linear regulators for supply-regulated phase-locked loops," in *IEEE J. Solid-State Circuits*, vol. 41, no. 2, pp. 413-424, Feb. 2006.

- [67] C.-H. Lee, K. McClellan, and J. Choma, "A Supply-Noise-Insensitive CMOS PLL With a Voltage Regulator Using DC–DC Capacitive Converter," in *IEEE J. Solid-State Circuits*, vol. 36, no. 10, pp. 1453-1463, Oct. 2001.
- [68] J. Liu et al., "A 0.012mm² 3.1mW Bang-Bang Digital Fractional-N PLL with a Power-Supply-Noise Cancellation Technique and a Walking-One-Phase-Selection Fractional Frequency Divider," in Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2014, pp. 268–269.
- [69] A. Elshazly *et al.*, "A 0.4-to-3 GHz Digital PLL With PVT Insensitive Supply Noise Cancellation Using Deterministic Background Calibration," in *IEEE J. Solid-State Circuits*, vol. 46, no. 12, pp. 2759-2771, Dec. 2011.
- [70] C.-W. Yeh, C.-E. Hsieh, and S.-I. Liu, "A 3.2GHz Digital Phase-Locked Loop with Background Supply-Noise Cancellation," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2016, pp. 332–333.
- [71] Y.-C. Huang, C.-F. Liang, H.-S. Huang, and P.-Y. Wang, "A 2.4GHz ADPLL with Digital-Regulated Supply-Noise-Insensitive and Temperature-Self-Compensated Ring DCO," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 270–271.
- [72] H. Kim, W. Jung, K. Kim, S. Kim, W.-S. Choi, and D.-K. Jeong, "A Low-Jitter 8-GHz RO-Based ADPLL With PVT-Robust Replica-Based Analog Closed Loop for Supply Noise Compensation," in *IEEE J. Solid-State Circuits*, vol. 57, no. 6, pp. 1712-1722, June 2022.
- [73] D.-H. Oh *et al.*, "A 2.8Gb/s All-Digital CDR with a 10b Monotonic DCO," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2007, pp. 222–223.

- [74] N. Xing, W.-Y. Shin, D.-K. Jeong, and S. Kim, "High-Resolution Time-to-Digital Converter Utilising Fractional Difference Conversion Scheme," in *IET Electron. Lett.*, vol. 46, no. 6, pp. 398-400, Mar. 2010.
- [75] A. Rylyakov *et al.*, "Bang-bang digital PLLs at 11 and 20GHz with sub-200fs integrated jitter for high-speed serial communication applications," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2009, pp. 94–95.
- [76] D. Kim and S. Cho, "A Supply Noise Insensitive PLL with a Rail-to-Rail Swing Ring Oscillator and a Wideband Noise Suppression Loop," in *Proc. IEEE Int. Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2017, pp. C180–C181.

초록

본 논문은 현대 시리얼 링크의 클락킹에 관여되는 주요한 문제들에 대하여 기술 한다. 준속도, 다중 표준 구조들이 채택되고 있는 추세에 따라, 기존의 클라킹 방법 은 낮은 비용의 구현의 관점에서 새로운 혁신을 필요로 한다. LC 공진기를 대신하여 능동 소자 발진기를 사용한 주파수 합성에 대하여 알아보고, 이에 발생하는 두가지 주요 문제점과 각각에 대한 해결 방안을 탐색한다. 각 제안 방법을 프로토타입 칩을 통해 그 효용성을 검증하고, 이어서 능동 소자 발진기가 미래의 고속 시리얼 링크의 클락킹에 사용될 가능성에 대해 검토한다.

첫번째 시연으로써, 고주파 고리 발진기의 높은 플리커 잡음을 완화시키기 위해 기준 신호를 배수화하여 뒷단의 위상 고정 루프의 대역폭을 효과적으로 극대화 시 키는 회로 기술을 제안한다. 본 기술은 지터를 누적 시키지 않으며 따라서 깨끗한 중간 주파수 클락을 생성시켜 위상 고정 루프와 함께 높은 성능의 고주파 클락을 합성한다. 기준 신호를 성공적으로 배수화하기 위한 타이밍 조건들을 먼저 분석하 여 타이밍 오류를 제거하기 위한 방법론을 파악한다. 각 교정 중량은 연역적 확률을 기반으로한 LMS 알고리즘을 통해 갱신되도록 설계된다. 교정에 필요한 시간을 최 소화 하기 위하여, 각 교정 이득은 타이밍 오류 근원들의 크기를 귀납적으로 추론한 값을 바탕으로 지속적으로 제어된다. 40-nm CMOS 공정으로 구현된 프로토타입 칩 의 측정을 통해 저소음, 고주파 클락을 빠른 교정 시간안에 합성해 냄을 확인하였다. 이는 177/223 fs의 rms 지터를 가지는 8/16 GHz의 클락을 출력한다. 두번째 시연으로써, 고리 발진기의 높은 전원 노이즈 의존성을 완화시키는 기술 이 포함된 주파수 합성기가 설계되었다. 이는 고리 발진기의 전압 헤드룸을 보존함 으로서 고주파 발진을 가능하게 한다. 나아가, 전원 노이즈 감소 성능은 공정, 전압, 온도 변동에 대하여 민감하지 않으며, 따라서 추가적인 교정 회로를 필요로 하지 않는다. 마지막으로, 위상 노이즈에 대한 포괄적 분석과 회로 최적화를 통하여 주 파수 합성기의 저잡음 출력을 방해하지 않는 방법을 고안하였다. 해당 프로토타입 칩은 40-nm CMOS 공정으로 구현되었으며, 전원 노이즈가 인가되지 않은 상태에서 289 fs의 rms 지터를 가지는 8 GHz의 클락을 출력한다. 또한, 20 mVrms의 전원 노이 즈가 인가되었을 때에 유도되는 지터의 양을 -23.8 dB 만큼 줄이는 것을 확인하였다.

주요어: 주파수 합성기, 위상 잡음, 지터, 올디지탈 위상 고정 루프, 고리 발진기, 다위상 클락, 디지털 변조 저항, 기준 클락 배수화, 전원 잡음 **학번**: 2017-28301