

Circuits and Techniques for All-Digital Frequency Synthesizers and Design Automation

by

Kyumin Kwon

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in the University of Michigan
2023

Doctoral Committee:

Professor David Wentzloff, Chair
Associate Professor Ronald Dreslinski
Professor Michael Flynn
Associate Professor Hun-Seok Kim

Kyumin Kwon

kmkwon@umich.edu

ORCID iD: 0000-0001-9880-0775

© Kyumin Kwon 2023

Table of Contents

List of Figures	iv
List of Tables	viii
Abstract	ix
Chapter 1 Introduction	1
1.1 Increasing Demand for Analog Design Automation in sub-20nm Technology	1
1.2 Next Step for Synthesizable Phase Locked Loops	3
1.3 Need for Low-Jitter Frequency Synthesizers	5
1.4 Thesis Contributions	6
Chapter 2 Design Automation of Synthesizable PLL and a Calibration-free Feedforward	
Technique	7
2.1 Introduction	7
2.2 Cell-based Architecture	7
2.3 Overall Design Automation Flow	9
2.3.1 Modeling DCO and PLL	10
2.3.2 Design Solution Searcher	13
2.3.3 Back-End Flow	13
2.4 Design Examples and Measurement Results	14
2.5 Calibration-free Feedforward Noise Cancellation	15
2.5.1 Edge Selecting Feedforward Scheme	19
2.5.2 Linearized Noise Analysis	20
2.6 Synthesizable Feedforward PLL Measurement Results	27
2.7 Conclusion	28
Chapter 3 PLL Fractional Spur's Impact on FSK Spectrum and a Synthesizable ADPLL for	
a Bluetooth Transmitter	29

3.1 Introduction	29
3.2 Prediction of BLE-TX Spurious Tones from PLL Fractional Spurs	32
3.2.1 Fractional Spur Positions	32
3.2.2 Semi-analytical Model of BLE Fractional Spurs	35
3.2.3 Experimental Results	39
3.3 Proposed All Digital BLE Transmitter	41
3.3.1 Proposed Two-step TDC	44
3.3.2 Proposed Calibration Scheme	46
3.3.3 Circuit Implementation	49
3.3.4 Switched Capacitor Power Amplifier	51
3.3.5 Design Automation and Open Source	52
3.4 Measurement Results	53
3.5 Conclusion	61
Chapter 4 Sub-400fs Low-Jitter Ring Oscillator Based Fractional-N MDLL with Reference Triggered Ring Oscillator	62
4.1 Introduction	62
4.2 Proposed Fractional-N MDLL	64
4.3 Circuit Implementation	67
4.4 Result and Discussion	69
4.5 Conclusion	74
Chapter 5 Conclusion	75
5.1 Summary of Contributions	75
5.2 Future Work	76
Appendix	78
BIBLIOGRAPHY	83

List of Figures

Figure 1.1	Trend of DRC rules and operations [1]	2
Figure 1.2	FoM and frequency multiplication ratio of published synthesizable PLLs.....	4
Figure 1.3	Jitter vs EVM plot for 5G communication system in FR1 band (<7GHz).....	5
Figure 2.1	Fully-synthesizable ADPLL architecture and the modeling approach for Digital and Analog portion.	8
Figure 2.2	Cell-based DCO architecture with its design variables and the 2 auxiliary cells.....	9
Figure 2.3	Overall automation flow.....	10
Figure 2.4	Design solution searching process with different modeling method for DCO-level and PLL-level. NTF1, 2, 3 are noise transfer functions from each noise source to the phase of the output clock. Q-noise indicates quantization noise.....	14
Figure 2.5	Example layouts of PLLs in GF 12nm (left) and TSMC 65nm (right).	15
Figure 2.6	Performance comparison of 8 Generated PLL designs for different input specifications.	16
Figure 2.7	Die photo of the fabricated PLL as part of SoC in TSMC 65nm.....	16
Figure 2.8	Phase noise comparison between measurement, model and simulation results at 840MHz.	17
Figure 2.9	Block diagram of previously published feedforward techniques.	17
Figure 2.10	Block diagram of the proposed synthesizable PLL with edge selection FNC technique.	18

Figure 2.11	Timing diagram of the edge selection process	18
Figure 2.12	Linearized phase domain model for the proposed PLL.....	20
Figure 2.13	Performance comparison of MP and FNCP in two different environments. PN plots when (a) DCO noise dominates, (b) TDC noise dominates. Time domain simulation plot when (c) DCO noise dominates, (d) TDC noise dominates	25
Figure 2.14	Contour plots of (a) jitter improvement by FNC with MP in-band PN level due to DCO on x-axis and TDC on y-axis, (b) required maximum delay per DCO stage for FNC to be 1.5x beneficial.....	26
Figure 2.15	Measured phase noise (meas.) in comparison with the analytical model (model) when (a) DCO noise dominates and (b) TDC noise dominates.	27
Figure 3.1	Phase Domain ADPLL block diagram with loop INL with respect to t_{err}	33
Figure 3.2	Conceptual illustration of (a) $t_{err}[N_{ref}]$, (b) PSD of CT PNN, (c) time domain of DT PNN, (d) PSD of DT baseband PNN.....	34
Figure 3.3	Conceptual illustration of the impact of modulation on fractional spurs of equation (8).	38
Figure 3.4	Simulated PLL and BLE spectrum and predicted spurs' positions and amplitudes when (a) $f_{CH} = 2.404\text{GHz}$, (b) $f_{CH} = 2.426\text{GHz}$	40
Figure 3.5	Simulated fractional spur attenuation for different N_s	40
Figure 3.6	Spectral mask for PLL spurious tones at frequencies $f_{frac,N}$, $N=1, 2, 3, 4, 5$ to satisfy BLE spectral mask	41
Figure 3.7	Block diagram of the proposed all-digital BLE-TX.....	42
Figure 3.8	Post-parasitic INL of 5 stage EMBTDC (left) and 45 stage DLTDC (right).	42

Figure 3.9 Behavioral simulation results of PLL (left) and BLE (right) output using (a) DLTDC INL and (b) TSTDC INL with calibration at $f_{CH}=2.402\text{GHz}$.	43
Figure 3.10 (a) block diagram, (b) signal waveforms of TSTDC	45
Figure 3.11 Example of calibrating for delay between DCO_PH[1] and DCO_PH[2]. (a) Locking to DCO_PH[2] by adding EMBTDC_OFFSET, (b) illustration of 2 cases that can happen due to random jitter	47
Figure 3.12 Conceptual illustration of (a) process of generating MAP_LUT and FINAL_LUT and an example operation when $\text{crs_idx}=2$ and $\text{fine_idx}=3$, resulting TSTDC transfer function (b) without calibration assuming equal delay between DCO_PHS, (c) with proposed calibration	48
Figure 3.13 VHDL simulation result of TSTDC INL with and without proposed calibration for 3 different corners.	49
Figure 3.14 Placement pattern for (a) DCO + EMBTDC and (b) DLTDC	50
Figure 3.15 Block diagram of the SCPA	52
Figure 3.16 Automated design flow for the PLL	52
Figure 3.17 Automated design flow for the PLL	53
Figure 3.18 Measurement results of SCPA. (a) Pout , (b) PAE with respect to CTRL and VDD.	54
Figure 3.19 Measurement results of standalone PLL Frequency vs time plot during calibration phase	54
Figure 3.20 Measurement results of standalone PLL. (a) phase noise plot and (b) spectrum for 2 different modes with $\text{FCW} = 60.0156$, $\text{FREF}=40 \text{ MHz}$, $\text{FOUT}=2.4006 \text{ GHz}$	55

Figure 3.21 (a) Measured worst-case fractional spurs and RMS jitters for different FCW_FRACs, (b) worst spurs for FOUT=2.40078 GHz for different VDD levels. (c) Simulated fractional spur levels depending on temperature for FOUT=2.40078 GHz.....	56
Figure 3.22 Measured BLE performance. (a) frequency vs time plot with eye diagram, (b) GFSK spectrum comparison between different modes.....	57
Figure 3.23 Measured PLL spectrum (left) and BLE spectrum (right) for fCH=2.452 GHz along with spur prediction from Section II.....	58
Figure 3.24 Measured BLE (a) spectrum for 3 different channels, (b) worst case spur margin to the spectral mask across BLE channels w/ and w/o calibration	58
Figure 3.25 FoM _N and Area comparison	60
Figure 4.1 Proposed 2-step DTC using RTRO.....	63
Figure 4.2 Block diagram of the proposed fractional-N MDLL.....	64
Figure 4.3 Timing diagram of major signals for edge replacement, FLL1/2, DLL1/2.....	65
Figure 4.4 Block diagram of (a) main MDLL and (b) RTRO.	67
Figure 4.5 Block diagram of RTRO edge selection block.....	68
Figure 4.6 Layout (left) and simulated power breakdown (right) of the proposed design.	69
Figure 4.7 Simulated INL of the fine DTC.....	70
Figure 4.8 Simulated phase noise performance of fractional-N operation (blue) and integer-N operation (red) and free-running mode (gray).	71
Figure 4.9 Output spectrum of MDLL when FCW is 60.....	71
Figure 4.10 Output spectrum of MDLL when FCW is $60+1/2^{10}$	72
Figure 6.1 Average error between (19) and (20) with respect to f_{spur} , $33\beta 2\pi$ ratio for different a_3 values.....	82

List of Tables

Table 2.1	DCO model accuracy for 125 designs compared to simulation results.....	13
Table 2.2	Spec comparison between simulation and measurement.....	16
Table 3.1	Performance summary and comparison with state-of-the-art fractional-N ADPLLs.	59
Table 4.1	Block specifications from transistor-level post parasitic simulation.....	70
Table 4.2	Comparison with recent frequency synthesizers with DTC range (DR) reduction technique.....	73

Abstract

As semiconductor fabrication process become complex to achieve target yield and performance in sub-20nm field-effect transistors (FETs), not only the number of design rule constraints (DRCs) exploded, but also the dependencies between different rules increased, which made manual layout design of custom chip more challenging and time consuming. While digital circuit design has been highly automated thanks to its better immunity to layout parasitic and mismatches, analog circuit design automation lags due to its layout sensitivity. As a solution for analog design automation, using standard digital flow to build a cell-based architecture has been suggested as one of few possibilities. This approach simplifies the design parameters to number of series/parallel cells from conventional width/lengths of every transistor, simplifying the modeling process. Also, it takes advantage of existing layout design engines that can handle complex DRCs, reducing one big step of design automation.

Among analog circuits, clock generators have been extensively explored in the area of cell-based architecture due to the digital nature of clock signal and early development of all-digital architectures. But prior arts showed limits in two areas: 1) fully automating the design process starting from a user given specification, 2) systematic solution to alleviate the degradation of analog performance due to the automatic routing.

In this dissertation, we propose an automated design flow for all digital phase locked loops (ADPLL) and architectural improvements including digital calibration scheme to push the performance limits of synthesizable clock generators. In chapter 2, design automation flow for

baseline ADPLL architecture and novel feedforward scheme that doesn't require gain calibration is proposed. By combining physics-based equation and simulation results, we show a sample efficient (3 sets of simulation required) modeling method that successfully predicts key metrics of digitally controlled ring oscillator (DCO) with error rate less than 1.5%. A prototype design was fabricated in 65nm process using the automation flow. We also propose a feedforward technique that selects the closest edge to the reference clock among interpolated DCO edges. This technique is amenable to PnR tool and reduces the jitter by 4.22x when the DCO noise dominates the TDC quantization noise.

Chapter 3 analyzes PLL fractional spur's impact on Bluetooth low energy (BLE) spectrum to define spectral mask for PLL that can satisfy that of the BLE. Also, we propose a novel two-step TDC architecture and calibration scheme to overcome the performance limits coming from random routings. The 1.8-2.7 GHz PLL was fabricated in 12nm FinFET technology, consuming 3.91 mW at 2.4006GHz achieving FoM of -220.7dB in fractional-N operation.

Finally, chapter 4 proposes an all-digital fractional-N multiplying delay locked loop (MDLL) that uses reference triggered ring oscillator (RTRO) as a coarse DTC that reduces fine DTC range by 9x. Prototype design was fabricated in 65nm CMOS process and measured integer-N result shows 325 fs jitter and 16.1mW power consumption, achieving FoM of -237.7 dB. Simulated fractional-N operation shows worst case fractional spur of -41.9 dBc and rms jitter of 507 fs.

Chapter 1 Introduction

1.1 Increasing Demand for Analog Design Automation in sub-20nm Technology

With ever-increasing demand for small and fast computers for deep learning and Internet of Things (IoT), chip fabrication industry brought commercialized transistor length down to 3nm. While this significantly improved computational power, it posed new challenges to circuit designers; the exponentially increased DRC rules complicates the manual layout design. To ensure manufacturability in such small scale in the presence of variation, edge placement error and variety of other issues, the DRC rules have grown drastically as shown in Figure 1.1 [1]. Dependencies between different rules complicates the design process even more. Some examples of restriction in finFET technologies are uniform width and pitch to reduce non-ideal lithographic effect, unidirectional orientation, double coloring, and gridded design [2]. These restrictions require designers to use patterned and segmented layout styles rather than conventional “analog style” layout, reducing the merit of custom layout compared to auto-PnR layout. Also, the complex DRC significantly increases the manual design time.

Solutions for analog circuit layout automation that are suggested to reduce the time-to-market of chip design industry can be categorized into two streams: 1) automation from the substrate level [3], [4] and 2) cell-based approach utilizing existing digital PnR tool [5]–[10]. The first method develops a software that automatically generate a DRC clean Graphic Design System

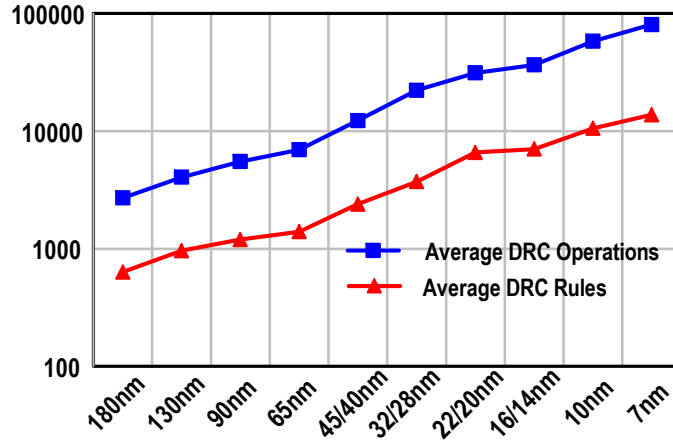


Figure 1.1 Trend of DRC rules and operations [1]

(GDS) for every layer in the PDK with given design rules and user directives. For advanced technology, a grid-based routing has been adapted to handle complex DRC rules with better portability. It successfully mimics the shape of conventional analog layout and achieves similar mismatch and parasitic performance with a manually designed one. But some manual effort of modifying the process-specific primitives when porting to a new process is required to implement a DRC/LVS clean layout. Also, generating a design for a given user specification requires a transistor-level solution searcher, which complexity increases exponentially with the number of transistors. While many research use machine-learning (ML) based model for schematic optimization [3], [4], the amount of training data required for modeling a large-scale analog circuit limits its practical usage due to the enormous number of simulations required. The second approach, on the other hand, uses cell-based architectures which design parameters are simple integer values (number of series/parallel cells) and therefore, simplifies the modeling process with or without ML model. Also, by taking advantage of the built-in PnR engine of digital layout tool, the cell-based approach has minimum porting cost if the unit cells are prepared. But the resulting

analog performance suffers from increased mismatches and parasitic due to random placements and routings.

In Chapter 2, design automation flow for synthesizable ADPLL is presented which requires only 3 sets of simulation results to make a model and takes <1.2 hours to generate a GDS from a given input specification. Also, a novel feedforward scheme that doesn't require gain calibration is proposed in the synthesizable PLL and its effect and limitations are analyzed. The analysis is compared to the measured results from a chip fabricated in TSMC 65nm.

1.2 Next Step for Synthesizable Phase Locked Loops

The PLL is one of the most popular building blocks being explored with the cell-based methodology for its early development of all-digital architectures and the digital nature of a clock signal [11]–[27]. The first generation of synthesizable PLLs focused on the cell-based implementations for integer-N operation. [11] proposed a tri-state inverter based DCO, [12] added a pulse-width modulation on the enable signal of the DCO cell to achieve higher frequency resolution. But the jitter power trade-off was inferior compared to manually designed integer-N PLLs, resulting in FoMs worse than -219 dB.

The second phase of the research explored fractional-N operation with aggressive noise suppression through edge replacement architectures such as injection-locked PLLs (ILPLL) [13], [17], [23], [26] and MDLLs [16], [19], [20]. While the best FoMs achieved are -247.2 dB for integer-N [18] and -234.4 dB for fractional-N [23], designs with FoM < -227dB have frequency multiplication ratios under 25 as shown in Figure 1.2. This limits the practical usage for higher frequency applications such as multi-gigahertz wireless communication circuits without using expensive high frequency crystal oscillators.

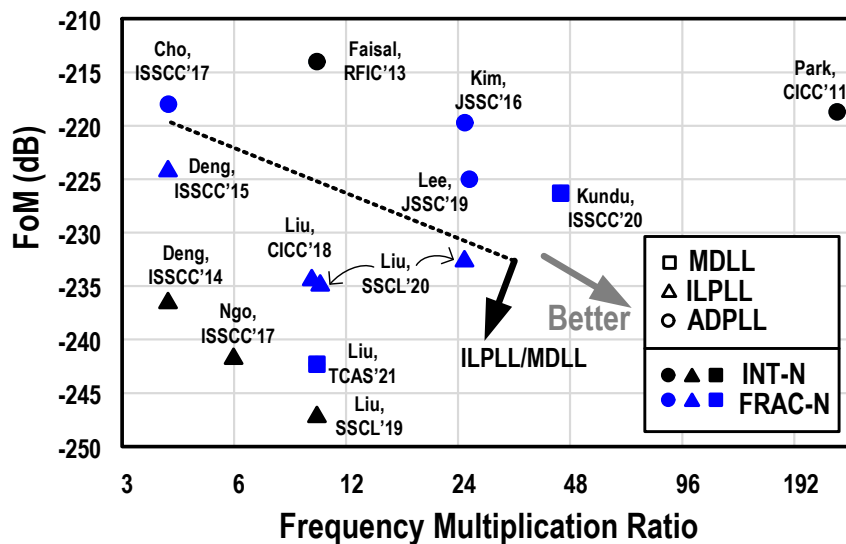


Figure 1.2 FoM and frequency multiplication ratio of published synthesizable PLLs

A PLL for a multi-gigahertz wireless transmitter is a good next target for cell-based PLL designs. For example, the BLE specifications require the 2.4GHz GFSK signal to have an eye-opening greater than 370KHz for 1Mbps data rate and spurious emission to be under the spectral mask. Random routing from a P&R tool increases delay mismatches in multi-stage timing control blocks such as the DCO, TDC and DTC, resulting in a high fractional spur due to the nonlinearity. The high bandwidth required for the data rate and RO noise suppression combined with this nonlinearity poses a challenge for synthesizable PLLs to meet the spectral mask of the BLE specification. A systematic solution for the problem is required to maximize the strength of short design and porting time of synthesizable circuits.

In Chapter 3, two-step TDC architecture is proposed as part of synthesizable ADPLL for BLE-TX in order to overcome the resolution limit of the baseline TDC. Also, a calibration scheme that alleviates coarse TDC nonlinearity caused by random routing is presented. The design was fabricated in GLOBAL FOUNDRY 12nm FinFET process and the 1.8-2.7 GHz PLL consumes

3.91 mW at 2.4006GHz achieving FoM of -220.7dB in fractional-N operation. Measured performance satisfies BLE standard requirements thanks to the proposed techniques.

1.3 Need for Low-Jitter Frequency Synthesizers

While ring oscillator (RO) has shown its potential of replacing some applications such as BLE that inductor-capacitance (LC) oscillator was conventionally used in, high data rate wireless communication systems still require LC oscillators to satisfy the stringent jitter requirements. For example, 5G communication system requires integrated phase noise (IPN), which is equivalent to error vector magnitude (EVM) in 4^M QAM scheme [28], less than -35 dBc, which translates to RMS jitter of 404 fs for 7 GHz. Also, fractional-N synthesizers are preferred for its ability to generate fine-spaced carriers, which varies between 15 kHz – 450 kHz for 5G, with a fixed reference clock, which typical range is 10-100 MHz. Therefore, pushing the limit of jitter

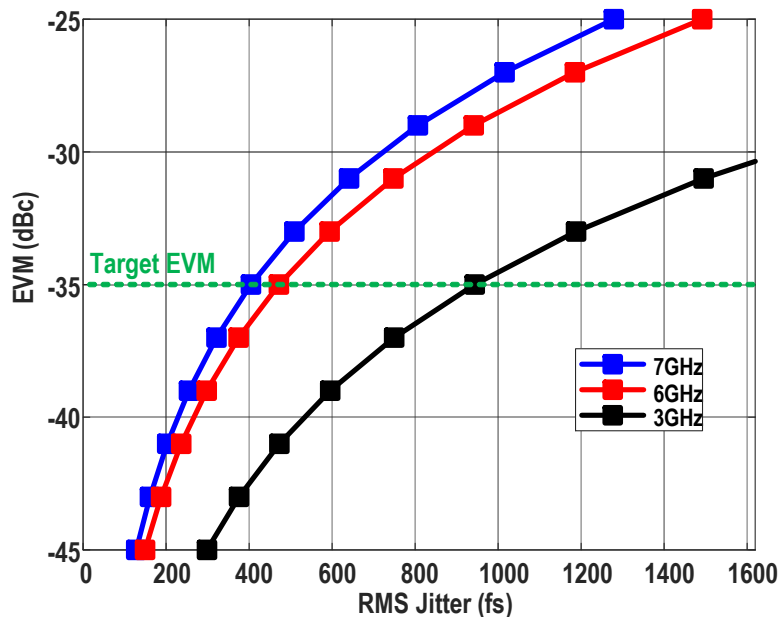


Figure 1.3 Jitter vs EVM plot for 5G communication system in FR1 band (<7GHz)

performance of RO-based fractional-N frequency synthesizers is the key next step to broaden its application.

Edge replacement circuits such as IL-PLL and MDLL are inherently excellent at suppressing DCO phase noise thanks to the large effective bandwidth achieved by the noise-resetting action every reference cycle [29]. In Chapter 4, a fractional-N MDLL that uses RTRO as a coarse DTC has been proposed. Fabricated in 65nm CMOS technology, it achieves RMS jitter of 325 fs for integer-N operation at 3.25 GHz. Simulated fractional-N operation shows RMS jitter of 353 fs with worst case fractional spur of -53 dBc.

1.4 Thesis Contributions

In summary, this dissertation proposes a design automation flow for PLL that is highly portable to reduce the time-to-market of IC industry amidst an exponentially increasing design complexity. Also, we present digital frequency synthesizer architectures that alleviates the analog issues caused by the automated routing, or pushes the performance limit of RO-based synthesizer. The automation flow only requires 3 sets of simulation results to generate a highly accurate model of error rate less than 1.5% that predicts the PLL performance and 8 PLLs are generated and the performances are compared to the input specifications to demonstrate the flow. Next, a fully-synthesized PLL in a BLE-TX with novel two-step TDC architecture and calibration scheme to reduce the PnR induced nonlinearity is proposed and analyzed. The measurement results satisfy the BLE requirements thanks to the proposed techniques. Finally, an all-digital fractional-N MDLL that uses RTRO as a coarse DTC to reduce the fine DTC range by 1/9 is proposed. Measured integrated jitter of integer-N operation is 325 fs, achieving FoM of -237.3 dB. Simulated fractional-N operation shows jitter of 507 fs, achieving FoM of -233.5 dB.

Chapter 2 Design Automation of Synthesizable PLL and a Calibration-free Feedforward Technique

2.1 Introduction

In this chapter, we present a synthesizable ADPLL generator that uses a simple analytical model for DCO characterization which only requires 3 sets of simulation results. Based on the DCO performance, PLL specifications are predicted by frequency-domain model using the known transfer functions of D-flip flop (DFF) based time to digital converter (TDC) and digital loop filter (DLF). The combination of human knowledge based model and existing digital synthesis tool incredibly fastens the process of characterization and design.

To prove the concept, 8 PLLs are generated from different input specifications in 65nm CMOS technology, and their performances are compared with the given requirements. One of the PLLs is fabricated as part of fully-synthesized SoC [5] and the measurement result is compared to the predicted values.

2.2 Cell-based Architecture

We employ an embedded TDC (EMBTDC) [30] based phase-domain ADPLL architecture [31] and the simplified block diagram is shown in Figure 2.1. DCO's phase information is captured in the form of digital words by EMBTDC and digital counter at the rising edge of reference clock,

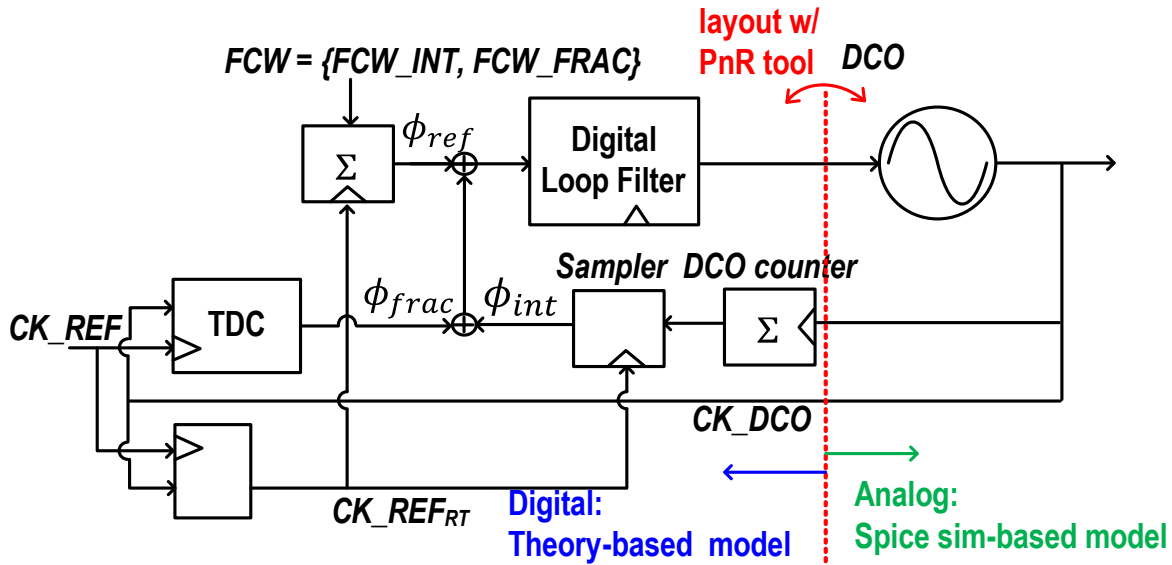


Figure 2.1 Fully-synthesizable ADPLL architecture and the modeling approach for Digital and Analog portion.

each generating fractional and integer phase information. Phase/frequency comparison is done in the digital domain by subtracting the accumulated phase information of DCO from that of the target phase. Target phase information is generated by accumulating the programmable frequency command word (FCW) every reference cycle. The phase error information is then processed through the digital loop filter and controls the DCO frequency to correct the phase/frequency error.

The DCO architecture is shown in Figure 2.2 and is composed of 2 custom-designed auxiliary cells (aux-cells): tri-state differential inverter (coarse controller: CC) and switched mosfet-capacitor (Fine controller: FC), which are digital standard cell grid compliant. As the name indicates, the former coarsely tunes the frequency while the latter tunes finely. The four design variables that determine the DCO performance are: 1. Number of tunable CC (N_{CC}), 2. Number of tunable FC (N_{FC}), 3. Number of always-on CC (N_{DRV}), 4. Number of stages (N_{STG}), as shown in Figure 2.2. Depending on these design variables, the frequency range, nominal frequency, phase

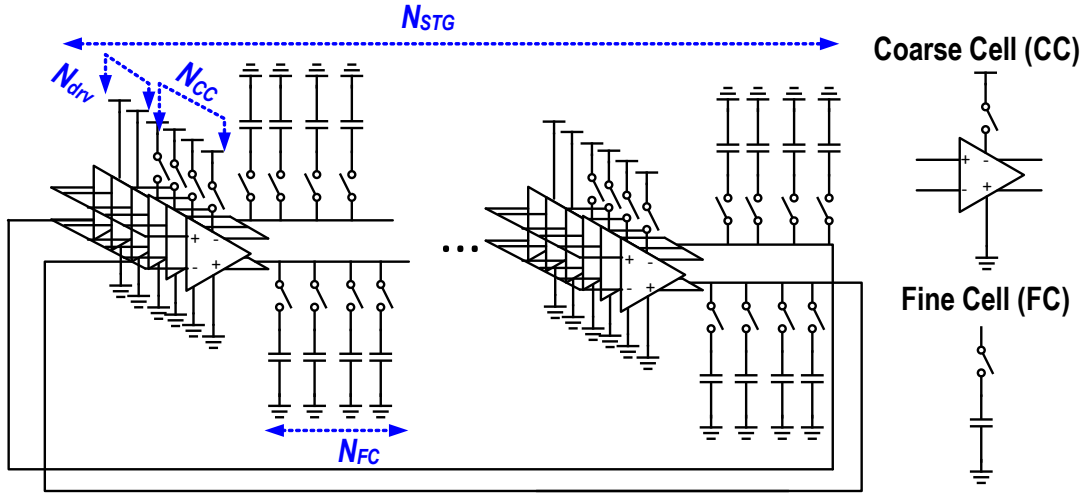


Figure 2.2 Cell-based DCO architecture with its design variables and the 2 auxiliary cells.

noise and power of the DCO varies. Detailed properties of operation will be discussed in the next section. The cell-based DCO can be laid out by a PnR tool, which enables the use of a standard digital flow for the whole PLL.

2.3 Overall Design Automation Flow

The overall automation flow is shown in Figure 2.3. The DCO modeling procedure is run in the absence of a model file. Generated model predicts the performance of the DCO and PLL from design parameters. Using the model, the design decision algorithm searches for the designs that satisfy the user given specifications. Once the design parameters are decided, Verilog source files and scripts for synthesis and PnR are generated. The tool then automatically runs the digital flow all the way through drc/lvs. Currently supported specifications are frequency range, nominal frequency, frequency resolution, DCO power consumption and in-band phase noise level. As an open-source project this list of specifications is expected to grow with future contributions. Details of Modeling, Design solution searcher and implementation are explained in the following sections.

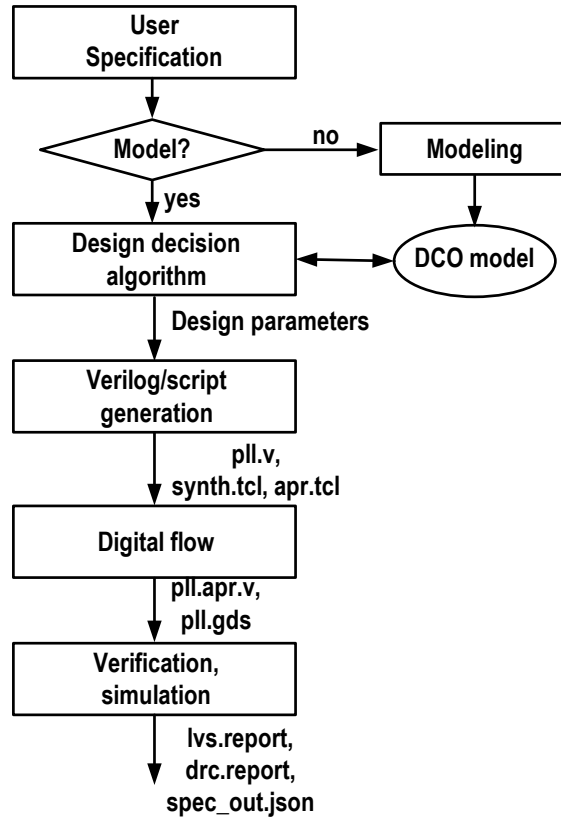


Figure 2.3 Overall automation flow

2.3.1 Modeling DCO and PLL

The objective of the modeling procedure is to generate a model that predicts the PLL performance for a pre-defined design space (currently 2.6M designs). The DCO performance determines not only the frequency range and resolution, but also the TDC quantization level since an EMBTDC is employed. Therefore, the in-band phase noise level of the PLL can also be calculated for a given bandwidth and DCO specifications and the DCO model plays a key role in the overall PLL model. Since the transistor level behavior impacts the DCO specs, the model is built from SPICE simulations. These simulations are automatically run for several different DCO

designs to capture the cell-level characteristics that impact the DCO-level specifications, without needing to know the physics-level parameters.

We use the analytical frequency equation proposed in [32] that uses PDK/aux-cell specific constants that represent the effective current to capacitance ratio for predicting the performance as a function of the aforementioned design parameters. The frequency of the DCO can be expressed as follows.

$$f_{dco} = \frac{N_{CC-on} + N_{DRV}}{((N_{CC} + N_{DRV}) \cdot \alpha + N_{sc} \cdot \beta + N_{sc-on} \cdot \gamma) \cdot N_{stg}} \quad (1)$$

$$\alpha = C_{CC} / I_{CC} \quad (2)$$

$$\beta = C_{FC} / I_{CC} \quad (3)$$

$$\gamma = C_{FC-on} / I_{CC} \quad (4)$$

Where I_{CC} is the driving strength of CC, N_{CC-on} is the number of enabled CCs per stage, N_{DRV} is the number of always-on CC per stage, C_{CC} is the parasitic capacitance per CC, C_{FC-on} and C_{FC} are the capacitance of FC when it is on and off, respectively. N_{CC-on} , N_{FC-on} are tunable values and used to change the frequency of the DCO during PLL operation. To acquire three constants α , β and γ , we need three simulation results with different design variables and tuning words. The modeling procedure automatically runs transient simulation for 3 different designs and extracts the constants by solving the equations. For the phase noise modeling of the DCO, we employ the analytical expression for ring oscillator's phase noise spectrum due to the transistor's white noise presented in [8]. We extract a phase noise constant K_{pn} with a noise simulation of one DCO design, where the phase noise at a certain frequency offset f_{offset} is expressed as:

$$L(f_{offset}) = \frac{K_{pn}}{N_{CC-on} + N_{DRV}} \left(\frac{f_{dco}}{f_{offset}} \right)^2 \quad (5)$$

Phase noise simulation is run by SPECTRE periodic noise simulation to extract the model constant K_{pn} for nominal frequency, which is the frequency when $N_{CC-on} = N_{CC}/2$ and $N_{FC-on} = N_{FC}/2$, with $f_{offset} = 1MHz$. Using previously obtained α, β and γ , we now know f_{dco} for a certain design and tuning word. So, using the $L(f_{offset})$ from only one simulation result, we can calculate the phase noise model constant K_{pn} . To evaluate the accuracy of the model, the flow runs simulations for user-defined range of designs and reports the maximum error rate for all the specs. Table 1 shows the maximum error rate of the model predicted values compared to the simulation results for 125 DCO designs. The strength of the equation-based model is the sample efficiency, which requires only 3 sets of simulation results per PDK. For post layout performance, we assume α, β and γ scale independently but constantly over different designs due to parasitic capacitance. The parasitic uncertainty is regulated by using an auto-generated placement scripts to place the DCO cells in a fixed pattern.

For PLL performance estimation, reference clock frequency and PLL bandwidth are assumed to be 10MHz and 1MHz, respectively. Since the bandwidth is programmable and the reference clock can be changed, the user can configure the dynamics of the output design as needed. The PLL in-band phase noise is estimated by applying noise transfer functions (NTFs) [33] to different noise sources based on model-predicted DCO specifications and adding them as shown in Figure 2.4. Assuming reference noise is negligible, we consider the noise sources: 1. DCO phase noise, 2. TDC quantization noise, which depends on the number of stages of the DCO, 3. Frequency control quantization noise, which depends on the frequency resolution of the DCO.

Model Verification	Specifications				
	Fmax	Fmin	Fres	PN@1MHz	Power
Max Error(%)	0.58	3.77	10.53	6.82	1.33

Table 2.1 DCO model accuracy for 125 designs compared to simulation results

2.3.2 Design Solution Searcher

This part of the tool searches for design solutions that satisfy user given specifications. This is done by sweeping a pre-defined design space with the model file and filtering the designs that satisfy the input specifications. Since the prediction through the model is simple mathematical calculations, it takes less than a minute to search 2.6M designs.

Finding a feasible set of specifications can be challenging from a user perspective. For example, DCO has a tradeoff between frequency, power consumption and phase noise. If the user requires a combination of the three specs that results in a figure of merit (FoM) better than the achievable value, no design will satisfy the given specifications. To give a sense of feasible ranges of specs to the user, the tool prints out the achievable range of specs for the failed categories if it cannot find a design that satisfies all the specs. This process is shown in Figure 2.4, where the tool provides a range of power consumption and Phase noise that are achievable so that the design solution can be met in the next iteration.

2.3.3 Back-End Flow

Once the design parameters are chosen, Verilog sources describing a particular PLL are generated automatically by changing the parameters of baseline Verilog files. The TCL scripts for digital flow are also generated correspondingly. Timing constraints and area of the design changes

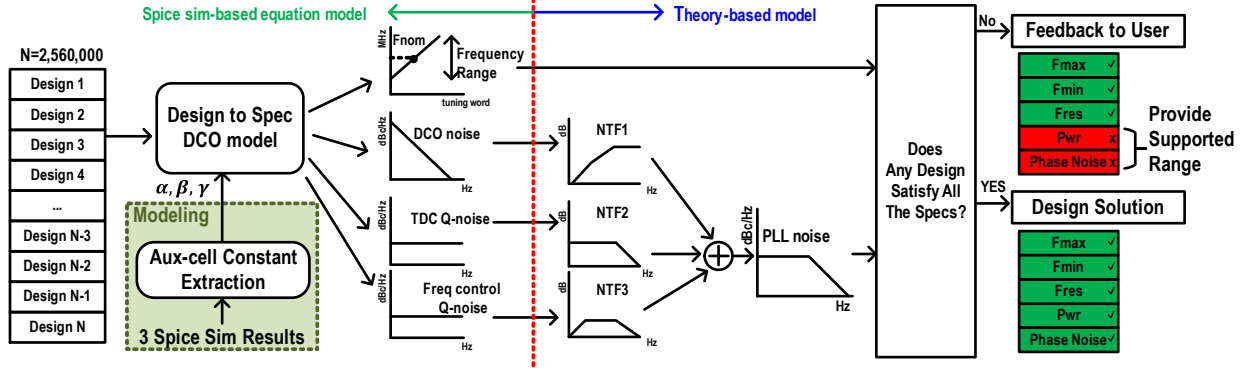


Figure 2.4 Design solution searching process with different modeling method for DCO-level and PLL-level. NTF1, 2, 3 are noise transfer functions from each noise source to the phase of the output clock. Q-noise indicates quantization noise.

according to the nominal frequency of DCO and estimated area from the synthesis. To minimize the impact of layout on oscillator’s performance, DCO is implemented separately with different power domain and used as a hard macro in the PLL layout. The placement of the DCO inside the PLL is described in a factorized number of core-size to ensure portability over designs and PDKs. Using the TCL scripts, digital flow is automatically run followed by a post-parasitic SPICE simulation of DCO to check its analog performance. Resulting DCO specifications are then used by a Matlab behavioral model to verify the PLL performance in a time-domain simulation and the results are written in a *spec_out.json* file together with the input specs, allowing the comparison between the two.

2.4 Design Examples and Measurement Results

The tool currently supports GF 12nm, TSMC 65nm and Intel 22nm technology for PLL generation. Examples of the generated PLL layouts are shown in Figure 2.5. To show design flexibility, we generated 8 PLLs with random input specifications in 65nm process. Frequency

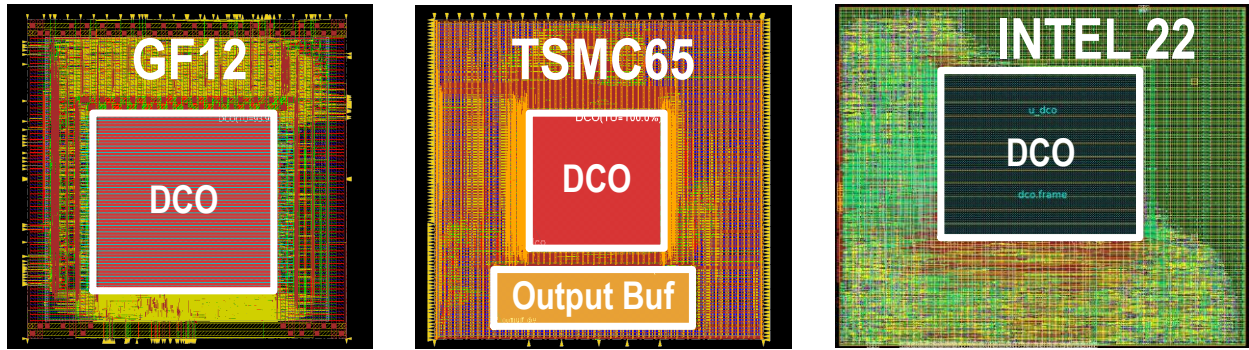


Figure 2.5 Example layouts of PLLs in GF 12nm (left) and TSMC 65nm (right).

range and nominal frequency are verified by post-pex SPICE simulation, while phase noise performance of the PLL is verified by Matlab time domain behavioral model using DCO performance extracted from analog simulation results. Figure 2.6 shows the input and output specifications. We can observe that the output frequency range is wider than the input, nominal frequency of input and output is approximate, while the output phase noise at 1MHz offset is lower than the required phase noise limit. As a prototype design, a PLL was fabricated and measured, which die photo is shown in Figure 2.7. Table 2.2 compares simulation results with the measured performance. Figure 2.8 shows the comparison between measurement, frequency domain model and simulation result of PLL phase noise at 840MHz with 20MHz reference clock and we can observe that the three agree in high accuracy.

2.5 Calibration-free Feedforward Noise Cancellation

One of the techniques to compensate the RO performance is feedforward noise cancellation (FNC). By feeding forward the phase error captured by the phase detector to the output clock, FNC reduces the noise level of the PLL without affecting the stability. Figure 2.9 shows 2 different

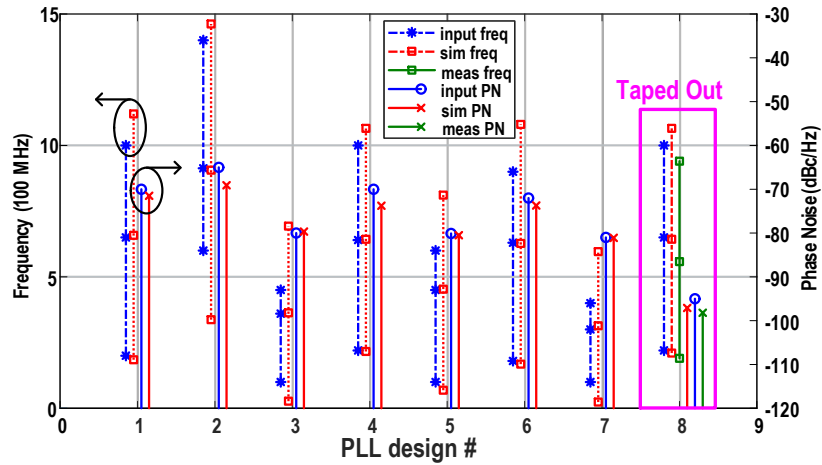


Figure 2.6 Performance comparison of 8 Generated PLL designs for different input specifications.

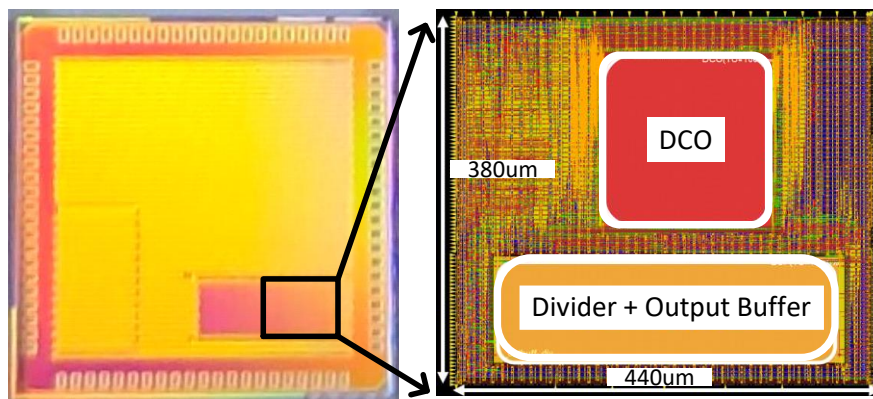


Figure 2.7 Die photo of the fabricated PLL as part of SoC in TSMC 65nm.

	Specifications				
	Fmax (MHz)	Fmin (MHz)	Fnom (MHz)	RMS Jitter @840MHz (ps)	DCO Power (mW)
Sim	1065	209	643	11.1	7.2
Meas.	940	190	558	11.5	6.9
Error (%)	11.73	9.09	13.2	3.7	4.16

Table 2.2 Spec comparison between simulation and measurement

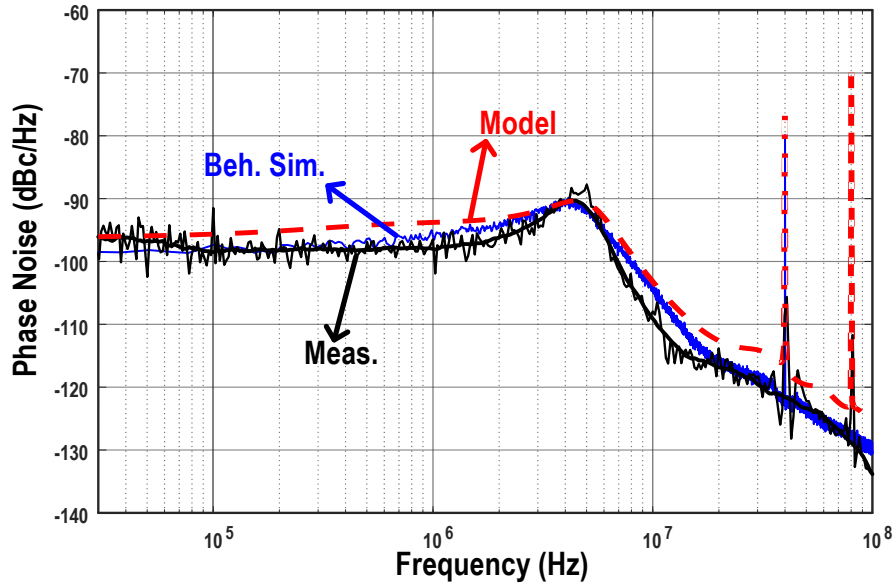


Figure 2.8 Phase noise comparison between measurement, model and simulation results at 840MHz.

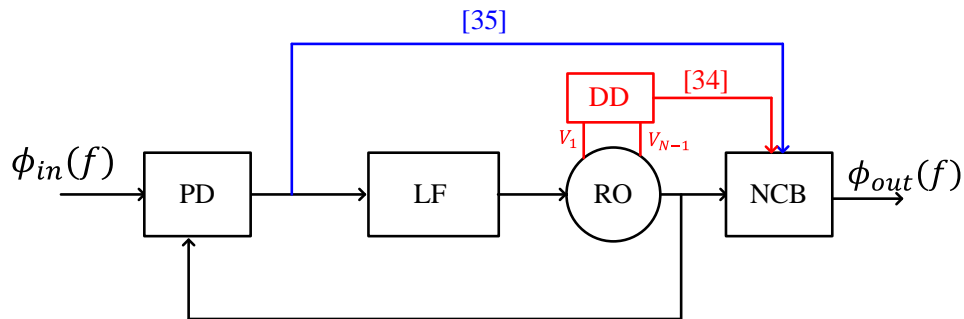


Figure 2.9 Block diagram of previously published feedforward techniques.

architectures of previously published FNC PLLs. [34] uses a delay-line discriminator (DD) embedded in the RO to extract the out-of-band PN and cancels the noise component using a voltage-controlled delay (VCD) element outside of the PLL. [35] uses a sub-sampling phase detector (SSPD) to capture the phase error and uses its output voltage to control the VCD. Both [34] and [35] achieved $>10\text{dBc/Hz}$ PN suppression with FNC. However, since the controllable

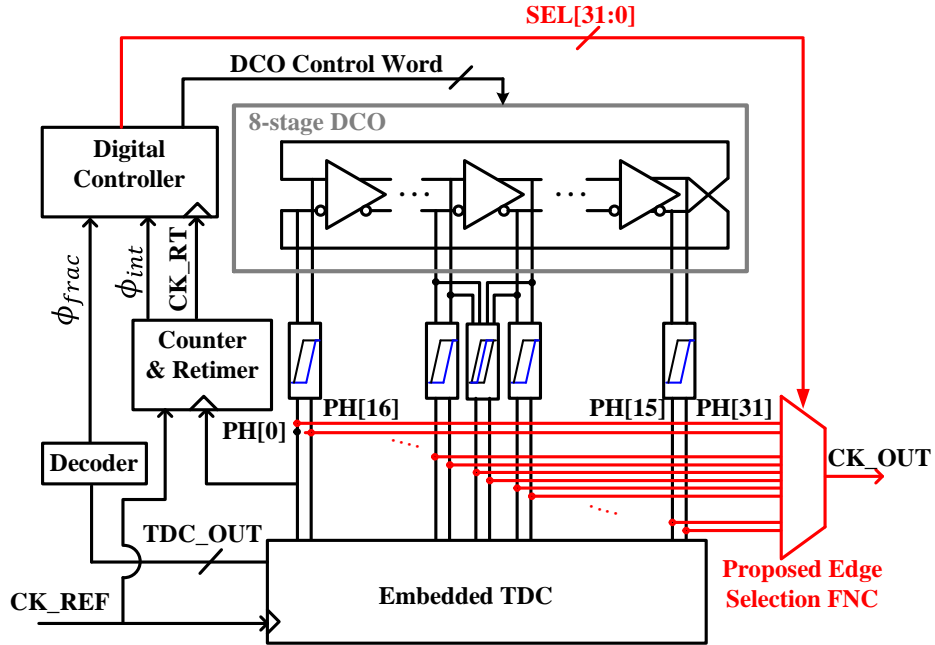


Figure 2.10 Block diagram of the proposed synthesizable PLL with edge selection FNC technique.

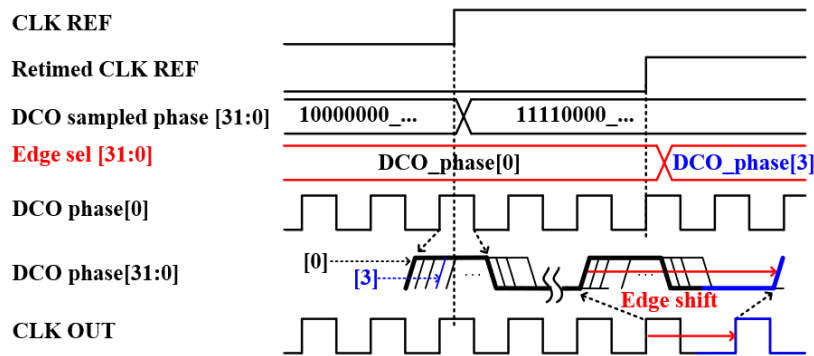


Figure 2.11 Timing diagram of the edge selection process

delay unit that cancels the noise is separate from the noise detection unit, both require gain calibration of the FNC path for accurate cancellation. We propose an FNC method that does not require any calibration, and is amenable to cell-based design and APR.

2.5.1 Edge Selecting Feedforward Scheme

A block diagram of the proposed feedforward ADPLL is shown in Figure 2.10. The baseline architecture is adapted from [31]. The phase information of the DCO is captured by an embedded TDC [30] and digital counter. The reference phase ramp is generated by integrating the frequency command word (FCW). Phase comparison and filtering is done in the digital domain, clocked by the retimed reference clock. The DCO is composed of 8 stages of tri-state differential inverter cells and switched capacitor cells and is controlled by the output of the loop filter. The coarse and fine-tuning words each controls the number of tri-state cells and switched capacitors that are enabled per stage. An embedded TDC latches the node voltages of each stage of the DCO on the rising edge of the reference clock, capturing the fractional phase error. In our design, 8 differential phase interpolators double the 16 phases of the differential DCO. So, the TDC quantizes the fractional phase error into 5-bits.

Figure 2.11 shows a signal diagram of edge selection logics. Initially, CLK_OUT is connected to DCO_phase[0], which was the last edge before CLK_REF on the previous cycle. But on the next rising edge of CLK_REF, due to DCO noise, DCO_phase[3] is the last edge before CLK_REF instead of DCO_phase[0]. This information is captured by embedded TDC, latching DCO_phase[3]=1 and DCO_phase[4]=0. Using this information, the code of the edge selection block gets updated on the retimed CLK_REF, which is approximately 4 DCO cycles after the reference edge. The edge selection block is composed of tri-state buffers from digital standard-cell library, functioning as a clock multiplexer that connects CLK_OUT to the desired DCO edge. Now CLK_OUT is connected to DCO_phase[3], effectively reducing the phase error by $3\Delta T_{tdc}$, where ΔT_{tdc} is the time resolution of the TDC.

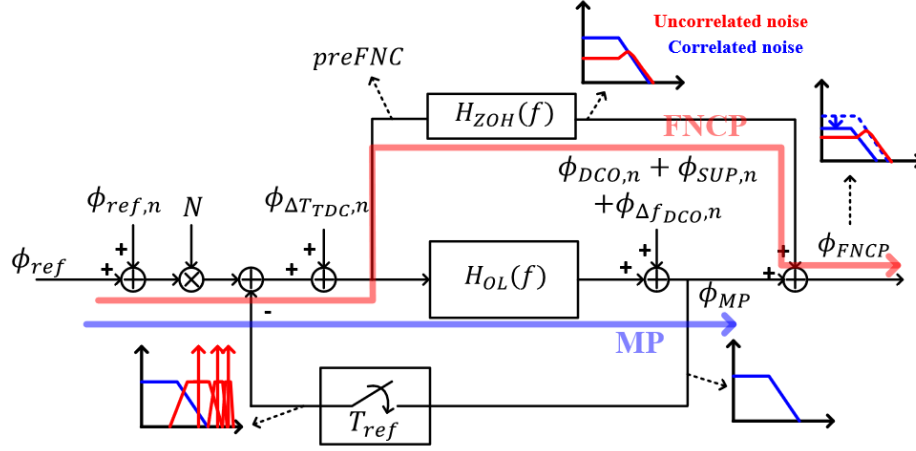


Figure 2.12 Linearized phase domain model for the proposed PLL

One drawback of both the embedded TDC and the proposed FNC scheme is the quantization noise due to this finite time resolution of $\Delta T_{tdc} = T_{dco}/N_{tdc}$ where T_{dco} is the period of the DCO clock, and N_{tdc} is the number of phases that the TDC latches (32 in the proposed design). The TDC resolution adds quantization noise into the loop and the FNC path, degrading the PN performance of the PLL.

2.5.2 Linearized Noise Analysis

Figure 2.12 illustrates a linearized phase domain model of the proposed PLL with different noise sources. The term main path (MP) and FNC path (FNCP) will be used to indicate the normal PLL loop and the feedforward path. $\phi_{pll_{out}}$ is the output phase of MP, which is sampled by the reference clock on every phase comparison event. In the frequency domain, this generates a train of copied spectra with f_{ref} spacings (red graphs). The loop filter in the MP suppresses these copied spectra while the FNCP only has filtering by zero-order hold (ZOH) action, which transfer function is

$$H_{ZOH}(f) = e^{-i\pi f T_{ref}} \text{sinc}(f T_{ref}) \quad (1)$$

Where $T_{ref} = 1/f_{ref}$. Since $H_{ZOH}(s)$ has $1/f^2$ roll off, it suppresses the out-of-band spectra, but not as sharp as the MP does, which transfer function is

$$H_{OL}(f) = \left(K_p + \frac{K_i}{1-z(f)^{-1}} \right) \cdot z(f)^{-1} \cdot H_{ZOH}(f) \cdot \frac{K_{DCO}}{1-z(f)^{-1}} \quad (2)$$

Where $z(f)^{-1} = \exp(-i2\pi f / f_{ref})$. Based on (1) and (2), bellow three sections will analyze the FNC effect on different noise sources.

2.5.2.A DCO noise shaping

In the following analysis, DCO noise implies the sum of 1) DCO random noise, $\phi_{DCO,n}$, 2) DCO phase deviation due to supply noise, $\phi_{SUP,n}$, and 3) DCO phase deviation due to DCO frequency resolution, $\phi_{\Delta f_{DCO,n}}$. $S_{\phi,DCO,n}(f)$ indicates the noise spectrum of DCO noise. The PN spectrum of the PLL output due to DCO noise is then expressed as

$$S_{\phi,MP}(f)|_{DCO,n} = S_{\phi,DCO,n}(f) \cdot \frac{1}{1+H_{OL}(f)} \quad (3)$$

The copied spectra generated from sampling action are not correlated with $\phi_{pll_{out}}(f)$. Thus, the FNCP adds uncorrelated noise shaped by $|H_{ZOH}(f)|^2$ while cancelling the correlated portion shaped by $|1 - H_{ZOH}(f)|^2$ [6]. Sum of the two is expressed as

$$\begin{aligned}
S_{\phi, FNCP}(f)|_{DCO,n} &= |1 - H_{ZOH}(f)|^2 \cdot S_{\phi, MP}(f)|_{DCO,n} \\
&+ |H_{ZOH}(f)|^2 \cdot \sum_{\substack{k=-\infty \\ k \neq 0}}^{\infty} S_{\phi, MP}(f - k \cdot f_{ref})|_{DCO,n}
\end{aligned} \tag{4}$$

Because the high frequency suppression of $|H_{ZOH}(f)|^2$ is less sharper than $|H_{OL}(f)|^2$, FNCP adds out-of-band noise to the MP. Figure 2.13(a) and (b) show a PN break-down of MP and FNCP, showing this DCO noise shaping of both paths. But the amount of noise cancelled by the FNCP is much larger than the added, reducing the overall jitter from DCO and power supply.

2.5.2.B TDC quantization noise shaping

Assuming white noise, the single-sided TDC quantization noise spectrum can be expressed as

$$S_{\phi, TDC,n}(f) = \frac{(2\pi)^2}{12} \left(\frac{\Delta T_{TDC}}{T_{DCO}} \right)^2 \frac{1}{f_{ref}} \tag{5}$$

The PN spectrum of MP due to TDC quantization is then expressed as

$$S_{\phi, MP}(f)|_{TDC,n} = S_{\phi, TDC,n}(f) \cdot \frac{H_{OL}(f)}{1+H_{OL}(f)} \tag{6}$$

The spectrum of TDC noise at node *preFNC* is then

$$S_{\phi, preFNC}(f)|_{TDC,n} = S_{\phi, TDC,n} \cdot \frac{1}{1+H_{OL}(f)} \tag{7}$$

The transfer function of the correlated path of FNCP for $S_{\phi,MP}(f)|_{TDC,n}$ is then

$$\frac{\phi_{FNCP,corr.}}{\phi_{MP}|_{TDC,n}} = \left| 1 + \frac{H_{ZOH}(f)}{H_{OL}(f)} \right|^2 \quad (8)$$

Unlike the transfer function of the DCO's correlated path in (4), (7) adds noise instead of cancelling it. Intuitively, since $S_{\phi,preFNC}(f)|_{TDC,n}$ is a high-passed spectrum of $S_{\phi,TDC,n}(f)$, the in-band correlated noises are not being cancelled as (4). For the uncorrelated noise, the transfer function is same as that of (4). Therefore, the FNCP increases the out-of-band noise as shown in Figure 2.13(a) and (b), while not cancelling the in-band portion. The total PN spectrum of FNCP from the TDC noise is

$$\begin{aligned} S_{\phi,FNCP}(f)|_{TDC,n} &= \left| 1 + \frac{H_{ZOH}(f)}{H_{OL}(f)} \right|^2 \cdot S_{\phi,MP}(f)|_{TDC,n} \\ &+ |H_{ZOH}(f)|^2 \cdot \sum_{\substack{k=-\infty \\ k \neq 0}}^{\infty} S_{\phi,MP}(f - k \cdot f_{ref})|_{TDC,n} \end{aligned} \quad (9)$$

2.5.2.C Total noise

Remaining noise source is the reference noise. Using the same principle as (4), the output PN spectrum of MP and FNCP due to reference noise can be written as

$$S_{\phi,MP}(f)|_{REF,n} = S_{\phi,REF,n}(f) \cdot \left| \frac{N \cdot H_{OL}(f)}{1 + H_{OL}(f)} \right|^2 \quad (10)$$

$$\begin{aligned} S_{\phi,FNCP}(f)|_{REF,n} &= |1 - H_{ZOH}(f)|^2 \cdot S_{\phi,MP}(f)|_{REF,n} \\ &+ |H_{ZOH}(f)|^2 \cdot \sum_{\substack{k=-\infty \\ k \neq 0}}^{\infty} S_{\phi,MP}(f - k \cdot f_{ref})|_{REF,n} \end{aligned} \quad (11)$$

Where N is the frequency command word. The contribution from reference clock in the proposed design has $<-8\text{dBc/Hz}$ contribution on the output noise compared to that of DCO or TDC. Therefore, the foregoing analysis assumes that the DCO random noise and TDC quantization noise are the only noise sources contributing to the output spectrum. The total noise of FNCP is approximately the sum of (4) and (9), which is

$$S_{\phi, FNCP}(f) = S_{\phi, FNCP}(f)|_{TDC, n} + S_{\phi, FNCP}(f)|_{DCO, n} \quad (12)$$

As observed from Sections II.B and II.C, FNCP reduces the overall noise contribution from DCO but increases the contribution from TDC. Therefore, the effect of FNC on the total noise performance depends on the relative levels of the two noises. When DCO noise dominates MP PN as in Figure 2.13(a) and (c), FNCP reduces the overall noise level. This can be understood intuitively in the time domain. As shown in (c), when random jitter of MP output is much larger than the resolution of FNC correction, the abrupt phase corrections hide inside the random jitter, improving the jitter performance. When TDC noise dominates, however, the correction of FNCP outlies the random jitter, adding more noise to the MP as shown in Figure 2.13(d). In the frequency domain, this appears as an increase in the out-of-band noise from TDC, as shown in Figure 2.13(b). Figure 2.14(a) shows a contour plot of the relationship between the jitter ratio of two modes, $\sigma_{rms, MP} / \sigma_{rms, FNCP}$, and the in-band PN level of MP due to DCO and TDC noise. The jitter values are results of behavioral simulation with $f_{ref} = 40\text{MHz}$, $f_{out} = 840\text{MHz}$, $\text{BW} = 2.8\text{MHz}$, sweeping ΔT_{TDC} and DCO noise level (assuming $1/f^2$ degradation). We can observe that the FNC has more effect with larger noise contribution from DCO and less from TDC. The difference should be approximately 10dBc/Hz or greater for FNC to have more than 1.5x of improvement.

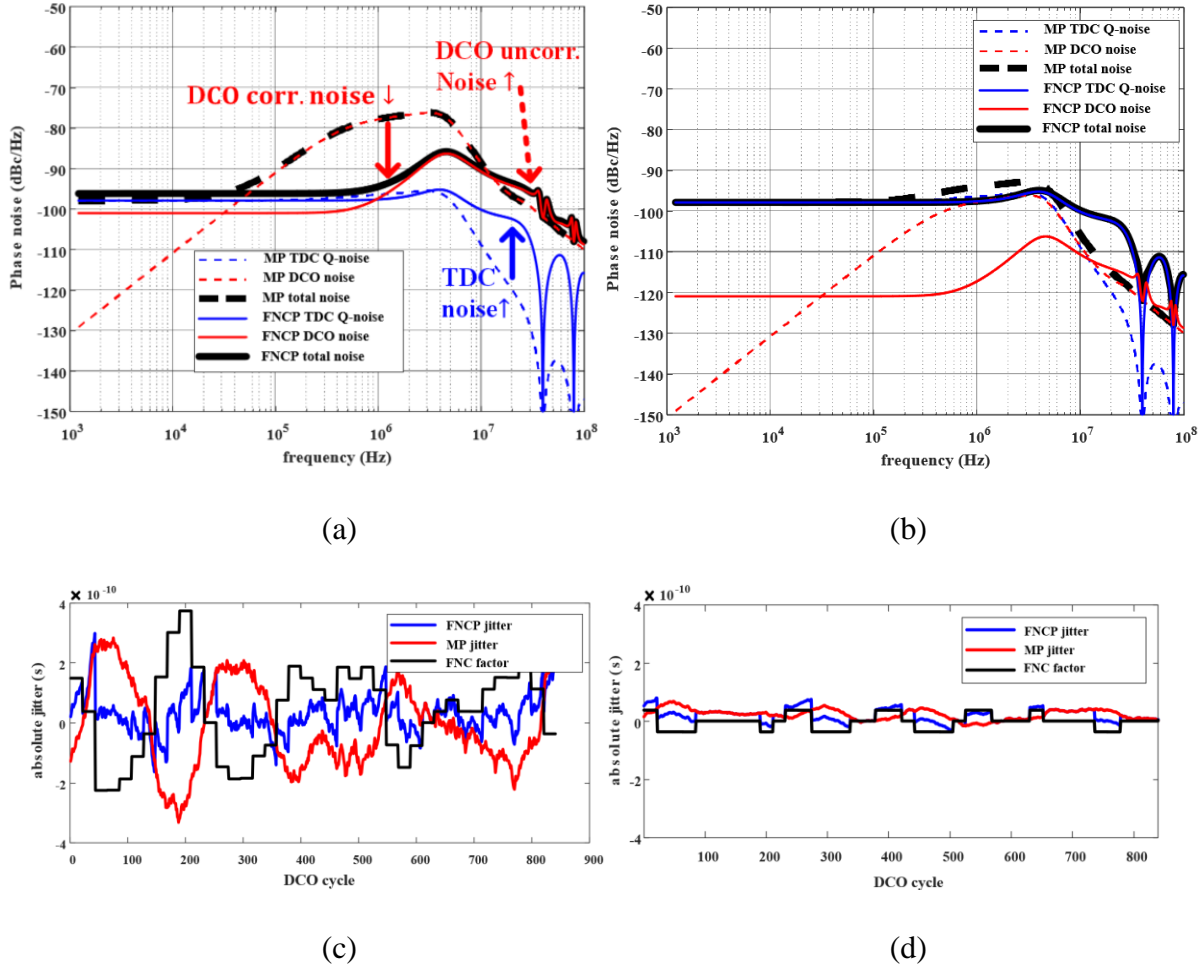


Figure 2.13 Performance comparison of MP and FNCP in two different environments. PN plots when (a) DCO noise dominates, (b) TDC noise dominates. Time domain simulation plot when (c) DCO noise dominates, (d) TDC noise dominates

Since $\Delta T_{TDC} = 1/(f_{DCO} \cdot 4 \cdot N_{stg})$, we can derive a condition of delay per DCO stage (ΔT_{stg}) which FNC is beneficial by 1.5x. Noise performance of the DCO will be characterized by FOM_{PN} , which is

$$FOM_{PN} = 10 \log_{10} \left(S_{\phi,DCO,n}(f_{off}) \cdot \frac{P}{1mW} \cdot \left(\frac{f_{off}}{f_{DCO}} \right)^2 \right) \quad (13)$$

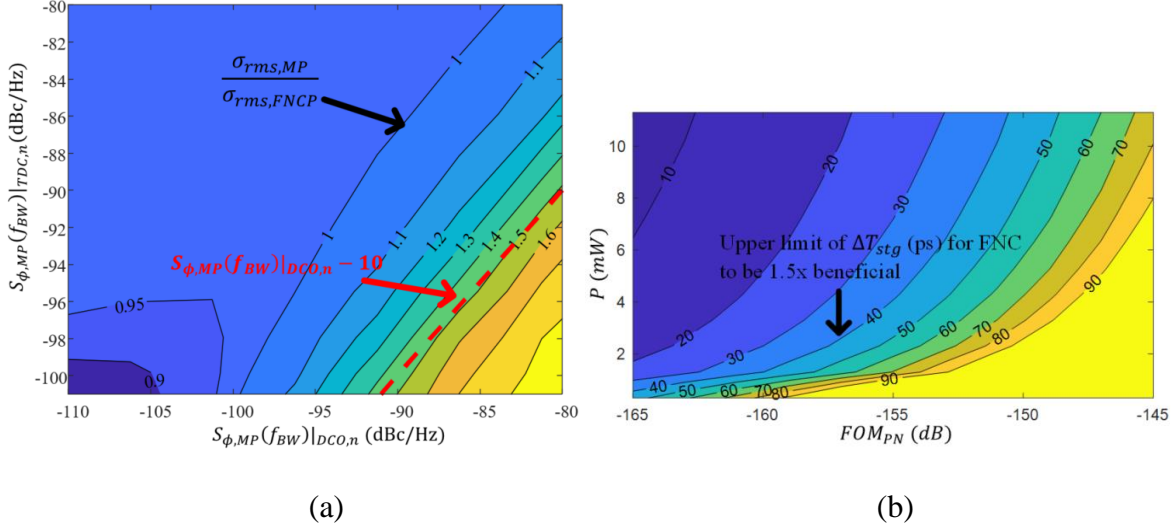


Figure 2.14 Contour plots of (a) jitter improvement by FNC with MP in-band PN level due to DCO on x-axis and TDC on y-axis, (b) required maximum delay per DCO stage for FNC to be 1.5x beneficial.

where f_{off} is the frequency offset from f_{DCO} and P is the power consumption. Assuming $S_{\phi,MP}(f_{BW})|_{DCO,n} = S_{\phi,DCO,n}(f_{BW})$, where f_{BW} is the PLL bandwidth, by combining $S_{\phi,MP}(f_{BW})|_{DCO,n} > 10 \times (5)$ with $f_{DCO} = 1/(\Delta T_{stg} \cdot N_{stg})$, the condition of ΔT_{stg} for $\sigma_{rms,MP}/\sigma_{rms,FNCP} > 1.5$ is

$$\Delta T_{stg} < \sqrt{\frac{1mW}{P} \cdot \frac{f_{ref} \cdot 10^{\frac{FOM_{PN}}{10}}}{0.49 \cdot f_{BW}^2}} \quad (14)$$

It is notable that N_{stg} is cancelled. Figure 2.14(b) shows a contour plot of this condition assuming $V_{dd} = 1.2V$, $f_{ref} = 40MHz$, $f_{BW} = 2MHz$, while sweeping FOM_{PN} and P of DCO. The restriction for FNC is stricter with DCO with better FOM_{PN} and power consumption.

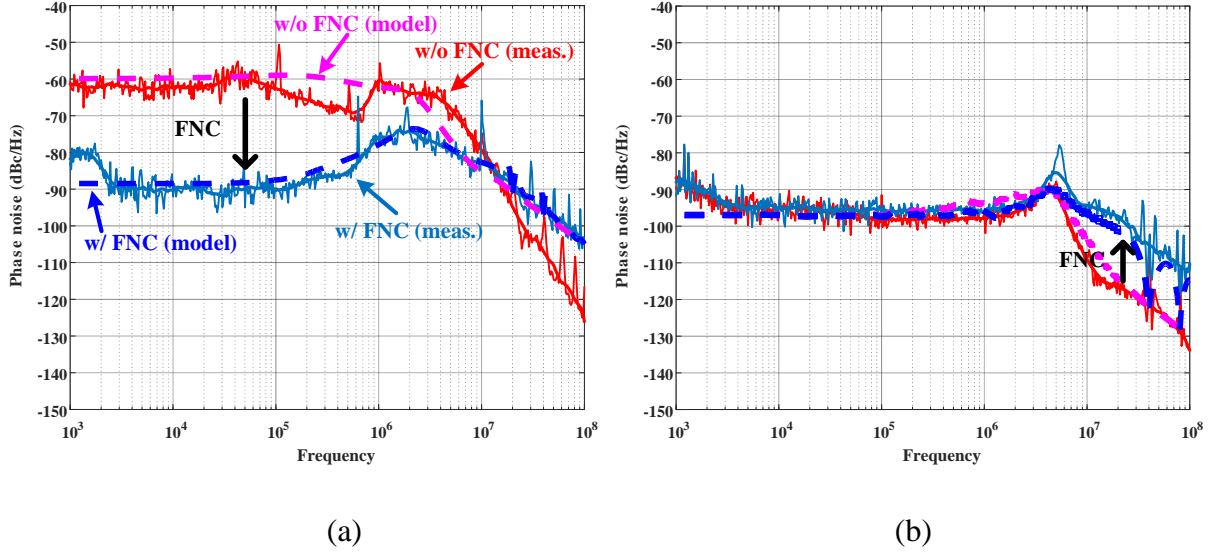


Figure 2.15 Measured phase noise (meas.) in comparison with the analytical model (model) when (a) DCO noise dominates and (b) TDC noise dominates.

2.6 Synthesizable Feedforward PLL Measurement Results

A test chip was fabricated in 65nm CMOS technology. The layout of the PLL is done by APR tool, where the auxiliary cells are designed manually but then placed and routed automatically. The PLL is measured in two different modes: 1) High DCO noise, and 2) Low DCO noise. The two modes are realized by turning on or off a free-running RO that shares a supply with the DCO in the PLL. When the free-running RO is on, the supply noise due to its oscillation increases the noise level of the main DCO. For each environment, the phase noise plots with and without FNC are compared in Figure 2.15 along with the analytical model results. In the first case, $f_{ref} = 20\text{MHz}$, $f_{out} = 800\text{MHz}$. From Figure 2.15(a), we can observe that FNC reduces the in-band phase noise by 15dBc/Hz at frequency offset of 1MHz, and the integrated jitter (1K – 10MHz) by 4.22x (291.8ps \rightarrow 69.0ps), for only a 3.54% increase in power (7.07 \rightarrow 7.32 mW). On the second case with low DCO noise, $f_{ref} = 40\text{MHz}$, $f_{out} = 840\text{MHz}$ and the power increased

from 8.78mW to 9.05mW. As we can see from Figure 2.15(b), TDC quantization noise dominates the PLL noise and the FNC increases the out-of-band noise compared to the main path, increasing the integrated jitter (10.3ps \rightarrow 18.6ps). The analytical model follows the overall tendency of the measurement results and accurately predicts both cases.

2.7 Conclusion

An automation flow for a synthesizable ADPLL has been proposed in this Chapter. The combination of human knowledge and simulation results fastened the modeling process to requiring only 3 sets of results. To prove the concept, 8 PLLs were generated, and their performances were compared to the input specifications. All output performances satisfied the input requirement and the phase noise prediction of behavioral model highly agreed with the measurement result. Also, a novel calibration-free feedforward implementation method for synthesizable ADPLL is proposed as part of the fabricated chip. The effects of FNC depending on the dominant noise sources are analyzed with frequency domain model and the results are compared with measurements. The condition for the proposed FNC method to be beneficial is derived.

Chapter 3 PLL Fractional Spur's Impact on FSK Spectrum and a Synthesizable ADPLL for a Bluetooth Transmitter

3.1 Introduction

The design and verification process is becoming more challenging for analog circuit designers due to the increased layout sensitivity and non-intuitive design rules from complex processes such as multi-patterning in sub-20nm nodes. Open-source analog generators (public cloud repositories for analog circuit design automation), have been developed over the past decade to assist analog circuit designers and reduce time-to-market of ASIC designs [5], [36], [37]. One of the approaches for automation is to adopt a cell-based analog circuit design and use existing digital synthesis tools to generate the layout [5]. By taking advantage of the built-in P&R engine in a digital layout tool, the cell-based approach has significantly lower porting cost once the cells are prepared.

While the PLL is one of the most popular building blocks being explored with the cell-based methodology, recent publications of cell-based PLLs focus on fractional-N frequency

multipliers to widen the applications [13], [16], [17], [21]–[23]. While the automatic routing of the P&R tool is the biggest contribution to the reduced design time, it poses a new challenge in implementing a fractional-N PLL by adding mismatches in timing control blocks such as the digitally controlled oscillator (DCO), TDC, and digital to time converter (DTC). Since the blocks process a saw-tooth shaped time error between the output and reference clock, their non-linearity over the range of at least one period of the DCO (T_{DCO}) results in high fractional spurs in the output spectrum. While the blocks are widely used in fractional-N synthesizable PLLs, only a few ([16], [17]) proposed a systematic solution for the routing uncertainty.

[13] and [20] use the internal phases of the DCO to process fractional phase information. [13] proposed a fractional-N injection-locked PLL (ILPLL) by injecting a reference edge to one of the interpolated internal phases of the DCO. A two-step DTC was introduced in [20] to delay the reference clock by a desired amount, where the internal phases of a replica DCO are used as coarse DTC steps so that the resolution is synchronized to the period of the main DCO. The DCO is especially vulnerable to routing mismatch since it requires a large number of cells and routing connectivity to cover a desired frequency range. While these architectures are highly sensitive to delay mismatch between DCO stages, both lack a circuit level solution for the problem.

[21] uses a direct-digital synthesizer driven by a free-running oscillator along with a D-flip-flop (DFF) based sub-sampling phase detector (SSPD). Fractional-N operation is achieved by adding a fractional code to the output of the SSPD. While the architecture alleviates the non-linearity coming from TDC/DTC by avoiding any D-to-A or A-to-D process during the phase error detection, the free-running oscillator and the phase interpolator are custom designed, degrading the merits of a synthesizable PLL. The PI is a source of non-linearity that directly impacts the output spectrum, which is very sensitive to the routing mismatch of P&R tool.

To solve the issue of routing mismatch in a two-step DTC, [16]–[18] compensate the coarse DTC’s INL by setting proper offsets on the fine DTC control words for each coarse step. This zero-order interpolation-based calibration scheme systematically cancels a discontinuous INL induced by the routing mismatches. However, the frequency command word (FCW) is limited to 10, which requires expensive reference clock with frequency (f_{ref}) higher than 100s of MHz to support a multi-gigahertz output frequency.

In this chapter, we present an open-source, fully synthesizable PLL driven by a 40MHz reference clock for a 2.4GHz BLE-TX (FCW > 60) with a calibration scheme that compensates the P&R induced non-linearity [27], [38]. The proposed synthesizable ADPLL employs a novel two-step TDC (TSTDC) and digital on-chip calibration scheme that reduces the fractional spurs. An embedded TDC (EMBTDC) is used for coarse quantization and Vernier delay line TDC (DLTDC) is used for fine quantization. This reduces the required number of stages of DLTDC as well as its peak INL value, while it is used to measure and compensate the EMBTDC non-linearity. The BLE-TX was fabricated in 12nm FinFET technology and the measured performance satisfies the BLE standard requirements for most of the channels. The standalone PLL supports an output frequency range of 1.8-2.7GHz, consuming 3.91mW at 2.4006 GHz, occupying an area of 0.063mm².

One of the major challenges of designing a PLL for BLE using a P&R tool is the degraded GFSK modulation performance caused by non-linearity which shows up as spurious tones in the frequency domain. To our knowledge, there is no existing analysis on the changes in the PLL fractional spurs before and during FSK modulation. In this work, we provide a detailed analysis of the PLL fractional spur’s impact on the FSK spectrum and derive a spectral mask for PLLs that will then satisfy the BLE mask based on a semi-analytical model. Prediction of the BLE spectrum

for a given PLL fractional spurs sets a clear linearity target for PLL designers and opens new possibilities for PLL designers to better optimize fractional-N PLLs for FSK modulation applications, and explore architectures that are more easily ported and amenable to design automation.

3.2 Prediction of BLE-TX Spurious Tones from PLL Fractional Spurs

The center frequencies of BLE channels, f_{CH} , are 2MHz apart starting from 2.402GHz to 2.480GHz [39]. With $f_{ref} > 2\text{MHz}$, a PLL needs to operate in a fractional-N mode in order to lock to a certain f_{CH} , and $f_{ref} = 40\text{MHz}$ is common for BLE transmitters. For data transmission, the PLL modulates the output frequency by +250kHz and -250kHz from f_{CH} for data 1 and 0, respectively. Spurious tones due to f_{CH} also are affected by this modulation, resulting in different positions and amplitudes in the FSK spectrum compared to those of a standalone PLL mode operating at f_{CH} . We investigate the effect of FSK modulation on spurious tones by defining the fractional spur positions due to different harmonics of periodic nonlinearity noise (PNN) of the PLL in Section II-A and using a mathematical model to derive the spurious tones' relative amplitudes and positions compared to the original values in Section II-B. Experimental results are shown in Section II-C to prove the derived model.

3.2.1 Fractional Spur Positions

Since the frequency modulation's effect of a fractional spur depends on which harmonic of the PNN that it originated from, we first need to relate each harmonic with the resulting fractional spur position. Unlike the case when the fractional frequency is realized by high-order multi-modulus dividers [40], [41], the fractional spur positions of a phase domain (PD) architecture [31] are more straight-forward to understand. As shown in Figure 3.1, in PD-PLLs, a saw-tooth

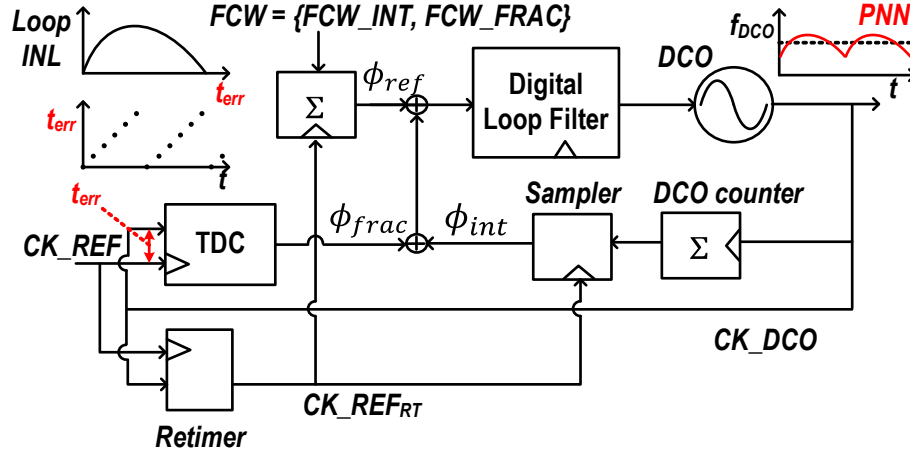


Figure 3.1 Phase Domain ADPLL block diagram with loop INL with respect to t_{err}

shaped fractional phase error is processed by the TDC and digital loop filter. The loop nonlinearity with respect to the ideal time error between the PLL output clock CK_DCO and reference clock CK_REF , denoted as t_{err} , results in a PNN in the ensemble average of CK_DCO frequency f_{DCO} , with a time period of $(1/FCW_FRAC) \cdot T_{ref}$, where T_{ref} is the period of CK_REF and FCW_FRAC is the fractional part of the FCW . The discrete time saw-tooth shaped signal t_{err} can be expressed as

$$t_{err}[N_{ref}] = MOD(FCW_FRAC \cdot T_{DCO} N_{ref}, T_{DCO}), \quad (1)$$

where N_{ref} is the current count of reference cycle, T_{DCO} is the period of CK_DCO , and $MOD(a, b)$ is a modulus function that returns the modulus value of a/b . $t_{err}[N_{ref}]$ can be viewed as a sampled version of a continuous time (CT) signal with period $(1/FCW_FRAC) \cdot T_{ref}$ as shown in Figure 3.2(a). The CT PNN can be expressed in Fourier-series (FS) as

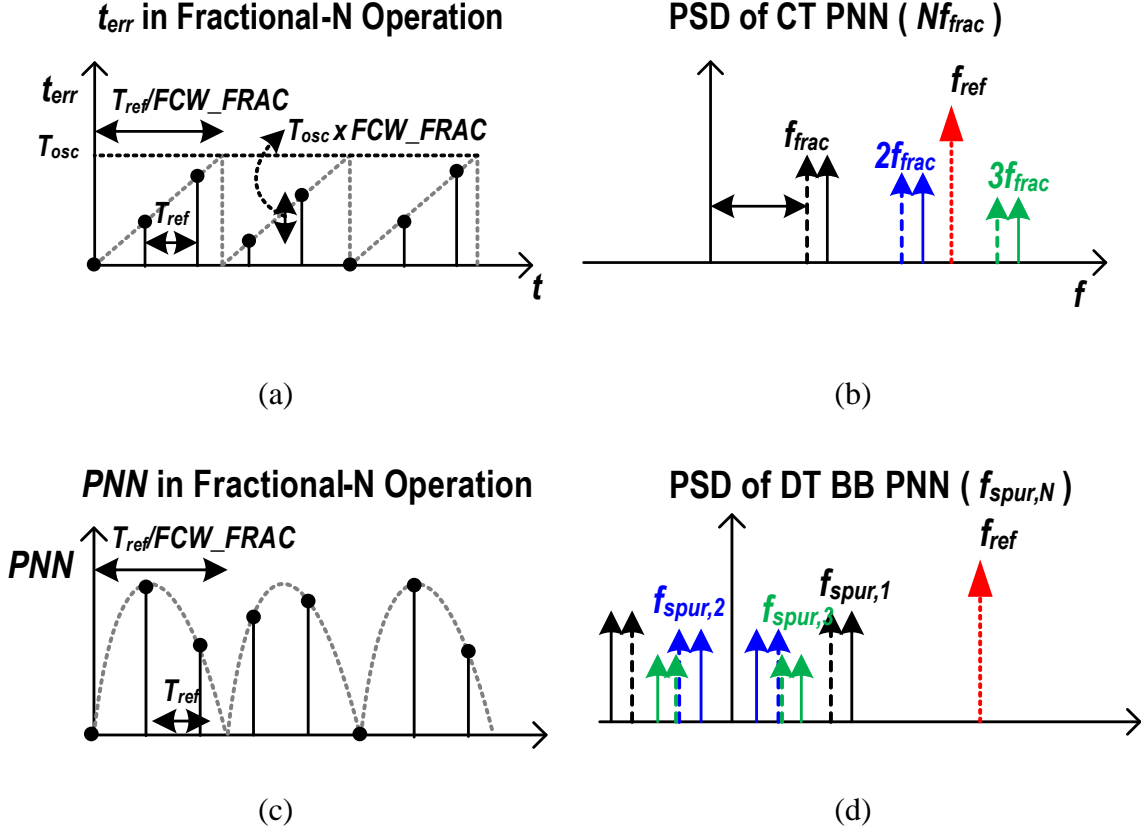


Figure 3.2 Conceptual illustration of (a) $t_{err}[N_{ref}]$, (b) PSD of CT PNN, (c) time domain of DT PNN, (d) PSD of DT baseband PNN.

$$s_{CT}(t) = \frac{1}{2}a_0 + \sum_{N=1}^{\infty} a_N \cos(2\pi N f_{frac} t) + \sum_{N=1}^{\infty} b_N \sin(2\pi N f_{frac} t), N = 1, 2, 3 \dots \quad (2)$$

where $f_{frac} = FCW_FRAC \cdot f_{ref}$. According to sampling theory, the baseband positions of the fractional spurs after being sampled at f_{ref} rate are the results of spurs at Nf_{frac} down converted by subtracting the nearest integer multiple of f_{ref} . Depending on which side of the closest multiple of f_{ref} that Nf_{frac} is, the spurious tone moves either closer or farther from the center frequency when FCW_FRAC increases, which will happen during FSK modulation for transmitting data 1,

as shown in Figure 3.2(b) and (d). Thus, we define the fractional spur position of the N^{th} harmonic of discrete time (DT) PNN as below

$$f_{spur,N} = Nf_{frac} - af_{ref}, a = 0, 1, 2, 3 \dots, \left(a - \frac{1}{2}\right) f_{ref} < Nf_{frac} < \left(a + \frac{1}{2}\right) f_{ref} \quad (3)$$

For example, with $f_{frac}=16\text{MHz}$ and $f_{ref}=40\text{MHz}$, $f_{spur,2} = 40\text{MHz} - 32\text{MHz} = 8\text{MHz}$, $f_{spur,3} = 40\text{MHz} - 48\text{MHz} = -8\text{MHz}$ as shown in Figure 2.2(d) as the dotted arrows. While $f_{spur,2}$ and $f_{spur,3}$ are in same absolute frequency offset, when f_{frac} changes by Δf , frequency offset of $f_{spur,2}$ reduces while it increases for $f_{spur,3}$ as shown as solid arrows in Figure 2.2(d). The frequency perturbation due to the PNN is held for T_{ref} in PD ADPLL, allowing us to express the resulting sampled and held periodic signal by simply replacing Nf_{frac} of (2) with $f_{spur,N}$ of (3), resulting in below equation. This definition will be used for the rest of the paper.

$$s(t) = \frac{1}{2}a_0 + \sum_{N=1}^{\infty} a_N \cos(2\pi f_{spur,N}t) + \sum_{N=1}^{\infty} b_N \sin(2\pi f_{spur,N}t), N = 1, 2, 3 \dots \quad (4)$$

3.2.2 Semi-analytical Model of BLE Fractional Spurs

In [42], the power spectral density (PSD) of a BFSK signal with random data was derived by obtaining an autocorrelation of the signal and then conducting a Fourier-transform. But the derivation assumes a spectrum without any spurious tone. By adding terms that mathematically represent PLL fractional spurs to the derivation, we analyze the impact of BFSK modulation on the spurious tones' positions and amplitudes and later prove that the maximum spur prediction is valid for GFSK modulation as well. This section shows a simplified version of the analysis while the details can be found in the Appendix. We assume $a_0 = b_N = 0$ from (4) in this section, since a_0 contributes only on the frequency offset, not the spurious tone, and the presence of b_N does not

change the result and the logical flow compared to only having a_N . Using the definition of frequency PNN in (4) and proper assumptions shown in the Appendix, a BFSK signal in the presence of fractional spurs can be written as

$$u_m(t) = A_u \cos \left(B_n(t) + \sum_{N=1}^{\infty} a'_N \sin \left(C_{N,n}(t) \right) \right),$$

$$\approx A_u \cos(B_n(t)) - A_u \sin(B_n(t)) \cdot \left\{ \sum_{N=1}^{\infty} a'_N \sin \left(C_{N,n}(t) \right) \right\}, \quad (5)$$

$$B_n(t) = 2\pi \cdot [f_{CH}t + x_{n+1}(250 \cdot 10^3)(t - nT)$$

$$+ (250 \cdot 10^3) \sum_{r=1}^n x_r], nT \leq t < (n+1)T, \quad (6)$$

$$C_{N,n}(t) = 2\pi \cdot [f_{spur,N}t + x_{n+1}N(250 \cdot 10^3)(t - nT)$$

$$+ N(250 \cdot 10^3) \sum_{r=1}^n x_r], nT \leq t < (n+1)T, \quad (7)$$

where x_n is the n^{th} transmitted data (1 or -1), $B_n(t)$ and $C_{N,n}(t)$ are the phase components due to the main signal and the fractional spur's N^{th} harmonic during frequency modulation for n^{th} data transmission, respectively. a'_N is a scaled version of a_N to accommodate the frequency to phase conversion of PNN. We use phase terms instead of frequency terms because the phase information has to be preserved between two consecutive data transmission as a continuous phase FSK modulation. Illustration of the derivatives of $B_n(t)$ and $C_{N,n}(t)$, which are the frequencies of each component, are shown in Figure 3.3. We can observe that $B_n(t)$ deviates by 250kHz for data encoding while $C_{N,n}(t)$ deviates by $N \cdot 250\text{kHz}$. The second term of (5) can be seen as a mixing between two signals: $A_u \sin(B_n(t))$ and $\left\{ \sum_{N=1}^{\infty} a'_N \sin \left(C_{N,n}(t) \right) \right\}$. Using sinusoidal properties, we can further expand the term as below.

$$\begin{aligned}
A_u \sin(B_n(t)) \cdot \left\{ \sum_{N=1}^{\infty} a'_N \sin(C_{N,n}(t)) \right\} \\
= \sum_{N=1}^{\infty} \{ a'_N \sin(B_n(t) + C_{N,n}(t)) + a'_N \sin(B_n(t) - C_{N,n}(t)) \}. \quad (8)
\end{aligned}$$

As (8) shows, the mixing results in two terms per harmonic; addition between $B_n(t)$ and $C_{N,n}(t)$, and subtraction between the two. For $N = 1$, the second term of (8) leads to a single tone at $f_{CH} - f_{spur,1}$ because both $B_n(t)$ and $C_{1,n}(t)$ have identical frequency modulation of 250kHz and they cancel each other. This can be applied to GFSK signal as well because the modulation terms are still identical for center frequency and the fundamental harmonic of the spurious tone.

Except for the second term of $N = 1$, other terms of (8) can be seen as separate FSK signals with different center frequencies and frequency deviations. The first term of $N = 1$ and both terms of other odd harmonics ($N = 3, 5, 7, \dots$) result in phase accumulation of integer multiples of π for one data period ($1\mu s$), which is expressed as $2\pi \cdot (N \pm 1) \cdot 250kHz \cdot 1\mu s$. In these special cases, continuous phase FSKs become identical to discontinuous FSKs because every data transmission ends with the same phase. The spectrum of discontinuous phase modulation contains two sinusoidal functions with amplitude degraded by 6.02 dB compared to the original signal and located at $\pm \Delta f$, where Δf is the frequency deviation for data 0 and 1, from the original position [42]. In (8), the original positions are at $f_{CH} \pm f_{spur,N}$, and FSK modulation will spread the tones to $f_{CH} + f_{spur,N} \pm (N + 1) \cdot 250kHz$ and $f_{CH} - f_{spur,N} \pm (N - 1) \cdot 250kHz$ with 6.02dB degraded amplitudes. This phenomenon is shown on the right side of Figure 3.3 for $N = 1$ and 3. This only applies to FSKs that the product of data duration ($1\mu s$) and frequency deviation (250kHz) is 0.5. For even N 's, both terms of (8) result in continuous phase modulation that are not a discontinuous phase modulation and the power of the spurs are attenuated greater than 15dB.

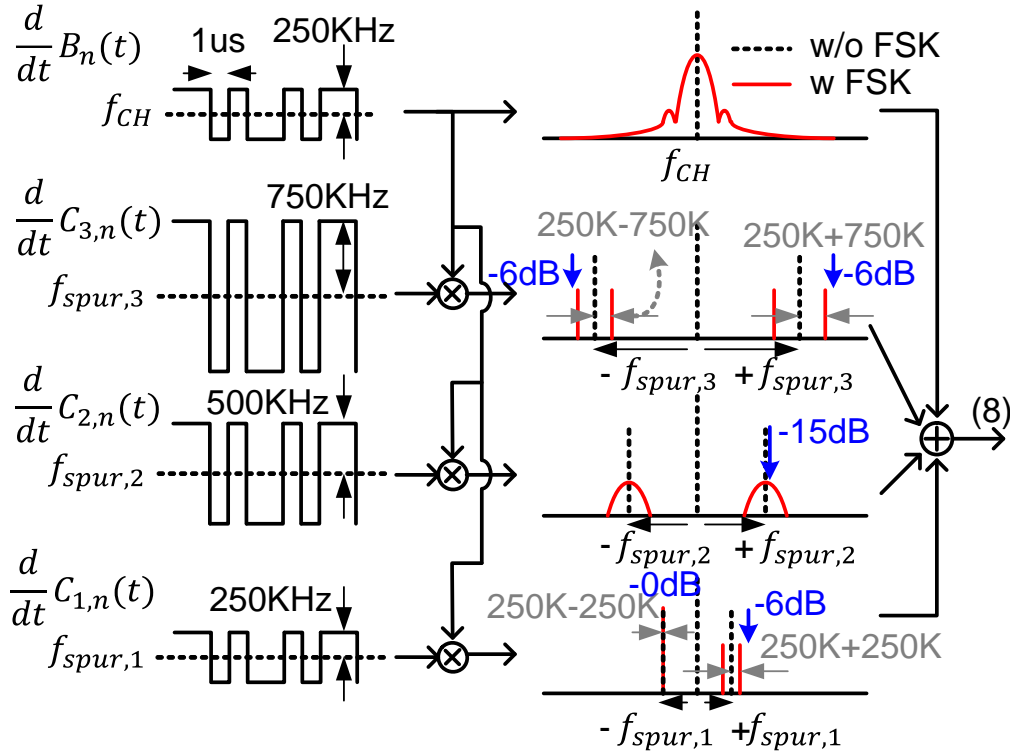


Figure 3.3 Conceptual illustration of the impact of modulation on fractional spurs of equation (8).

A special case occurs when $|f_{spur,1}| < 250kHz$, where a_N, b_N changes for data 1 and 0, resulting in a spread spectrum even for odd value of N 's, showing attenuation greater than 15dB according to experimental results. Therefore, to focus on the worst-case spurious tones at the output spectrum, we consider the cases when $|f_{spur,1}| \geq 250kHz$. In summary, the group of offset positions and amplitudes of the modulated signal's spurious tones resulting from odd harmonics of a PNN can be expressed as follows.

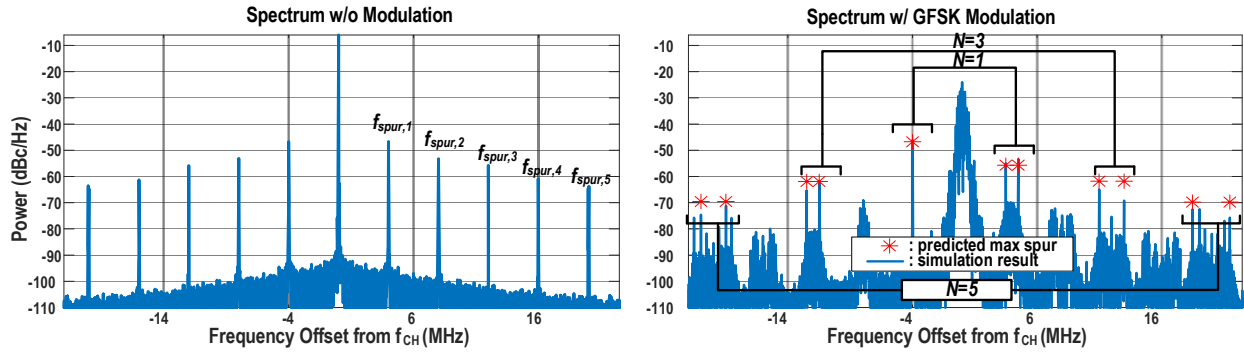
$$f_{ble,spur,N} = \begin{cases} \begin{pmatrix} f_{spur,N+(1+N)\Delta f} \\ f_{spur,N-(1+N)\Delta f} \\ -f_{spur,N} \end{pmatrix}, & N = 1 \\ \begin{pmatrix} f_{spur,N+(1+N)\Delta f} \\ f_{spur,N-(1+N)\Delta f} \\ -f_{spur,N+(1-N)\Delta f} \\ -f_{spur,N-(1-N)\Delta f} \end{pmatrix}, & N = 3, 5 \dots \end{cases} \quad (9)$$

$$S_{ble,spur,N} = \begin{cases} \begin{pmatrix} S_{pll}(f_{spur,N})/4 \\ S_{pll}(f_{spur,N})/4 \\ S_{pll}(f_{spur,N}) \end{pmatrix}, & N = 1 \\ \begin{pmatrix} S_{pll}(f_{spur,N})/4 \\ S_{pll}(f_{spur,N})/4 \\ S_{pll}(f_{spur,N})/4 \\ S_{pll}(f_{spur,N})/4 \end{pmatrix}, & N = 3, 5 \dots \end{cases} \quad (10)$$

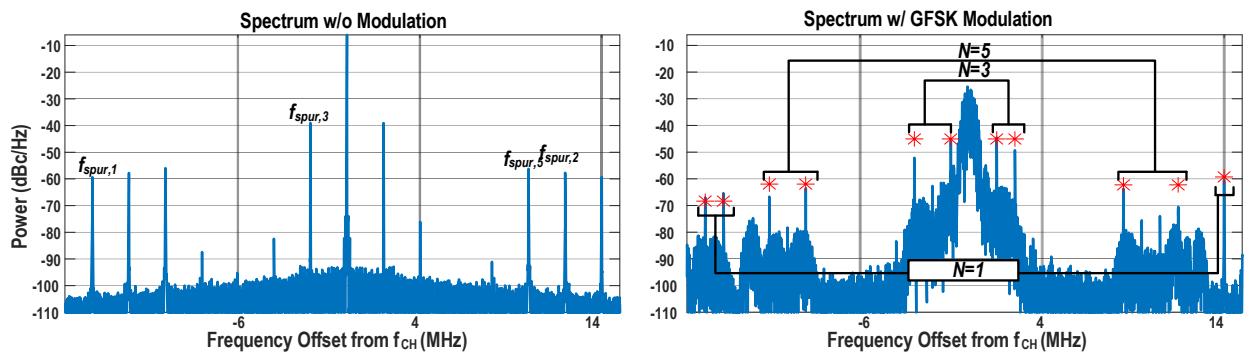
where $f_{ble,spur,N}$ is a group of frequency offsets from the center frequency f_{CH} due to spurious tone at $f_{spur,N}$, $S_{ble,spur,N}$ is a group of amplitudes of spurious tones at each frequency offset and $\Delta f = 250\text{kHz}$.

3.2.3 Experimental Results

To verify the above model, we simulated a GFSK modulated signal with added random phase noise and PNN. $f_{ref} = 40\text{MHz}$, $a_N = b_N = 126 \cdot 10^3$, $N=1, 2, 3, 4, 5$ are used for $s(t)$ from (4) in a Matlab simulation with voltage-time pair vectors. GFSK modulation is realized by assigning proper sequences of frequencies for given data pattern. Since the absolute values of a_N and b_N do not affect the relative amplitudes and positions of spurs after modulation, we used arbitrary values that lead to reasonable spur amplitudes. Figure 3.4(a) and (b) show the spectrum plots of the signal with and without the modulation for $f_{CH} = 2.404\text{GHz}$ and 2.426GHz ,



(a)



(b)

Figure 3.4 Simulated PLL and BLE spectrum and predicted spurs' positions and amplitudes when

(a) $f_{CH} = 2.404\text{GHz}$, (b) $f_{CH} = 2.426\text{GHz}$.

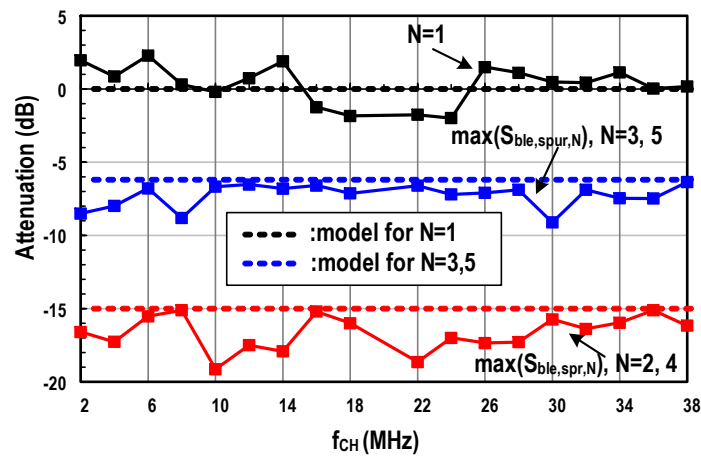


Figure 3.5 Simulated fractional spur attenuation for different N_s .

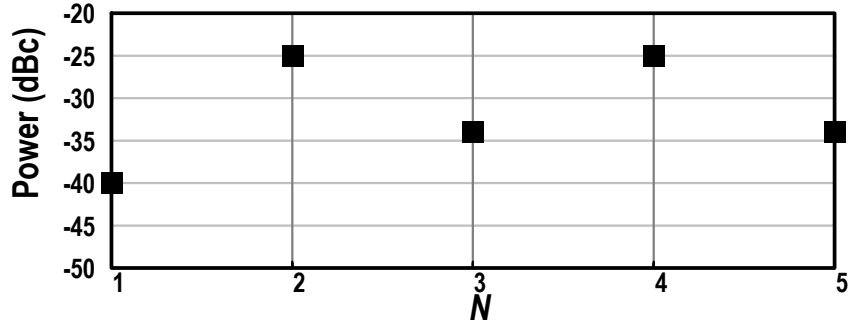


Figure 3.6 Spectral mask for PLL spurious tones at frequencies $f_{frac,N}$, $N=1, 2, 3, 4, 5$ to satisfy BLE spectral mask

respectively. In both cases, spurs at $-f_{spur,1}$ is maintained, which are -4MHz and +14MHz, respectively. For $f_{spur,3}$ and $f_{spur,5}$, the predicted spur positions from (9) are accurate while the amplitudes from (10) match the maximum spur values of each group. Spurs at $f_{spur,2}$ and $f_{spur,4}$ are attenuated more than 15dBc due to the spread spectrum effect of the modulation. Figure 3.5 shows the minimum spur degradation values from GFSK modulation to check the worst case for each group of N's: $N=1, N=3,5, N=2, 4$. Based on the worst-case attenuation of each channel, we derive the spectral mask for PLL fractional spur with respect to N that satisfies the BLE mask, which is shown in Figure 3.6.

3.3 Proposed All Digital BLE Transmitter

The block diagram of the proposed BLE-TX [27], [38] is shown in Figure 3.7. An 8x oversampled BLE finite state machine (BLE_FSM) produces the required $FCWs$ for the fractional-N ADPLL according to the BLE packet data. An embedded TDC (EMBTDC) [30] quantizes the output phases ($DCO_PH[4:0]$) of a 5-stage DCO by latching the voltages of $DCO_PH[4:0]$ at the rising edge of the reference clock (CK_REF) using 5 D-flip-flops (DFFs). Unlike using a separate

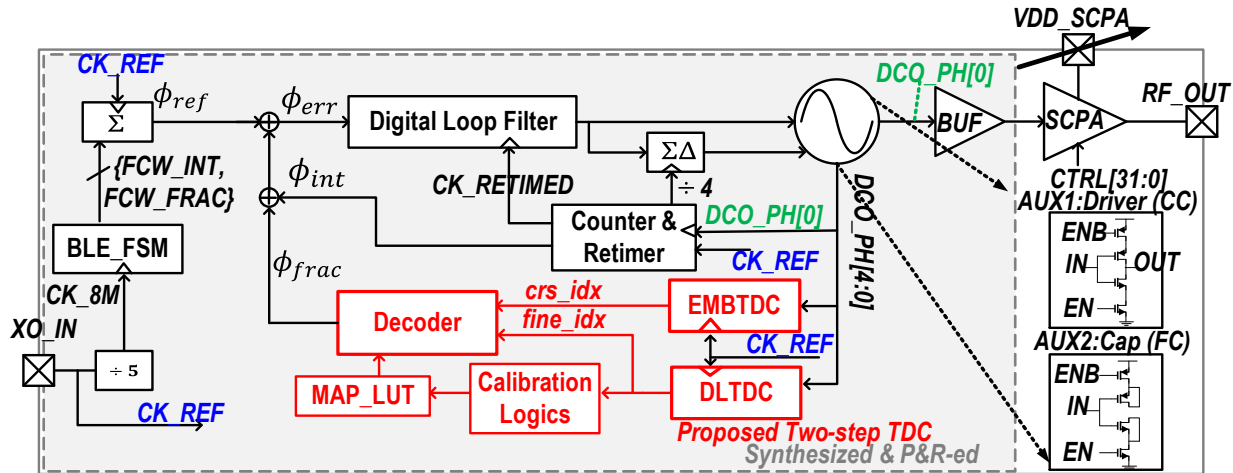


Figure 3.7 Block diagram of the proposed all-digital BLE-TX.

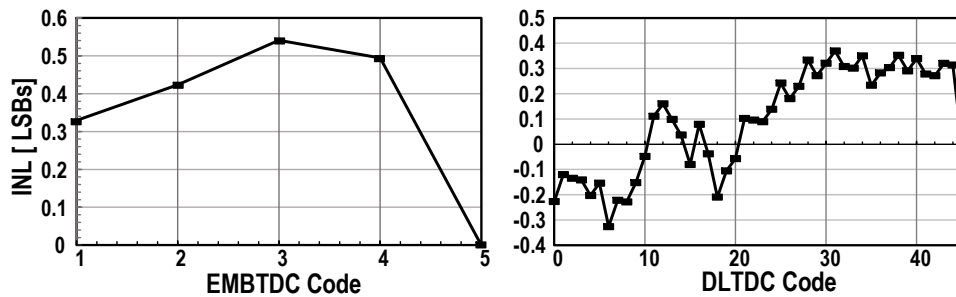
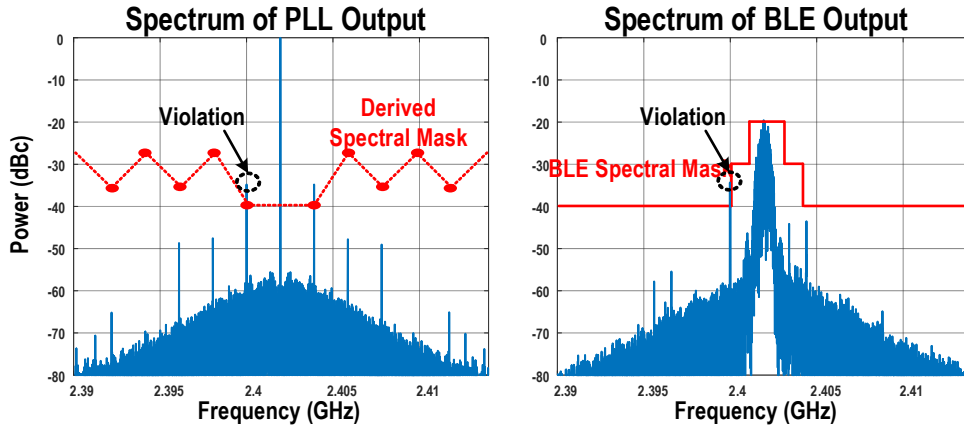
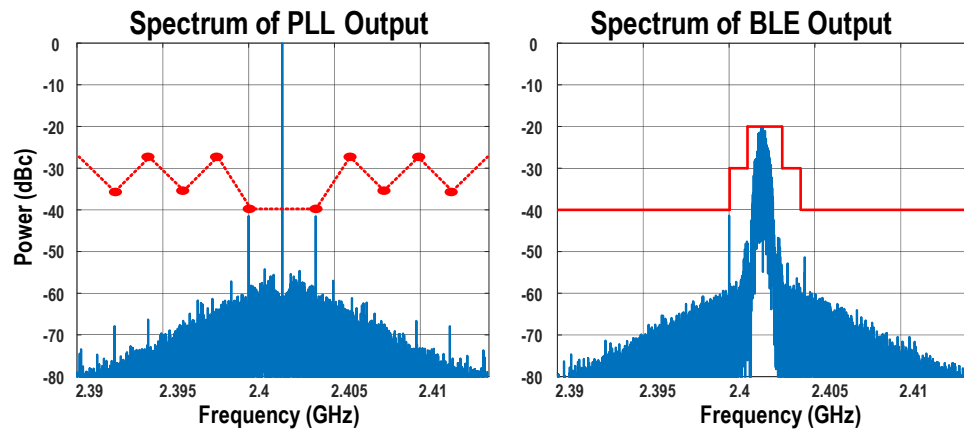


Figure 3.8 Post-parasitic INL of 5 stage EMBTDC (left) and 45 stage DLTDC (right).

TDC, the EMBTDC eliminates the need to calibrate the TDC gain since it captures the internal phase information of the DCO. However, the TDC resolution is limited by the number of DCO stages, and the linearity is directly impacted by the delay mismatches between the DCO stages. A large number of DCO cells are required to cover the target tuning range (1.8-2.7GHz), which increases the uncertainty of routing by the P&R tool, resulting in significant delay mismatches. Compared to a DCO, a Vernier delay-line TDC (DLTDC) [43], requires fewer cells per stage to generate a delay difference between two delay lines and the impact of routing uncertainty is trivial with a small number of stages. But to cover the input time range of the DCO period T_{DCO} ($\approx 416ps$



(a)



(b)

Figure 3.9 Behavioral simulation results of PLL (left) and BLE (right) output using (a) DLTDC INL and (b) TSTDC INL with calibration at $f_{CH}=2.402\text{GHz}$.

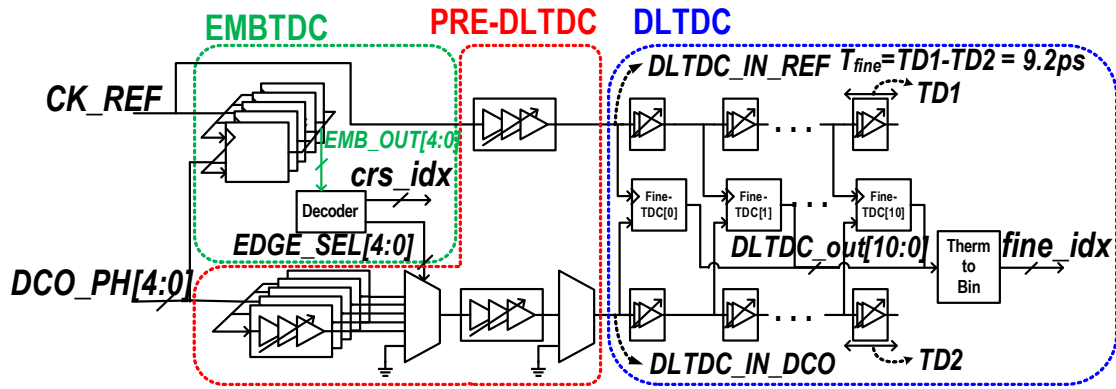
for 2.4GHz) with a DLTDC time resolution T_{DLTDC} ($\approx 9.2\text{ps}$ according to post-parasitic simulation) alone, a large number of delay stages (45) is required, which increases area and non-linearity over the range. Simulated post parasitic INLs of a synthesized 5 stage EMBTDC and 45 stage DLTDC are shown in Figure 3.8. The PLL and BLE's PSD from behavioral simulation result with the DLTDC INL is shown in Figure 3.9(a). The PLL fractional spur violates the mask defined in

Section II, resulting in violation in BLE as well. To utilize the DLTDC's small time resolution and good linearity for a small number of stages, the proposed TSTDC uses an EMBTDC and DLTDC as coarse and fine TDCs, respectively. This combination relaxes the layout requirements for linearity of the DLTDC since the required input time range is reduced by a factor of EMBTDC quantization steps (5) and post parasitic extraction simulation results verify that the linearity of the synthesized DLTDC is adequate for the BLE standard requirements. The EMBTDC's non-linearity is compensated by measuring each quantization level with the DLTDC and using the resulting LUT of DLTDC values as a decoder. Details about calibration are further discussed in Section III-B. The modulated ADPLL output drives a custom designed switched-capacitor power amplifier (SCPA) which is matched on-chip to a 50Ω antenna.

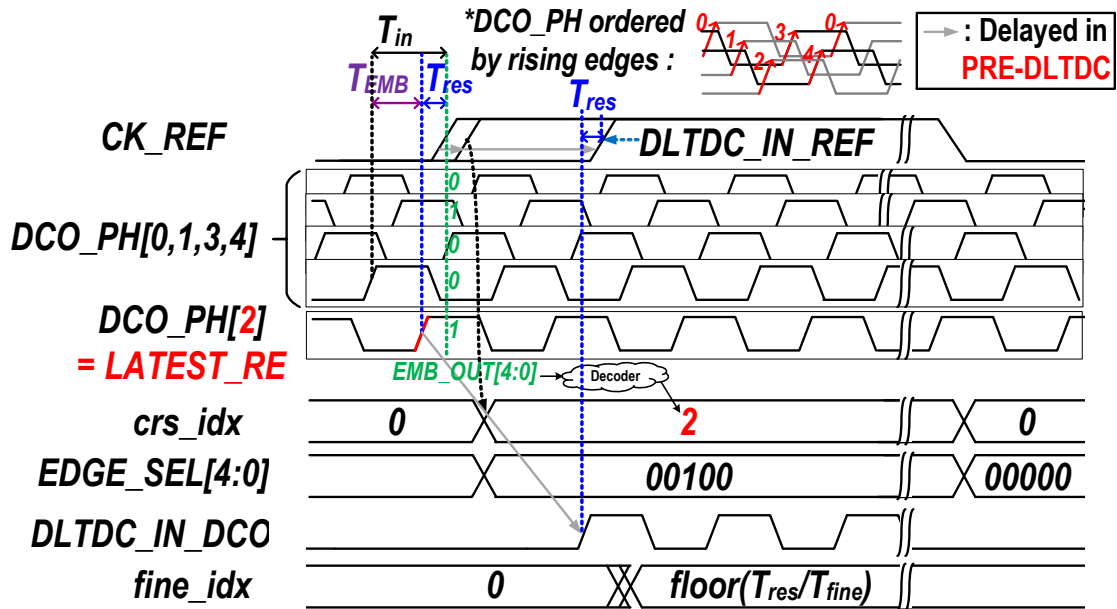
3.3.1 Proposed Two-step TDC

Figure 3.10 shows the block diagram and signal waveforms of the proposed TSTDC. The TDC only uses the rising edges of the DCO and $DCO_PH[4:0]$ is ordered accordingly as shown in Figure 3.10(b). $LATEST_RE$ indicates the latest rising edge among $DCO_PH[4:0]$ on the rising edge of CK_REF and crs_idx is the corresponding index ($LATEST_RE = DCO_PH[crs_idx]$). In this example, $DCO_PH[2]$ was the last phase to rise before $LATEST_RE$ and crs_idx is 2. The index is computed from EMBTDC outputs, and has the information of the time difference between the main clock output $DCO_PH[0]$ and $LATEST_RE$, noted as T_{EMB} in the figure. The index will be later used to pick a digital word from a coarse-fine mapping look up table (MAP_LUT) to represent T_{EMB} in the fine TDC domain. For example, if $T_{EMB} = 120\text{ps}$ and DLTDC time resolution $T_{fine} = 10\text{ps}$, $MAP_LUT(crs_idx) = T_{EMB}/T_{fine} = 12$. The LUT is generated during the calibration phase and will be discussed in III.B. $LATEST_RE$ is then selectively passed to the DLTDC by a 5 bit

one-hot signal $EDGE_SEL[4:0]$ that is also computed using EMBTDC outputs. CK_REF gets delayed by the same amount as $LATEST_RE$ in PRE-DLTDC, preserving the residue time error (T_{res}) to the DLTDC. The delivered two rising edges of CK_REF and $LATEST_RE$ paths are shown as $DLTDC_IN_REF$ and $DLTDC_IN_DCO$ in the figure. $EDGE_SEL[4:0]$ is latched by a delayed CK_REF to prevent glitch propagation. T_{res} is then finely quantized by the DLTDC. The



(a)



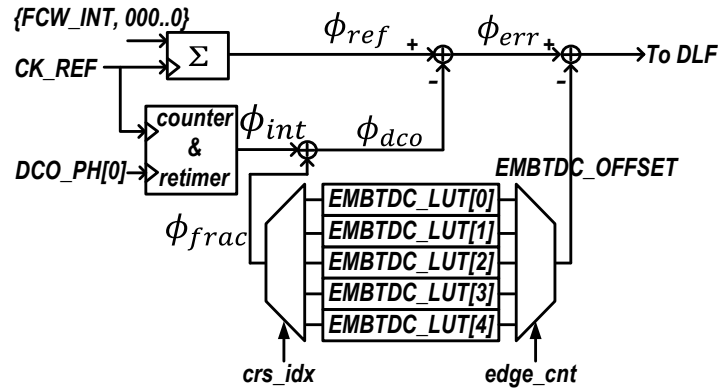
(b)

Figure 3.10 (a) block diagram, (b) signal waveforms of TSTDC

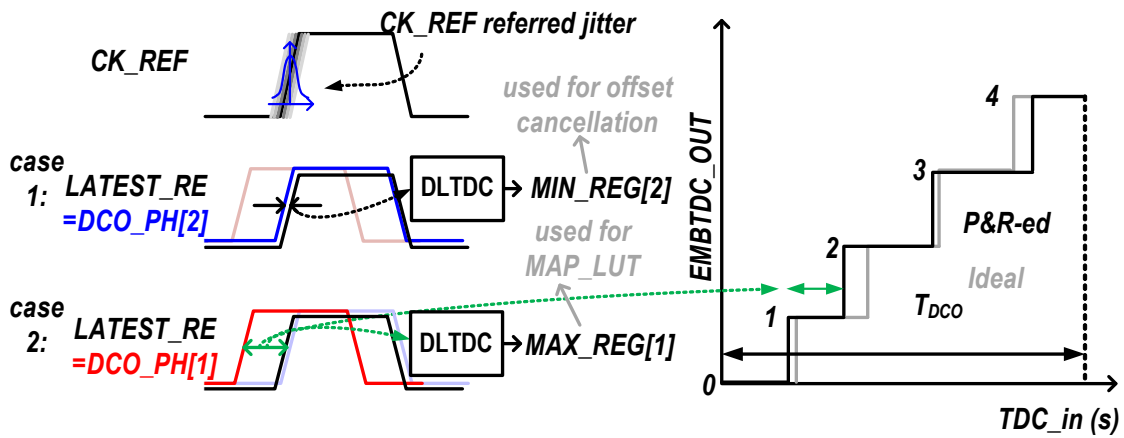
thermometer-coded output $DLTDC_OUT[10:0]$ is converted to binary $fine_idx$. $fine_idx + MAP_LUT(crs_idx)$ thus represents T_{in}/T_{fine} , where T_{in} is the input time difference of the TSTDC.

3.3.2 Proposed Calibration Scheme

During the calibration phase, the PLL only uses the integer part of FCW (FCW_INT) and operates in EMBTDC mode to measure the delay mismatches between adjacent DCO_PHs using the DLTDC. Pre-set values are used for the EMBTDC_LUT in this mode. Since the TDC acts like a bang-bang phase detector in the integer-N mode, the mismatches between EMBTDC_LUT values and actual quantization steps have a trivial effect on the jitter. The PLL finite state machine aligns $DCO_PH[edge_cnt]$ with CK_REF using $EMBTDC_LUT[edge_cnt]$ as an offset on ϕ_{err} as shown in Figure 3.11(a), where $edge_cnt$ increases by 1 for the next edge when the calibration is done for the current one. Once the PLL is locked, $LATEST_RE$ varies between two adjacent DCO_PHs due to jitter: $DCO_PH[edge_cnt]$ when jitter > 0 and $DCO_PH[edge_cnt-1]$ when jitter < 0 . In the first case, the minimum value of $DLTDC_OUT$ is preserved by a bitwise-AND operation in MIN_REG and is used for PRE-DLTDC offset cancellation to match the delays between CK_REF path and $LATEST_RE$ path. In the second case, the maximum value is preserved by a bitwise-OR operation in MAX_REG and the calibration for this edge continues for a programmable number of cycles. With enough samples, $MAX_REG[edge_cnt-1]$ represents the time difference between $DCO_PH[edge_cnt-1]$ and $DCO_PH[edge_cnt]$ quantized by the DLTDC. Illustration of this process is shown in Figure 3.11(b). After the process is completed for all edges, MAP_LUT is generated by accumulating $MAX_REG[4:0]$ to map the coarse quantization steps (crs_idx) with DLTDC indices ($fine_idx$) as shown in Figure 3.12(a). The sum of MAX_LUT (MAX_final_idx) represents the total steps of $fine_idx$ that covers T_{DCO} and is used to automatically select one of the on-chip pre-registered $FINAL_LUTs$ that have fractional digital values for each



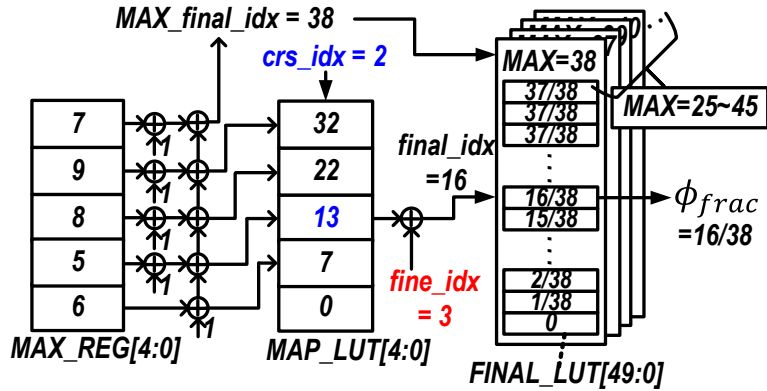
(a)



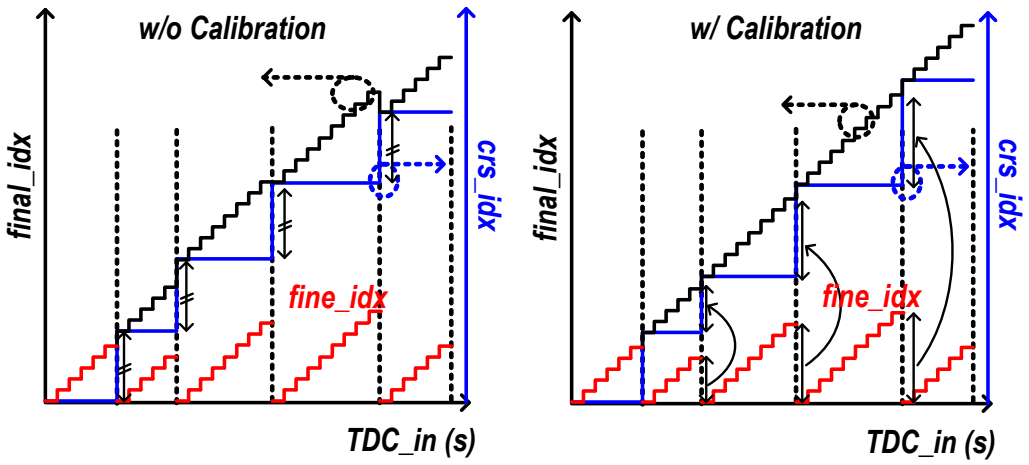
(b)

Figure 3.11 Example of calibrating for delay between DCO_PH[1] and DCO_PH[2]. (a) Locking to DCO_PH[2] by adding EMBTDC_OFFSET, (b) illustration of 2 cases that can happen due to random jitter

MAX_final_idx . Figure 3.12(b) and (c) illustrate the effect of using MAP_LUT . Without calibration, the EMBTDC quantization steps are assumed even and equal to $T_{DCO}/5$, abrupt non-linear jump appears in the transition of crs_idx due to the DCO delay mismatches. When MAP_LUT , which has delay mismatch information, is used after calibration, the non-linear jumps are reduced as shown.



(a)



(b)

(c)

Figure 3.12 Conceptual illustration of (a) process of generating MAP_LUT and FINAL_LUT and an example operation when crs_idx=2 and fine_idx=3, resulting TSTDC transfer function (b) without calibration assuming equal delay between DCO_PHs, (c) with proposed calibration

Figure 3.13 shows the VHDL simulation result of TSTDC INL comparison between before and after the calibration for 3 different corners. EMBTDC and DLTDC INLs are extracted from a post-parasitic spice simulation. As the process corner moves from *ss* to *ff*, the DLTDC resolution becomes finer, and more codes are required to cover T_{DCO} while the EMBTDC resolution is

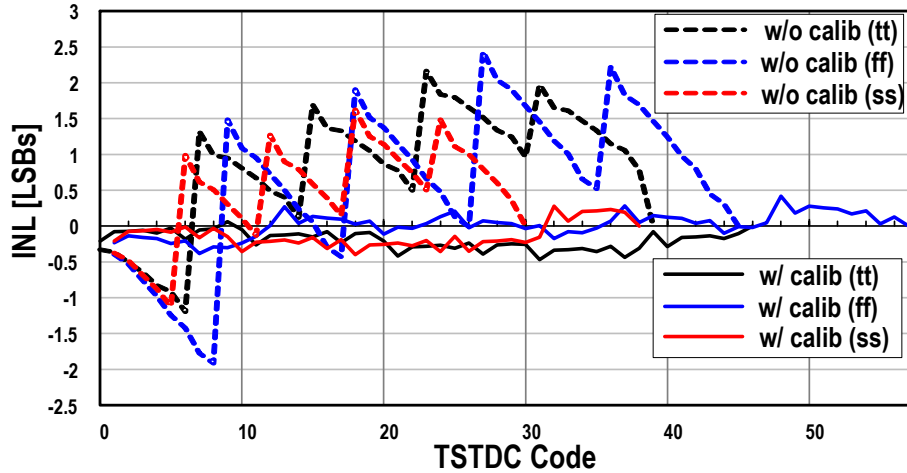


Figure 3.13 VHDL simulation result of TSTDC INL with and without proposed calibration for 3 different corners.

regulated by the PLL. The proposed foreground calibration scheme compensates for the process variation, resulting in similar INL peak values for different corners. But the current version does not have the ability to track real time voltage or temperature drift after the calibration. Behavioral simulation result using the calibrated INL value of typical corner is shown in Figure 3.9(b), satisfying both spectral masks from section II and BLE specification.

3.3.3 Circuit Implementation

A 5 stage RO is used for the DCO, where the coarse-tuning cells (CC) and fine-tuning cells (FC) are realized using tri-state inverter and switched MOS-cap cells as shown in Figure 3.7. The two auxiliary cells are custom designed to match the pitch of standard cells and are placed and routed by the P&R tool. The DCO is composed of 52 CCs and 28 FCs per stage. The design parameters are chosen to satisfy a frequency range of 2.0 ~ 2.5 GHz to cover the BLE channels with sufficient margin, frequency resolution < 150KHz for noise and spur performance and phase noise at 1MHz offset < -85 dB to satisfy frequency drift performance of BLE [44].

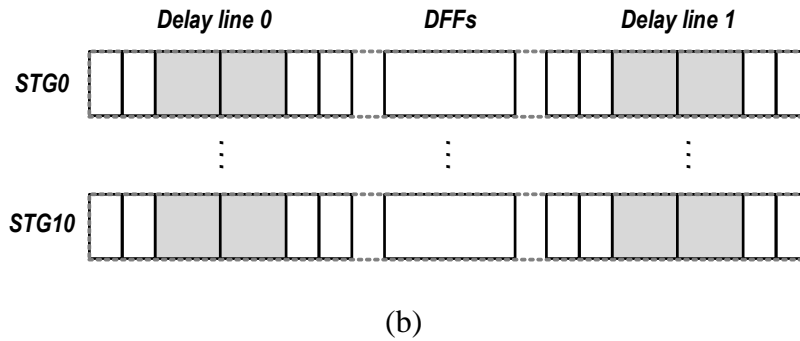
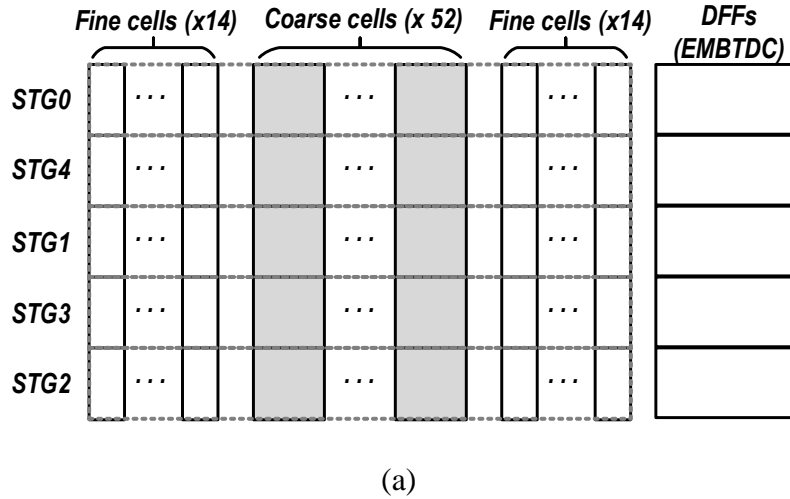


Figure 3.14 Placement pattern for (a) DCO + EMBTDC and (b) DLTDC

DLTDC is composed of 11 stages with each stage having 2 CCs and 4 FCs. The delay difference between the two lines is realized by turning on 1 CC and 2 FCs in the reference path delay line, and 2 CCs and 4FCs in the feedback path delay line. The resulting time resolution is 9.2ps from post-parasitic simulation and it needs to cover the largest EMBTDC quantization step, which is 94.3ps.

The DCO and DLTDC are placed using automatically generated placement scripts that scale with design parameters (number of stages, CCs and FCs). The placement patterns are shown in Figure 3.14. Cells in each stage are placed in one row to minimize the parasitic capacitance

within a stage, while different stages are placed in the shown order to minimize the load mismatch between the stages.

3.3.4 Switched Capacitor Power Amplifier

A switched capacitor digital power amplifier architecture [45] was employed in the BLE-TX for the high switching speed of MOS transistors in FinFET 12nm technology [46]. Figure 3.15 shows the block diagram of the cell-based SCPA. A 32-bit thermometer coded sliced array is chosen to achieve the required tuning range for the output power level. The unit capacitor size is designed to be 750fF. 4 stages of fan-out of 2 inverters are used for each clock driver path, which is preceded by a NAND gate for the *CTRL* signal that enable/disables the unit clock driver. The parasitic capacitance and the wire-bonding inductance are absorbed by the output matching network. In a SPCA, the top plates of all capacitors are connected and matched to a 50 Ω antenna through a band-pass impedance matching network. The bottom plates of the capacitors are individually connected through high-speed RF switches to either the supply voltage *VDD* or the ground *GND*. The total capacitance seen by the output impedance matching network is independent from *CTRL* value because the top-plate is never switched [47]. Therefore, the band-pass matching network does not require tuning at different output power levels. A cell-based design approach was utilized, allowing a potential for design automation in the future.

CTRL signal is used to achieve a certain output power, which controls the number of bit-slices that are switching the bottom plates of the capacitors between the *VDD* and *GND* at the desired frequency. The ratio of switching capacitors to the total number of capacitors dictates the output power level. The peak output power is a function of *VDD* and the optimal load resistance

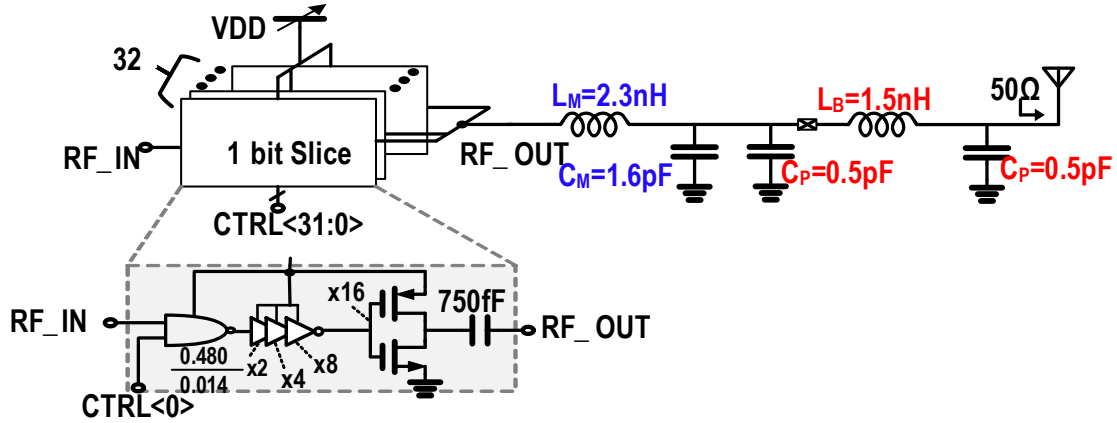


Figure 3.15 Block diagram of the SCPA

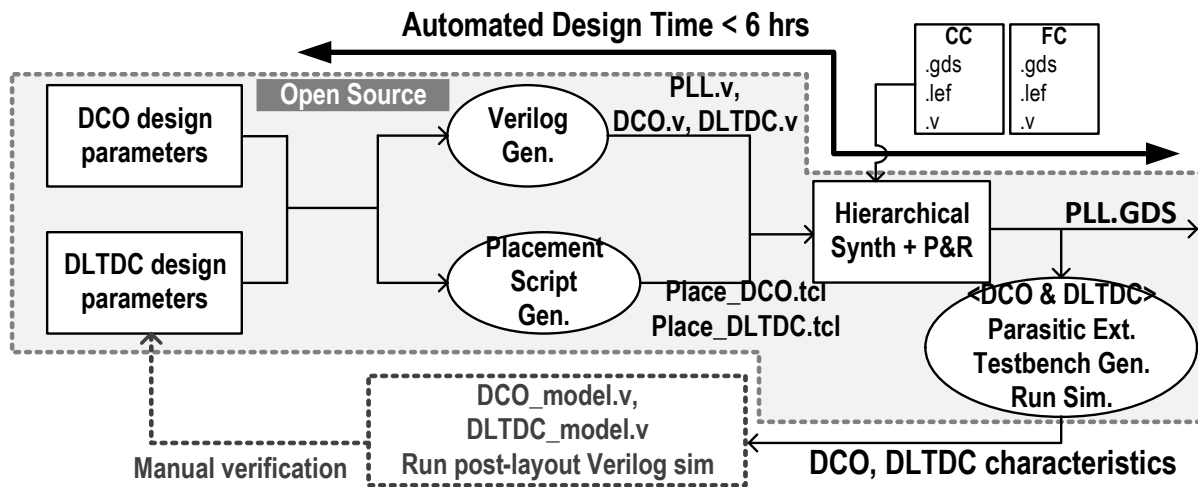


Figure 3.16 Automated design flow for the PLL

(R_{opt}) of the SCPA. Therefore, we employed VDD voltage as a secondary control knob to achieve fine/course tuning of the output power level while optimizing for high efficiency

3.3.5 Design Automation and Open Source

The design process for the BLE-TX is shown in Figure 3.16. The SCPA is custom designed, along with the two auxiliary cells CC and FC. The entire PLL is described in VHDL and the verilog files and TCL scripts including the placement scripts are automatically generated using

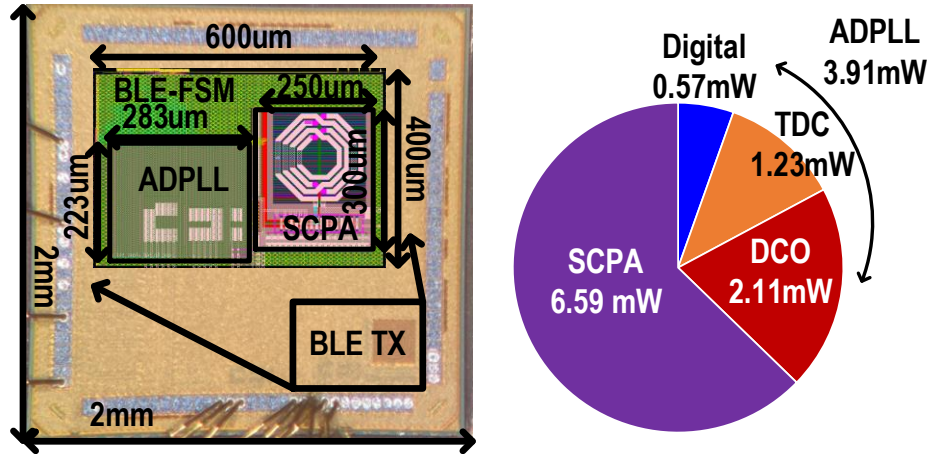


Figure 3.17 Automated design flow for the PLL

the design parameters of the DCO and DLTDC. These files are used in a standard digital flow, ultimately producing a gds file. Using the automatically generated post-layout netlist and testbench, a designer can change the design parameters and run the iteration again in under 6 hours, to achieve the desired functionality and specification.

The grey portion of Figure 3.16 is open source [48]. Users can download and regenerate a PLL design using any design parameters. Modifications can be made and redistributed by pushing the changes to the repository. The automation flow uses commercial tools for the digital flow and simulations. In the future these will be modified to support open-source EDA tools [49], [50].

3.4 Measurement Results

The BLE TX was fabricated in a 12nm FinFET process and it occupies an area of $400\mu\text{m} \times 600\mu\text{m}$. The die micrograph and power breakdown are shown in Figure 3.17. At V_{DD} of 0.8V and maximum $CTRL$ of 32, the measured SCPA output power is 0.8dBm, satisfying class 3 power requirement. Higher output power levels can still be achieved by increasing the supply

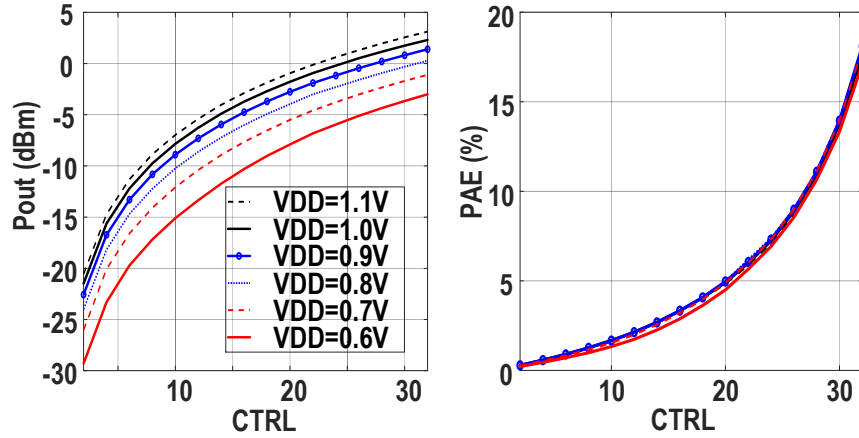


Figure 3.18 Measurement results of SCPA. (a) Pout , (b) PAE with respect to CTRL and VDD.

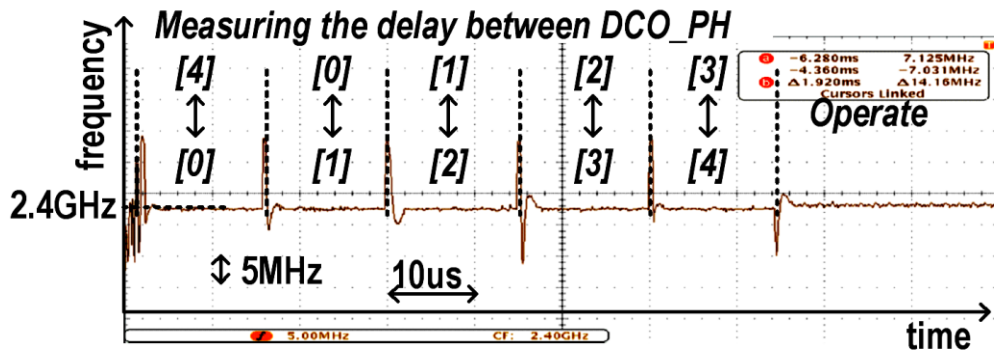
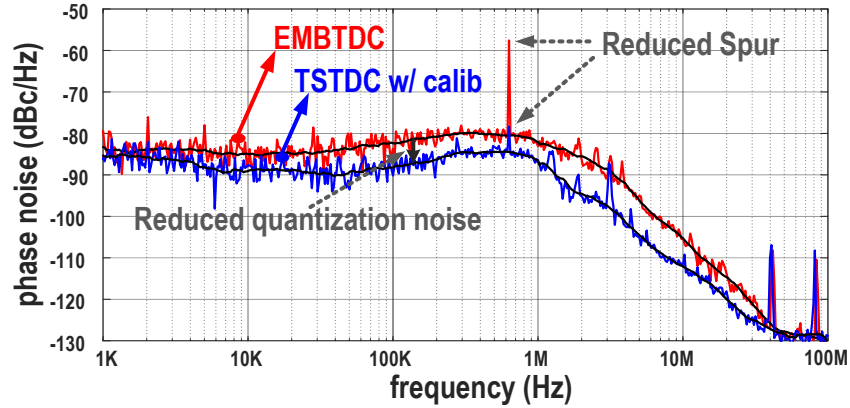
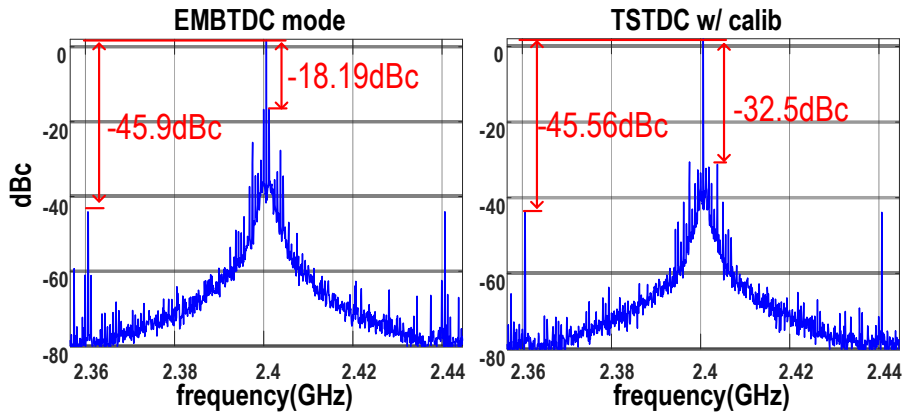


Figure 3.19 Measurement results of standalone PLL Frequency vs time plot during calibration phase

voltage further. Figure 3.18 shows the Pout and PAE with respect to the supply voltage and CTRL value. We can observe that VDD has insignificant impact on the PAE at power back-off but with limited Pout's tuning range. Scaling VDD redefines the P_{sat} of the SCPA and therefore the peak PAE can be maintained at power back-off. The achieved peak PAE is limited to $\sim 18\%$ due to the low quality factor of large matching inductance in finFET process compared to conventional planar CMOS technology with RF flavors.



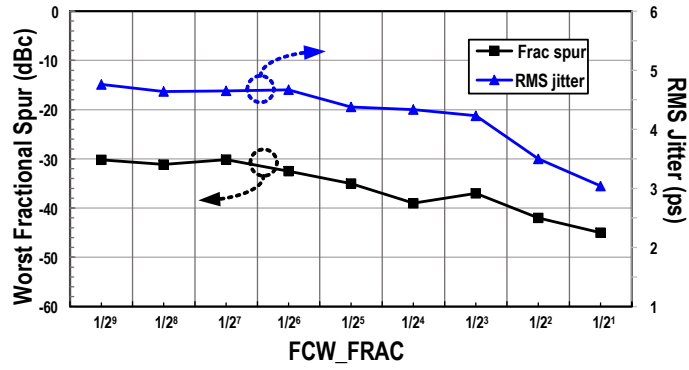
(a)



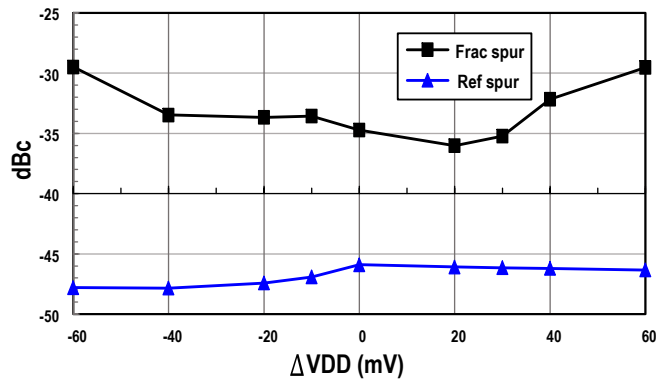
(b)

Figure 3.20 Measurement results of standalone PLL. (a) phase noise plot and (b) spectrum for 2 different modes with FCW = 60.0156, FREF=40 MHz, FOUT=2.4006 GHz

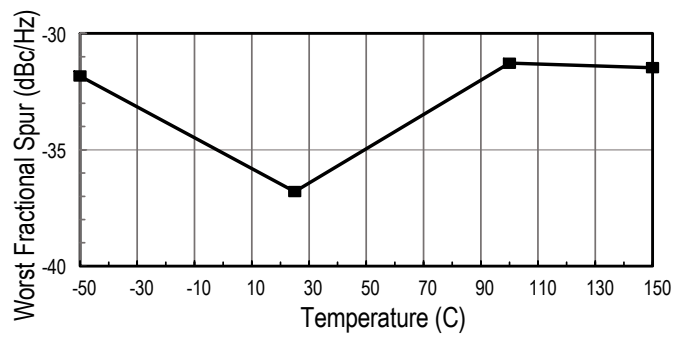
Measurement results of standalone PLL mode are shown in Figure 3.19 and 3.20. The frequency locking behavior during the calibration phase is shown in Figure 3.19. Measured calibration time is around 78us. It is mainly composed of repeating the process of locking a certain DCO_PH to CK_REF and measuring the EMBTDC quantization step for a programmed cycle (64 reference cycles). Phase noise plot for FCW=60.0156 is shown in Figure 3.20(a) and the spurious tone improvements are shown in Figure 3.20(b). From the phase noise plot, we can observe that both the random noise and spurious tones are improved by the proposed technique. From the near



(a)

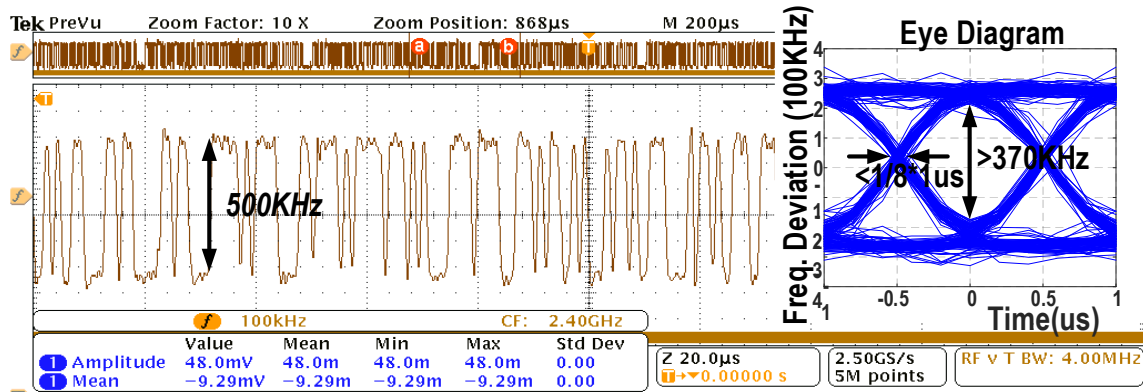


(b)

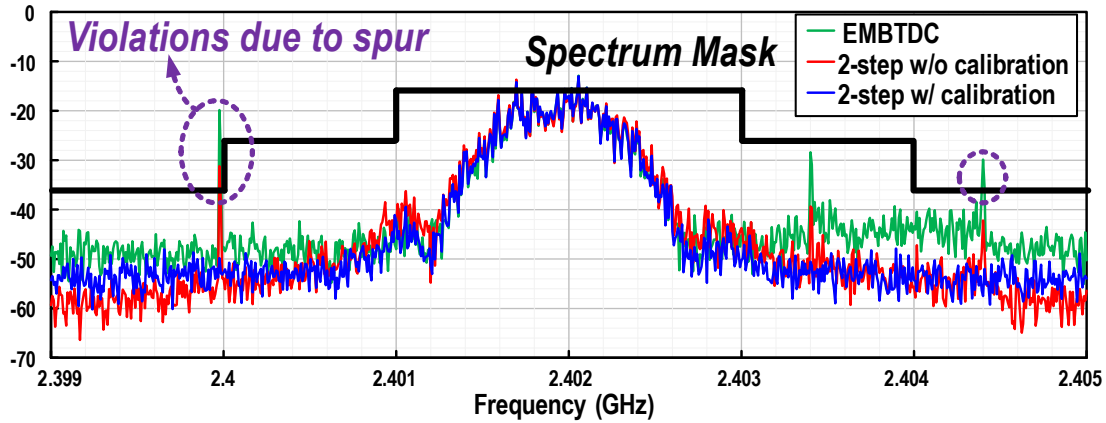


(c)

Figure 3.21 (a) Measured worst-case fractional spurs and RMS jitters for different FCW_FRACs, (b) worst spurs for FOUT=2.40078 GHz for different VDD levels. (c) Simulated fractional spur levels depending on temperature for FOUT=2.40078 GHz.



(a)



(b)

Figure 3.22 Measured BLE performance. (a) frequency vs time plot with eye diagram, (b) GFSK spectrum comparison between different modes.

integer-N spectrum, the largest fractional-spur is improved by 14.3dB from EMBTDC to TSTDC with calibration. The worst-case fractional spurs and integrated jitter (range: 10 kHz ~ 10 MHz) for different FCW_FRACs are shown in Figure 3.21(a). The worst-case spur is still higher than those of [20], [51], mainly due to the limited DLTDC resolution (9.2ps) compared to DTCs (<1ps) and the limited tuning resolution for offset cancellation of each DCO_PH path in PRE-DLTDC, resulting in non-linearity near the coarse step conversions. Figure 3.21(b) shows fractional spur

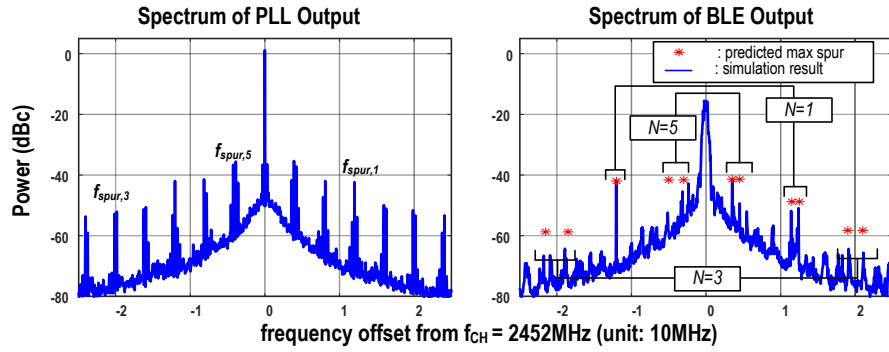
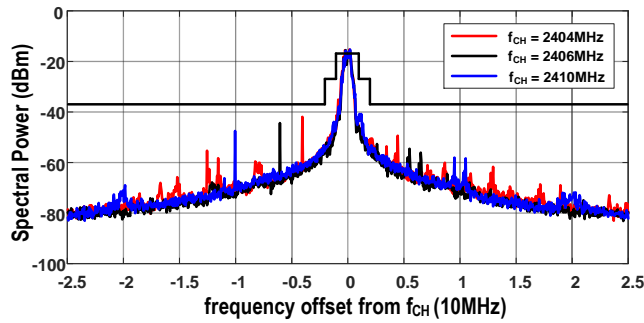
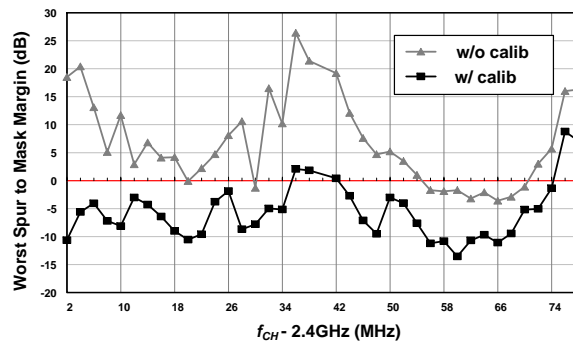


Figure 3.23 Measured PLL spectrum (left) and BLE spectrum (right) for $f_{CH}=2.452$ GHz along with spur prediction from Section II.



(a)



(b)

Figure 3.24 Measured BLE (a) spectrum for 3 different channels, (b) worst case spur margin to the spectral mask across BLE channels w/ and w/o calibration

	THIS WORK	RO-ADPLLs				PLLs for BLE		
		JSSC'22 [30]	JSSC'16 [12]	JSSC'21 [7]	ISSCC'15 [6]	JSSC'19 [23]	JSSC'18 [31]	JSSC'18 [32]
Architecture	2-step TDC	DTC+MDLL	TDC	TPC+MDLL	ILPLL	ADPLL+EC	TDC	DTC+TDC
Synthesizable?	Yes	No	Yes	Yes	Yes	No	No	No
Oscillator type	RO	RO	RO	RO	RO	RO	LC	LC
Standard	BLE	N/A	N/A	N/A	N/A	BLE	BLE	BLE
Process (nm)	12	65	65	22	65	40	28	65
Reference (MHz)	40	50	80	80	380	37.5	40	26
Output Freq. Range (GHz)	1.8~2.7	0.8-2.0	0.6-1.7	1.2-3.8	0.8-1.7	N/A	2.05-2.55	2.0-2.8
Meas. Output	2.4006	1.5195	2	3.6175	1.5222	2.402	2.44	2.442
N	60.015	30.39	25.0	45.219	4.006	64.053	61.0	93.923
In-Band PN (dBc/Hz)	-90.2 @40KHz	-104 @100KHz	-103 @4MHz	-98.2 @100KHz	-95 @10KHz	-85 @1MHz	-106 @100KHz	103.7 @100KHz
In-Band PN _n ^{*1} (dBc/Hz)	-90.2 @40KHz	-100.0 @100KHz	-101.4 @4MHz	-101.8 @100KHz	-91 @10KHz	-85.0 @1MHz	-106.1 @100KHz	-103.8 @100KHz
Reference Spur (dBc)	-45.6	-44	N/A	-53	N/A	-55	-78	-72
Power (mW)	3.91	11.95	10.8	3.19	3	0.34	0.5	0.98
FoM ^{*2}	-220.7	-224.8	-219.7	-226.3	-224.2	-208.5	-239.2	-246
FoM _N ^{*3}	-238.5	-239.6	-233.7	-242.9	-230.2	-226.6	-257.1	-265.7
Core Area (mm ²)	0.063	0.18	0.047	0.0052	0.048	0.0166	0.33	0.23

¹ PN_n =normalized phase noise to F_{out} = 2.4006GHz

$$^2 \text{FoM} = 10 \log \left(\sigma_{jitter}^2 \left(\frac{P_{DC}}{1mW} \right) \right)$$

³ FoM_N = 10log (σ_{jitter}² (P_{DC}/1mW) /N) from Megawer, ISSCC'18

Table 3.1 Performance summary and comparison with state-of-the-art fractional-N ADPLLs

and reference spur for different VDDs for F_{OUT}=2.40078 GHz. The LUT is generated for ΔVDD=0 and the supply voltage is tweaked during the operation to investigate the supply sensitivity of the spurious tones. It shows 4 dB measured maximum spur degradation for ΔVDD= -60mV ~ 60mV. Figure 3.21(c) shows the simulated fractional spur levels for different temperatures. Delay difference of the two paths of DLTDC changes by 4% when the temperature rises from 25°C to 150°C, degrading the fractional spur by 5.33dB. The ADPLL consumes 3.91mW with 0.8V supply voltage and the measured PLL FoM is -220.7 dB. Figure 3.22 shows the BLE packet transmission performance in the advertising channel #37 (f_{CH}=2.402 GHz). The measured transient response of the DCO frequency during a packet transmission demonstrates that the frequency drift is below the BLE spec of 50 kHz. The eye-diagram of two transmitted packets demonstrates an eye opening

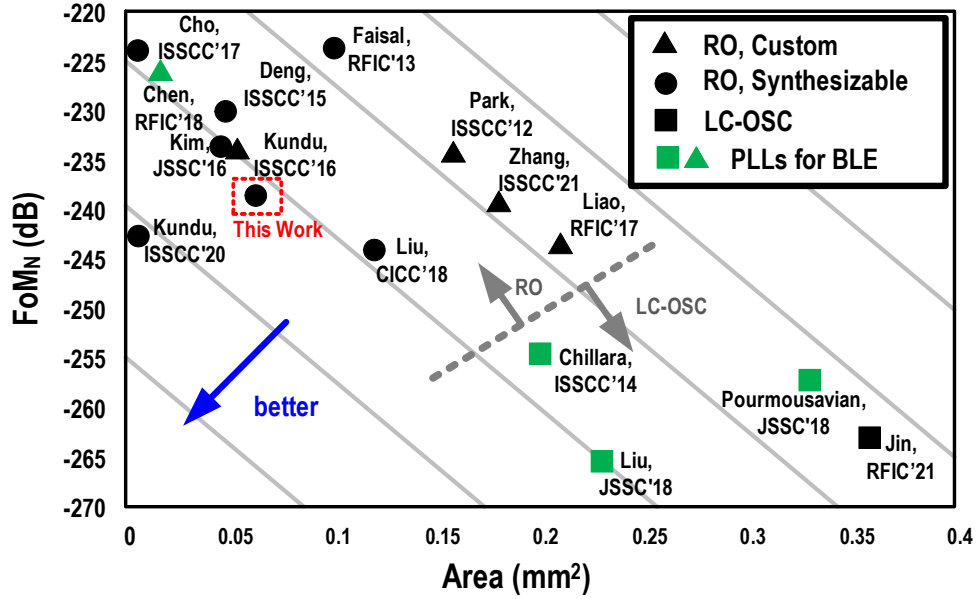


Figure 3.25 FoM_N and Area comparison

greater than 370 kHz. Figure 3.23 shows the measured PLL and BLE output spectrums at $f_{CH} = 2.452\text{GHz}$. The model derived from Section II accurately predicts the spur positions and the worst-case amplitudes for $N=1$ and 5. Figure 3.24(a) shows BLE spectrum for 3 different channels and Figure 3.24(b) shows the worst fractional spur margin to the spectral mask measured across the BLE channels. Without calibration, the mask is violated for 29 channels while calibration reduced it to 5, showing average of 12.48 dB improvement in the worst spurs. A comparison table with state-of-the-art fractional-N ADPLLs is shown in Table I. The PLL has a relatively small total area of 0.063mm^2 compared to other ADPLLs for BLE thanks to finFET technology and the RO-based architecture. It also achieves a comparable FoM_N with state-of-the-art RO-based fractional-N ADPLL/MDLLs and is integrated for the first time in a BLE-TX among synthesizable PLLs. The proposed PLL has competitive FoM_N and area trade-off as shown in Figure 3.25.

3.5 Conclusion

In this Chapter, we predict the positions and amplitudes of BLE spurious tones originated from PLL fractional spurs by investigating the equations of a FSK signal when a PNN is present. With reasonable assumptions, one term of FSK signal can be approximated to the result of mixing between the main modulated signal and a modulated spurious tone. PNN's odd harmonics greater than 1 result in only 6.02 dB attenuation compared to the original spur amplitude, and a 0 dB attenuation for the first harmonic. The model is compared with simulation results and shows accurate prediction for positions and worst-case amplitudes of the fractional spurs. Based on the prediction, we propose a spectral mask for the PLL fractional spurs to satisfy that of the BLE.

A fully synthesizable ADPLL based BLE-TX is proposed that meets the derived spectral mask. The highly automated design flow is open source so that it can be ported to other technology with minimal effort. A novel TSTDC and calibration scheme is proposed as a solution for increased fractional spur due to P&R-ed layout to achieve both BLE standard requirements and fast design time. The combination of EMBTDC and DLTDC improves the time resolution of EMBTDC and reduces the input time range of the DLTDC. By measuring each EMBTDC step with the DLTDC and using the resulting LUT as a decoder, delay mismatches in the DCO have been compensated. The all-digital BLE-TX is fabricated in 12nm and the proposed technique reduced the fractional spur by 14.3dB, playing a critical role in meeting the BLE standard requirements. The standalone PLL consumes 3.91mW in 2.4006GHz, achieving a FoM_N of -238.5 dB with $f_{ref} = 40\text{MHz}$.

Chapter 4 Sub-400fs Low-Jitter Ring Oscillator Based Fractional-N MDLL with Reference Triggered Ring Oscillator

4.1 Introduction

Research in fractional-N frequency synthesizers has been surging due to the stringent jitter and spurious tone requirements for applications such as 5G and high-speed wireline communication systems. DTCs are widely used for cancelling the deterministic time error between reference and output clocks in fractional-N frequency synthesizers for its superior time resolution compared to TDCs [20], [52], [53]. Since the non-linearity and noise increase with the delay range of a DTC, various range reduction techniques have been proposed to reduce this range, and thus reducing the in-band phase noise and fractional spurs [54], [55]. [54] used both positive and negative edges of the reference clock to reduce the DTC range (DR) by 1/2. But for a 3 GHz output clock, the required DR of 167ps is still high, rendering it very difficult to achieve both high resolution and linearity. [55] reduced the DR by 1/8 by utilizing the DCO's 8 output phases as coarse DTC steps. While the embedded nature of this technique prevents the need for gain calibration, it still requires calibration for the delay mismatches between DCO phases, increasing design complexity and requiring extra hardware. In addition, the DR reduction is limited by the number of stages of the DCO.

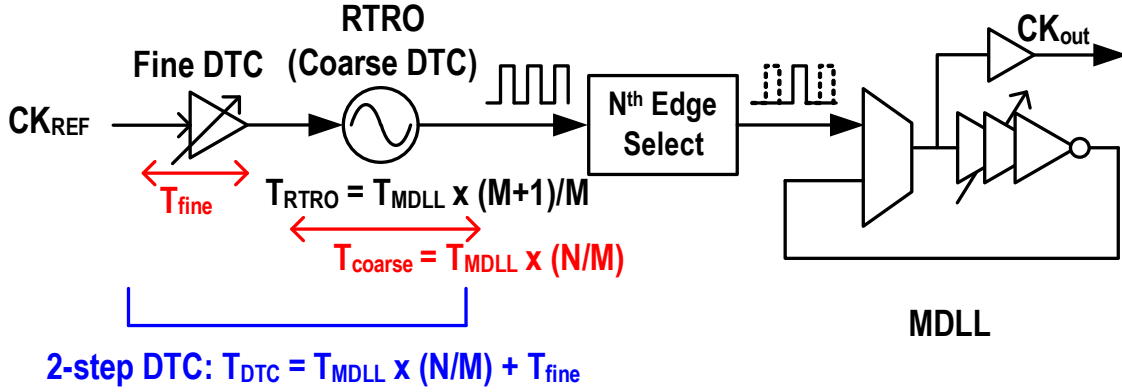


Figure 4.1 Proposed 2-step DTC using RTRO

In this work, we use RTRO [56] as a coarse DTC to achieve DR reduction of 1/10 by setting RTRO period to $(11/10) \cdot T_{MDLL}$, where T_{MDLL} is a period of MDLL. While the time resolution achieved by only using the RTRO can be improved by reducing the period offset between T_{RTRO} and T_{MDLL} , evaluation of input time error requires larger number of RTRO cycles [56], resulting in higher loop latency. To mitigate the dependency between loop latency and the time resolution, this work uses a fine DTC in addition to the coarse RTRO-based DTC. By setting the resolution with a separate fine DTC, high resolution and small loop latency can be achieved at the same time.

Figure 4.1. illustrates the concept of the proposed 2-step DTC. Reference clock CK_{REF} is first delayed by a switched capacitor based fine-DTC by T_{fine} and triggers the RTRO. When $T_{RTRO} = T_{MDLL} \cdot (M+1)/M$, Nth cycle of the RTRO then effectively generates a delay of $T_{MDLL} \cdot (N/M)$, denoted as T_{coarse} . Therefore, injecting the Nth edge of RTRO to the main MDLL effectively delays the relative position of CK_{REF} 's rising edge to that of CK_{OUT} by $T_{fine} + T_{coarse}$. The fine DTC only needs to cover one coarse step T_{MDLL}/M , where M is a programmable value.

A prototype design was designed in 65nm CMOS process and consumes 10.7 mW with 0.294 mm^2 active area. The simulated MDLL shows 239.9fs RMS jitter at 3.0001 GHz with a 50

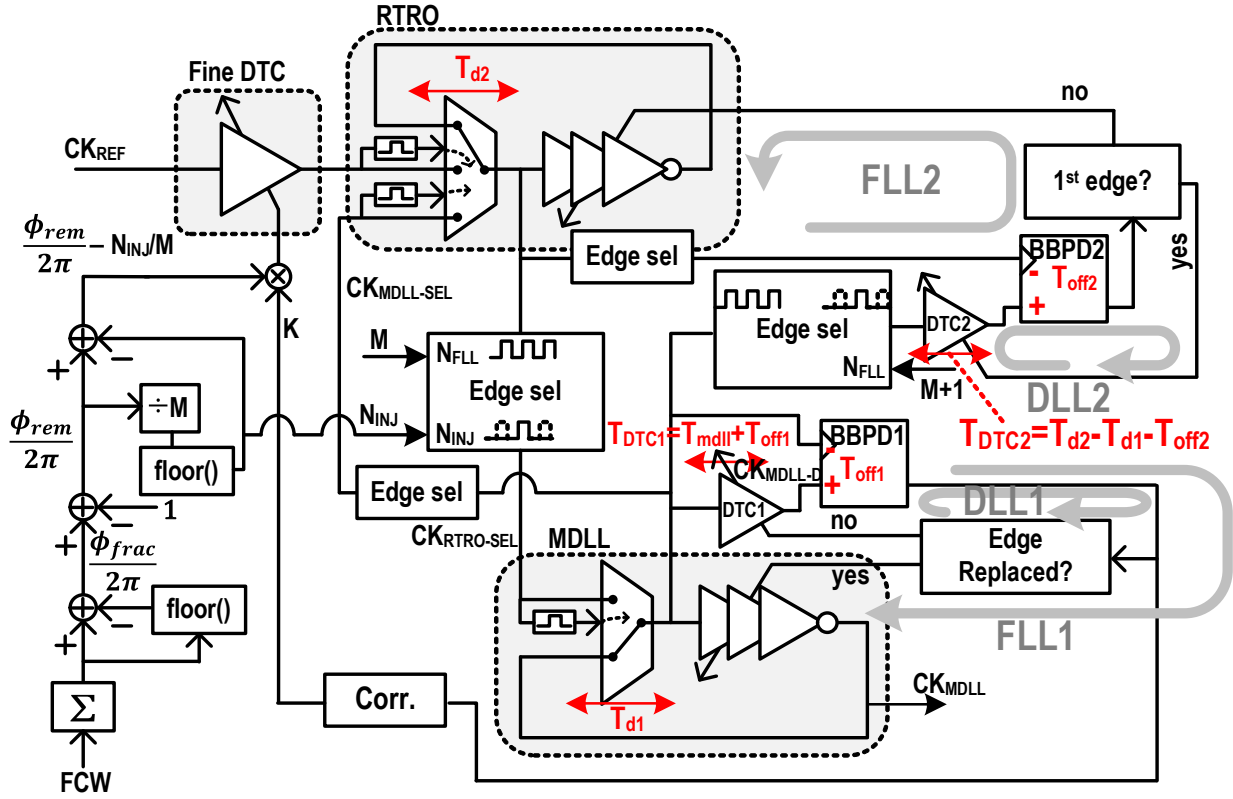


Figure 4.2 Block diagram of the proposed fractional-N MDLL

MHz reference clock, achieving -242.1 dB FoM. The worst case fractional spur and reference spur are -69.3 dBc and -56.7 dBc, respectively.

4.2 Proposed Fractional-N MDLL

A block diagram of the proposed fractional-N MDLL is shown in Figure 4.2. The 2-step DTC's objective is to delay CK_{REF} edge relative to the CK_{MDLL} edge, by $\phi_{rem} = 2\pi - \phi_{frac}$ of the MDLL's period. ϕ_{frac} is the fractional component of the accumulated FCW, which is the ideal position of the CK_{REF} edge relative to the CK_{MDLL} edge. This ensures that the output of the 2-step DTC aligns with the next immediate rising edge of CK_{MDLL} . The RTRO delays CK_{REF} by $\text{floor}(\phi_{rem}/(2\pi/M)) \cdot 2\pi/M$, where $\text{floor}()$ is a flooring function and $2\pi/M$ is the phase resolution of

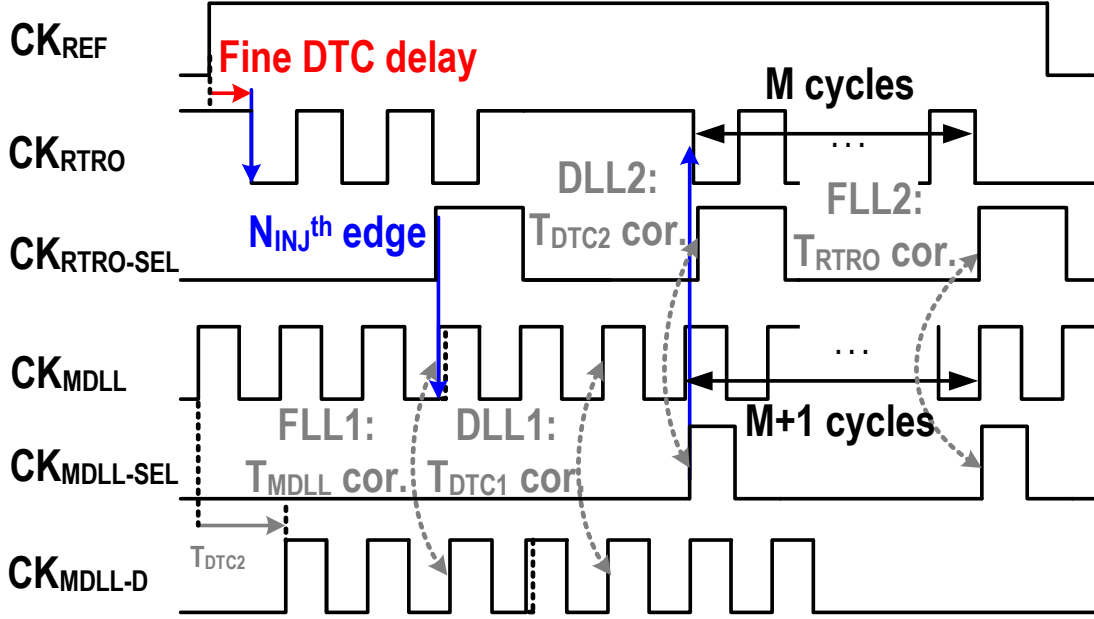


Figure 4.3 Timing diagram of major signals for edge replacement, FLL1/2, DLL1/2.

the coarse DTC when $T_{RTRO} = T_{MDLL} \cdot (M+1)/M$. This delay is realized by selectively passing the N_{INJ}^{th} edge of the RTRO through the edge selection block once CK_{REF} triggers the oscillator, where $N_{INJ} = \text{floor}(\phi_{rem} \cdot M/2\pi)$. The remaining delay $\phi_{rem} - 2\pi \cdot N_{INJ}/M$ is generated by the fine DTC. Therefore, the DTC only needs to cover T_{MDLL}/M . The edge selection blocks work as a mediator between the RTRO and MDLL, selectively passing the edges required for edge replacement and FLL operation. The period error of the MDLL or RTRO introduces periodic disturbance of the output phase, resulting in increased reference spurs or fractional spurs. Therefore, delay and frequency control loops are used to cancel period errors.

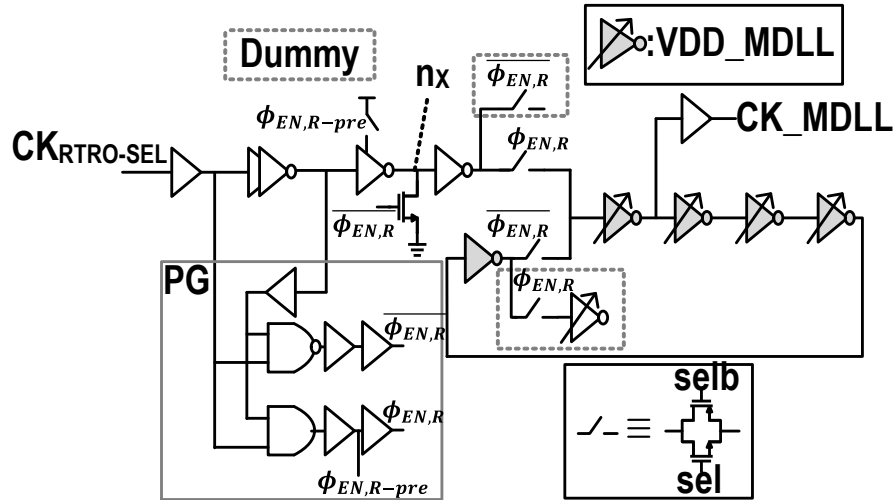
FLL1 and DLL1 from Figure 4.2 regulate T_{MDLL} and T_{DTC1} , respectively. The latter compensates for the BBPD1 offset. The operation in the time domain is shown in Figure 4.3. The MDLL frequency error is detected by a time-period comparator method shown in [20]. If CK_{MDLL-D} is delayed by $T_{MDLL} + T_{off1}$ from CK_{MDLL} , where T_{off1} is the offset of BBPD1, the BBPD1 output at the incident of edge replacement represents the polarity of MDLL period error. FLL1 uses this

output to tune the MDLL frequency in the corresponding direction. This method can only correct a phase error in the range of $-\pi < \phi_{err} < \pi$. Therefore, a coarse FLL that compares the frequency of the divided CK_{MDLL} and CK_{REF} [57] is used to bring T_{MDLL} into this range. DLL1 then uses the BBPD1 output after the edge replacement to control T_{DTC1} , which compensates the time offset of BBPD1.

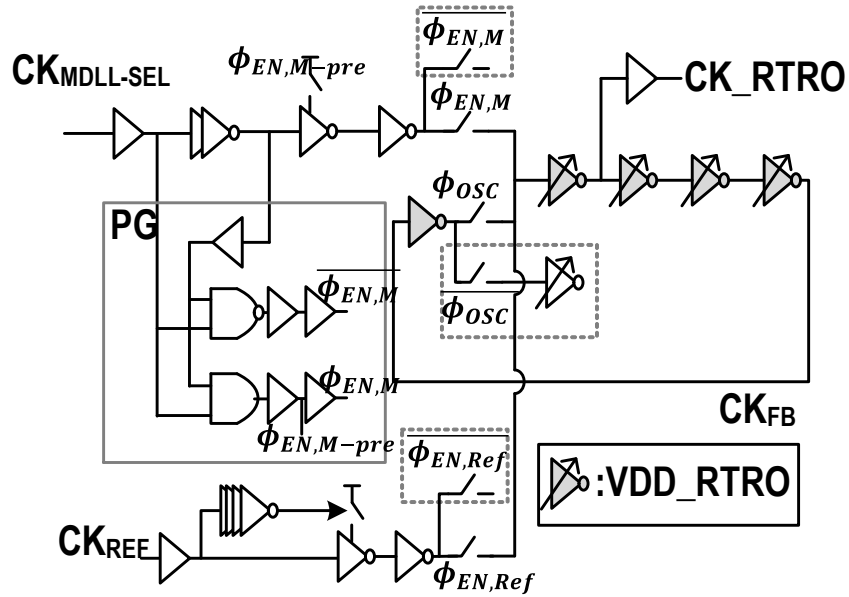
After the operation of FLL1 and DLL1, FLL2 and DLL2 work to control T_{RTRO} and T_{DTC2} , respectively. The effective time offset of FLL2, depicted as $T_{d2}-T_{d1}-T_{off2}$ in Figure 4.2, is cancelled by DTC2, where T_{off2} is the time offset of BBPD2, T_{d1} and T_{d2} are the multiplexer delays of MDLL and RTRO, respectively. A rising edge of CK_{MDLL} is passed through the edge selection block and triggers the RTRO to correct T_{RTRO} . BBPD2 then compares the arrival time of $CK_{MDLL-SEL}$ and $CK_{RTRO-SEL}$, which are the outputs of edge selection blocks that take CK_{MDLL} and CK_{RTRO} as inputs. The MDLL edge selection block selectively passes two edges that are $M+1$ cycle apart, while the 1st and M^{th} edge of CK_{RTRO} is passed after $CK_{MDLL-SEL}$ triggers RTRO. The 1st BBPD2 output is used to correct T_{DTC2} for offset cancellation, while the 2nd output is used for T_{RTRO} correction, since $M+1$ cycle of T_{MDLL} is being compared with M cycle of T_{RTRO} in the latter edge.

The four control loops, two FLLs and two DLLs, have one solution. Therefore, they converge to the desired period and delay values as long as the solution is in the range of each block. MDLL and RTRO are controlled by 5-to-1 bit dithering to improve effective delay resolution. Each loop is controlled by a single integral path with programmable gain.

The Fine DTC gain is calibrated with a least-mean square (LMS) algorithm that correlates the BBPD2 output with $\phi_{rem} - 2\pi N_{INJ}/M$.



(a)



(b)

Figure 4.4 Block diagram of (a) main MDLL and (b) RTRO.

4.3 Circuit Implementation

Figure 4.4(a) and (b) show the detailed block diagram of the MDLL and RTRO, respectively. Frequency control of the two oscillators is done digitally using switched inverters for

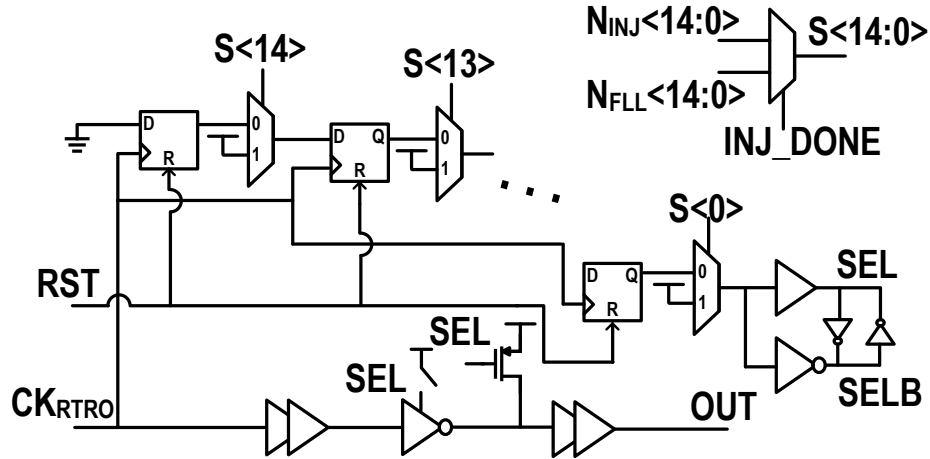


Figure 4.5 Block diagram of RTR0 edge selection block.

coarse control and switched capacitor for fine control. The enable signals $\phi_{EN,M}$ and $\phi_{EN,R}$ are pulses triggered by the incoming edges and only the negative edges are used for both edge replacements. To prevent clock feedthrough on the positive edge of $CK_{RTR0-SEL}$ into CK_{MDLL} , the gate voltage of the edge replacing inverter, noted as n_x in Figure 4.5(a), is pulled-down by a weak NMOS after the edge is replaced so that the voltage transition is slow. Since the varying load can lead to periodic supply ripple due to irregular charge-dump and increase the reference spur, dummy transmission gates are placed in parallel to the main transmission gates. The varying load can lead to periodic supply ripple due to irregular charge-dump and increase the reference spur. The 2 oscillators in MDLL and RTR0 each have separate supply nodes to isolate from the supply ripple on a rising reference clock. The RTR0 is first triggered by CK_{REF} and replaces the MDLL's edge with $CK_{RTR0-SEL}$. After the replacement, the RTR0 stays idle until it is triggered by $CK_{MDLL-SEL}$ for frequency correction of RTR0 as explained in Section 4.2.

Figure 4.5 shows the block diagram of the edge selection block for CK_{RTR0} . One-hot signals $N_{INJ}<14:0>$ and $N_{FLL}<14:0>$ have the information of N and $M+1$ from Figure 4.1, respectively. Therefore, $N_{INJ}<14:0>$ is selected as selection signal $S<14:0>$ for edge replacement,

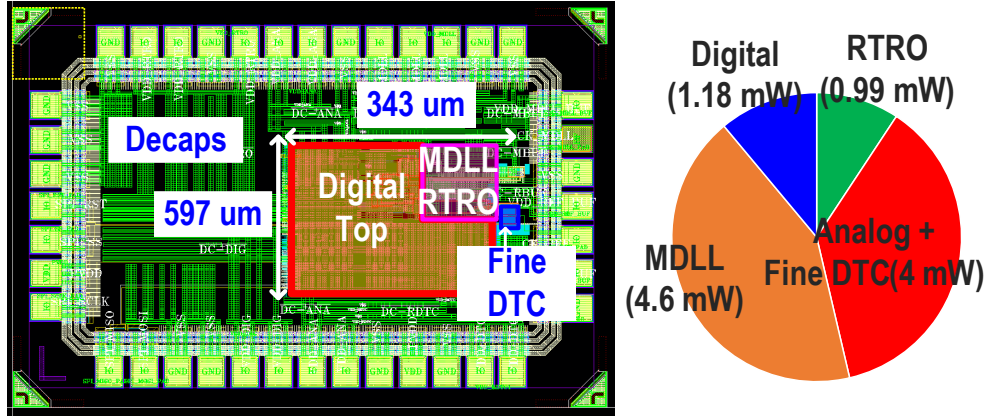


Figure 4.6 Layout (left) and simulated power breakdown (right) of the proposed design.

while $N_{FLL}<14:0>$ is selected for FLL operation after the replacement. The one “high” bit of $S<14:0>$ is shifted every input clock cycle until it reaches SEL, which passes the incoming edge to $CK_{RTRO-SEL}$. The enable signal SEL stays high for one CK_{RTRO} cycle and returns to low and the flip flops get reset once the edge replacement is done. Then the RTRO goes to an idle state and the SEL signal is set to high to pass the first edge of CK_{RTRO} that is triggered by $CK_{MDLL-SEL}$ for FLL operation. After the first edge is passed, $N_{FLL}<14:0>$ selects the $(M+1)^{th}$ edge to be passed next.

4.4 Result and Discussion

A prototype was designed in a 65nm CMOS process, and its performance was evaluated with simulations. Top level layout and power breakdown are shown in Figure 4.6. The active area is 0.294 mm² and total power consumption during fractional-N steady state operation is 10.68 mW. The verilog models of the block-level performances were characterized from transistor level post-parasitic simulation. The models were used in the top-level verilog simulation with the control logics. Key metrics that were used to characterize the block level models are shown in Table 4.1. σ_{T_d} represents period jitter for the MDLL and RTRO and edge-to-edge delay jitter for the DTCs.

	MDLL	RTRO	fine DTC	DTC1/2
σ_{T_d} (fs)	14.5	16.1	126.8	41.3
Pwr (mW)	4.55	4.04	1.26	1.56
ΔT_d (ps)	0.0095	0.0224	0.125	0.113
ΔT_{inj} (ps)	0.22	N/A	N/A	N/A
peak INL(LSB)	N/A	N/A	0.66	N/A

Table 4.1 Block specifications from transistor-level post parasitic simulation.

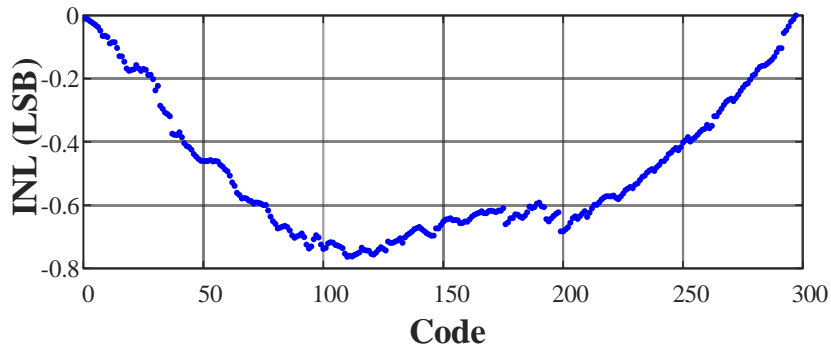


Figure 4.7 Simulated INL of the fine DTC.

Power of the MDLL and RTRO are reported during free-running oscillation at their target frequency. The power consumed by the RTRO during fractional-N operation is 0.99mW, less than the value shown in TABLE I, because oscillation is enabled roughly 1/4th of the time on average for coarse delay generation and FLL operation. ΔT_d is the fine delay resolution of the blocks. ΔT_{inj} is the period difference of the MDLL at the event of edge replacement. This is mainly caused by the slope difference of the replaced edge and the oscillating edge. The fine DTC architecture is similar to the DTC proposed in [57], but without the coarse bank and the replica DTC that prevents the code dependent supply ripple. Instead, we separated the supply of the first and second inverter of the DTC to reduce the non-linearity from the ripple. The fine DTC has a 298-bit thermometer-

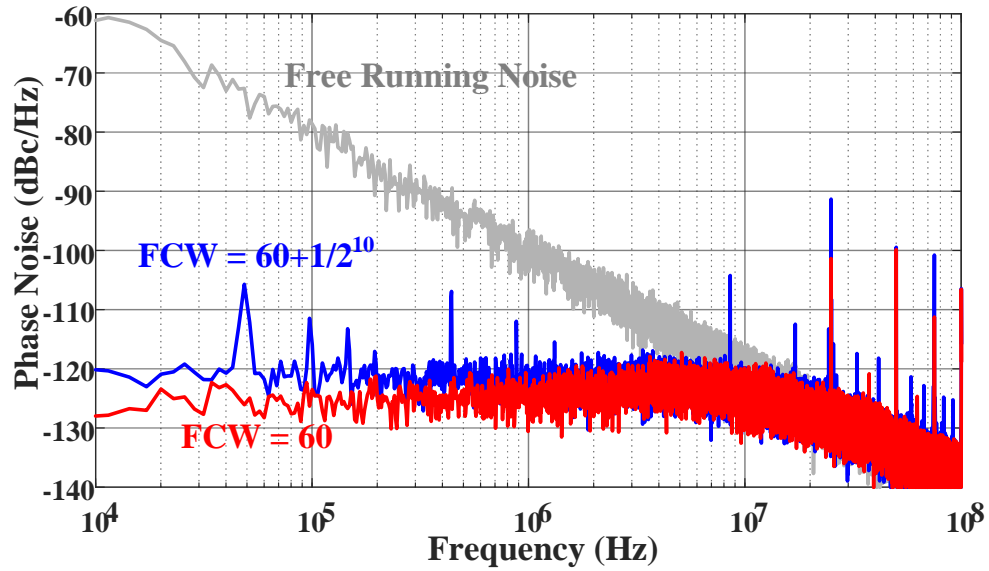


Figure 4.8 Simulated phase noise performance of fractional-N operation (blue) and integer-N operation (red) and free-running mode (gray).

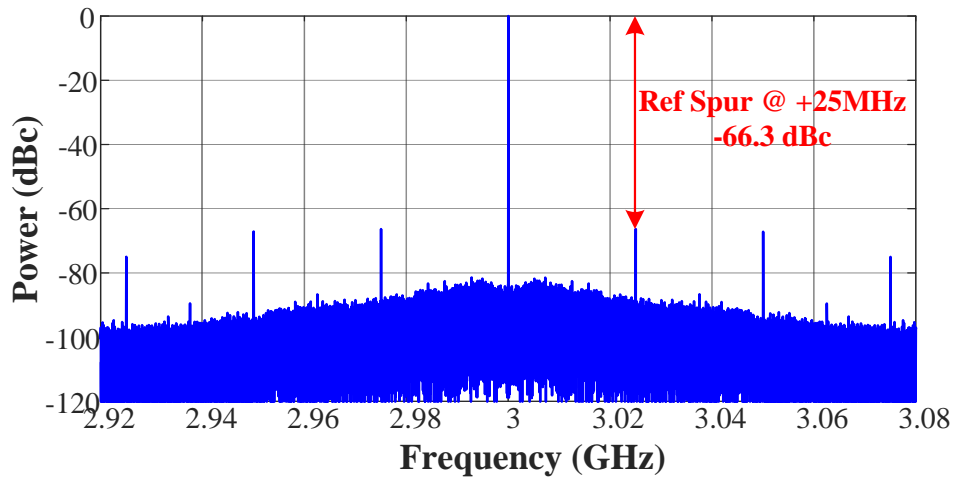
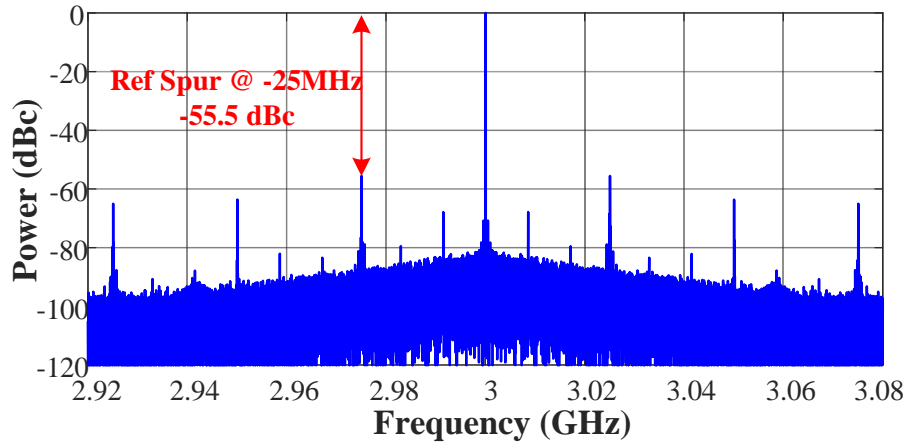


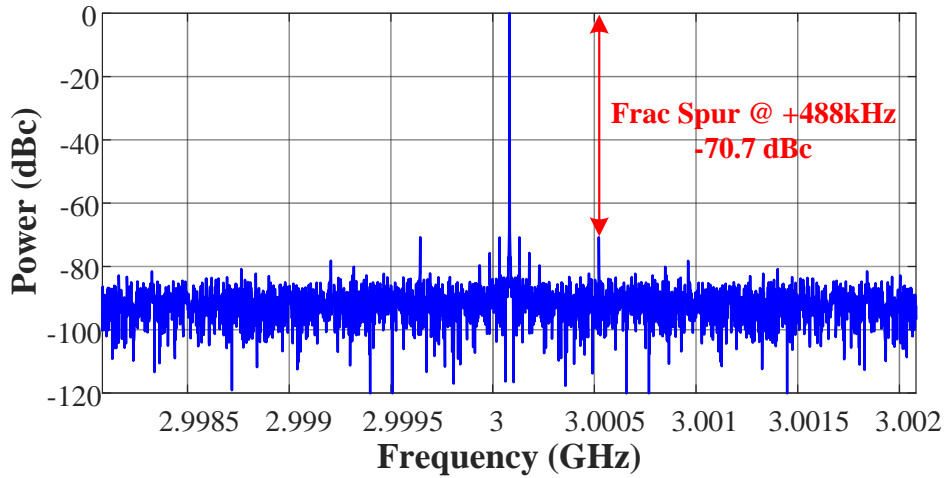
Figure 4.9 Output spectrum of MDLL when FCW is 60.

coded control word and covers a delay range of 38ps. As shown in Figure 4.7, The peak INL value is 0.76 LSB, where the average LSB is 125 fs.

A 50 MHz reference frequency is used to generate an output frequency of 3.0 GHz. Simulated output phase noise plot for FCW 60 and 60+1/210 are shown in Figure 4.7. The



(a)



(b)

Figure 4.10 Output spectrum of MDLL when FCW is $60+1/2^{10}$.

integrated jitter values are 136.9 fs and 141.3 fs (10 kHz ~ 100 MHz), respectively. The in-band noise level of the fractional-N operation is higher due to the added noise from the RTRO and fine DTC. Output spectrums for two FCWs are shown in Figure 4.9 and Figure 4.10, respectively. The worst reference spur for integer-N operation is -66.3 dBc at 25 MHz offset, which is mainly due to the limit cycle of the FLLs and DLLs. A reference spur at 50 MHz offset is due to ΔT_{inj} . The

	¹ This work	W. Wu JSSC'21	H. Park JSSC'22	A. Santiccioli JSSC'19
DR Reduction Technique	Cyclic DTC	VCO Double Edge	RO Phase Selector	VCO Double Edge
DR Reduction Factor	10	2	8	2
Requires NL cal.?	No	Yes	Yes	Yes
Fine DTC Tres (fs)	125	330	200	² N/S
Fine DTC range (ps)	38	320	70	N/S
f _{REF} (MHz)	50	76.8	100	100
f _{OUT} (GHz)	3	3.1	5.2	3.05
Worst frac.spur (dBc)	-69.3	-72	-59	-54.8
Ref. spur (dBc)	-56.7	-69.6	-64	-54.5
rms jitter (fs)	239.9 (10k-100M)	83.4 (10k-100M)	188 (1k-30M)	376 (30k-30M)
Power (mW)	10.68	14.2	15.67	3.35
³ FoM _{jitter} (dB)	-242.1	-250.1	-242.6	-243.2
⁴ FoM _{jitter,N} (dB)	-259.9	-266.1	-259.8	-258.0
Area (mm ²)	0.294	0.31	0.139	0.0275
Oscillator type	RO	LC	RO	RO

¹This performances are based on VHDL simulation with block specs extracted from post-layout transistor-level simulation

²Not shown ³FoM_{jitter} = $10 \log(\sigma_t^2) + 10 \log\left(\frac{PWR}{1mW}\right)$ ⁴FoM_{jitter,N} = FoM_{jitter} - 10log(N)

Table 4.2 Comparison with recent frequency synthesizers with DTC range (DR) reduction technique.

simulated worst-case fractional spur for the FCW of $60+1/2^{10}$ is -70.7 dBc. The low fractional spur was achievable thanks to the intrinsically linear RTRO-based coarse DTC and the reduced delay range of the fine DTC.

Table 4.2 shows the performance comparison with the state-of-the-art DTC-based fractional-N frequency synthesizers. The proposed 2-step DTC achieves highest DR reduction and thus improving linearity of the fine DTC. Also, the cyclic nature is inherently linear, preventing the need of non-linearity calibration for the coarse DTC.

4.5 Conclusion

In this chapter, we introduce a novel DTC range reduction technique using an RTRO as a cyclic coarse DTC. The period offset between the RTRO and MDLL is the coarse resolution, where each cycle adds linear delay to the reference clock without requiring non-linearity calibration. Hence the required fine DTC range is reduced by 10x, contributing to low fractional spur of the proposed fractional-N MDLL. The concept was proven through VHDL simulation with block level models characterized from post-layout simulation. The simulated MDLL achieves a worst-case fractional spur of -69.3 dBc with 239.9 fs jitter.

Chapter 5 Conclusion

5.1 Summary of Contributions

This dissertation proposes design automation methodology for ADPLL and RO-based digital frequency synthesizers for various applications. Two synthesizable PLLs designed by synthesis and PnR tools are presented along with analysis and measurement results. A custom designed all-digital MDLL, which can potentially be built by PnR tool, has been shown for low-jitter performance.

In chapter 2, design automation flow for synthesizable ADPLL is introduced along with a calibration-free feedforward technique. A physics-based model is used to optimize the simulation time, which abstracts PDK/aux-cell characteristic into few constants that are automatically extracted from 3 sets of simulations. The model shows error rate less than 2.2% for 125 designs and 8 PLL examples show that the resulting specs satisfy the input requirements. Edge-selecting feedforward scheme, which is to select the closest edge among 32 interpolated DCO phases to the reference clock to instantly cancel existing jitter, is demonstrated as part of the synthesizable PLL and the effect is shown in measurement results. The technique reduced RMS jitter by 4.04x when the DCO random noise dominates the in-band noise, while it increases it by 1.8x when the TDC noise dominates. By frequency domain analysis, we derived the condition of which the technique can be beneficial and proven by behavioral simulation result.

Chapter 3 demonstrates a two-step TDC based synthesizable PLL for BLE-Tx. The proposed two-step TDC and calibration scheme improved the TDC resolution and compensated non-linearity induced by PnR tool. This reduced the output fractional spur of both PLL and BLE, enabling it to meet the standard requirements. The 1.8-2.7 GHz PLL consumed 3.91 mW at 2.4006 GHz, achieving FoM of -220.7dB. The proposed calibration scheme reduced the worst-case fractional spur by 12.48 dB in average over all BLE channels. Also to demystify the relationship between PLL fractional spur and the resulting spurs in BLE spectrum, we derive the relative position and amplitude change for each fractional spur harmonic. This sets a clear linearity target for PLL designers for FSK applications.

In chapter 4, a 3.25 GHz fractional-N MDLL has been proposed which uses RTRO as a coarse DTC of 2-step DTC. The architecture reduces fine DTC range by a programmable factor, which typical value is 9, while alleviating the trade-off between latency and resolution when RTRO is used alone. Fabricated chip achieved 325 fs jitter in integer-N operation, while the reference spur is relatively high compared to state-of-the-art frequency synthesizers. Simulation suggests that supply and ground inductances over 500 pH and 100 pH each can lead to the reference spur due to the sudden change in current drain at the incident of edge replacement. The lesson is that such architecture requires careful layout and packaging that can optimize supply parasitic (e.g., ball grid array (BGA) packaging).

5.2 Future Work

In this section, we discuss remaining problems and possible solutions to broaden the use of cell-based, synthesizable method for analog circuits along with the improvements needed for the fractional-N MDLL proposed in Chapter 4.

In Chapter 3, we proposed a digital calibration method to compensate the routing uncertainties introduced by APR tool. But reducing the routing uncertainty in the first place is a more desired solution for the reduced the power, area and complexity of the digital calibration blocks and for the support of more various architectures. While the APR tool allows a patterned and predictable routing for power net, its ability to do so for signal net is significantly limited. The predictable "analog" type routing for signal nets can be implemented either by an automation wrapper around the APR tool to calculated coordinates and width of the routings and use the power routing commands for the desired signal, or the APR tool to develop a routing option for such purpose. This will improve parasitic resistance and capacitance, linearity of delay or timing control blocks, mismatch of differential circuits without additional cost.

The fractional-N MDLL proposed in Chapter 4 suffers from high reference spur (-45 dBc/Hz without edge replacement) due to power and ground bounce. One of the main sources of the supply ripple is the large reference buffer, which consumes $>1\text{mW}$ to minimize additional jitter, that shares the same ground node with the rest of the circuits. Large instantaneous current ($>10\text{mA}$) drained through the ground causes ripple, which propagates to other blocks. Also the design of output buffer that drives the pad and external devices such as SMA Cable didn't consider the effect of non-ideal supply characteristics. This has led to large supply ripples ($>300\text{mV}$ according to simulation with 500pH and 100pH supply and ground inductances), that can deteriorate the operation of the rest of the blocks. To reduce the effect of supply noise, large buffers that are driving or driven by external components should have isolated grounds. The proposed two-step DTC can be used in other architectures such as fractional-N PLL where a linear delay generation is required.

Appendix

Detailed Derivation for Chapter 3.2.2

This appendix shows the justifications of the approximations that are made in Section II-B with a more generalized expression for FSK modulated signal. A BFSK signal $u(t)$ is defined in [42] as

$$u(t) = A_u \cos(B_n(t)), nT \leq t < (n+1)T, n = 0, 1, 2, \dots$$

(11)

$$B_0(t) = (\alpha + x_1\beta)t + \phi$$

(12)

$$B_n(t) = \alpha t + x_{n+1}\beta(t - nT) + \phi + \beta T \sum_{r=1}^n x_r, n > 0.$$

(13)

where A_u is the voltage amplitude of the signal, n is the current count of the data being transmitted, T is the data duration period, α is the center frequency of FSK in radians/s, β is the frequency deviation for data encoding of FSK in radians/s, x_n is n^{th} data being transmitted (1 or -1) and ϕ is the initial phase of the signal in radians.

Let's investigate how $s(t)$ from (2) changes during BFSK modulation. For $x_n = \pm 1$, f_{frac} changes by $\pm\beta/2\pi$ and therefore Nf_{frac} changes by $\pm N\beta/2\pi$. Thus, the resulting deviation of $f_{spur,N}$ with respect to x_n is

$$\Delta f_{spur,N} = \begin{cases} N\beta/2\pi, & x_n = 1 \\ -N\beta/2\pi, & x_n = -1 \end{cases}$$

(14)

Using (14), we are now able to define the modulated version of $s(t)$, denoted as $s_m(t)$, as

$$s_m(t) = \frac{1}{2}a_0 + \sum_{N=1}^{\infty} a_N \cos(C_{N,n}(t)) + \sum_{N=1}^{\infty} b_N \sin(C_{N,n}(t)),$$

$$nT \leq t < (n+1)T,$$

$$n = 0, 1, 2, \dots$$

(15)

$$C_{N,0}(t) = (\omega_{spur,N} + x_1 N\beta)t + \phi_2$$

(16)

$$C_{N,n}(t) = \omega_{spur,N}t + x_{n+1}N\beta(t - nT) + \phi_2 + N\beta T \sum_{r=1}^n x_r, n > 0$$

(17)

where $\omega_{spur,N} = 2\pi f_{spur,N}$. Since $s_m(t)$ is the frequency nonlinearity from PNN, the resulting phase is the integral of it as follows.

$$\phi_m(t) = 2\pi \int_0^t s_m(t') dt' = \pi a_0 t + \sum_{N=1}^{\infty} \frac{a_N}{f_{spur,N} + x_{n+1}N\beta/2\pi} \sin(C_{N,n}(t))$$

$$+ \sum_{N=1}^{\infty} \frac{b_N}{f_{spur,N} + x_{n+1} N \beta / 2\pi} \cos(C_{N,n}(t)), nT \leq t < (n+1)T, \quad n = 0, 1, 2, \dots$$

(18)

a_0 results in a frequency offset of the main BFSK signal and we assume $a_0 = 0$ since we are interested in the spurious tones not the center frequency drift. Note that $\phi_m(t)$ is composed of sinusoidal waves with amplitude modulation (AM) along with frequency modulation (FM) due to BFSK. By adding $\phi_m(t)$ in the phase term of $u(t)$ of (11), we can express the modulated signal with fractional spurs as

$$\begin{aligned} u_s(t) &= A_u \cos(B_n(t) + \phi_m(t)) \\ &= A_u \{ \cos(B_n(t)) \cos(\phi_m(t)) - \sin(B_n(t)) \sin(\phi_m(t)) \} \end{aligned}$$

(19)

If $\phi_m(t) < 1$ radian, which is true for spurious tones less than -10 dBc, the approximations $\cos(\phi_m(t)) \approx 1$, $\sin(\phi_m(t)) \approx \phi_m(t)$ can be made. Also, if $f_{spur,N} \gg \frac{N\beta}{2\pi}$, $\phi_m(t)$ can be approximated as a constant amplitude signal. With these assumptions, (19) can be approximated as

$$\begin{aligned} u_s(t) &\approx A_u \cos(B_n(t)) - A_u \sin(B_n(t)) \phi_m(t) \\ &\approx A_u \cos(B_n(t)) - \sum_{N=1}^{\infty} \frac{A_u a_N}{f_{spur,N}} \sin(B_n(t)) \cos(C_{N,n}(t)) \\ &\quad - \sum_{N=1}^{\infty} \frac{A_u b_N}{f_{spur,N}} \sin(B_n(t)) \sin(C_{N,n}(t)) \\ &= A_u \cos(B_n(t)) - \sum_{N=1}^{\infty} \frac{A_u}{2f_{spur,N}} \left[a_N \sin(D_{N,n}(t)) - b_N \cos(D_{N,n}(t)) \right] \end{aligned}$$

$$-\sum_{N=1}^{\infty} \frac{A_u}{2f_{spur,N}} \left[a_N \sin(E_{N,n}(t)) + b_N \cos(E_{N,n}(t)) \right],$$

(20)

$$D_{N,n}(t) = B_n(t) + C_{N,n}(t)$$

(21)

$$E_{N,n}(t) = B_n(t) - C_{N,n}(t)$$

(22)

Figure 6.1. shows the simulated average error between (19) and (20) for different spur amplitudes and $f_{spur,N}/(N\beta/2\pi)$ of $s_m(t)$. The smallest $f_{spur,N}/(N\beta/2\pi)$ is 2.6 when $N=3$, $f_{CH} = 14\text{MHz}$ and $f_{ref} = 40\text{MHz}$, so the error plot is shown for $N=3$ to check the worst-case approximation error mainly due to the AM components of (18), which is 4dB. We can observe that most cases when $f_{spur,N}/(N\beta/2\pi) \geq 8$ results in error less than 2dB.

(20) is equivalent to (8) from Section II-B where $\alpha_N' = \frac{A_u a_N}{f_{spur,N}}$. The rest of the derivation in Section

II-B are valid for (20) as well, resulting in the conclusion of (10). Although the complete PSD of $u_s(t)$ should be calculated by deriving the auto-correlation function $R_{u_s}(\tau)$ followed by a Fourier-transform, the maximum spur values of each function are preserved in the final spectrum. This is because the autocorrelation of a sum of different functions includes the autocorrelation of each function with additional terms derived from inter-function correlations. Thus, the dominant value for a certain frequency among individual terms is maintained in the final PSD.

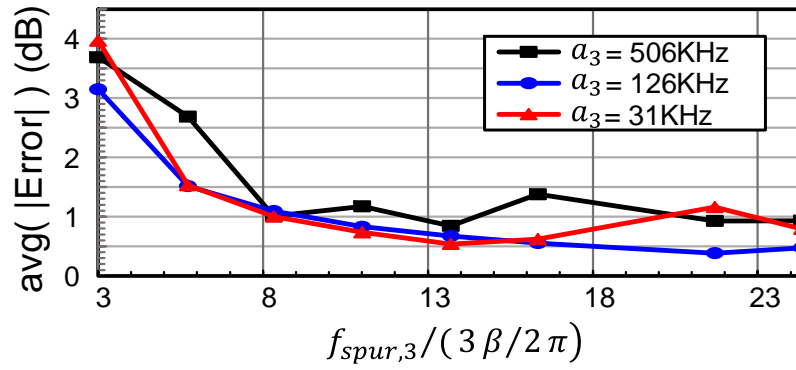


Figure 6.1 Average error between (19) and (20) with respect to $\frac{f_{spur,3}}{\frac{3\beta}{2\pi}}$ ratio for different a_3 .

BIBLIOGRAPHY

- [1] Ed Sperling, “Design Rule Complexity Rising.” <https://semiengineering.com/design-rule-complexity-rising> (accessed Aug. 01, 2023).
- [2] R. O. Topaloglu, “Design with FinFETs: Design rules, patterns, and variability,” in *2013 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2013, pp. 569–571. doi: 10.1109/ICCAD.2013.6691172.
- [3] K. Hakhamaneshi, N. Werblun, P. Abbeel, and V. Stojanović, “Late Breaking Results: Analog Circuit Generator based on Deep Neural Network enhanced Combinatorial optimization,” in *2019 56th ACM/IEEE Design Automation Conference (DAC)*, 2019, pp. 1–2.
- [4] M. B. Alawieh, S. A. Williamson, and D. Z. Pan, “Rethinking Sparsity in Performance Modeling for Analog and Mixed Circuits using Spike and Slab Models,” in *2019 56th ACM/IEEE Design Automation Conference (DAC)*, 2019, pp. 1–6.
- [5] T. Ajayi *et al.*, “An open-source framework for autonomous SoC design with analog block generation,” in *2020 IFIP/IEEE 28th International Conference on Very Large Scale Integration (VLSI-SOC)*, IEEE, 2020, pp. 141–146.
- [6] Y. K. Cherivirala and D. D. Wentloff, “A Capacitor-less Digital LDO Regulator with Synthesizable PID Controller Achieving 99.75% Efficiency and 93.3ps Response Time in 65nm,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2023.

- [7] Y. K. Cherivirala, M. Saligane, and D. Wentzloff, "An Open Source Compatible Framework to Fully Autonomous Digital LDO Generation," in *2023 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2023.
- [8] S. Kamineni, S. Gupta, and B. H. Calhoun, "MemGen: An Open-Source Framework for Autonomous Generation of Memory Macros," in *2021 IEEE Custom Integrated Circuits Conference (CICC)*, 2021, pp. 1–2. doi: 10.1109/CICC51472.2021.9431501.
- [9] S. Kamineni, A. Sharma, R. Harjani, S. S. Sapatnekar, and B. H. Calhoun, "AuxcellGen: A Framework for Autonomous Generation of Analog and Memory Unit Cells," in *2023 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2023, pp. 1–6. doi: 10.23919/DATE56975.2023.10137270.
- [10] K. Kwon and D. Wentzloff, "Synthesizable ADPLL Generator: From Specification to GDS," in *International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design*, Jul. 2023.
- [11] Y. Park and D. D. Wentzloff, "An all-digital PLL synthesized from a digital standard cell library in 65nm CMOS," in *Proceedings of the Custom Integrated Circuits Conference*, 2011. doi: 10.1109/CICC.2011.6055347.
- [12] M. Faisal and D. D. Wentzloff, "An automatically placed-and-routed ADPLL for the medradio band using PWM to enhance DCO resolution," in *Digest of Papers - IEEE Radio Frequency Integrated Circuits Symposium*, 2013, pp. 115–118. doi: 10.1109/RFIC.2013.6569537.
- [13] W. Deng *et al.*, "A 0.048mm² 3mW synthesizable fractional-N PLL with a soft injection-locking technique," in *Digest of Technical Papers - IEEE International Solid-State Circuits*

- Conference*, Institute of Electrical and Electronics Engineers Inc., Mar. 2015, pp. 252–253. doi: 10.1109/ISSCC.2015.7063021.
- [14] B. Liu *et al.*, “An HDL-described fully-synthesizable sub-GHz IoT transceiver with ring oscillator based frequency synthesizer and digital background EVM calibration,” in *2019 IEEE Custom Integrated Circuits Conference (CICC)*, IEEE, 2019, pp. 1–4.
- [15] B. Liu *et al.*, “An HDL-described fully-synthesizable sub-GHz IoT transceiver with ring oscillator based frequency synthesizer and digital background EVM calibration,” in *2019 IEEE Custom Integrated Circuits Conference (CICC)*, IEEE, 2019, pp. 1–4.
- [16] B. Liu *et al.*, “A Fully Synthesizable Fractional-N MDLL with Zero-Order Interpolation-Based DTC Nonlinearity Calibration and Two-Step Hybrid Phase Offset Calibration,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 2, pp. 603–616, Feb. 2021, doi: 10.1109/TCSI.2020.3035373.
- [17] B. Liu *et al.*, “A 1.2ps-jitter fully-synthesizable fully-calibrated fractional-N injection-locked PLL using true arbitrary nonlinearity calibration technique,” in *2018 IEEE Custom Integrated Circuits Conference, CICC 2018*, Institute of Electrical and Electronics Engineers Inc., May 2018, pp. 1–4. doi: 10.1109/CICC.2018.8357041.
- [18] B. Liu *et al.*, “A 0.4-ps-Jitter -52-dBc-Spur Synthesizable Injection-Locked PLL with Self-Clocked Nonoverlap Update and Slope-Balanced Subsampling BBPD,” *IEEE Solid State Circuits Lett*, vol. 2, no. 1, pp. 5–8, 2019, doi: 10.1109/LSSC.2019.2910470.
- [19] S. Kundu, L. Chai, K. Chandrashekar, S. Pellerano, and B. Carlton, “25.5 a self-calibrated 1.2-to-3.8 GHz 0.0052 mm² synthesized Fractional-N MDLL using a 2b time-period comparator in 22nm FinFET CMOS,” in *2020 IEEE International Solid-State Circuits Conference-(ISSCC)*, IEEE, 2020, pp. 276–278.

- [20] S. Kundu, L. Chai, K. Chandrashekar, S. Pellerano, and B. R. Carlton, "A Self-Calibrated 2-bit Time-Period Comparator-Based Synthesized Fractional-N MDLL in 22-nm FinFET CMOS," *IEEE J Solid-State Circuits*, vol. 56, no. 1, pp. 43–54, Jan. 2021, doi: 10.1109/JSSC.2020.3021279.
- [21] H. Cho *et al.*, "A 0.0047mm² highly synthesizable TDC- and DCO-less fractional-N PLL with a seamless lock range of f_{REF} to 1GHz," in *Digest of Technical Papers - IEEE International Solid-State Circuits Conference*, Institute of Electrical and Electronics Engineers Inc., Mar. 2017, pp. 154–155. doi: 10.1109/ISSCC.2017.7870307.
- [22] Y. He *et al.*, "An Injection-Locked Ring-Oscillator-Based Fractional-N Digital PLL Supporting BLE Frequency Modulation," *IEEE J Solid-State Circuits*, vol. 57, no. 6, pp. 1765–1775, Jun. 2022, doi: 10.1109/JSSC.2022.3154752.
- [23] B. Liu *et al.*, "A Fully-Synthesizable Fractional-N Injection-Locked PLL for Digital Clocking with Triangle/Sawtooth Spread-Spectrum Modulation Capability in 5-nm CMOS," *IEEE Solid State Circuits Lett*, vol. 3, pp. 34–37, 2020, doi: 10.1109/LSSC.2020.2967744.
- [24] S. Kim *et al.*, "A 2 GHz Synthesized Fractional-N ADPLL With Dual-Referenced Interpolating TDC," *IEEE J Solid-State Circuits*, vol. 51, no. 2, pp. 391–400, Feb. 2016, doi: 10.1109/JSSC.2015.2494365.
- [25] M. Lee, S. Kim, H. J. Park, and J. Y. Sim, "A 0.0043-mm² 0.3-1.2-V frequency-scalable synthesized fractional-N Digital PLL with a speculative dual-referenced interpolating TDC," *IEEE J Solid-State Circuits*, vol. 54, no. 1, pp. 99–108, Jan. 2019, doi: 10.1109/JSSC.2018.2876464.

- [26] H. C. Ngo, K. Nakata, T. Yoshioka, Y. Terashima, K. Okada, and A. Matsuzawa, "A 0.42ps-jitter -241.7dB-FOM synthesizable injection-locked PLL with noise-isolation LDO," in *Digest of Technical Papers - IEEE International Solid-State Circuits Conference*, Institute of Electrical and Electronics Engineers Inc., Mar. 2017, pp. 150–151. doi: 10.1109/ISSCC.2017.7870305.
- [27] K. Kwon, O. Abdelatty, and D. Wentzloff, "Open-Source Fully-Synthesizable ADPLL for a Bluetooth Low-Energy Transmitter in 12nm FinFET Technology," in *2022 IEEE Radio Frequency Integrated Circuits Symposium (RFIC)*, IEEE, 2022, pp. 155–158.
- [28] M. Oveisi and P. Heydari, "A Study of BER and EVM Degradation in Digital Modulation Schemes Due to PLL Jitter and Communication-Link Noise," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 69, no. 8, pp. 3402–3415, 2022, doi: 10.1109/TCSI.2022.3172707.
- [29] A. Santiccioli, C. Samori, A. L. Lacaita, and S. Levantino, "Time-Variant Modeling and Analysis of Multiplying Delay-Locked Loops," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 10, pp. 3775–3785, 2019, doi: 10.1109/TCSI.2019.2918027.
- [30] M. S.-W. Chen, D. Su, and S. Mehta, "A calibration-free 800 MHz fractional-N digital PLL with embedded TDC," *IEEE J Solid-State Circuits*, vol. 45, no. 12, pp. 2819–2827, 2010.
- [31] R. B. Staszewski and P. T. Balsara, "Phase-domain all-digital phase-locked loop," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 52, no. 3, pp. 159–163, 2005.
- [32] D. M. Moore, "Circuits and Techniques for Cell-based Analog Design Automation in Advanced Processes," University of Michigan, 2018.
- [33] M. H. Perrott, M. D. Trott, and C. G. Sodini, "A modeling approach for Δ fractional-N frequency synthesizers allowing straightforward noise analysis," *IEEE*

- J Solid-State Circuits*, vol. 37, no. 8, pp. 1028–1038, 2002, doi: 10.1109/JSSC.2002.800925.
- [34] S. Min, T. Copani, S. Kiaei, and B. Bakkaloglu, “A 90-nm CMOS 5-GHz Ring-Oscillator PLL With Delay-Discriminator-Based Active Phase-Noise Cancellation,” *IEEE Journal of Solid-State Circuits*, vol. 48, no. 5, pp. 1151–1160, May 2013.
- [35] S. S. Nagam and P. R. Kinget, “A –236.3dB FoM sub-sampling low-jitter supply-robust ring-oscillator PLL for clocking applications with feed-forward noise-cancellation,” in *2017 IEEE Custom Integrated Circuits Conference (CICC)*, 2017, pp. 1–4. doi: 10.1109/CICC.2017.7993677.
- [36] K. Kunal *et al.*, “ALIGN: Open-source analog layout automation from the ground up,” in *Proceedings of the 56th Annual Design Automation Conference 2019*, 2019, pp. 1–4.
- [37] E. Chang *et al.*, “BAG2: A process-portable framework for generator-based AMS circuit design,” in *2018 IEEE Custom Integrated Circuits Conference (CICC)*, IEEE, 2018, pp. 1–8.
- [38] K. Kwon, O. A. B. Abdelatty, and D. D. Wentzloff, “PLL Fractional Spur’s Impact on FSK Spectrum and a Synthesizable ADPLL for a Bluetooth Transmitter,” *IEEE J Solid-State Circuits*, pp. 1–14, 2023, doi: 10.1109/JSSC.2023.3236640.
- [39] O. Abdelatty, A. Alghaihab, Y. K. Cherivirala, S. Kamineni, B. Calhoun, and D. D. Wentzloff, “A $300\ \mu\ \mathrm{W}$ Bluetooth-Low-Energy Backchannel Receiver Employing a Discrete-Time Differentiator-Based Coherent GFSK Demodulation,” in *2021 IEEE Radio Frequency Integrated Circuits Symposium (RFIC)*, 2021, pp. 239–242. doi: 10.1109/RFIC51843.2021.9490429.

- [40] Y. Donnelly and M. P. Kennedy, "Prediction of Phase Noise and Spurs in a Nonlinear Fractional- N Frequency Synthesizer," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 11, pp. 4108–4121, 2019.
- [41] F. Bizzarri, A. M. Brambilla, and S. Callegari, "On the mechanisms governing spurious tone injection in fractional PLLs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 11, pp. 1267–1271, 2016.
- [42] W. R. Bennett and S. O. Rice, "Spectral density and autocorrelation functions associated with binary frequency-shift keying," *Bell system technical journal*, vol. 42, no. 5, pp. 2355–2385, 1963.
- [43] P. Dudek, S. Szczepanski, and J. V Hatfield, "A high-resolution CMOS time-to-digital converter utilizing a Vernier delay line," *IEEE J Solid-State Circuits*, vol. 35, no. 2, pp. 240–247, 2000.
- [44] X. Chen *et al.*, "Analysis and design of an ultra-low-power bluetooth low-energy transmitter with ring oscillator-based ADPLL and $4\times$ frequency edge combiner," *IEEE J Solid-State Circuits*, vol. 54, no. 5, pp. 1339–1350, 2019.
- [45] J. S. Walling, S.-M. Yoo, and D. J. Allstot, "Digital power amplifier: A new way to exploit the switched-capacitor circuit," *IEEE communications magazine*, vol. 50, no. 4, pp. 145–151, 2012.
- [46] P. Madoglio *et al.*, "13.6 A 2.4 GHz WLAN digital polar transmitter with synthesized digital-to-time converter in 14nm trigate/FinFET technology for IoT and wearable applications," in *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, IEEE, 2017, pp. 226–227.

- [47] S.-M. Yoo, J. S. Walling, E. C. Woo, B. Jann, and D. J. Allstot, “A switched-capacitor RF power amplifier,” *IEEE J Solid-State Circuits*, vol. 46, no. 12, pp. 2977–2987, 2011.
- [48] “FASoC: Fully-Autonomous SoC Synthesis,” <https://github.com/idea-fasoc/fasoc>.
- [49] “OpenROAD: an integrated chip physical design tool that takes a design from synthesized Verilog to routed layout,” <https://github.com/The-OpenROAD-Project/OpenROAD>.
- [50] “Ngspice: a mixed-level/mixed-signal circuit simulator,” <https://github.com/ngspice/ngspice>.
- [51] Q. Zhang, S. Su, C. R. Ho, and M. S. W. Chen, “A Fractional-N Digital MDLL with Background Two-Point DTC Calibration,” *IEEE J Solid-State Circuits*, vol. 57, no. 1, pp. 80–89, Jan. 2022, doi: 10.1109/JSSC.2021.3098009.
- [52] D. Tasca, M. Zanuso, G. Marzin, S. Levantino, C. Samori, and A. L. Lacaita, “A 2.9–4.0-GHz Fractional-N Digital PLL With Bang-Bang Phase Detector and 560- $\{\rm fs\}_{\rm rms}$ Integrated Jitter at 4.5-mW Power,” *IEEE J Solid-State Circuits*, vol. 46, no. 12, pp. 2745–2758, 2011, doi: 10.1109/JSSC.2011.2162917.
- [53] S. Levantino, G. Marzin, and C. Samori, “An Adaptive Pre-Distortion Technique to Mitigate the DTC Nonlinearity in Digital PLLs,” *IEEE J Solid-State Circuits*, vol. 49, no. 8, pp. 1762–1772, 2014, doi: 10.1109/JSSC.2014.2314436.
- [54] W. Wu *et al.*, “32.2 A 14nm Analog Sampling Fractional-N PLL with a Digital-to-Time Converter Range-Reduction Technique Achieving 80fs Integrated Jitter and 93fs at Near-Integer Channels,” in *2021 IEEE International Solid-State Circuits Conference (ISSCC)*, 2021, pp. 444–446. doi: 10.1109/ISSCC42613.2021.9365850.
- [55] H. Park, C. Hwang, T. Seong, and J. Choi, “A Low-Jitter Ring-DCO-Based Fractional-N Digital PLL With a 1/8 DTC-Range-Reduction Technique Using a Quadruple-Timing-

- Margin Phase Selector,” *IEEE J Solid-State Circuits*, vol. 57, no. 12, pp. 3527–3537, 2022, doi: 10.1109/JSSC.2022.3200475.
- [56] H. S. Kim *et al.*, “A Digital Fractional-N PLL With a PVT and Mismatch Insensitive TDC Utilizing Equivalent Time Sampling Technique,” *IEEE J Solid-State Circuits*, vol. 48, no. 7, pp. 1721–1729, 2013, doi: 10.1109/JSSC.2013.2253407.
- [57] 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,” *IEEE J Solid-State Circuits*, vol. 55, no. 12, pp. 3349–3361, 2020, doi: 10.1109/JSSC.2020.3019344.