An Ultra-Low-Power ADPLL for WPAN Applications

Design, Implementation, and Validation

Vamshi Krishna Chillara

Master of Science Thesis

TU Delft Delft University of Technology

Electronics

An Ultra-Low-Power ADPLL for WPAN Applications

Design, Implementation, and Validation

Master of Science Thesis

For the degree of Master of Science in Electrical Engineering
at Delft University of Technology

Vamshi Krishna Chillara

November 15, 2013

Faculty of Electrical Engineering Mathematics and Computer Science (EEMCS) ·
Delft University of Technology

Delft University of Technology
Department of
Electrical Engineering

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering Mathematics and Computer Science (EEMCS) for acceptance a thesis entitled

An Ultra-Low-Power ADPLL for WPAN Applications

by

Vamshi Krishna Chillara

in partial fulfillment of the requirements for the degree of

Master of Science Electrical Engineering

Dated: November 15, 2013

Supervisor(s):

Prof. Dr. Robert Bogdan Staszewski

Dr. Yao-Hong Liu

Reader(s):

Dr. Ir. Wouter A. Serdijn

Prof. Dr. Edoardo Charbon

Ir. Ying Wu

Abstract

RF PLLs for frequency synthesis and modulation consume a significant share of the total transceiver power, making sub-mW PLLs key to realize ULP WPAN radios. Compared to analog PLLs, all-digital phase-locked loops (ADPLLs) are preferred in nanoscale CMOS, as they offer benefits of smaller area, programmability, capability of extensive self-calibrations, and easy portability. However, analog PLLs dominate the ULP arena, since the time-to-digital converter (TDC) of an ADPLL has traditionally been power hungry. In this work [1], an ultra-low power 2.1 GHz – 2.7 GHz fractional-N ADPLL is presented for wireless personal area network (WPAN) applications. A DTC-assisted snapshot TDC and a DC-coupled DCO buffer with a tunable voltage transfer characteristic (VTC) are proposed to lower the power consumption.

The ADPLL prototype fabricated in TSMC LP 40 nm CMOS process consumes only 860 μW at 1 V supply, and has a measured rms jitter of 1.71 ps (integrated from 1k to 100MHz), leading to a state-of-the-art jitter^2-power FoM of -236 dB. The frequency modulation capability is also demonstrated with a 2 Mcps HS-OQPSK modulation for IEEE 802.15.4 (*ZigBee*) and 1 Mbps GFSK for *Bluetooth Smart*. This work presents the first-ever wireless ADPLL to break the 1mW barrier and consumes at least five-times lower power compared to state-of-the-art ADPLLs. The presented low-power techniques enable the adoption of ADPLLs in the emerging ultra-low-power applications.

Acknowledgements

I would like to thank my supervisor, Prof. Robert Bogdan Staszewski, who introduced me to this exciting field of Digital-RF. Needless to say, his constant guidance, encouragement, and immense knowledge were indispensable for successful completion of this work. I am glad (rather relieved) that I did not mar his immaculate record of first-time silicon success with ADPLLs.

I want to thank Dr. Wouter Serdijn, Prof. Edoardo Charbon, and Ying Wu for reading my thesis and serving on my committee.

I would like to thank Dr. Kathleen Philips for providing the opportunity to work at imec-nl. My daily-supervisors, Dr. Maja Vidojković and Dr. Yao-Hong Liu, had been of great support and guidance through out, and I cannot thank them enough. Maja got me started, helped me with the layout during the tape-out crunch, and was always there for any advice—technical or otherwise. Yao-Hong was the go-to guy for most part of my work, and we spent a lot of time having detailed discussions on every conceivable issue. He coordinated this two-student project, and also brought delicious home-cooked lunches when we had to work over weekends.

I owe special thanks to Bindi Wang for her meticulous design of the TDC and DTC, and importantly, for being a great company during times we had to work long. I should thank Xiongchuan (Coby) Huang and Ao Ba for their help in designing the DCO. Ao also designed the PA used in this work. I want to acknowledge the generous support of Benjamin Büsze in synthesizing the digital logic. I am grateful to Sandra and Marion for helping with the administrative tasks.

Ini and Sangan deserve special mention, for one cannot endure microelectronics at Delft, and remain sane without amazing friends. Also, Satoshi should be thanked for organizing get-togethers, visits, and ensuring we felt at home amidst all the dutch-eccentricities.

Last thanks, as always, goes to my family for their unconditional love and support. My brother has been a great friend and mentor through out my life, and I will always strive to emulate him. My mom and dad did all they could to ensure I have a better life; I am fortunate to have them in my life. I should acknowledge Google Hangouts for connecting us from across three continents.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless communications have seen an explosive growth in the last decade. Anytime anyplace connectivity is now a reality with smart phones becoming integral to our daily lives. Although proliferation of smart mobile devices will continue to propel the semiconductor market, many untapped opportunities exist in the realm of short-range low-cost wireless networks. These small, low-cost autonomous sensor-nodes have applications ranging from health care monitoring to home automation and environment sensing. It is these networks of tiny sensor-nodes that have the potential to trigger the next wireless revolution after cell phones and internet. Low-cost high-volume requirements of these sensor nodes fare well with the integration and scaling trends of CMOS. However, the autonomous operation requires the power consumption to be as low as possible. Power budget analysis of such sensor nodes reveals that the wireless transceiver dominates the overall power. This, together with the need for large volume, low-cost, highly integrated solutions warrant the need for ultra-low-power RF transceivers in nanometer scale digital CMOS technologies. This thesis is an effort in that direction and deals with the design of an ultra-low power all-digital phase-locked loop (ADPLL) for frequency synthesis in wireless transceivers.

1-1 Frequency synthesis in wireless transceivers

Figure 1-1 shows a simplified block diagram of a typical low-power transceiver. A low-IF receiver and direct modulation transmitter are typically the preferred architectures in low-power low-cost applications owing to their power and cost efficiency [2–5].

In a low-IF receiver, the received radio frequency (RF) signal from the antenna is down-converted to a low intermediate frequency (IF) by a quadrature mixer after amplification by a low-noise amplifier (LNA). After low pass filtering and boosting the signal amplitude to the required level by means of a programmable gain amplifier (PGA), the signal is fed to an analog-to-digital converter (ADC). The digitized data is then processed by the digital baseband which recovers the transmitted bits. On the transmit side, the digital baseband converts the user data to symbols, which are

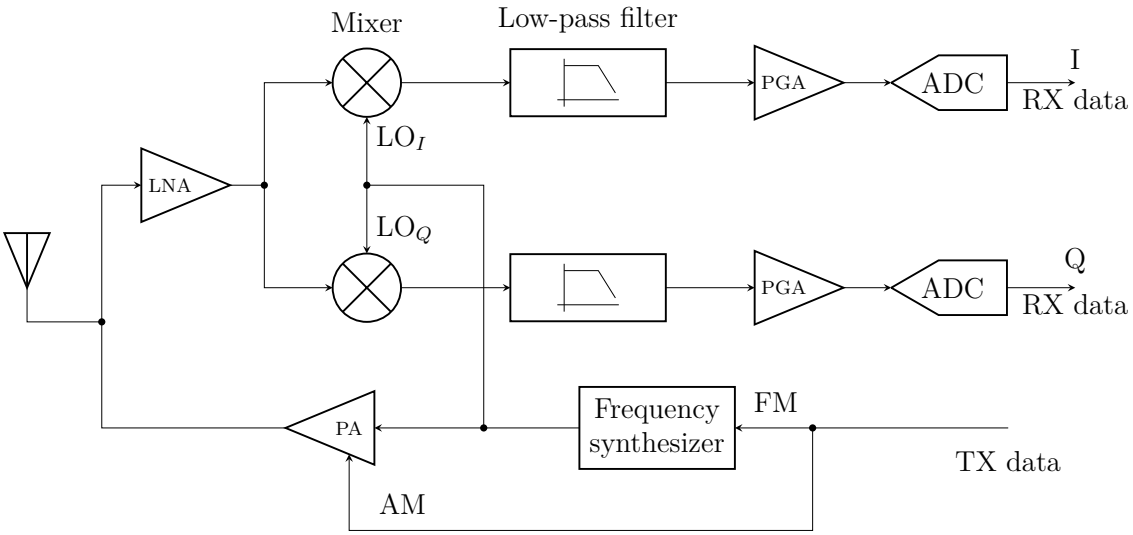


Figure 1-1: Low-power transceiver architecture.

pulse shaped to obtain I and Q digital samples. These I and Q samples are then transformed to polar form with amplitude and phase components. The frequency data—obtained from differentiating[1]the phase data—then modulates the synthesizer while the amplitude data controls the power amplifier. Thus, polar architecture avoids the need for up-conversion mixers and instead uses direct modulation, thereby saving power and area. Thus, a frequency synthesizer is an integral part of the transceiver and plays a key role in both receive path—to generate local oscillator (LO) signal for down-conversion—as well as in the transmit path, for direct frequency modulation.

1-1-1 Metrics of a frequency synthesizer

A local oscillator (LO) that generates a signal with programmable frequency enables up/down conversion in a wireless transceiver. A frequency synthesizer is normally used as it can provide signals with high spectral purity over wide range of frequencies. Moreover, it enables direct frequency modulation in polar transmitters. Figure 1-2 shows the functional block diagram of such a frequency synthesizer. A

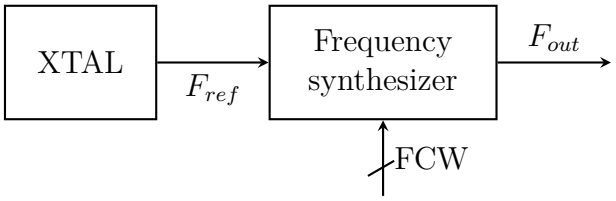


Figure 1-2: Frequency synthesis.

frequency synthesizer takes a low frequency clean reference from a crystal oscillator[2]

[1]implemented as difference between consecutive samples.

[2]On chip replacements for crystal oscillators such as thermal diffusivity based [6] and MEMS based [7] references is an active area of research but crystal based references currently dominate [8].

as an input and generates an output signal with a frequency determined by the frequency command word (FCW). Some common metrics to gauge the performance of a frequency synthesizer are discussed here.

Frequency tuning range and resolution

This denotes the range of frequencies that the synthesizer can generate with acceptable spectral purity and resolution. For instance, a frequency synthesizer designed for a 2.4 GHz *Bluetooth Smart* transceiver should cover the frequencies from 2.402 GHz to 2.480 GHz with a step of 2 MHz. Moreover, the accuracy with which the required channel frequency can be acquired is also crucial and is specified as parts-per-million. Variations over process, voltage, and temperature (PVT) should be considered while formulating the synthesizer specifications.

Channel switching time

Another important constraint on the synthesizer is its switching time—the time it takes to switch from one channel to another with a required accuracy. This can be crucial for some applications requiring fast switching such as frequency hopping spread spectrum. Moreover, the PLL consumes more power in the transient state compared to the locked state. Thus, fast switching helps to save power in ultra-low-power applications such as *Bluetooth Smart*.

Spectral purity

The required spectral purity is usually specified as a spectral mask that the synthesizer should satisfy. Both random and periodic fluctuations corrupt the spectral purity of a frequency synthesizer. The output of a frequency synthesizer can be represented in time-domain by Eq. (1-1).

$$v(t) = A\sin(\omega_c t + \psi(t)) \tag{1-1}$$

where A represents the amplitude of the synthesized signal with frequency ω_c. $\psi(t)$ captures the phase fluctuations. The impact of amplitude noise is generally neglected as it can be alleviated by a limiter circuit. Phase fluctuations in turn are divided into random and periodic variations as shown in Eq. (1-2).

$$\psi(t) = \Delta\psi\sin(\omega_m t) + \phi(t) \tag{1-2}$$

The first term denotes the periodic variations, which appear as a spurious tone at a frequency ω_m from the carrier. The random fluctuations, $\phi(t)$, manifest themselves as a noise-skirt around the desired frequency (see Figure 1-3).

1-1-2 Impact of phase noise on transceiver operation

Both the spurious tones and the noise skirt degrade the performance of the transceiver. The spot value of the phase noise spectrum at a particular frequency and the integral

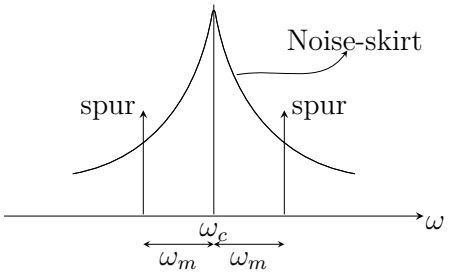


Figure 1-3: Phase noise and spurious tones.

of phase spectral density over a frequency range can both impair the performance of the transceiver. On the transmit side, noise-skirt on the LO signal can make it a potential interferer (blocker) to a near-by receiver detecting a signal in the adjacent frequency channel (see Figure 1-4).

Figure 1-5 shows the impact of the phase noise of the frequency synthesizer on the receiver in the presence of a large blocker in the near-by frequency channel. If the

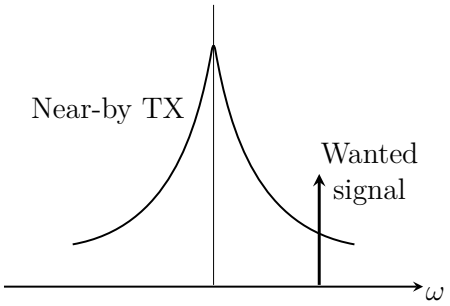


Figure 1-4: Effect of TX phase noise.

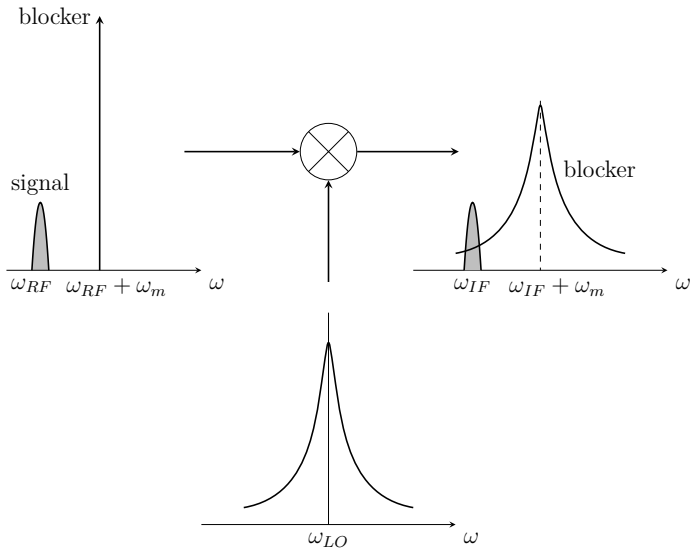


Figure 1-5: Effect of LO phase noise on receiver.

LO is ideal and has a spectrum of Dirac delta function, δ, then the down-converted blocker is at offset ω_m from the wanted signal and hence can be filtered off. However, if LO signal has a phase-noise skirt or a spectral tone at that offset, the blocker and

wanted signal are down-converted to the same IF frequency, thereby degrading the signal-to-noise ratio (SNR).

1-2 PLL based frequency synthesis

Although other methods such as direct digital synthesis or look-up table synthesis exist, phase-locked loop (PLL) based indirect frequency synthesis is widely used in wireless applications owing to its spectral purity and energy efficiency. Figure 1-6 depicts a simple block diagram of a PLL based frequency synthesizer.

PLL is a negative feedback system that ensures that the phase of the desired output signal is locked to that of a known stable reference. The phase detector compares the phase of the reference signal with the output signal and generates the phase error. This phase difference is processed by the loop filter—to set the loop dynamics—and is then used to control the oscillator. In the steady state, the error signal in the loop is forced to a constant value ensuring that the phases of two signals are locked and hence the frequencies.[3] In some implementations, a frequency divider is placed in the feedback path so that a high frequency signal can be generated from a lower reference frequency.

The frequency synthesis in wireless transceivers is dominated by charge-pump phase-locked loops (CP-PLLs) and all-digital phase-locked loops (ADPLLs), which differ in the implementation of the aforementioned building blocks. With the advent of nano-scale CMOS, ADPLLs have begun to supplant the traditional CP-PLLs in most applications.

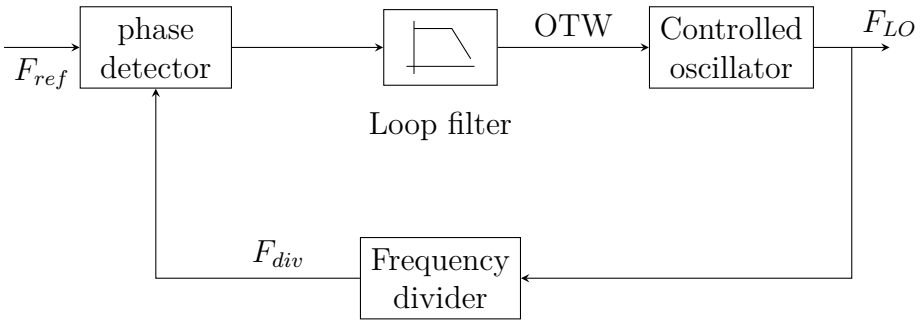


Figure 1-6: PLL based frequency synthesizer.

1-2-1 Charge-pump phase-locked loop

Figure 1-7 shows a simplified block diagram of the conventional CP-PLL. The controlled oscillator of Figure 1-6 is implemented as a voltage-controlled oscillator (VCO)—controlled by an analog input voltage. The phase frequency detector (PFD) compares the phase of the reference signal and the divided down output signal, F_{div}, to generate an up/down pulse with a width proportional to the phase error. The

[3] frequency being the derivative of phase, a constant phase difference translates to zero frequency difference.

output of the phase detector controls the magnitude and direction of the charge-pump current that is pumped into or out of the loop filter. The passive loop-filter converts this current into a tuning voltage for the VCO while suppressing the noise from the reference signal and the phase frequency detector. The output frequency of the VCO is adjusted such that the steady-state phase difference becomes constant and the output frequency is N times the input reference frequency.

In integer-N synthesizers, the reference frequency limits the frequency resolution. A fine frequency resolution requires a small reference frequency which limits the maximum loop-bandwidth that can be attained owing to the concerns of loop stability [9]. Loop-bandwidth is usually chosen not greater than one-tenth of the reference frequency. A wide loop bandwidth might be needed for higher suppression of oscillator phase noise or for faster switching speeds. Also, a small reference frequency needs a large multiplying factor N. Noise on the reference signal is amplified by N^2 and hence a very large value of N is not desired.

To decouple this bandwidth-resolution trade-off, fractional-N synthesizers evolved. The division ratio of the multi-modulus divider toggles between two integer values e.g., N and N+1, to achieve the required fractional division on average. The divider control is generated by a $\Sigma\Delta$ modulator to scramble the control pattern so that the resulting quantization noise is high-pass shaped. Consequently, $\Sigma\Delta$ control of the divider ratio enables a fine frequency resolution with a large reference frequency. However, the bandwidth cannot be too large since the quantization induced phase noise should be sufficiently filtered.

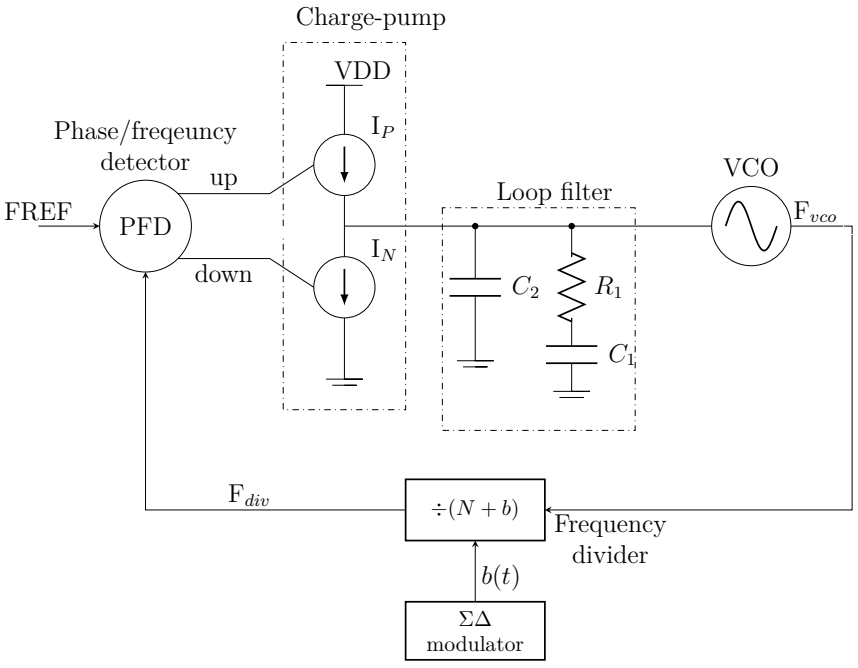


Figure 1-7: Simplified block diagram of charge-pump phase-locked loop (CP-PLL).

Implementation difficulties in nanoscale CMOS

CP-PLLs with sub-mW power consumption have been realized for use in ultra-low-power applications [5]. However, the analog intensive nature of the CP-PLL makes

its implementation difficult in advanced process nodes. Technology scaling shrinks the transistor dimensions to enable high integration densities. Supply voltages are also reduced to limit the electric fields so as to ensure reliable operation. However, this supply reduction makes it challenging to design traditional analog circuits such as a cascoded current source used in the charge-pump. Moreover, short channel effects such as drain-induced barrier lowering (DIBL) lead to reduced drain-source impedance of the transistor further complicating the design of current sources. Also, the smallest transistor dimensions are usually not used in the charge-pump design to minimize noise and to achieve good matching between up and down current sources. Hence a charge-pump occupies a significant area. Also, since the passives do not scale well with the technology, the loop-filter used in CP-PLL occupies large area, increasing the overall system costs. In addition, the reduced gate-oxide thickness in the nanoscale CMOS technologies results in significant current leakage through the integration capacitor of the loop-filter. This leakage current increases the total PLL jitter, thereby degrading the performance.

1-3 Motivation and thesis objectives

The need for low-power low-cost transceivers in large volumes favors the realization in the most advanced CMOS node available. The problems with analog design in low-voltage nano-meter scale CMOS processes and the possibility to integrate RF front-end with baseband processor favors the digitally intensive approach. Furthermore, a digital intensive approach provides numerous other benefits—better testability, reconfigurability, lower area, and high degree of integration. Moreover, the digital approach allows for extensive calibrations; for instance it is possible to track and correct the variations of the digitally controlled oscillator (DCO) gain over process, voltage, and temperature (PVT) [10]. This enables to accurately implement two-point frequency modulation required for a polar transmitter.

Accordingly, a digitally intensive PLL architecture called ADPLL emerged for frequency synthesis in wireless applications. Since its inception in [11], ADPLLs have garnered the attention of numerous researchers because of their financial benefits and amenability to scaling. ADPLLs for frequency synthesis in different wireless standards namely Bluetooth, GSM, and EDGE have been successfully demonstrated [12–14] with a performance comparable to or even better than that of conventional charge-pump phase-locked loops (CP-PLLs). DCO and time-to-digital converter (TDC) being the main bottle-necks justifiably have seen a number of efforts to counter the limitations arising from their finite resolution. While [15–17] improve the DCO's performance, [18–21] address the problems that arise from TDC non-idealities. The fractional spurs that impede the realization of a wide-band ADPLL are the focus of [22,23]. Thanks to these improvements, nearly one-third of the new mobile phones produced use ADPLLs [24].

In contrast, ADPLLs are hardly used in the ultra-low power arena with charge-pump PLLs dominating the frequency synthesis in these transceivers [25–29]. This is evident from Figure 1-8 which shows a plot of fractional-N ADPLLs published in the last five years at ISSCC and VLSI symposium. The published works are plotted with power consumption in mW as abscissa and the jitter variance in ps^2 as the ordinate. The

jitter^2-power FoM[4] is also indicated in the graph. It can be noticed that there are no fractional-N ADPLLs in the sub-mW vicinity, a key requirement for *Bluetooth Smart* and *ZigBee* applications. The main objective of this thesis is to break the sub-mW barrier for ADPLLs so as to enable their adoption in ultra-low-power applications. The targeted specifications and scope of this thesis are discussed in the next section.

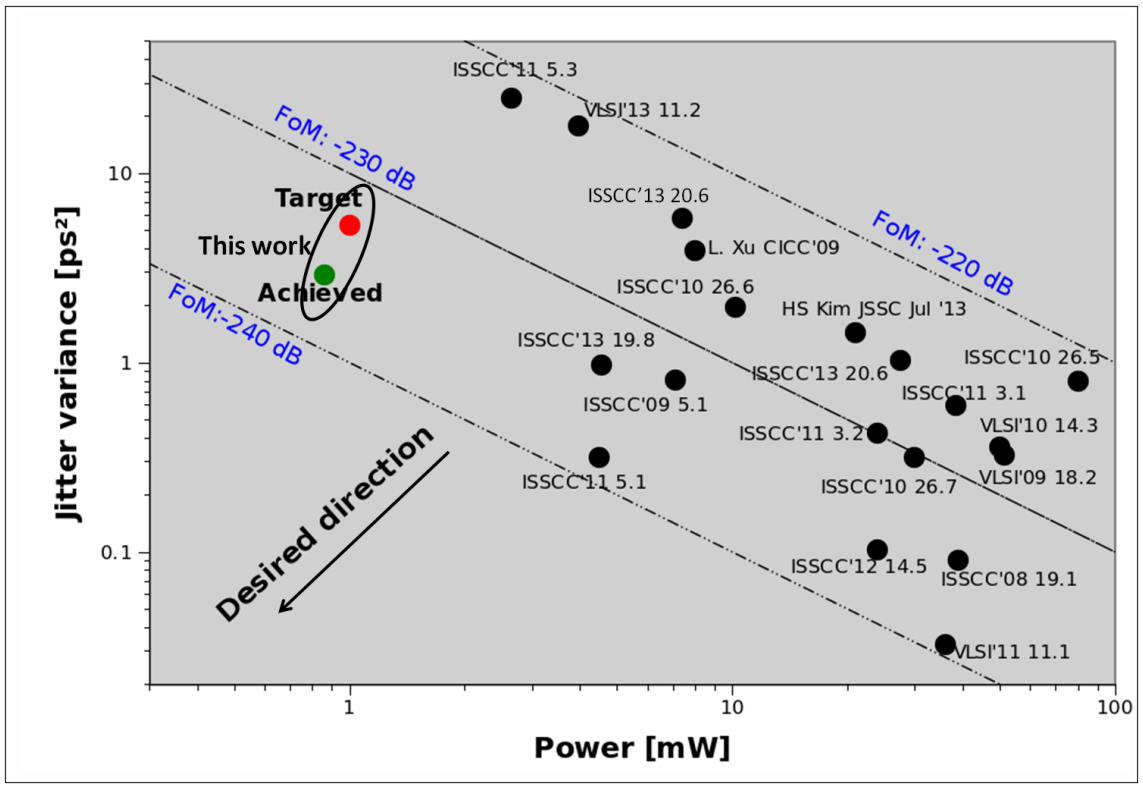


Figure 1-8: Benchmark of fractional-N ADPLLs.

1-3-1 Target specifications

The specifications of the targeted frequency synthesizer are shown in Table 1-1. In addition to generating the LO signal with the required frequency, this ADPLL should also act as a direct frequency modulator in the transmitter path. Accordingly, an ADPLL with a two-point modulation capability satisfying the tabulated specifications is targeted.

This all-digital synthesizer designed in 40 nm digital CMOS technology targets *Bluetooth Smart* and *ZigBee* standards around the 2.4 GHz ISM band. It is desirable if this very synthesizer can also support 400 MHz medical implant communication service (MICS) applications. The MICS standard has two bands of frequencies: 402–405 MHz and 420–450 MHz. Hence, the synthesizer is targeted to cover the frequency range of 2.4–2.7 GHz (400–450 MHz after division by 6) to accommodate all the three aforementioned standards. In addition, the synthesizer should be able to handle data rates of up to 1 Mbps for *Bluetooth Smart* when acting as a frequency

[4] $FoM = 10log_{10}\Big(\big(\sigma^2_{jitter}\big) * \big(\frac{P}{1mW}\big)\Big)$ [30]

Table 1-1: Target specifications of the frequency synthesizer.

Parameter	Requirement
Frequency range	2.4–2.7 GHz
Frequency accuracy	60 KHz
Switching time	40 μs
Phase noise @ 1 MHz offset	-110 dBc/Hz
Integrated RMS phase error	2°
Modulation data rate	1 Mbps
Reference frequency	32 MHz
Power consumption	<1 mW
Technology	40 nm digital CMOS

modulator. Finally, sub-mW power consumption is crucial to accommodate its use in the autonomous sensors deployed in wireless personal area networks (WPANs) or wireless body area networks (WBANs).

1-4 Research contribution

It is clear from the earlier discussion that the analog charge-pump phase-locked loops (CP-PLLs) dominate the ultra-low-power landscape. ADPLLs not only ease the design challenges in advanced process nodes but also facilitate additional benefits that accrue from digitization. However, ADPLLs have not been explored for the ultra-low-power applications, which require sub-mW power consumption. This work presents the first-ever multi GHz sub-mW fractional-N ADPLL by using a combination of digital-to-time converter (DTC) and TDC for fractional phase detection.

A TDC in a conventional ADPLL [31] needs to cover one full DCO period sensing the DCO clock at its full rate, thus consuming several mW. In this work, TDC snapshotting is implemented to reduce the sampling rate to FREF, while the DTC reduces the detection range to less than one-tenth of the DCO clock period, leading to a significant power reduction.

In addition, a power efficient DCO buffer with a tunable voltage transfer characteristic (VTC) is presented. It is DC coupled to the low-swing output of the DCO to avoid driving bulky resistor-biased decoupling capacitors. This approach reduces the overall power consumption of the DCO plus buffers.

In this two-student ultra-low-power ADPLL project at imec-nl, my work deals with the system level analysis of ADPLL, RTL and back-end design of low-speed digital circuitry, and circuit-level design of DCO and other high frequency blocks. The TDC, DTC, and the necessary calibration circuitry are designed by a colleague, Bindi Wang. A silicon prototype that can support *Bluetooth Smart* and *ZigBee* standards while consuming just 860 μW is implemented to demonstrate the feasibility of adapting ADPLLs for frequency synthesis in ultra-low power transceivers.

1-5 Thesis organization

This thesis is organized as follows. Chapter 2 begins with a background on ADPLLs while reviewing the current state-of-the-art and highlighting the challenges in adapting ADPLLs for ultra-low-power applications. Then the DTC assisted snapshot TDC based ADPLL implemented in this work is presented. Chapter 3 describes the RTL design of this ADPLL. The circuit design of DCO and other high speed blocks—divider-by-2, phase incrementer, and DCO buffer—is described in Chapter 4 along with simulation results. The measurement set-up, PCB layout, and the measurement results of the implemented silicon prototype are presented in Chapter 5. Chapter 6 is reserved for conclusions and provides suggestions for future work.

Chapter 2

All-digital phase-locked loops

This chapter reviews the common architectures of all-digital phase-locked loops (ADPLLs) from the literature. The power consumption of each of these architectures is analyzed to identify the power-hungry blocks. The possible solutions to reduce the power at architectural level are explored. Finally, the digital-to-time converter (DTC) assisted snapshot time-to-digital converter (TDC) based ADPLL implemented in this work is described.

2-1 Existing architectures

Digital processing is cheaper, faster, and consumes less power in advanced CMOS nodes. However, scaling is less attractive for analog circuits as they are limited by the electronic noise and low supply voltages make it difficult to meet the accuracy requirements. ADPLLs evolved to address the challenges of implementing the traditional charge-pump phase-locked loops (CP-PLLs) in advanced process nodes.

In an ADPLL all the analog blocks of the CP-PLL are replaced either by their digital equivalents or by mixed signal blocks with digital interfaces at the input and output. A voltage-controlled oscillator (VCO) of the conventional CP-PLL is replaced by a digitally controlled oscillator (DCO), which deliberately avoids any analog tuning. The DCO produces a frequency proportional to its input digital word plus an offset. The charge-pump/phase-frequency detector combination got replaced by a TDC, which digitizes the time difference (phase) between its input signals. Since the phase error is digitized, a bulky passive loop filter of the CP-PLL can be replaced by a compact digital one, resulting in considerable cost savings. A review of current literature reveals various implementations to realize an ADPLL, mostly differing in the way phase to digital conversion is achieved. They can be broadly classified into the following two types: divider-less and divider-based ADPLLs.

2-1-1 Divider-less ADPLL

This is the architecture that triggered the research in digital-intensive frequency synthesis for wireless transceivers and is first presented in [31]. Figure 2-1 shows

a simplified block-diagram of the divider-less ADPLL [32]. The output frequency termed as the variable frequency, F_v, is set by the frequency command word (FCW).

$$FCW = \frac{F_v}{FREF} \tag{2-1}$$

Equivalently, FCW denotes the number of clock cycles of variable clock (CKV) in one reference clock. Hence, accumulating the FCW at reference rate provides the reference phase (Φ_R) that is normalized to CKV clock period. The variable phase, Φ_V, is obtained by counting the number of output clock cycles. Phase incrementer gives the integer part of the variable phase, Φ_{V_int}. A TDC is used to detect the phase with a resolution finer than one integer cycle.

A typical TDC consists of a chain of delay elements that delay the CKV signal. The delayed versions of the CKV are sampled by FREF to obtain the time delay between the edges in terms of number of delay elements [18]. Hence the resolution of the fractional phase detection is limited by the intrinsic delay of the unit cells in the TDC. The fractional phase is obtained by normalizing the output of the TDC to the CKV period.

The variable phase and reference phase, which are available as digital words can now be subtracted by digital logic to generate the phase error. This phase error is passed through a digital loop filter to obtain the oscillator tuning word (OTW). OTW controls the DCO frequency in a negative feedback configuration such that the phases of the CKV and reference clock FREF are either equal or have a constant difference in the steady state, depending on the type of loop. The frequency of the variable clock (CKV) is hence equal to FCW times the reference frequency.

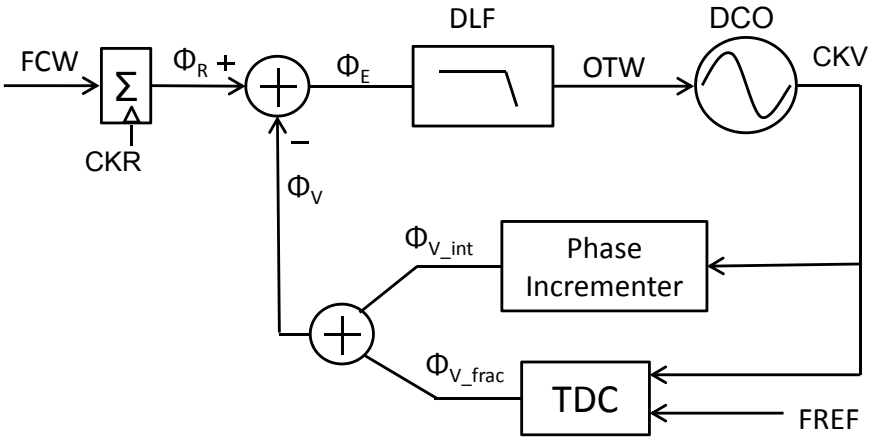


Figure 2-1: Architecture of divider-less ADPLL.

Since the clock signals CKV and FREF are asynchronous, re-timing is needed to avoid metastability. The re-timed reference clock CKR is generated by sampling the reference clock FREF with the variable clock (CKV). Since this re-sampling itself may introduce metastability, sampling is done both at the rising edge and the falling edge of CKV, and the one farthest from the metastability is picked [32]. The re-timed reference clock CKR is then used as the clock for the low frequency digital logic: phase detector, digital loop filter, and gain normalization blocks.

2-1-2 Divider-based ADPLL

The divider-based ADPLL architecture, shown in Figure 2-2, is an exact block-by-block replacement of the analog blocks in a traditional CP-PLL with their digital equivalents. In this architecture, the variable clock (CKV) is passed through a fractional divider before it is fed to the TDC. The required output frequency is set by the divider ratio. To realize a fractional division ratio, a $\Sigma\Delta$ modulator dithers the divider ratio between two integer values. The TDC compares the time difference between the significant (rising or falling) edges of the stable reference clock FREF and the divided down output clock F_{div} to generate the phase error directly as a digital word. After passing through a digital filter, the phase error controls the DCO frequency to align it to the reference clock.

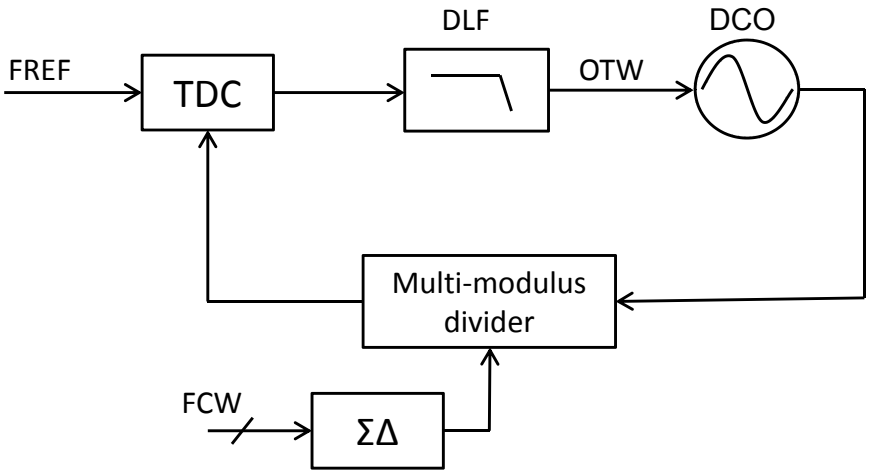


Figure 2-2: Architecture of divider-based ADPLL.

2-1-3 Comparison of divider-less and divider-based architectures

The phase incrementer and the divider play a similar role and consume comparable power and area, leading to a similar performance in either architecture. However, multi-modulus divider is accompanied with fractional spurs requiring additional circuitry to mitigate them [33]. Moreover, divider-based architectures do not operate in phase-domain and hence should be realized as type-II[1] systems to ensure the divided down CKV and FREF edges are closely aligned. On the other hand, the divider-less architectures [31,34] operate in the phase-domain but both the TDC and phase incrementer operate at the high output frequency (typically in the GHz range) consuming significant power. The reduced operating frequency of TDC in a divider-based architecture is offset by its increased range requirement—one reference clock period—as opposed to one CKV duration in a divider-less architecture. Thus, in either architecture, TDC consumes a significant power. Section 2-4 explores different approaches in the literature to tackle this issue.

[1] *type* indicates the number of poles at the origin

2-2 Requirements on TDC and DCO

Like any mixed-signal system, ADPLL also suffers from the effects of the inevitable quantization noise resulting from analog-to-digital and digital-to-analog conversion. The TDC which converts the analog phase of the CKV to a digital word limits the in-band phase-locked loop (PLL) noise, while the DCO that converts the digital control word to an analog signal with desired frequency affects the out-of-band noise. Fortunately, both the TDC resolution—usually limited by the inverter delay—and the DCO frequency step improve with technology scaling and hence the performance of the ADPLL improves with each advanced node. The requirements on the resolution of TDC and DCO for targeted applications are calculated here.

2-2-1 Impact of TDC on phase noise

The finite resolution of TDC results in quantization noise whose power is given by Eq. (2-2) [35], where Δ_{TDC} is the quantization step of the time-to-digital converter (TDC).

$$\sigma_t^2 = \Delta_{TDC}^2/12 \tag{2-2}$$

This noise is distributed over the frequency band $(0, \frac{f_s}{2})$, where f_s is the sampling frequency which is the reference frequency, f_R, of the ADPLL. By normalizing it to the unit interval and multiplying by 2π radians phase noise is obtained [32].

$$\sigma_\Phi = 2\pi\frac{\sigma_t}{T_v} \tag{2-3}$$

Assuming an ideal TDC, the noise distribution can be considered white giving rise to a single-sided noise spectral density of

$$\mathcal{L} = \frac{\sigma_\Phi^2}{f_R} \tag{2-4}$$

From Eq. (2-3) and Eq. (2-4), the single sided phase noise of the ADPLL due to quantization of TDC is given by

$$\mathcal{L} = \frac{(2\pi)^2}{12}\left(\frac{\Delta_{TDC}}{T_v}\right)^2\frac{1}{f_R} \tag{2-5}$$

This noise from the TDC has a low pass transfer function and hence dominates the in-band PLL noise. In addition to the quantization noise, the finite TDC step and mismatch between the delay elements also result in spurs at the frequency, $FCW_{frac} * FREF$, further degrading the phase noise of the ADPLL [22].

In the TSMC 40 nm process in which the targeted synthesizer is implemented, an inverter delay of 20 ps is achievable without consuming too much power. For this value of $\Delta_{TDC} = 20\ ps$, the in-band PLL noise floor given by Eq. (2-5) is equal to -95 dBc/Hz with 2.7 GHz operating frequency. This value is sufficient for the targeted applications and hence a delay-line based TDC suffices. Note that, it is possible to achieve sub-inverter delay resolution, although at the expense of increased power consumption. In [19] time amplification is used to achieve sub-ps resolution but consumes a staggering 60 mW power. Vernier delay-line based architecture [36] uses the difference between individual delay elements to achieve fine resolution but requires a large number of cells and hence consumes a lot of power.

2-2-2 Impact of DCO on phase noise

DCO is the core of the ADPLL and produces an output frequency proportional to its input digital word in addition to an offset. The frequency resolution of the DCO also adds additional quantization noise which can be modeled with white noise spectral characteristics provided the input tuning word spans multiple quantization levels. The variance of the quantization error in output frequency is then given by

$$\sigma_f^2 = \Delta f_{res}^2/12 \tag{2-6}$$

Since this phase noise power is spread uniformly from DC to Nyquist frequency, the single sided spectral density is given by

$$\frac{1}{2}S_{\Delta f} = \frac{\sigma_f^2}{f_R} \tag{2-7}$$

This can be converted to phase variation by multiplying the DCO's frequency to phase conversion transfer function, $\frac{2\pi}{s}$. The single-sided phase noise power spectral density at the output is then given by

$$\mathcal{L}(\Delta f) = \frac{1}{12}\left(\frac{\Delta f_{res}}{\Delta f}\right)^2 \frac{1}{f_R} \tag{2-8}$$

Since the DCO input tuning word is held constant between two different values, the white noise assumption is not fully justified. Hence, the above equation is multiplied by *sinc* function to account for the zero-order hold operation on the input tuning word of DCO [32].

$$\mathcal{L}(\Delta f) = \frac{1}{12}\left(\frac{\Delta f_{res}}{\Delta f}\right)^2 \frac{1}{f_R}\left(sinc\frac{\Delta f}{f_R}\right)^2 \tag{2-9}$$

The quantization noise of the DCO has a 20 dB/decade attenuation, similar to that of the up-converted thermal noise from the oscillator. As long as this quantization noise is kept sufficiently low compared to the inherent phase noise of the oscillator (resulting from thermal and flicker noise of active part or the passive LC tank), the overall phase noise is not significantly affected. The noise from the DCO is high-pass filtered by the loop and hence has a band-pass characteristic, degrading the out-of-band phase noise of the synthesizer.

A spot noise of -110 dBc/Hz at 1 MHz offset from the carrier frequency is targeted indicating that the phase noise from the DCO quantization should be lower than -120 dBc/Hz. A 50 kHz accuracy is required in channel selection, which corresponds to a DCO quantization noise of -112 dBc/Hz from Eq. (2-9), which is comparable to the targeted phase noise. Hence a 5-bit $\Sigma\Delta$ modulator is used to dither the DCO control word to minimize the quantization noise [17]. In this way, the quantization step is reduced to $\Delta_{fres}/2^5$ and so is the quantization noise. However, the $\Sigma\Delta$-modulator itself introduces spurs somewhat limiting its improvement. Either a high dithering rate or higher order dithering can be employed to counter the issue of spurs.

2-3 Direct frequency modulation

Direct frequency modulation of the frequency synthesizer simplifies the transmitter architecture, saving power and area. In traditional fractional-N synthesizers, the divider ratio is modulated by the TX data, achieving the desired frequency deviation. However, the maximum rate of modulation is limited by the loop bandwidth as the modulation signal is filtered by the low-pass characteristic of the PLL. A narrow loop bandwidth is usually chosen to minimize the high frequency quantization noise injected by the $\Sigma\Delta$ divider as well as to filter off the noise from phase detector or the reference clock. Thus it is difficult to achieve high data rates by directly modulating the divider ratio. To overcome this issue, techniques such as pre-emphasis [37] and two-point modulation [38, 39] are used.

In the approach using pre-emphasis, the frequency control is multiplied by inverse of the PLL transfer function to compensate the low-pass filtering by the loop. However, it requires accurate knowledge of the loop transfer function to ensure the pre-emphasis does not distort the transmission data. In two-point modulation method, the modulation data is added directly to the oscillator control signal in addition to the divider. If the oscillator gain is accurately known, this method removes the loop dynamics from the transmitted data path. The digital implementation of the PLL eases the implementation challenges of both these approaches. However, since the pre-emphasis method needs extra hardware it is not preferred.

2-3-1 Two-point frequency modulation

Figure 2-3 shows the implementation of the two-point modulation in the divider-less ADPLL architecture. The modulation data is added to the FCW in the low frequency

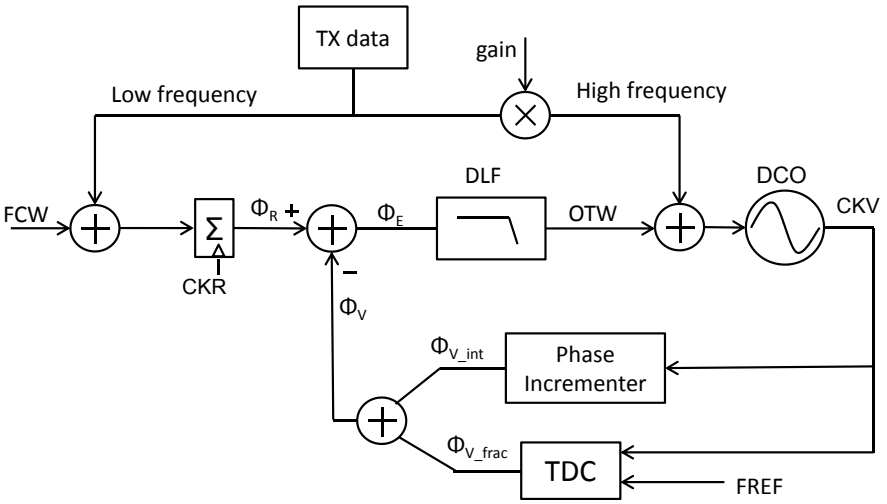


Figure 2-3: Two-point frequency modulation in divider-less ADPLL architecture.

path and directly to the DCO control word in the high frequency path. The scaling factor in the high-frequency path ensures that the LSB of the modulation data corresponds to same frequency step in both high and low frequency paths. The step in the low-frequency path is set accurately by the reference frequency and is

independent of the variations in process, voltage, and temperatures (PVTs). On the other hand, the frequency step of the high frequency path depends on the resolution of DCO, K_{DCO}, and hence is different to that of the low-frequency path.

When the modulation data increases, the frequency of the DCO and hence the variable phase, Φ_V, becomes higher. On the low-frequency path, the FCW is also increased which increases the reference phase, Φ_R, keeping the phase error unchanged. However, the calculation of the gain factor on the high frequency path of the TX data requires the knowledge of the DCO gain, K_{DCO}. It can be calculated *just-in-time* as discussed in [10]. Since the OTW is available as a digital signal, the change in OTW for a known deviation in frequency can be observed to estimate the value of K_{DCO} over PVT variations. Thus ADPLL can perform direct frequency modulation with a data-rate independent of the loop band-width while consuming little additional hardware.

2-4 Challenges for ultra-low-power applications

DCO and TDC consume most of the power in an ADPLL. The power consumption of DCO depends on its phase noise requirements, and hence can be scaled down in short-range wireless applications. The design of DCO is discussed in Chapter 4. On the other hand, the power consumption of TDC is large, mainly owing to its high frequency operation or large range requirement depending on the architecture.

The similarity of TDC to that of an analog-to-digital converter (ADC) inspired research in the lines of $\Sigma\Delta$ data-conversion. However, a coarser TDC resolution requirement does not necessarily translate to lower power beyond that available from minimally sized inverter delays. As a result, $\Sigma\Delta$ frequency discriminator based architectures [40, 41] do not help unless a bang-bang phase detector is adopted or sub-inverter delay resolutions are required. Also, this approach does not take advantage of the high resolution inverter delays available in the advanced CMOS nodes and expends considerable power in digital processing. A fine resolution TDC allows wide loop bandwidth, thereby filtering more noise from DCO. Since the power consumption of DCO is proportional to its phase noise, total ADPLL power can thus be reduced for a given jitter requirement.

In this work, the techniques to reduce the required phase detection range and operating frequency of TDC are explored instead and are discussed here.

2-4-1 Reducing the TDC detection range

TDC needs to cover a range equal to the CKV period with a fine resolution, thereby consuming considerable power. A bang-bang phase detector is adopted in [42] to replace the TDC, thereby alleviating the issue of large range requirement. A bang-bang phase detector is essentially a D flip flop determining whether the reference signal leads or lags the divided down signal. The spurs that are inherent to the bang-bang phase-locked loops (BBPLLs) are minimized by operating it in the random-noise regime. To achieve this, a true fractional divider using a 10-bit DTC is proposed.

Compared to a TDC, DTC consumes less power as it does not need power-hungry sense-amplifier based flip-flops [18] required by a TDC. That architecture achieves an impressive figure-of-merit (FoM)[2] of -238 dB while consuming 4.5 mW. However, the integrated jitter cannot be traded to achieve the sub-mW power levels required for our targeted applications as it would mean moving away from the random-noise regime, thereby exacerbating the issue of spurious tones resulting from limit cycles. Moreover, the loop bandwidth of that circuit depends on the noise at the input of the bang-bang phase detector and hence additional circuitry is required for loop bandwidth regulation and frequency acquisition.

A divider-less ADPLL with DTC and a bang-bang phase detector is presented in [43]. In [44], the authors propose to use the DTC in order to reduce the required TDC range rather than replacing it by a bang-bang phase detector. In both these architectures, the accumulated value of fractional FCW is used to delay the reference signal FREF such that the delayed reference signal, FREF_{dly} is almost aligned to the CKV once the loop is locked. This idea of phase prediction is depicted in Figure 2-4 for the case when FCW = 2.25. As can be seen, the required delay to align the FREF edge to the next CKV edge can be obtained by subtracting the accumulated value of fractional FCW from 1. The delayed reference signal and the CKV can be then fed to the narrow-range TDC, the output of which directly gives the fractional phase error. This approach helps in reducing the TDC range by as much as ten times.

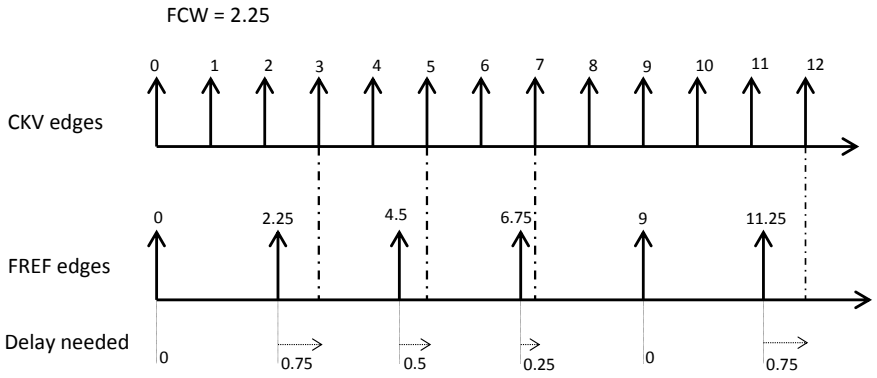


Figure 2-4: Principle of phase-prediction to reduce fractional-phase detection range.

In yet another approach [45], the quadrature phases of the DCO are used to reduce the operating range of TDC. Again, the accumulated value of FCW_{frac} is used to select one of the four CKV phases that is closest to the FREF signal. In this way, the required detection range is reduced to one-fourth of CKV period. However, the TDC is clocked at CKV frequency (usually in the order of GHz), thereby increasing the power consumption.

2-4-2 Lowering the operating frequency of TDC

The TDC compares the phase difference between the CKV signal and FREF and hence operates at high frequency, consuming significant power. However, the phase

[2] $FoM = 10log_{10}\Big(\big(\sigma^2_{jitter}\big) * \big(\frac{P}{1mW}\big)\Big)$ [30]

difference is calculated only close to the rising edge of FREF i.e., the TDC needs to operate only at the reference rate, typically tens of MHz. This is the basis of the time-windowed TDC based architecture presented in [46] where the TDC is gated to reduce its operating frequency and hence power consumption. However, this approach requires a TDC with a detection range equal to the CKV period and hence cannot be combined with one of the methods that reduce the detection range. In [44], DTC based phase-prediction is used to simultaneously reduce the operating frequency and detection range. Since the phase-prediction maintains a fixed phase difference between the FREF_{dly} and CKV signals, the first CKV pulse after the rising edge of FREF_{dly} is picked and is fed to the TDC for phase detection. In this way, total power consumption can be reduced by FCW times—typically on the order of 30–40 from the reduced operating frequency and nearly ten times from the reduced detection range.

2-5 DTC-assisted snapshot TDC based ADPLL

Figure 2-5 depicts the top-level block diagram of the ADPLL with two-point frequency modulation capability designed in this work. The blocks with gray background—TDC, DTC, snapshotting, K_{DTC} calibration, and power amplifier (PA)—are designed by colleagues at imec-nl and the description of their implementation is not a part of this thesis. The blocks with blue background are the analog/high-speed blocks that warrant custom design whose circuit level implementation and layout are described in Chapter 4. The remaining blocks comprise the low-speed digital logic responsible for processing the phase information to generate the appropriate control signals for the DCO. The functional description of the implemented low-speed digital blocks is provided in this section. The detailed RTL-level description is deferred until Chapter 3.

A divider-less ADPLL architecture with a difference mode phase detector [32] is used. To meet stringent sub-mW PLL power constraints, three low-power techniques are employed:

1. The biggest power-saving is obtained by using a DTC assisted snapshot TDC for fractional phase detection. DTC helps in reducing the detection range of DTC by using the principle of phase-prediction described in Section 2-4-1. Snapshotting reduces the operating frequency to reference rate.

2. Secondly, a power efficient DCO buffer with a tunable voltage transfer characteristic (VTC) is used to isolate the DCO from load variations in PA as well as to boost the low-swing DCO output to rail-to-rail voltage levels. Unlike the conventional approach, the buffer is DC-coupled to the DCO to avoid driving bulky resistor-biased decoupling capacitors. The implementation details of the DC-coupled DCO buffer are presented in Section 4-2-2.

3. Finally, a frequency divider (/2) reduces the operation of phase detection circuitry to half the DCO rate, CKVD2. This saves power at the expense of doubling the required detection range.

2-5-1 Phase prediction

By virtue of phase-prediction discussed in Section 2-4-1, DTC helps in reducing the required detection range of TDC. Thus, the DTC/TDC combination acts as a coarse-fine structure with the *coarse* DTC assisting the *fine* TDC. A 64-stage DTC is implemented so as to cover the maximum value of CKVD2 period with sufficient margin over PVT variations. The digitally controlled delay line of the DTC is similar to the one in [47]. The control word of the DTC is obtained from the accumulated value of FCW_{frac}, PHR_F, according to Eq. (2-10). The DTC gain, K_{DTC}, is the ratio of time-step of the DTC to the time period of CKVD2 .

$$DTC_{ctrl} = \frac{1 - PHR_F}{K_{DTC}} \tag{2-10}$$

For correct phase prediction, the value of K_{DTC} should be tracked accurately over PVT variations. Even with incorrect phase prediction, the ADPLL can lock but will have an increased phase noise as it tries to track the incorrectly predicted reference phase. The estimation is accomplished by the K_{DTC} calibration block of Figure 2-5 which monitors the sign of the phase error to correct the estimated value of K_{DTC} [48].

The basic idea of estimation is as follows: When the DTC gain is underestimated, the predicted variable phase is a saw tooth waveform with a slope larger than the ideal (i.e., reference) phase and a frequency f_{PE} equal to $f_R * min(FCW_{frac}, 1 - FCW_{frac})$ as shown in Figure 2-6. Similarly, when overestimated, the predicted phase has a slope smaller than the ideal one. Consequently, the polarity of ($(PHR_F\text{-}0.5)*PHE$) can be used to predict if the DTC gain is under/overestimated and thus can be corrected using least mean square algorithm.

Residue correction

Even when the DTC gain is correctly estimated, the accuracy of phase prediction is limited by the quantization step of the DTC. From Eq. (2-10), it can be seen that the DTC_{ctrl} can be a fractional number. However, only integer part is used to control the DTC delay and the residue that is left behind results in an error in phase prediction. However, since this error is known beforehand, it can be corrected by subtracting it from the output of the TDC. Thus this residue correction ensures that the quantization step of DTC has no impact on the overall phase noise. The exact implementation is described in Section 3-1-3.

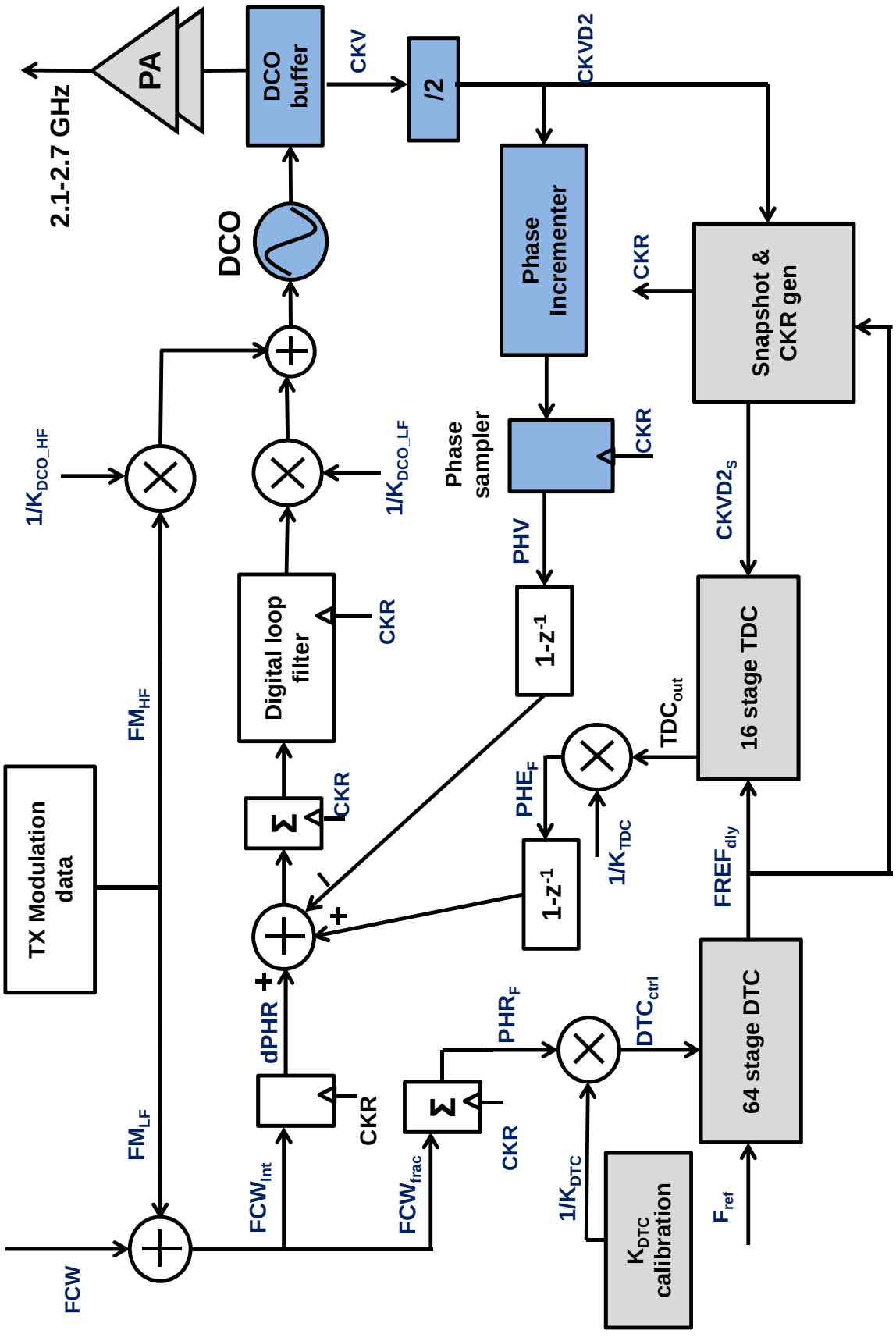
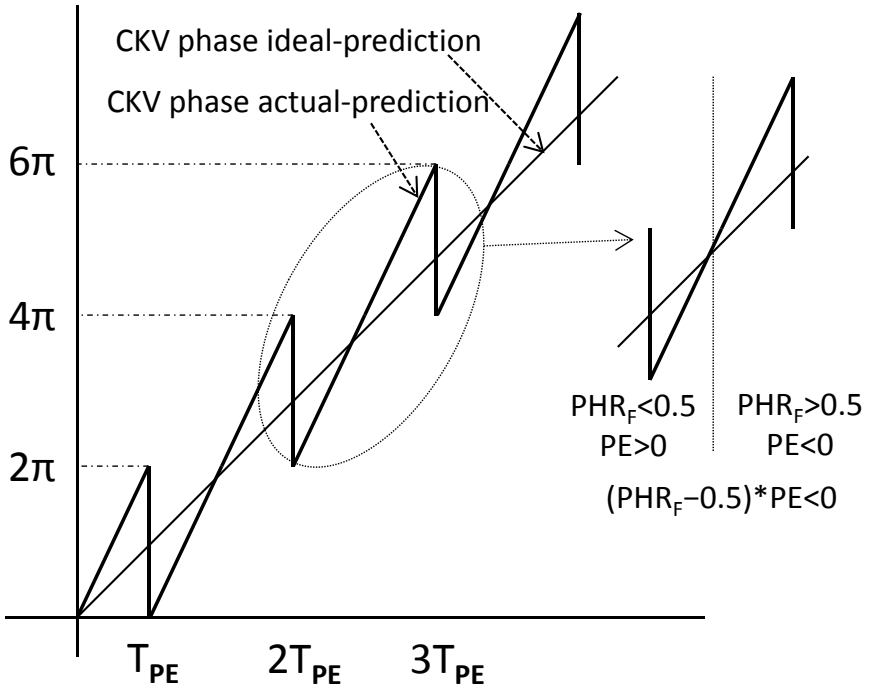
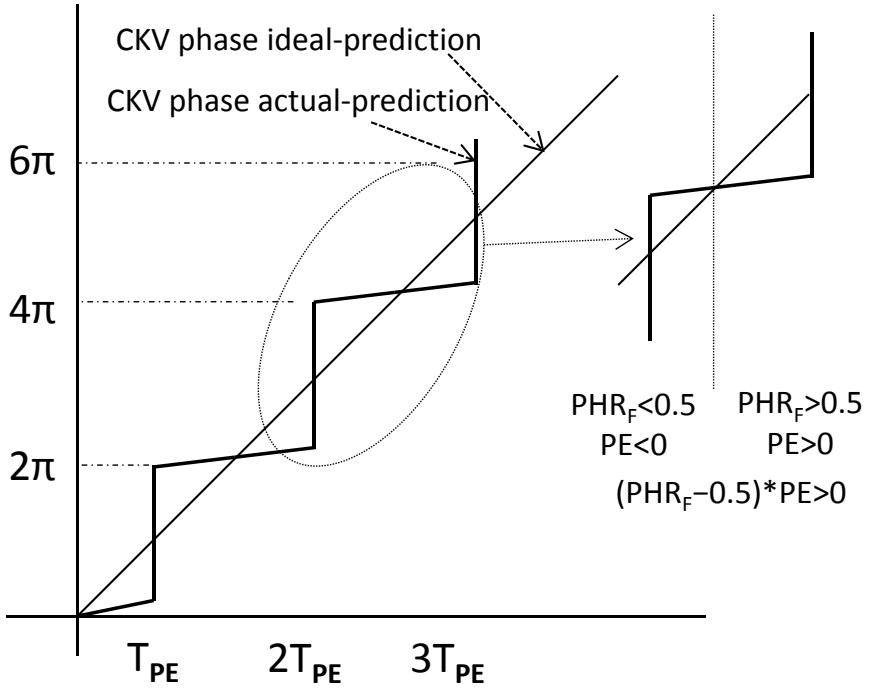


Figure 2-5: Top-level block diagram of the ADPLL.



K_{DTC} underestimation



K_{DTC} overestimation

Figure 2-6: Idea of DTC gain calibration [48].

2-5-2 Snapshotting and CKR generation

TDC snapshotting is implemented to reduce the operating frequency of TDC from CKVD2 to FREF. The delayed reference, FREF_{dly}, which is aligned to CKVD2 by means of phase prediction is used to trigger the snapshot circuit. As shown in Figure 2-7, FREF_{dly} deasserts the CKVENB signal which takes the snapshot of the first CKVD2 edge, CKVD2_{S}. Thus, at every rising edge of the delayed reference clock

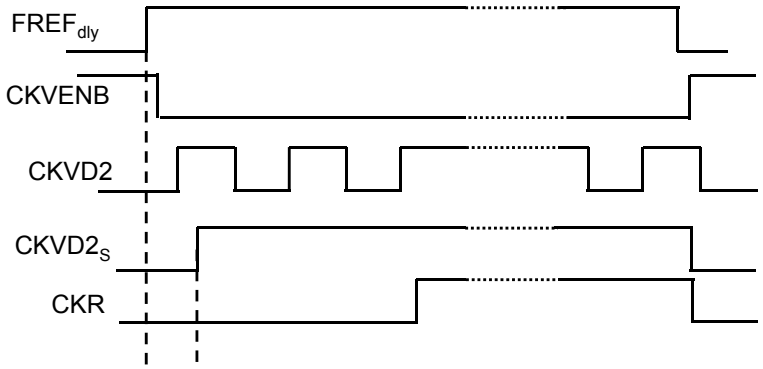


Figure 2-7: TDC snapshotting and CKR generation.

FREF_{dly}, the snapshot of the CKVD2 rising edge is taken. Hence, the frequency of the snapshot signal is equal to the reference rate.

The snapshot signal CKVD2_{S} and the delayed reference FREF_{dly} are fed to the TDC. Since these two signals are aligned by the DTC, a narrow range TDC suffices. Also, the two inputs to the TDC—$\text{FREF}_{\text{dly}}$and CKVD2_{S}—toggle at reference rate, reducing the operating frequency and hence power consumption of the TDC. When the ADPLL is locked, the phase prediction is accurate and hence the two inputs to the TDC have a constant phase difference. Any deviation from this phase difference indicates an incorrect output frequency and the TDC output directly gives the fractional phase error PHE_{F}. Note that in the conventional architecture [31], the output of the TDC gives the fractional value of variable phase which is then compared to the fractional reference phase to obtain the phase error. In this architecture, the fractional reference phase is used to delay the reference clock and the TDC output directly gives the phase error. The output of the TDC (PHE_{F}) is assigned zero when the time (phase) difference between its inputs is equal to half the TDC range. A smaller time difference gives a negative phase error and a larger time difference results in a positive phase error.

The snapshot signal CKVD2_{S} has some constant offset from the original CKVD2 signal, which can be greater than the narrow range of the TDC used. To compensate this propagation delay, a constant delay offset is introduced in the FREF_{dly} path before it is fed to the TDC. A set of inverters implement this offset delay. The value of the required offset delay is estimated from post-layout simulations and ensures the operation of TDC at the midpoint of its range in steady state. Before the ADPLL locks, the phase prediction is not accurate. Consequently, the time difference between FREF_{dly} and CKVD2_{S} is greater than the narrow range of the TDC. Thus, the TDC acts as a bang-bang phase detector, and brings the output frequency to the neighborhood of the target frequency. Eventually, the phase difference between the

TDC inputs falls within its linear range. To avoid long settling-time due to the initial bang-bang operation of the TDC, a 16-stage TDC covering one-fifth of CKVD2 period is implemented. Once the ADPLL is locked, the phase difference of the TDC inputs is small as the DTC aligns the FREF_{dly} and CKVD2_{S} accurately, and hence falls within its range. The TDC is realized by pseudo-differential inverter based delay lines and sense-amplifier based flip-flops with identical rising and falling edge metastability windows [18].

Reference clock retiming

As discussed in Section 2-1-1, the reference clock should be retimed by the CKVD2 to avoid metastability arising from multiple clock domains. In the conventional architecture [32], the FREF signal is sampled by both the rising and falling edges of the CKVD2 and the one farthest from metastability is selected. This increases the complexity and hence power consumption. In this work, the CKR generation circuitry is simplified to just a flip-flop, since FREF_{dly} and CKVD2_{s} are synchronized owing to phase prediction. Thus, FREF_{dly} is sampled directly by CKVD2_{s} to generate CKR two CKVD2 delays after its rising edge. This provides enough processing time for TDC and phase incrementer.

2-5-3 Low-speed digital logic

The low-speed digital logic is qualitatively discussed here. Chapter 3 provides a detailed description of each of the building blocks of the low-speed logic. The low-speed logic comprises a finite state machine, interface to TX data, phase detector, and a loop filter for each of the DCO's three banks: PVT, acquisition, and tracking. The DCO is made of three banks to cover a wide frequency range with a fine resolution. The required frequency is achieved successively with PVT bank bringing the output frequency to the neighborhood of the required channel, followed by acquisition bank which acquires the required channel with a medium resolution (~2 MHz in this work). Finally, the tracking bank locks to the required frequency channel with fine resolution. The state machine (not shown in Figure 2-5) controls the sequence in which the banks are activated and the duration for which they remain active.

The FCW is used to generate the required frequency according to Eq. (2-11).

$$F_{out} = FCW * FREF \tag{2-11}$$

The integer part of the FCW poses an upper-limit on the frequency that the ADPLL can generate for a given reference clock. In this work, a 32 MHz reference is used and a maximum frequency of 2.7 GHz is required. Since the loop operates at half the DCO frequency CKVD2, $\text{F}_{\text{out}} = \frac{2.7\ GHz}{2}$. Thus the required number of integer bits of FCW is obtained as $\frac{2.7e9/2}{32e6} = 42.2$, implying that 6 integer bits suffice. However, to enable the use of a lower reference frequency if required, 7 integer bits are allotted to the integer part of FCW. The number of fractional bits of FCW determine the average resolution with which the loop can lock to the required frequency. For a 1 kHz resolution, 16 fractional bits are used ($2*32e6/2^{16}$).

The TX modulation data is added to the FCW in the low frequency path (FM_{LF}) and to the DCO control word in the high frequency path (FM_{HF}). This two-point modulation allows the modulation bandwidth to exceed the PLL loop bandwidth. A scaling factor is added in the high frequency path to ensure the modulation data corresponds to same frequency deviation in both high and low frequency paths. It is required to compensate for the variations in the DCO gain, which is susceptible to PVT variations.

A difference-mode architecture is used for the phase detector. It is functionally similar to the one described in Section 2-1-1 except for the separation of phase error accumulation from the phase detection. In this structure, the variable frequency is obtained first from the variable phase, PHV, to compute the frequency error. The phase error is then obtained by accumulating the frequency error. It eases the implementation of zero-phase restart which is required at the mode switch-over from PVT to acquisition bank and from acquisition to tracking bank (see Section 3-1-4). Moreover, unwanted perturbations on the frequency error can be neglected by freezing the phase error.

The DTC control is generated from the accumulated fractional FCW, PHR_F. The required delay is normalized to the loop-operation frequency CKVD2 using K_{DTC} obtained from the DTC gain calibration block. The digital loop-filter of Figure 2-5 is actually a set of three loop-filters, one for each bank. The loop filter for PVT and acquisition banks is simply a proportional path leading to a type-I PLL operation in these modes. Type-I operation results in a faster frequency acquisition but poor phase-noise. Since the phase noise is not important in this interval, type-I operation is preferred to improve settling time. The loop-filter for the tracking bank has proportional and an integral path, leading to a type-II operation in this mode. An IIR filter is also implemented to provide an additional out-of-band noise filtering. All three banks have gain normalization blocks to remove the impact of the DCO gain on the loop parameters.

Chapter 3

RTL design

This chapter describes the RTL design of the low-speed digital logic in Verilog. The synthesizable low-speed logic takes the variable phase, frequency command word (FCW) and other parameters for loop filter as the input and generates the control word for digital-to-time converter (DTC) and digitally controlled oscillator (DCO).

3-1 RTL design of low-speed digital logic

The digital logic of the Figure 2-5 required for processing the variable phase information to generate appropriate control signals is described at the RTL level here. Verilog HDL and VHDL are the two primary hardware languages predominantly used for digital design. Verilog HDL is chosen for this project as it is easy to synthesize the logic using the existing design flow at imec-nl. A top level block diagram of the low-speed logic is shown in Figure 3-1.

The low-speed logic comprises a finite state machine, interface to TX data, phase detector, and a loop filter for each of the DCO's three banks: PVT, acquisition, and tracking. The state machine generates appropriate control signals to the rest of the blocks. It determines the duration and order in which each of the three banks of DCO are activated after the frequency search is triggered. Also, it ensures that only one of three banks of DCO is active at any given instant.

The block "TX interface" receives the TX data and the FCW from the digital baseband and generates the input control to the low-frequency and high frequency paths to facilitate two-point frequency modulation (see Figure 2-5). The phase detector generates the integer phase error by comparing the integer part of variable phase from the phase incrementer with the required reference phase. In addition, the phase detector also computes the value of DTC control from the fractional part of FCW as discussed in Section 2-5-1 and reads the fractional phase error from the time-to-digital converter (TDC). A separate loop filter for each of the three banks then processes the phase error to generate the appropriate control words for the DCO. The implementation details of each of these blocks is described in the following section.

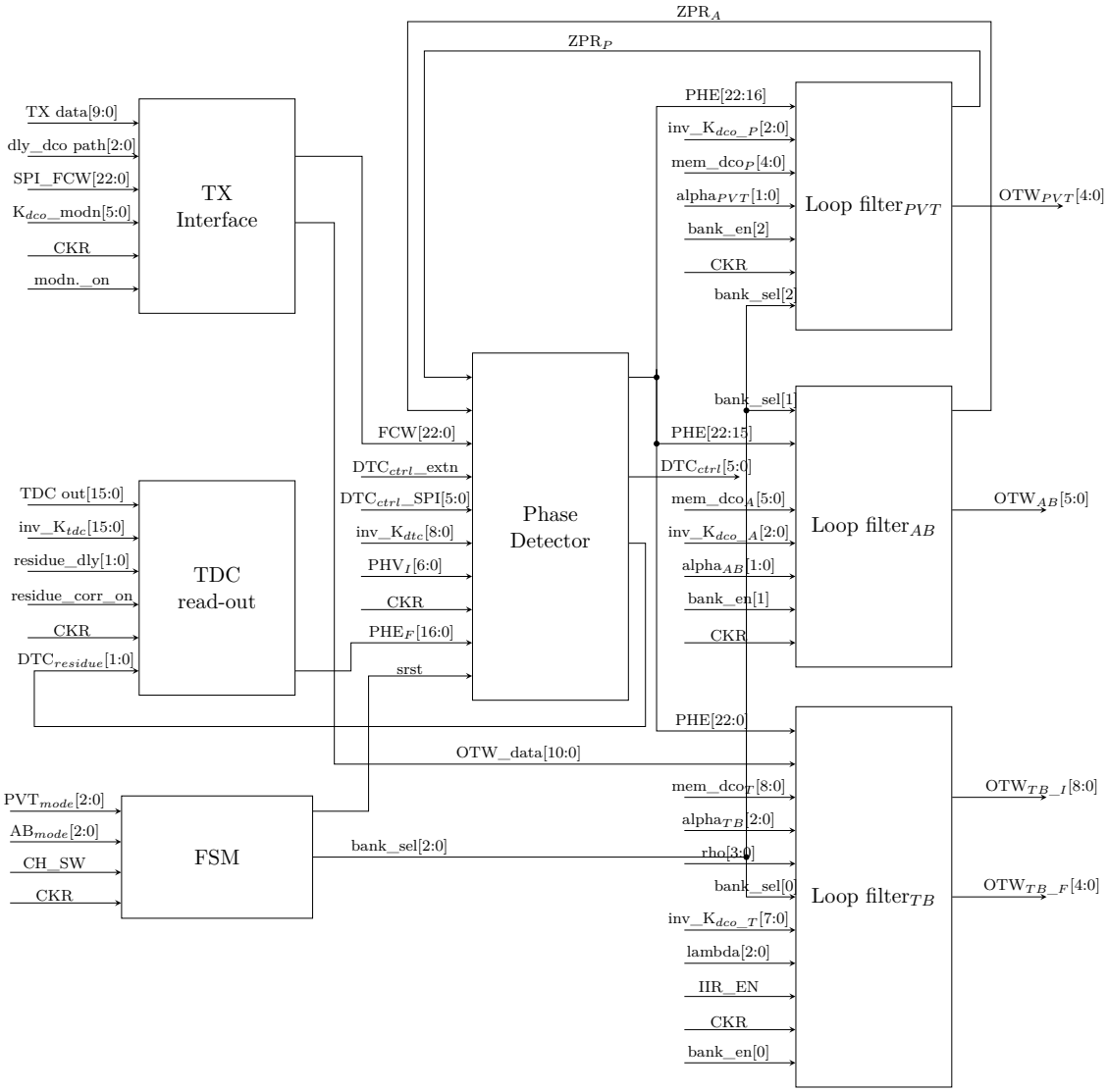


Figure 3-1: Top-level block diagram of low-speed logic.

3-1-1 Finite state machine

Figure 3-2 shows the flow chart that represents the functional behavior of the state machine.

The input channel switch, CH_SW, triggers the frequency search of the ADPLL. It is negative edge sensitive and starts the state machine in state: "000". The state machine waits for two CKR cycles in this state and moves to state: "001" while asserting the synchronous reset signal "srst". The state machine then moves to PVT mode after two CKR cycles. In the PVT mode, represented by state: "010" in the Figure 3-2, the srst signal is deasserted and the bank_en signal is set to "100". The MSB of the bank_en signal enables the PVT bank while the bit 1 enables acquisition bank and the LSB enables the tracking bank. In the PVT mode, PVT bank is enabled while the other two banks are disabled. The input, pvt_mode determines the duration of the ADPLL in PVT mode according to Eq. (3-1).

$$Time_{PVT} = (\text{PVT\_}mode + 1) * \frac{1}{FREF} * 30 \tag{3-1}$$

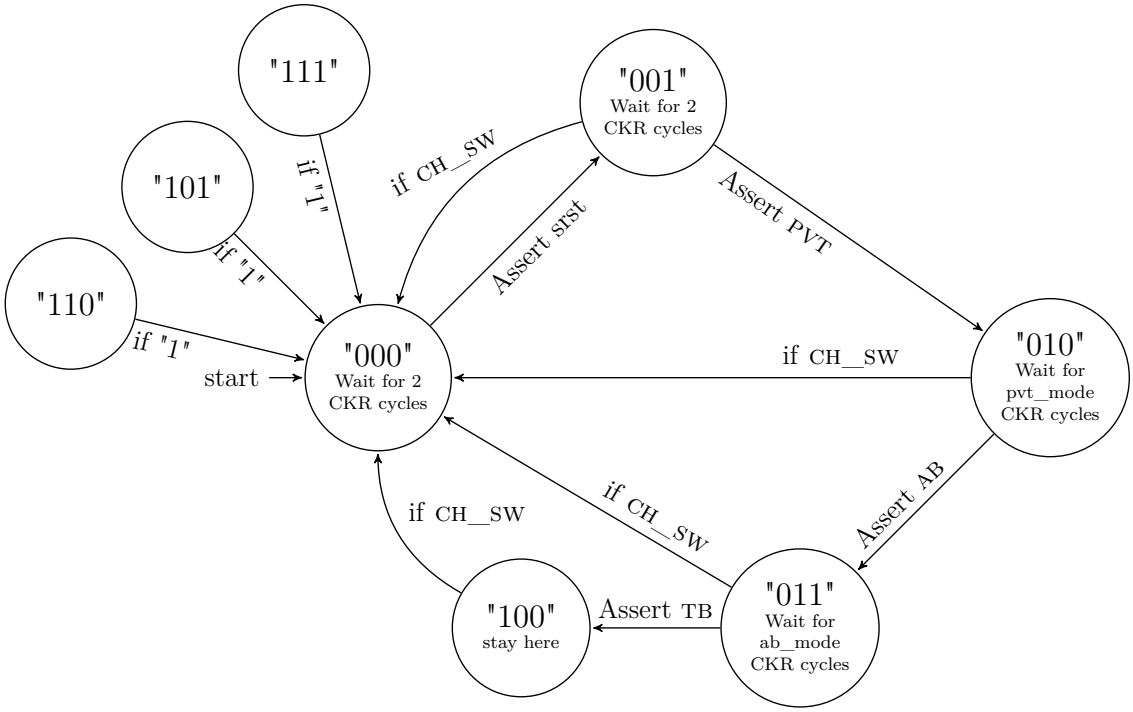


Figure 3-2: FSM of the implemented ADPLL.

In this work, a 32 MHz reference is used which corresponds to a period of around 30 ns. Thus with a 3 bit control, the duration of PVT mode can vary from 1 μs to 8 μs with a step of 1 μs and is typically set to the smallest value. After this time, the state changes to acquisition mode, state: "011". In this mode, only the acquisition bank is activated by setting the bank_en signal to "010". The input to the PVT bank is held at the previous state and the input to the acquisition bank now changes depending on the phase error. The duration of this state is controlled in a way similar to the PVT mode, with a three bit word, ab_mode. After the specified duration the state changes to tracking mode denoted as state: "100".

In the tracking mode, tracking bank is activated by setting the bank_en signal to "001" while disabling the other two banks. The inputs to the other two banks are frozen at the preceding value. The ADPLL remains in this state until the CH_SW signal is activated to acquire a new channel.

Activating the CH_SW signal, switches the ADPLL to initial state irrespective of the current state. Also, ADPLL transitions to the initial state from the unused states, "101"–"111" to avoid undefined states.

3-1-2 TX interface

Figure 3-3 shows the implementation details of the "TX interface" block of Figure 3-1. The inputs to this block are the 10-bit modulation data, "TX data", and the channel FCW that selects the channel frequency. Using these inputs, it generates appropriate inputs to the high frequency and low frequency paths.

On the low-frequency path, the TX data is added to the FCW directly to modulate the frequency. On the high frequency path, a DCO with a 60 kHz resolution and a

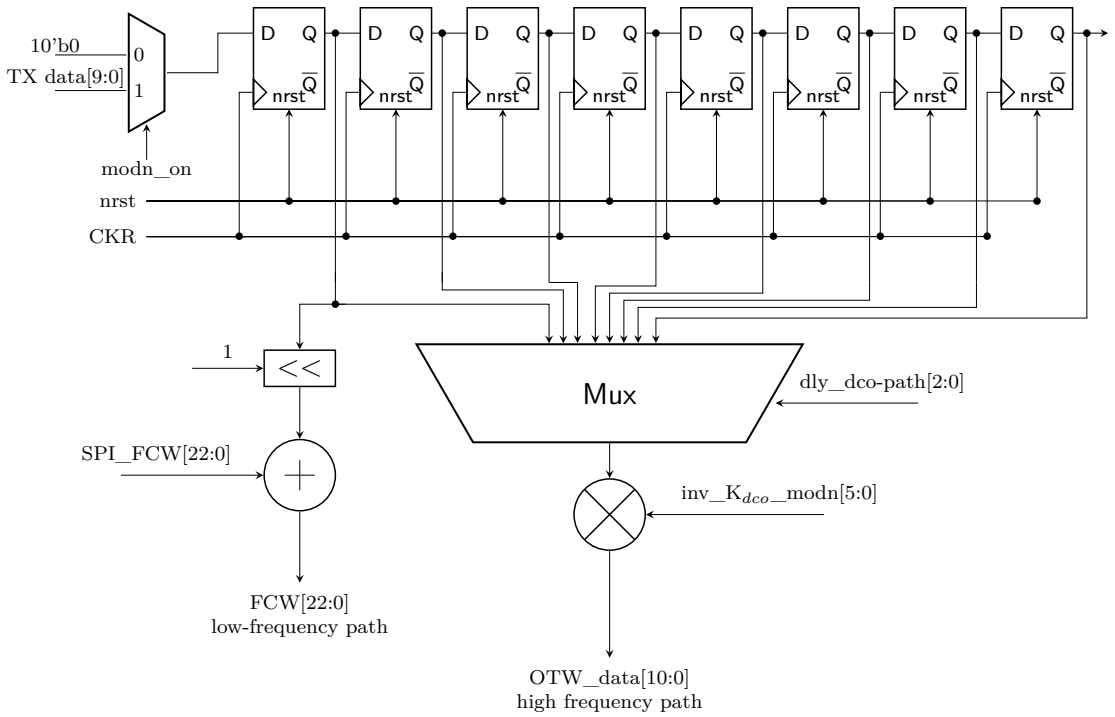
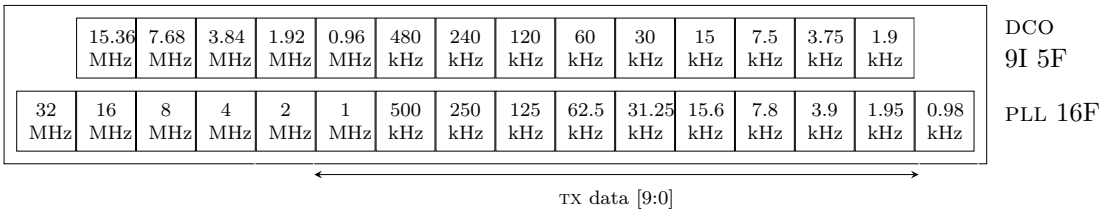


Figure 3-3: Implementation of TX interface block.

5-bit ΣΔ modulator that dithers the DCO input are used. This corresponds to an effective resolution of roughly 2 kHz ($60/2^5$). Consequently, 10-bit modulation data can support a frequency deviation of ±1 MHz with a 2 kHz resolution. Since 1 LSB of FCW corresponds to 1 kHz, the modulation data is shifted left by 1 bit before it is added to the FCW on the low frequency path. The figure below illustrates how the modulation data controls the low and high frequency paths.



The frequency corresponding to the modulation data on the low-frequency path is fixed by the reference frequency and is independent of PVT variations. However, the same is not true of the high frequency path as the DCO resolution is sensitive to PVT variations. Hence the modulation data is scaled by inv_K_{dco}_modn before it is added to the DCO control word. The scaling factor is calculated from the DCO gain according to Eq. (3-2). A 6-bit word with 5 fractional bits is used to cover a range of 0.75–2.7 with 3% resolution. Since value of K_{dco} typically reduces compared to the simulation results, more margin is kept on the higher side by adding an offset of 0.75 to the scaling factor.

$$inv\_K_{dco}\_modn = \frac{62.5\ kHz}{K_{dco}} \tag{3-2}$$

Additional delay elements (D flip-flops) are placed in the high frequency path to ensure the delay of the modulation data is same via both the paths. The 3-bit input control "dly_dco_path" selects the appropriate delayed version to be sent to the DCO in the high frequency path.

3-1-3 TDC readout

Figure 3-4 shows the internal details of the TDC readout block of Figure 3-1.

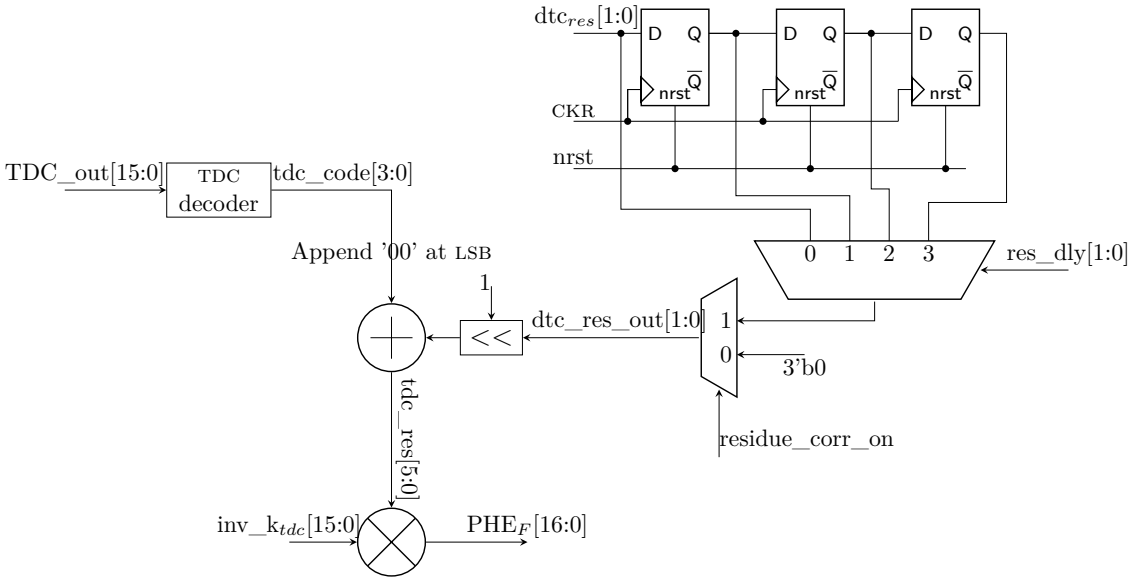


Figure 3-4: Implementation of TDC readout block.

The TDC gives the time difference between the snapshot of the CKVD2 signal, CKVD2$_S$, and the delayed reference signal FREF$_{dly}$. In the TDC, FREF$_{dly}$ signal is delayed by a series of inverters and the output of each inverter is sampled by the CKVD2$_S$ signal. Figure 3-5 shows the internal waveforms of an eight-stage TDC. The actual waveforms could be inverted when inverters are used as the delay elements. The output of the TDC is "11110000". The transition from '1' to '0' indicates the rising edge of the delayed FREF$_{dly}$ signal which occurs after four inverters in this example. Hence, the snapshot signal lags the FREF$_{dly}$ signal by a delay equal to 4 inverter delays.

In this architecture, the snapshot signal is generated by the rising edge of FREF$_{dly}$ signal and hence always lags behind it. Hence the TDC decoder output is assigned a value of 0 when the time difference is equal to half the TDC range (shown in Figure 3-5). Time difference less than this corresponds to negative phase error and more corresponds to a positive error. In the actual implementation, a 16-stage TDC is used and the output of TDC decoder, tdc_code[3:0], ranges from -8 to 7.

As discussed in Section 2-5-1, residue correction is employed to eliminate the impact of DTC quantization step on the phase noise of ADPLL. The residue corresponds to the fractional part of delay normalized to the DTC step. The DTC employs a cascade of two inverters as the delay element while the TDC uses a single inverter. Thus the DTC step is roughly twice that of the TDC and tracks over PVT variations.

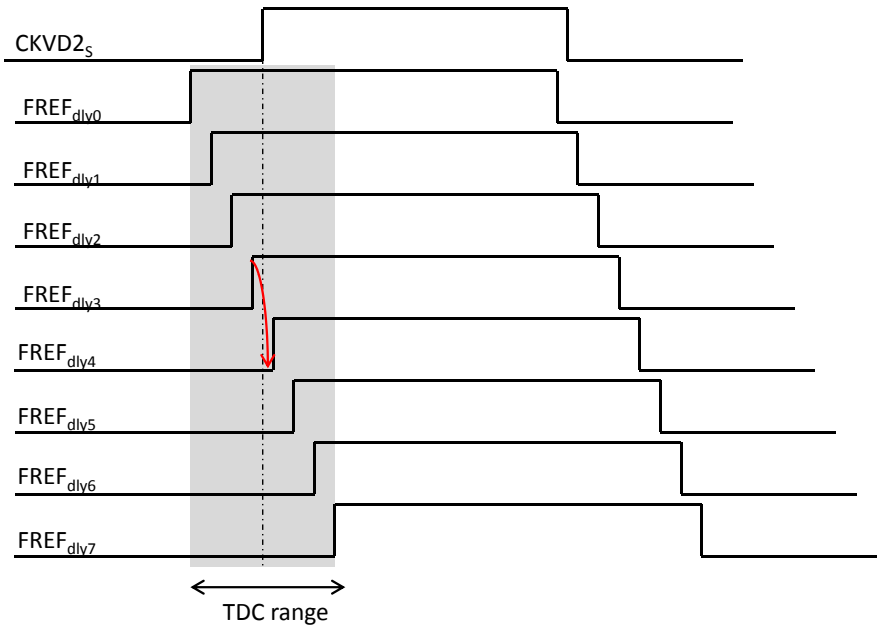


Figure 3-5: Decoding the output of TDC.

Consequently, the residue is shifted left by 1 place (multiply by 2) and two zeros (corresponding to fractional bits) are appended to the LSB of the "tdc_code" signal before they are added together. Moreover, to ensure that both the residue and the TDC output have same delay, a set of three D flip-flops are used to delay the residue. The signal, "res_dly" selects one of these delayed versions. Residue correction can be turned off by setting the res_corr_on signal to '0'.

Finally, the corrected output of TDC is normalized to the CKVD2 period to obtain the phase error. This is accomplished by multiplying it by "inv_k$_{tdc}$" which is the ratio of tdc step to CKVD2 period. Since, a fixed relation (2 times) exists between the TDC step and that of DTC, inv_k$_{tdc}$ can be deduced from the value of inv_k$_{dtc}$ which is estimated by the k$_{dtc}$ calibration block.

3-1-4 Phase detector

The phase detector implemented in this ADPLL is shown in Figure 3-6. A difference mode architecture is used in which the frequency information is first extracted from the phase to generate frequency error which is then accumulated to produce the phase error. The integer part of the variable frequency, dPHV$_I$, is obtained from the difference of two successive samples of the phase incrementer. It is compared with FCW to obtain the frequency error which is accumulated to produce the integer part of phase error.

For the fractional part, a combination of DTC and TDC is used. As discussed in Section 2-5-1, the accumulated value of the FCW$_{frac}$ is used to generate the DTC$_{ctrl}$ as well as the residue that needs to be corrected from the output of TDC. The overflow resulting from the accumulation of FCW$_{frac}$ is added to the integer FCW. The scaling by "inv_k$_{dtc}$" is required to convert the normalization factor of the delay from CKVD2 period to the DTC unit delay. To cover the required range with 1%

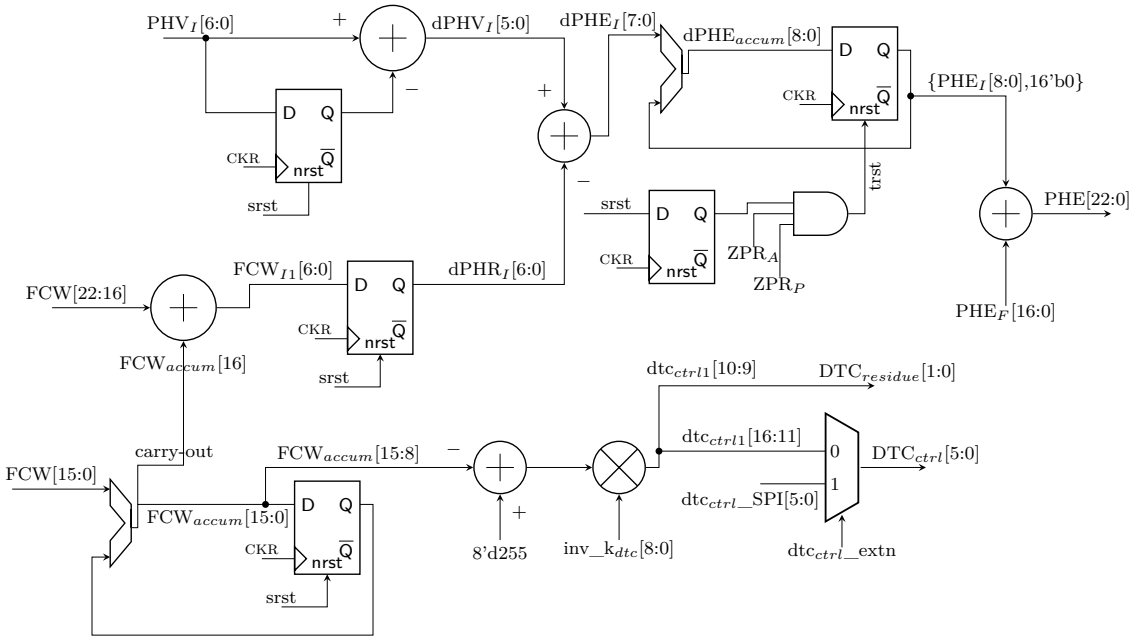


Figure 3-6: Implementation of phase detector block.

resolution, 6 integer bits and 3 fractional bits are chosen. This value of "inv_k$_{dtc}$" is obtained from the k$_{dtc}$ calibration block whose basic principle is described in Section 2-5-1. The total phase error is obtained by adding the output of TDC to the integer part of the phase error.

An additional signal "dtc$_{ctrl}$_extn" is used to facilitate the stand-alone measurements of DTC. When set to '1', the external control to the DTC, dtc$_{ctrl}$_SPI, overrides the control signal generated by the phase detect logic.

Zero-phase restart

The wide tuning range is covered with fine frequency resolution (60 kHz) by dividing the DCO into three banks: PVT, acquisition, and tracking with increasing resolutions and decreasing tuning ranges. Only one of the three banks is active at a time and the inputs to the banks are frozen at their previous values once the DCO mode is switched. Thus each successive bank operates on the excess phase error left by its immediate coarse-bank. Consequently, the phase error that maintains the input to the coarser bank is not needed as this input is frozen at the instant of mode switchover. Hence, the accumulator that integrates the frequency error (see Figure 3-6) is reset when the DCO mode is switched from PVT to acquisition or from acquisition to tracking. The signals, "ZPR$_P$" and "ZPR$_A$" indicate the mode switchover and reset the accumulator accordingly. The flip-flop used to delay the integer variable phase, PHV$_I$ is reset by a synchronous reset signal which is delayed by 1 CKR period to generate "trst". This delayed reset signal, trst, resets the accumulator that integrates the frequency error. This ensures that the accumulator starts to add only after the correct frequency error is available, irrespective of the instant at which the phase incrementer is reset.

3-1-5 Loop filter—PVT bank

Since the three banks of the DCO have different range and resolution, different bits of the phase error are used to generate their input control words. The tuning word of the PVT bank depends on the MSB bits of the phase error while that of the tracking bank is derived from the LSB bits. Also, since the PVT bank and acquisition bank are used only for frequency acquisition, the locking time is crucial while the phase noise does not matter during this time. Hence these two banks have just a proportional path resulting in a type-I PLL. Compared to a type-II PLL with an additional integral path, type-I systems lock faster but the DCO phase noise is suppressed to a lesser extent. A type-II system suppresses the DCO phase noise with a 40 dB/decade slope whereas a type-I system does so with a 20 dB/decade. Figure 3-7 shows top-level view of the loop filter implemented for each bank. The

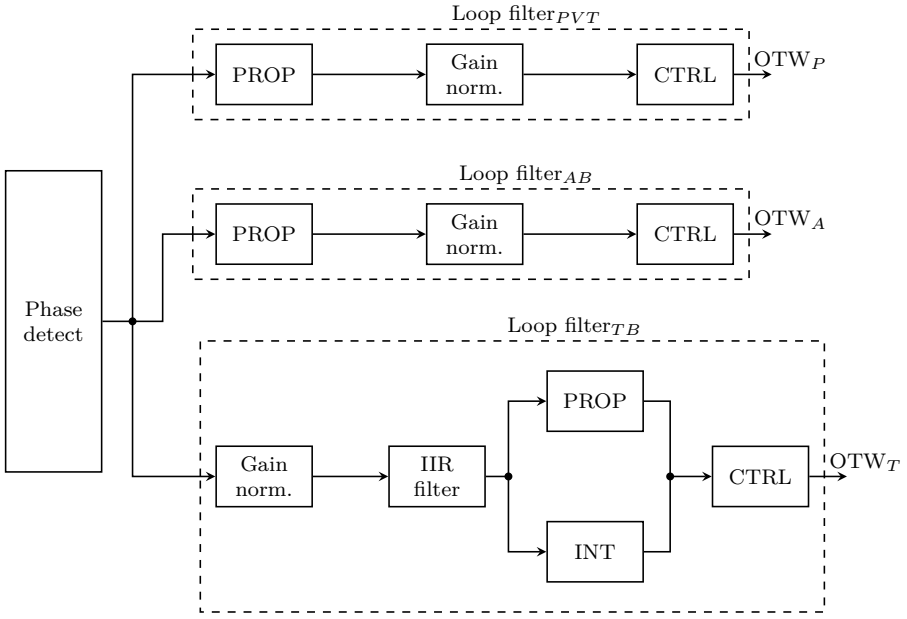


Figure 3-7: Loop filter—top level block diagram.

PVT and acquisition banks have a proportional path followed by a normalization and a control block. The tracking bank has an IIR filter and an integral path in addition. The gain normalization blocks normalize the DCO control word with the K_{dco} of the corresponding bank to avoid the impact of DCO gain variations on the loop transfer function. The control blocks implement the functionality to generate the zero-phase restart signals at mode switch-over and also to freeze the tuning word once the DCO mode is changed.

Figure 3-8 shows the implementation of the proportional, normalization, and control blocks of the PVT bank. The proportional constant, α, is chosen as a negative power of 2 as this reduces the hardware complexity by replacing the multiplication by a simple right shift. The number of right shifts can be programmed from 1 to 3 by setting the value of "alpha$_P$". The fractional bits of the phase error are neglected as they do not change the tuning word even with the minimum value of $\alpha = 1/2^3$.

The DCO gain normalization is done by multiplying the output of proportional path with $inv\_k_{dco\_P} = FREF/k_{dco\_P}$. The value of k_{dco} should be evaluated at half the DCO frequency as the loop operates at CKVD2. This normalizes the frequency

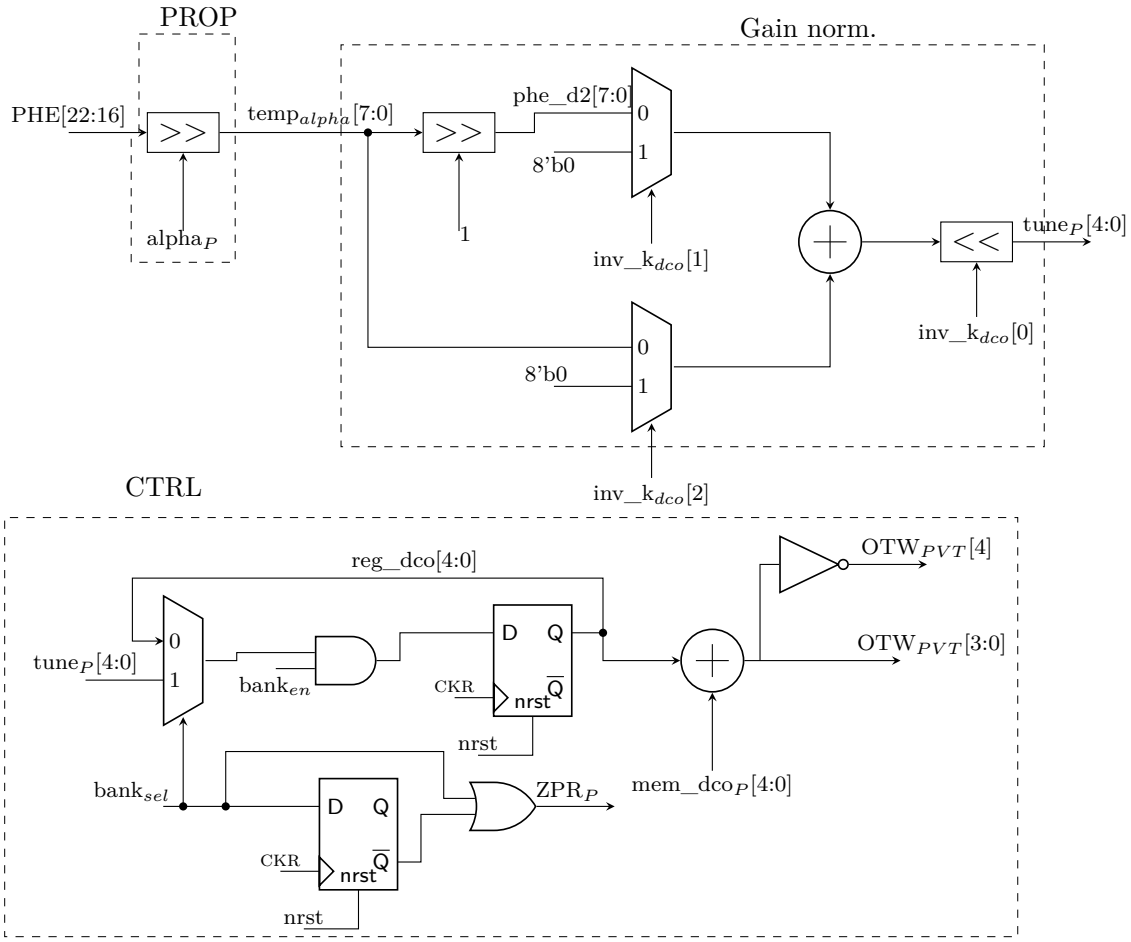


Figure 3-8: Loop filter and normalization for PVT bank.

information to reference frequency and decouples the loop transfer function from PVT variations. Since high accuracy is not required in the PVT bank, this normalization is implemented by a mantissa multiplier. A series of shifters and adders comprise the multiplier as shown in Figure 3-8. It implements the following relation:

$$multiplier = \left(inv\_k_{dco\_P}[2] + \frac{inv\_k_{dco\_P}[1]}{2}\right) * 2^{inv\_k_{dco\_P}[0]} \tag{3-3}$$

The LSB of the normalization factor is the exponent while the two MSB form the mantissa. For typical values of FREF = 32 MHz and k_{dco_P} = 21.5/2 MHz, $inv\_k_{dco\_P}$ is around 3.

The control block has the following three functions:

Freeze the PVT input on mode switch-over

The signal $bank_{sel}$, when deasserted, freezes the value of tuning word at reg_dco. It is held permanently until reset or $bank_{sel}$ is reasserted.

Generate zero-phase restart signal

The output of the OR gate in the Figure 3-8 is always '1' except the instants when the value of $bank_{sel}$ changes. Thus, a negative pulse of width equal to the CKR period is generated to indicate the mode switch-over. This negative pulse is used to clear the contents of the phase error accumulator.

Facilitate open-loop testing of the DCO

The signal bank_{en}, when zero, sets the value of tuning word calculated by the preceding blocks to 0. Consequently, the loop has no effect on the control word of DCO and is effectively open. The DCO can then be tested by varying the value of mem_dco_P externally for instance via SPI. mem_dco_P can also be used to start the DCO at a frequency different from the center frequency on reset, thereby reducing the lock time.

Finally, the MSB of the tuning word is inverted to convert the signed data into unsigned which then controls the PVT bank of the DCO.

3-1-6 Loop filter—Acquisition bank

To generate the tuning word of acquisition bank, only the MSB of the fractional phase error is used along with 3 integer LSB bits. Before the PLL is locked, the edge prediction is not accurate and hence the TDC goes out of its narrow range, thereby behaving like a bang-bang phase detector. Hence, only the MSB indicating the sign of the fractional phase error is used. The proportional, normalization, and control blocks are similar to those of the PVT bank except for changes in bit lengths (see Figure 3-9). Since the value of "alpha_A" is at least 1, it is absorbed into inv_k_{dco_A} to reduce its bit length. The normalization factor is hence $\text{FREF}/(2*\text{k}_{dco\_A})$, where k_{dco_A} should be evaluated at half the DCO frequency. The proportional path then shifts the input phase error right by (alpha_A-1).

The multiplication is similar to that of Eq. (3-1-5), except for an additional offset of 4.

$$multiplier = 4 + \left(inv\_k_{dco\_A}[2] + \frac{inv\_k_{dco\_A}[1]}{2}\right) * 2^{inv\_k_{dco\_A}[0]} \tag{3-4}$$

The control block is also similar and generates the zero-phase restart signal on mode switch-over, freezes the tuning word at the instant of switch-over, and facilitates the open-loop testing.

3-1-7 Loop filter—Tracking bank

After the acquisition bank is disabled, the frequency is locked to within one LSB of the acquisition bank. The tracking bank then remains active until a new frequency channel is required. The phase noise of the ADPLL is crucial in this interval and hence a type-II loop filter is used. The control word for the tracing bank is generated by the 9 MSB bits of the fractional phase error. Figure 3-10 shows the implementation details of the loop filter, normalization, and control block for the tracking bank.

$\mathcal{Z}$-domain analysis of the phase-locked loop (PLL) gives the dependence of loop bandwidth on the proportional constant, α, according to the Eq. (3-5) [32]. This assumes that the DCO again is accurately estimated.

$$BW = \frac{\alpha}{2\pi} f_R \tag{3-5}$$

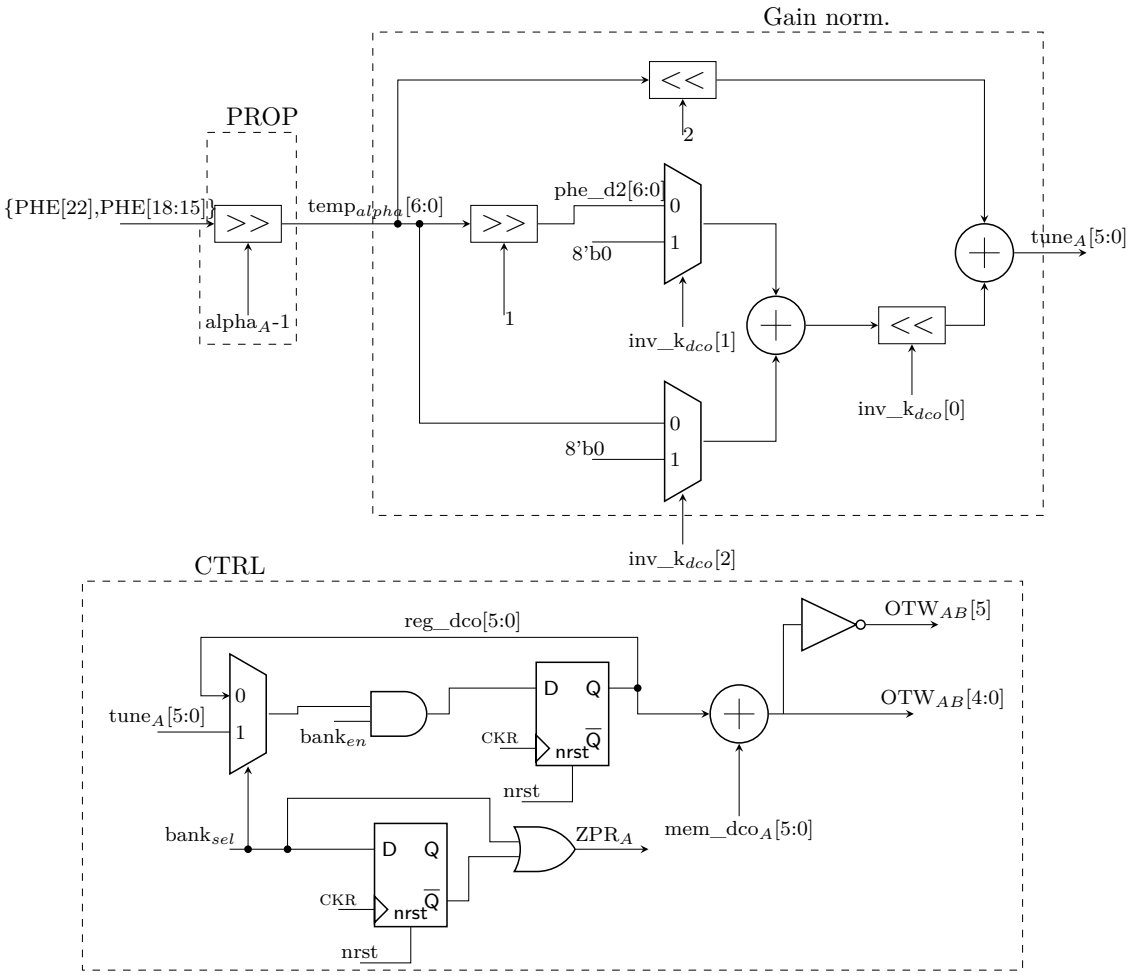


Figure 3-9: Loop filter and normalization for acquisition bank.

In this work, $\alpha = 2^{-alpha_T}$ and is restricted to negative powers of 2 to simplify the hardware implementation. It can take values from 2^{-3}–2^{-7}. Accordingly, the loop bandwidth can vary from 637 kHz–40 kHz. The value of integral loop constant, ρ, affects the settling behavior of the PLL. The damping factor ζ of the system is given by:

$$\zeta = \frac{1}{2}\left(\frac{\alpha}{\sqrt{\rho}}\right) \tag{3-6}$$

For an optimum settling behavior, $\zeta = 1/\sqrt{2}$, and hence ρ should vary from 2^{-7}–2^{-15}. The value of $\rho = 2^{-rho}$ is made programmable and can take only negative powers of 2. The integral path can be disabled by setting the signal T2 to zero.

Since the lowest value of alpha$_T$ is fixed at 3, these 3 right shifts are absorbed into the value of inv_k$_{dco_T}$ to reduce its bit length. Hence, the value of normalization factor is divided by 2^3 as shown in Eq. (3-7). The alpha$_T$ and rho are accordingly subtracted by three before they shift the input tuning word.

$$inv\_k_{dco\_T} = \frac{f_R}{8 * k_{dco\_T}} \tag{3-7}$$

An additional IIR filter is used to further suppress the out-of-band phase noise. The IIR filter implements the following difference equation and the implementation is

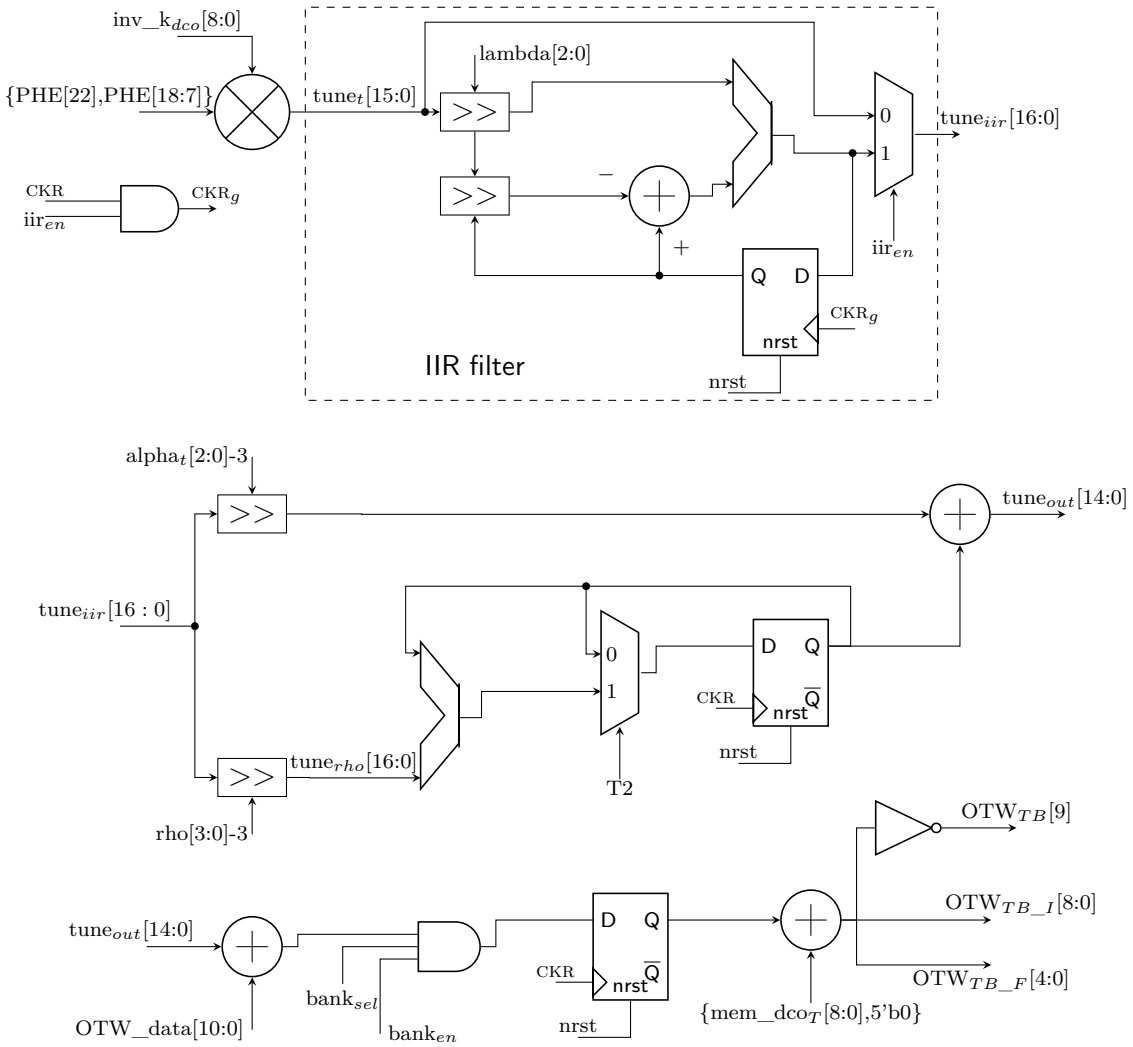


Figure 3-10: Loop filter and normalization for tracking bank.

shown in Figure 3-10.

$$y[k] = (1 - \lambda)y[k-1] + \lambda x[k] \tag{3-8}$$

where the value of $\lambda = 2^{-lambda}$ is limited to the negative powers of 2 and determines the location of the additional pole that is introduced by the IIR filter according to Eq. (3-9).

$$f_P = \frac{\lambda}{2\pi} f_R \tag{3-9}$$

As a rule of thumb, this pole is usually set at a frequency ten times to that of the pole introduced by the proportional path. If needed, the IIR filter can be by-passed by setting the signal iir$_{en}$ to 0. The clock is gated with this enable signal to ensure that the IIR filter does not burn any power when disabled.

The two-point modulation requires that DCO gain be estimated and normalized accurately. Hence, a full multiplier is used for gain normalization unlike the simplified add and shift structure used for the other two banks. The output of the proportional and integral path is summed to obtain the control word for the tracking bank. The modulation data, if any, is added to this word to directly modulate the DCO.

Also, bank_{en} signal can be set to '0' to facilitate open-loop testing similar to the other banks. The tracking bank can then be tuned by changing the external word, mem_dco_T.

The MSB is inverted to convert the signed word into unsigned and the integer bits are then sent to the tracking bank of the DCO. The 5 fractional bits are fed to the first order ΣΔ modulator to improve the DCO resolution and hence the quantization noise as discussed in Section 2-2-2

ΣΔ modulator

A 5-bit first order ΣΔ modulator is used to reduce the quantization noise from the DCO. Figure 3-11 shows its block diagram. A multiplexer selects the dithering clock

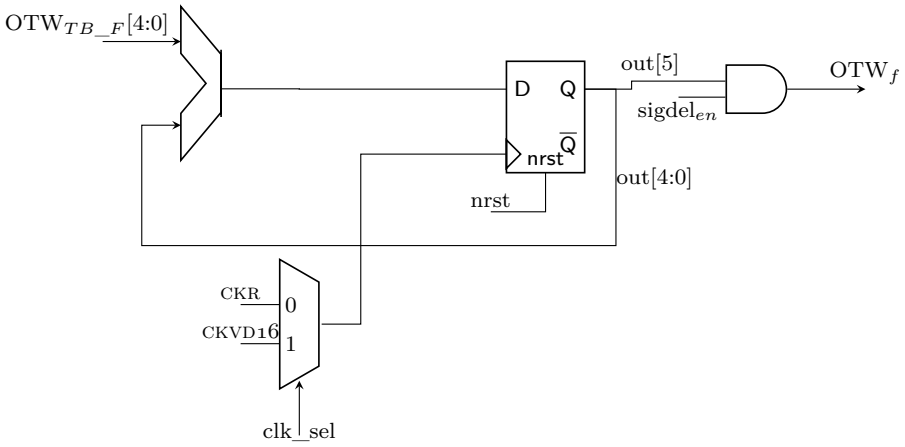


Figure 3-11: First order ΣΔ modulator.

between CKR or CKVD16 depending on the requirements. For instance, in receive mode, good out-of-band phase noise is required and hence a higher dithering rate may be preferred. The signal clk_sel selects one of these two clocks. The divide-by-2^n signals of the DCO output signal are available from the phase incrementer used to generate variable phase.

3-2 System simulation in Verilog

The high-speed blocks—DCO, divider, and phase incrementer—are modeled behaviorally in Verilog. Time-domain model of DCO as described in [49] is used for this purpose. The low-speed blocks requiring custom design i.e., TDC, DTC, and snapshot circuits are also modeled. These models are used together with the RTL level description of low-speed logic to perform system level simulations in Verilog.

Figure 3-12 shows important signals from a Verilog simulation of ADPLL. The signal "bank_sel" changes from '4' to '2' to '1', indicating the transition of ADPLL from PVT to acquisition to tracking state. The zero-phase restart pulses at each transition can also be seen. The frequency of the ADPLL and input controls to each bank are also shown. The input controls to each bank change only when the corresponding bank is active and are frozen afterwards.

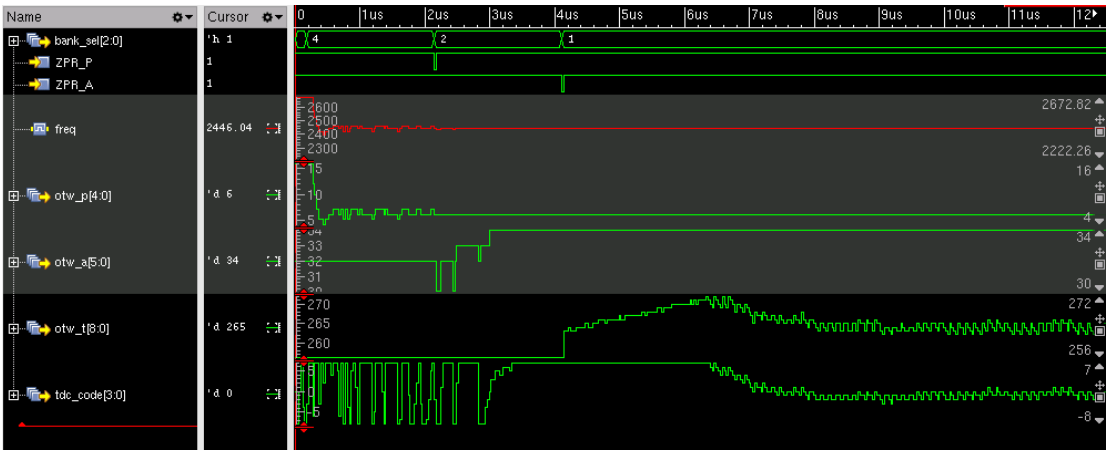


Figure 3-12: System level ADPLL simulation.

Figure 3-13 shows the zoom-in of Figure 3-12 for the PVT state. The PVT bank brings the output frequency to within one LSB of the PVT bank to the required value. Figure 3-14 shows the zoom-in of acquisition state and that of tracking state is shown in Figure 3-15.

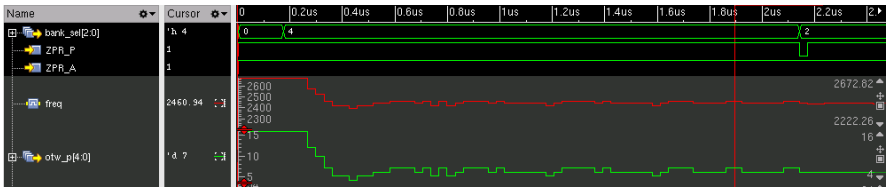


Figure 3-13: System level ADPLL simulation—PVT state.

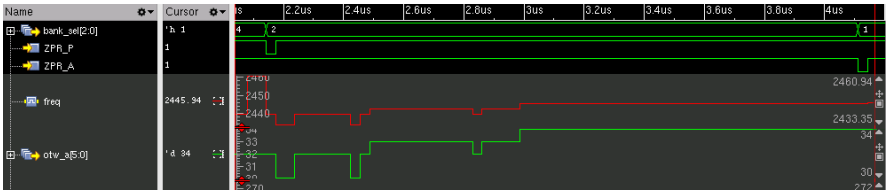


Figure 3-14: System level ADPLL simulation—acquisition state.

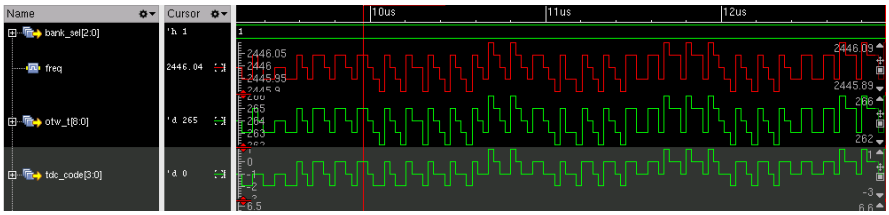


Figure 3-15: System level ADPLL simulation—tracking state.

The TDC output code and hence the tracking bank input deviate by large values at the start of frequency acquisition. This is the result of incorrect phase prediction at the start which leads to bang-bang operation of the narrow-range TDC. As the output frequency becomes closer to the target frequency, the phase prediction becomes accurate and the TDC operates at its center. Eventually, the output code of TDC

oscillates around zero. This is observed in the simulation as shown in Figure 3-15.

3-3 Estimated power consumption

Cadence Encounter is used to synthesize the implemented RTL level low-speed logic. After place and route, the extracted netlist is used along with the activity file generated from the system level simulation to estimate power consumption using Synopsis PrimeTime. The estimated power breakdown of the low-speed digital logic is shown in Table 3-1.

Table 3-1: Estimated power breakdown of the low-speed digital logic.

Block	Power consumption (μW)
TX interface	14
TDC readout	9
FSM	2
Phase detector	17
Loop filter—PVT	4.5
Loop filter—acquisition	5
Loop filter—tracking	125
Digital loop filter	50
Gain normalization	25
IIR filter	40
K_{DTC} calibration	35
DTC decoder	35
Total	250

Chapter 4

Circuit design

In this chapter, the circuit design and physical layout of the analog/high-speed blocks: digitally controlled oscillator (DCO), DCO buffer, divider, and phase incrementer, are described with a focus on lowering the overall power consumption.

4-1 Digitally controlled oscillator (DCO)

As discussed in Section 2-1-1, a DCO is at the heart of an all-digital phase-locked loop (ADPLL). It takes a digital input control word called the oscillator tuning word (OTW) and generates a sinusoidal output with a frequency proportional to OTW plus an offset. The performance of the frequency synthesizer—area, power, and spectral purity—depends to a large extent on that of the DCO. Hence considerable effort is put into its design, which is discussed in this section. The general design considerations, namely power, phase noise, and tuning range, are first discussed followed by a description of its building blocks —inductor, active part, and capacitor bank. The physical layout, which affects the overall performance is also discussed and the simulation results with post-layout parasitics are presented.

An LC tank based oscillator with a fixed inductor and digitally tunable capacitor banks is used to realize the DCO. The oscillation occurs at the resonance frequency of the LC tank. A positive-feedback or a negative resistance is used to compensate the energy-loss resulting from the parasitics in the LC tank, thereby sustaining the oscillation. The frequency tuning can be achieved either by changing the inductor or the capacitor. However, it is easier to change the capacitor rather than an inductor in monolithic implementations. Moreover, adding switches to the inductor will further degrade the already low quality factor of the on-chip inductors. Hence, a large number of capacitor cells controlled by the OTW is used to achieve the digital control of the oscillator frequency. The frequency of the DCO with N capacitor cells and an inductance L is given by Eq. (4-1).

$$f = \frac{1}{2\pi\sqrt{L * \sum_{i=0}^{N-1} C_k}} \tag{4-1}$$

In this work, a DCO covering a frequency range of 2.4–2.7 GHz with a phase noise of -110 dBc/Hz at 1 MHz offset from the oscillating frequency and a raw frequency resolution of 60 kHz is targeted. The design goal is to achieve this phase noise performance over the required tuning range with minimum power consumption. The design variables are the choice of inductor and capacitor values, width and length of the transistors in the active part and bias current.

4-1-1 Inductor

The power dissipated in the LC tank needs to be replenished so as to sustain the oscillation, which places a lower limit on the power consumption of the DCO. This loss power can be calculated by a simple tank model as shown in Figure 4-1. In

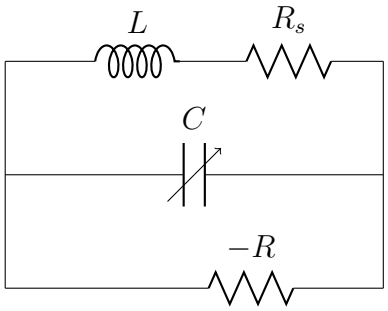


Figure 4-1: RLC model of the LC tank.

the figure, the losses in the inductor due to the finite resistance of the metal is represented as a series resistance R_s. The quality factor of the on-chip inductors is usually worse than that of the capacitor banks and hence they dominate the overall energy loss of the LC tank. Hence, the capacitor losses are neglected. The negative resistance is used to model the active element that compensates the LC tank loss. The maximum energy stored in the inductor and capacitor is equal and is given by Eq. (4-2).

$$E_{max} = \frac{1}{2}LI_{peak}^2 = \frac{1}{2}CV_{peak}^2 \tag{4-2}$$

where, V_{peak} and I_{peak} represent the peak values of the voltage across the LC tank and current through it. The current I_{peak} flows through R_s resulting in a power loss equal to

$$P_{loss} = \frac{1}{2}I_{peak}^2 R_s = \frac{1}{2}\frac{CV_{peak}^2 R_s}{L} \tag{4-3}$$

The operating frequency is usually fixed and hence Eq. (4-3) can be rewritten as

$$P_{loss} = \frac{1}{2}\frac{V_{peak}^2 R_s}{L^2\omega^2} = \frac{1}{2}\frac{V_{peak}^2}{L\omega Q_L} \tag{4-4}$$

where Q_L is the quality factor of the inductor defined as the ratio of the imaginary part of the impedance to the real part as shown in Eq. (4-5).

$$Q_L = \frac{\omega L}{R} \tag{4-5}$$

From Eq. (4-4) it is clear that for a given loss resistance and operating frequency of the oscillator, the larger the inductance the lower is the power consumption. Also, decreasing the series resistance or increasing the quality factor of the inductance lowers the power consumption.

The largest value of the inductor that can be chosen is usually limited by the requirement on tuning range or the area. For a given operating frequency, a large value of inductance results in a small value of capacitance, limiting the tuning range that can be achieved. Moreover, a large inductor is accompanied by a large coupling capacitance to the substrate, thereby decreasing the self-resonant frequency. Beyond this self-resonant frequency, the coil can no longer be used as an inductor thus imposing an upper limit on the achievable inductance value at a certain frequency in standard CMOS process. Also, it should be noted that the larger the inductance the bigger is the frequency step for a fixed capacitance step. On the other hand, the quality factor is limited by the technology (thickness and conductivity of available metal layers).

Finding the optimum value of inductance, which is just enough to satisfy the tuning range with a reasonably large self-resonant frequency is non-trivial. After estimating the amount of fixed parasitic capacitance from the capacitor banks and active part, a value of 7.7 nH is chosen after some iterations. The inductance available from the TSMC library is used in this work. The inductance with top most metal layers—which are the thickest and hence have lowest resistivity—is chosen to maximize the quality factor. Moreover, it results in lower coupling capacitance as it is farthest from the substrate.

Figure 4-2 shows the plot of the inductance as a function of frequency across typical, fast, and slow corners. The self-resonance frequency—beyond which an inductor

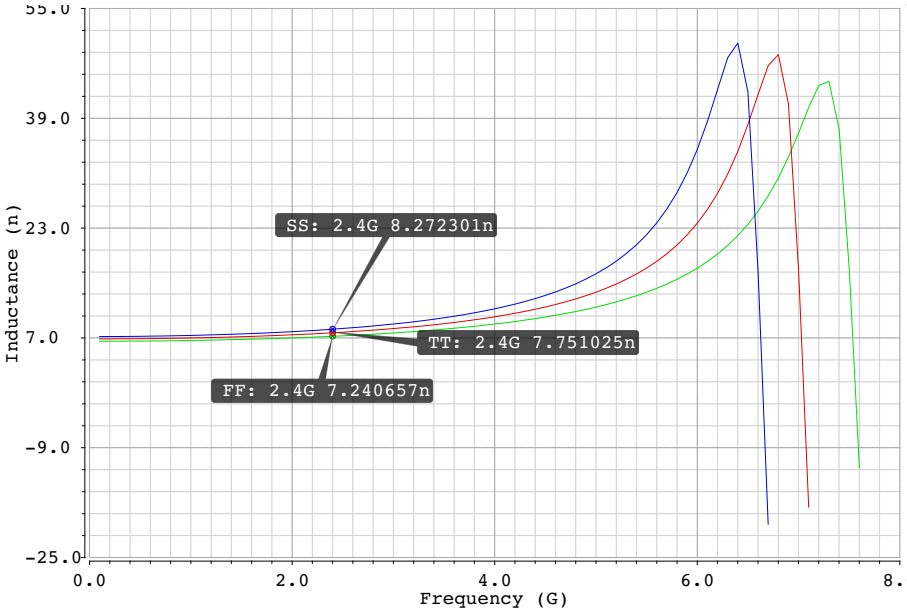


Figure 4-2: Inductance vs frequency across process corners.

behaves like a capacitor—is sufficiently far (>6 GHz) from the required operating frequencies (2.4–2.7 GHz). The value of inductance varies by almost 1 nH among different corners. This corresponds to a frequency deviation of nearly 150 MHz at

2.4 GHz. To ensure DCO supports the required tuning range across process variations, a margin of 150 MHz on either side is added to its tuning range.

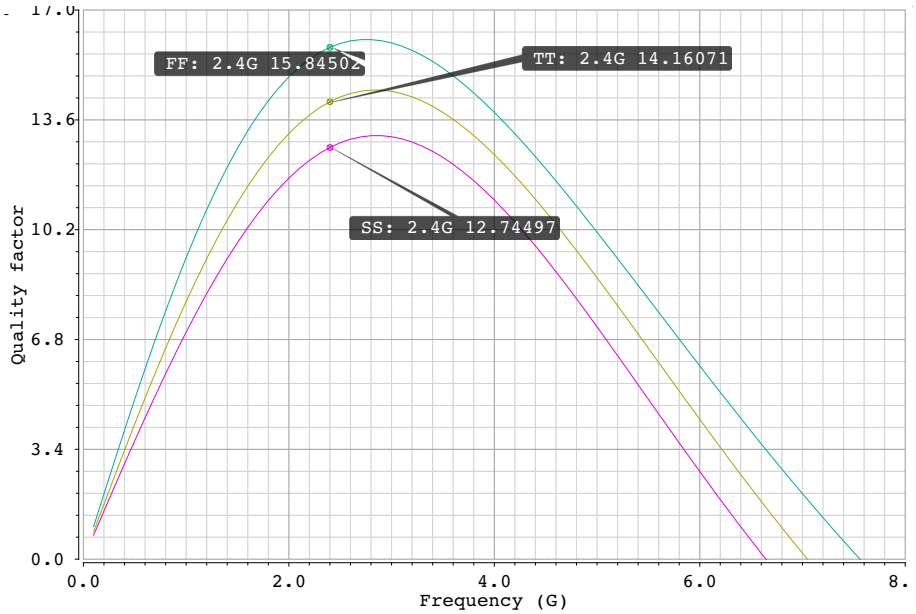


Figure 4-3: Q-factor vs frequency across process corners.

Figure 4-3 shows the plot of the quality factor of the inductance with frequency across process corners. The quality factor is around 14 in the typical case with a 10% variation across corners.

4-1-2 Active part and biasing

A complementary cross-coupled topology is used to provide the negative resistance in the circuit of Figure 4-1, which replenishes the tank losses. Compared to the structure with just an NMOS or a PMOS, complementary structure reuses the bias current, thereby providing roughly twice the transconductance for the same current. It provides a negative resistance of

$$R = -2/(g_{m,n} + g_{m,p}) \tag{4-6}$$

where $g_{m,n}$ and $g_{m,p}$ denote the transconductance of NMOS and PMOS transistors respectively.

Figure 4-4 shows the circuit schematic of the implemented DCO. Instead of a MOS current source, a poly-silicon resistor is used for biasing the DCO [50]. The current source is usually the dominating source of phase noise by means of flicker-noise up-conversion. Using a resistor avoids this and the thermal noise associated with it is filtered by means of a tail capacitor, C_{tail}.

The bias current is controlled by 5 parallel resistors connected to the supply rail by PMOS switches. A binary 5-bit digital word controls these switches, and hence the bias current. In addition, another resistor in parallel is always turned on providing a minimum offset current. The DCO current can vary from 80 μA to 450 μA at 1 V supply. The bias control can be used to implement swing voltage regulation in future designs to ensure optimum current consumption over the entire frequency range.

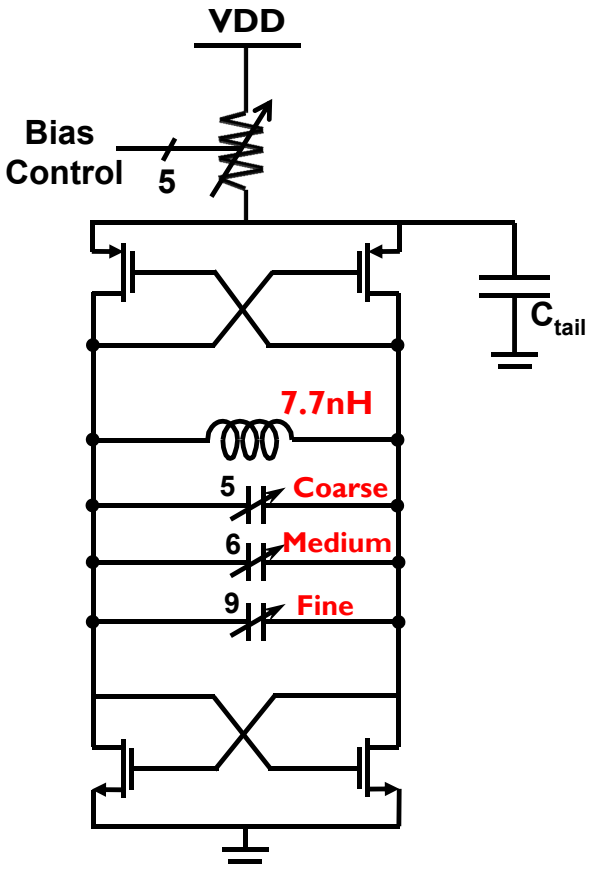


Figure 4-4: Circuit schematic of the implemented DCO.

The length of the transistors in the cross-coupled pair is chosen at 100 nm (minimum length: 40 nm) to minimize the flicker noise. The width of the transistors is increased until the g_m is sufficient to sustain the oscillation. The PMOS transistors are made thrice as wide as the NMOS devices to account for lower hole mobility. Using a large aspect ratio pushes the transistors into weak-inversion region offering a better g_m–I_d efficiency. However, it also increases the capacitance contribution from the active part to the tank, limiting the achievable tuning range. Moreover, since this capacitance comprises junction capacitance, it depends non-linearly on the node voltage, making the oscillator susceptible to supply pushing.

Figure 4-5 shows the layout of the active part of the DCO. The fingers of the cross-coupled pair are arranged in a common-centroid configuration to average out the linear process variations. Also, the transistors are laid out to have same orientation for better matching. The layout is made compact to minimize the routing parasitics at the drain terminal of the cross-coupled pair as they are absorbed into the LC tank, reducing the maximum frequency attainable.

4-1-3 Capacitor banks

Design of capacitor banks is challenging in that they need to be tunable over a large frequency range with a fine step. A tuning range of 2.25 GHz to 2.85 GHz with a 60 kHz step is targeted to leave enough margin for process variations to cover

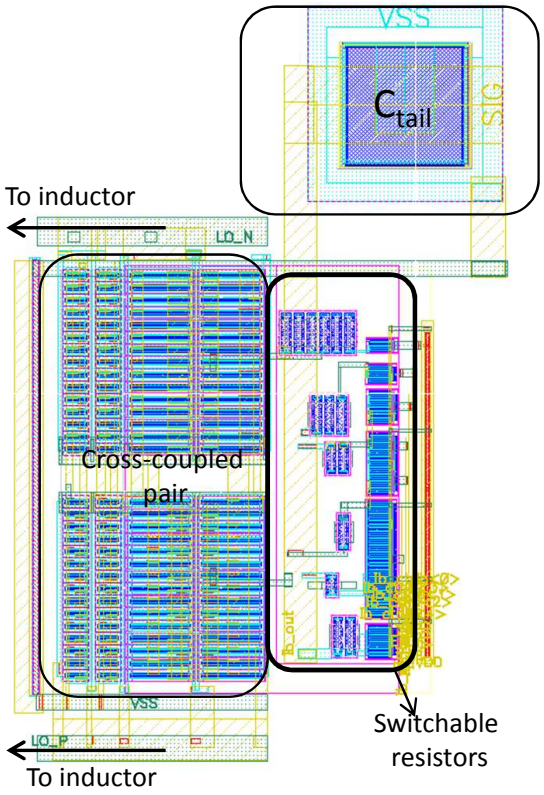


Figure 4-5: Layout of the active part of the DCO.

the required 2.4 GHz to 2.7 GHz range. This results in a resolution requirement of nearly 14 bits, which is difficult to achieve even with advanced component matching techniques. Generally, a 9-bit resolution is possible without resorting to matching techniques or digital calibration [32]. Fortunately, the fine 60 kHz resolution needs to cover a frequency deviation of at most 2 MHz corresponding to the frequency deviation required by the direct frequency modulation. Thus, a segmented approach can be used where three banks—coarse, medium, and fine—cover the desired tuning range with required resolution. The coarse, medium, and fine banks are referred to as PVT, acquisition, and tracking banks following the nomenclature used in [32].

The PVT bank covers the whole tuning range with very coarse resolution. The acquisition bank then brings the output frequency closer to the required channel frequency. The PVT and acquisition banks are active only at the start of the frequency acquisition as discussed in Section 3-1-1. Finally, the tracking bank locks to the target frequency channel with a fine resolution and needs to cover a range equal to either the step of medium bank or the modulation range, whichever is larger. On top of this, it should have sufficient margin to correct the frequency drifts in the oscillator over time.

The range and resolution of each bank is determined as follows. The range of the PVT bank is made equal to the required tuning range. The resolution of the tracking bank is determined by the required frequency resolution according to Eq. (4-7)

$$\Delta f = (2\pi^2 L f^3)\Delta C \tag{4-7}$$

A 60 kHz resolution corresponds to a frequency step of roughly 20 aF at 2.7 GHz output and 7.7 nH inductance. To have sufficient overlap between two consecutive capacitor banks, a ratio of 7–8 is chosen between the step of the coarser bank to the range of its immediate fine bank. Using these criteria, a 9-bit tracking bit with a 60 kHz step, 6-bit acquisition bank with a 2.5 MHz step, and a 5-bit PVT bank with a 20 MHz step are chosen.

Another important requirement of the capacitor banks is their quality factor. The total tank quality factor is given by the mathematical parallel combination of the inductor and capacitor quality factors. Hence, quality factor of the capacitor banks should be made large compared to that of the inductor so that overall Q is limited by that of the on-chip inductance. As mentioned earlier, a Q of 15 is achievable for on-chip inductors in the TSMC 40 nm technology. Hence, a quality factor of 50 is targeted for the capacitor banks in both ON and OFF states.

PVT bank

The PVT bank determines the tuning range of the DCO. A capacitor bank with a large frequency range, i.e., large $\left(\frac{C_{max}}{C_{min}}\right)$ ratio is desired as it minimizes the off-state capacitance. This in-turn allows the use of a larger inductance and hence results in lower power consumption as discussed in Section 4-1-1. A switched MOM configuration shown in Figure 4-6 is used a unit cell of the PVT bank as it can provide high tuning range. When the switch is enabled, the series combination of

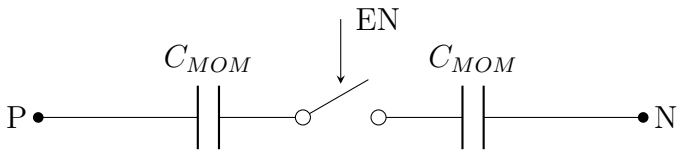


Figure 4-6: Switched MOM configuration.

two MOM capacitors, $C_{MOM}/2$, appears across the terminals P and N and the unit cell is considered to be in ON state. On the other hand, when the switch is open, the capacitance that appears across P and N is determined by the parasitics of the switch and the MOM capacitors.

Quality factor

In the ON state, the switch of Figure 4-6 can be replaced by a resistance R_{ON} of the transistor used to implement the switch. The impedance across P and N is then given by $Z = R_{\mathrm{ON}} + \frac{1}{j\omega\left(\frac{C_{\mathrm{MOM}}}{2}\right)}$ giving rise to an on-state quality factor of

$$Q_{on} = \frac{1}{\omega R_{on}\left(\frac{C_{MOM}}{2}\right)} \tag{4-8}$$

It is clear that R_{on} of the switch should be made as small as possible to have a good Q. Hence an NMOS is used instead of PMOS as the mobility and hence conductance of the electrons is 2–3 times that of holes. The ON resistance of a MOS switch can be expressed as

$$R_{on} = \frac{1}{\mu_n c_{ox} \frac{W}{L}(v_{gs} - v_{th})} \tag{4-9}$$

Eq. (4-9) indicates that the aspect ratio $\left(\frac{W}{L}\right)$ and the overdrive voltage $(v_{gs} - v_{th})$ should be maximized to reduce the on resistance. However, increasing the aspect ratio also increases the parasitic capacitance of the MOS switch and hence the off-state capacitance, thereby degrading the capacitance range. To maximize the overdrive voltage, two pull-down NMOS transistors are used to pull the DC voltage of source and drain terminals of the MOS switch to ground in ON state. The aspect ratio of the MOS switch is then increased until its Q_{on} reaches 50.

In the off-state the parasitic capacitance between drain and source terminals of the switch appears between the two MOM capacitors leading to an off-state capacitance of

$$C_{off} = \frac{C_{MOM}C_{par}}{C_{MOM} + 2C_{par}} \tag{4-10}$$

which is nearly equal to the value of parasitic capacitance. The capacitance has the following contributions: drain and source junction capacitors of the NMOS switch, drain junction capacitance of the pull-down transistors, and the parasitics from the MOM capacitor to substrate. The quality-factor in the off-state is largely determined by that of the MOM capacitors which is quite high.

The DC-voltage at the drain and source terminals of the NMOS switch is set by the leakage current through the pull-down transistors which are now turned-off. Since the DCO output swing is typically around 300 mV – 400 mV, the maximum negative voltage at the source terminal of the NMOS switch is around 200 mV. Hence, the gate-to-source voltage of the NMOS switch is always less than its threshold voltage, ensuring the switch remains off.

Figure 4-7 shows the circuit implementation of the unit cell of the PVT bank. An

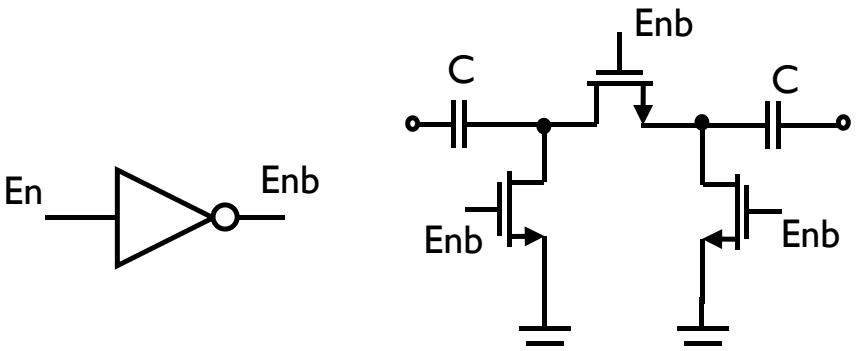


Figure 4-7: Circuit schematic of the unit cell of PVT bank.

inverter is placed between the input control and the switch-enable so that capacitance is lowest when the bank is turned on. This results in a positive k_{dco}, i.e., the frequency increases with increasing input control.

The 5-bit PVT bank is binary weighted and the cells corresponding to the higher order bits are made from multiple unit cells. The cell corresponding to bit 1 comprises 2^1 unit cells and bit 2 comprises 2^2 unit cells and so on. This ensures that the capacitance decreases monotonically with the input control. Also, the linearity is improved as systematic mismatches are reduced.

Figure 4-8 shows the physical layout of the 5-bit PVT bank. The unit cells are placed in a 4x8 matrix with 1 dummy cell. The cells are laid out such that they have a common centroid to suppress the mismatches resulting from linear gradients across the die. The layout parasitics are extracted using Calibre and the post-layout

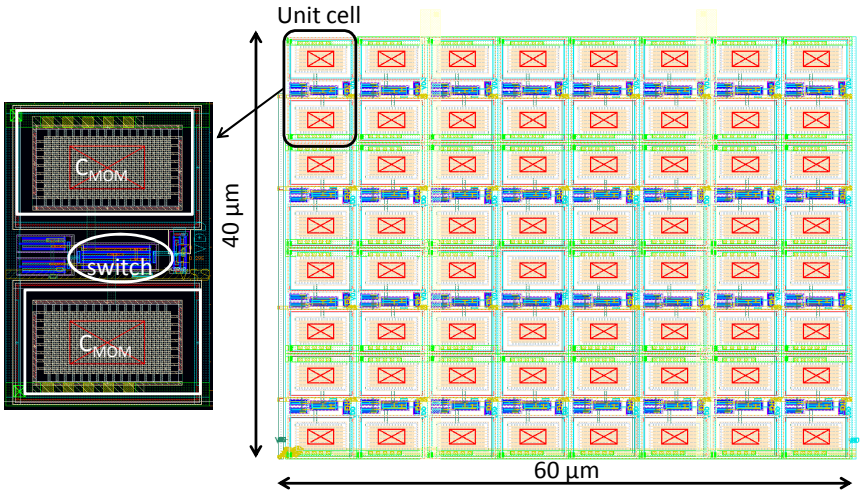


Figure 4-8: Layout of the PVT bank.

simulations to estimate the Q-factor and capacitance range are done using Cadence Spectre. The capacitance and Q-factor are calculated from the input impedance of the bank. Figure B-1 shows the C–V curve of the unit cell of the PVT bank.

The capacitance step is 8.8 fF with an off-state capacitance of 4.5 fF. The quality factor in ON and OFF states is shown in Figure B-2. As can be seen, the quality factor is limited in the ON-state and is 48, sufficiently higher than that of the inductor. Figure 4-9 shows the total capacitance range of the PVT bank including the routing parasitics. The PVT bank has a minimum capacitance step of 140 fF and covers a capacitance range of 273 fF with a 8.8 fF step.

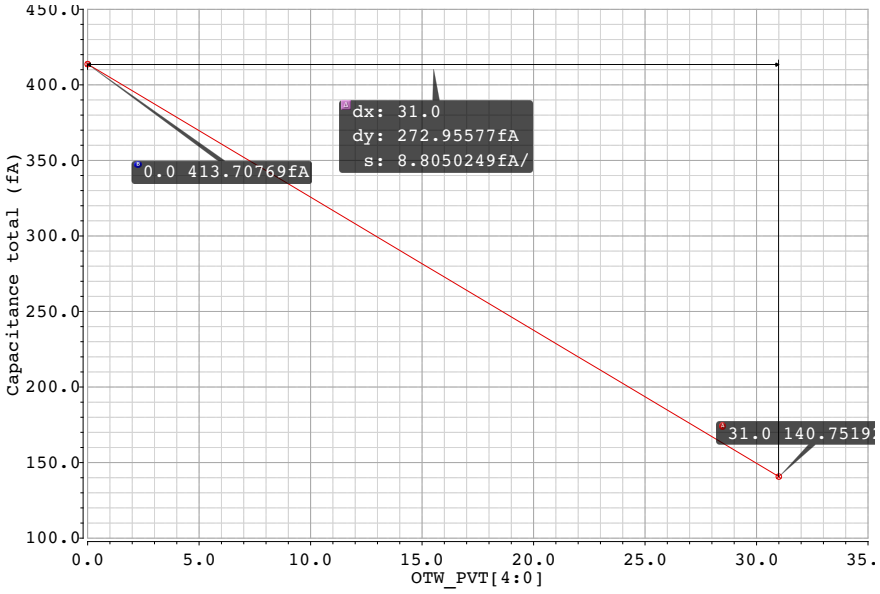


Figure 4-9: Capacitance range of the PVT bank.

Acquisition bank

The acquisition bank is implemented in a similar way as the PVT bank with smaller MOM capacitors. From Eq. (4-8), it can be seen that the on-state Q-factor increases when C_{MOM} decreases. Consequently, smaller switches are sufficient to achieve the required quality factor, resulting in a lower off-state capacitance. The 6-bit acquisition bank is realized from 63 unit cells controlled in a binary manner. The layout of the bank is shown in Figure 4-10. The unit cells are arranged in a matrix of 8x8 while ensuring a common-centroid. The acquisition bank occupies lesser area than PVT bank due to the lower C_{MOM} used. The post-layout simulations are performed

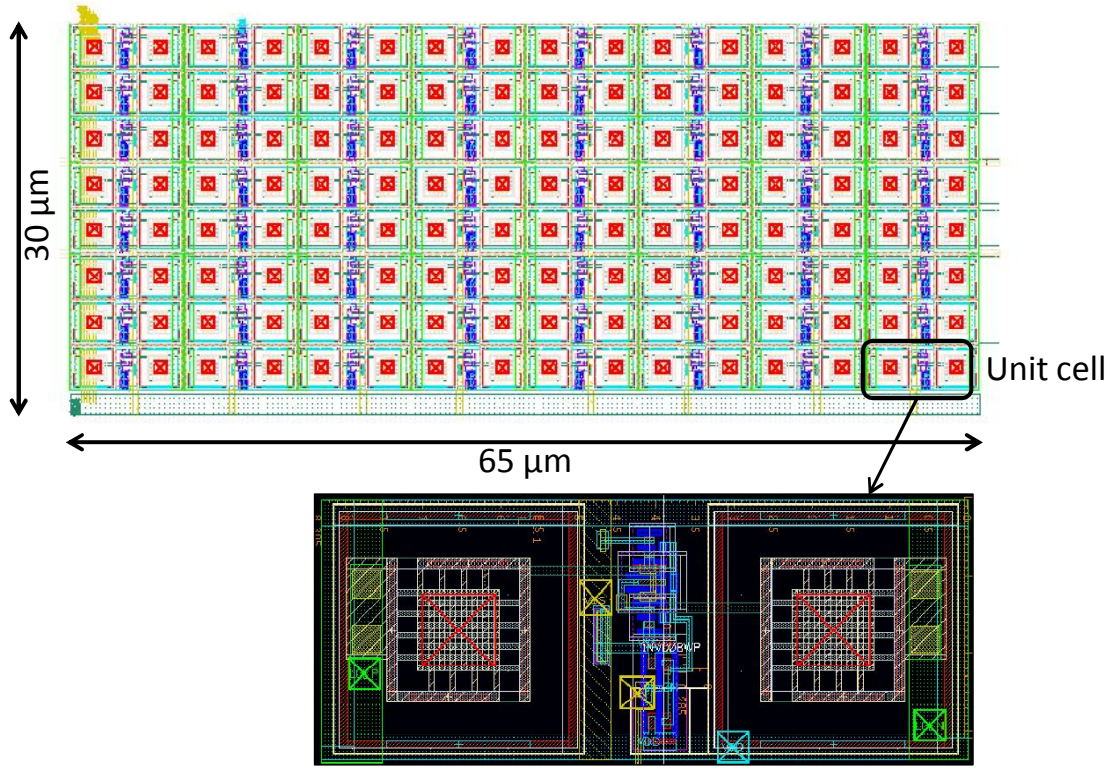


Figure 4-10: Layout of the acquisition bank.

to estimate the value of capacitance step, quality factor and capacitance range, and are shown in Figure B-3 and Figure B-4 respectively. The capacitance step of the unit cell is 1.2 fF with an off-state capacitance of 0.7 fF. The quality factor is greater than 50 in both ON and OFF states. Note that due to the presence of the inverter at the input of the switch, the ON state corresponds to low enable voltage in the simulation plots.

The total capacitance of the acquisition bank as a function of its 6-bit input control word is shown in Figure 4-11. The capacitance decreases monotonically with OTW_AB[5:0]. The minimum capacitance from the acquisition bank is only 44 fF, thanks to the smaller dimensions of the NMOS switch. The 6-bit acquisition bank has a tuning range of 76 fF.

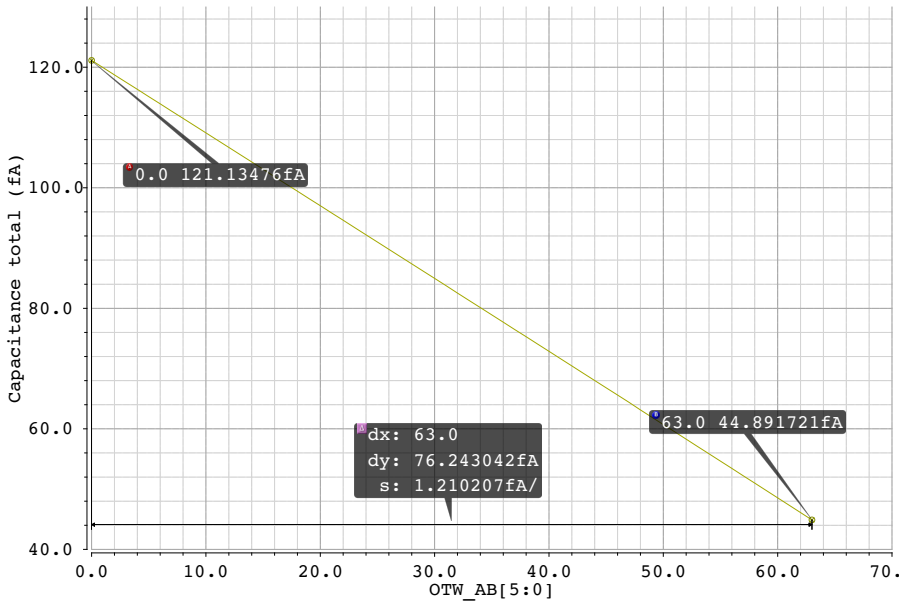


Figure 4-11: Capacitance range of the acquisition bank.

Tracking bank

Tracking bank locks the PLL frequency to the target frequency with the required resolution. The frequency resolution of this bank determines the resolution of the DCO. To achieve a 60 kHz resolution, a 20 aF capacitance is required as given by Eq. (4-7). The MOS varactors in an accumulation mode or depletion mode are the common choices to achieve this fine resolution [17, 51]. However, this requires accurate models for the varactor mode operation of the MOS transistor. Also, the C-V transfer curve of the MOS varactor is susceptible to PVT variations. Hence, fine bank is also realized using MOM capacitors which are better modeled and less prone to PVT variations.

The circuit implementation of the unit cell of the tracking bank is shown in Figure 4-12 [52]. When the control signal is high, the input to the NMOS switch is low and

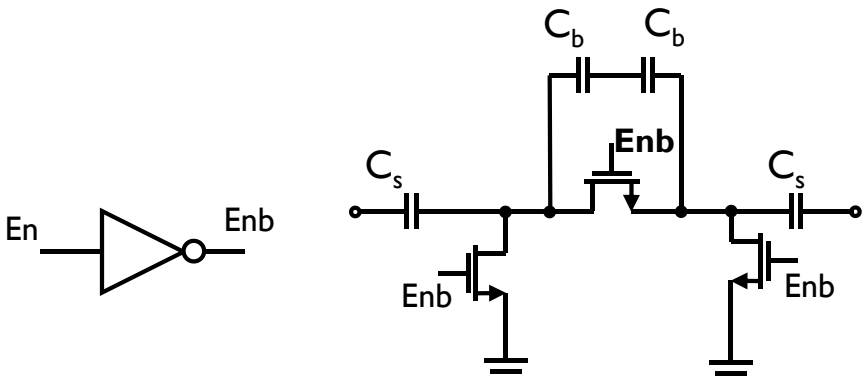


Figure 4-12: Circuit schematic of the unit cell of tracking bank.

is hence turned off. The capacitance in this state is given by the series combination

of capacitances C_b and C_s.

$$C_{low} = \frac{C_b C_s}{2(C_b + C_s)} \tag{4-11}$$

When the input control is low, the input to the switch is high leading to a higher capacitance of $\frac{C_s}{2}$. The capacitance step is then obtained by

$$\Delta C = \frac{C_s^2}{2(C_b + C_s)} \tag{4-12}$$

Thus a smaller capacitance step can be obtained by choosing a large value of C_b and a small value of C_s. In the implementation, the minimum available capacitance from the library for the MOM capacitors is chosen for C_s and the value of C_b is increased until the required capacitance step of 20 aF is achieved. This resulted in a value of 45 fF for the big capacitance C_b.

The tracking bank is used for direct frequency modulation and hence its linearity is crucial. Hence, the tracking bank is thermometer coded. This allows implementation of dynamic element matching for improved matching performance in future designs. For a 9-bit tracking bank, 511 unit cells are needed. With a large C_b of 45.5 fF, a prohibitively large area is required. This not only increases the cost but also results in increased routing parasitics. The increased parasitics limit the maximum operating frequency warranting a smaller value of the inductor and increased power consumption.

To address this issue, tracking bank is segmented into MSB cell with a step equal to eight times that of the unit cell and LSB cell. The 6 MSB bits control the MSB cells while the LSB cells are controlled by 3 LSB bits. In this way, only 70 cells—63 MSB and 7 LSB—are required as opposed to 511, which is the case without segmentation. In addition, the larger step size requirement of the MSB cells results in a smaller value for C_b according to Eq. (4-12), further reducing the area of the tracking bank.

A 6:64 thermometer decoder is required to control the MSB cells which is both power and area hungry. To avoid this, the MSB cells are arranged in a matrix of 8x8 and are controlled by two 3:8 decoders instead. The control logic along with the MSB cell is shown in Figure 4-13. The bits <8:6> of the control word are converted to 8 row signals and the next 3 bits, <5:3> control the columns. When the "row" signal is high, all the banks in that row are turned on irrespective of the value of the column signal. Only when a given row is not enabled, do the column signals determine whether the cell is enabled. Another 3:8 decoder is used to control the LSB bits. An additional LSB cell is used for $\Sigma\Delta$-dithering. In this way, a high-speed adder required to add the control word of the tracking bank and the output of the $\Sigma\Delta$ modulator is avoided. Thus 63 MSB cells and 8 LSB cells are used for implementing the tracking bank.

Figure 4-14a and Figure 4-14b show the physical layout of the LSB and MSB cells of the tracking bank. The layout of the total bank is shown in Figure 4-15. The LSB cells are placed in the first row, above the MSB cells. Again the unit cells are laid out to have a common centroid to minimize the effects of linear gradients across the die. The tracking bank occupies an area of 90 μm x 55 μm which is only slightly larger than that of the PVT bank.

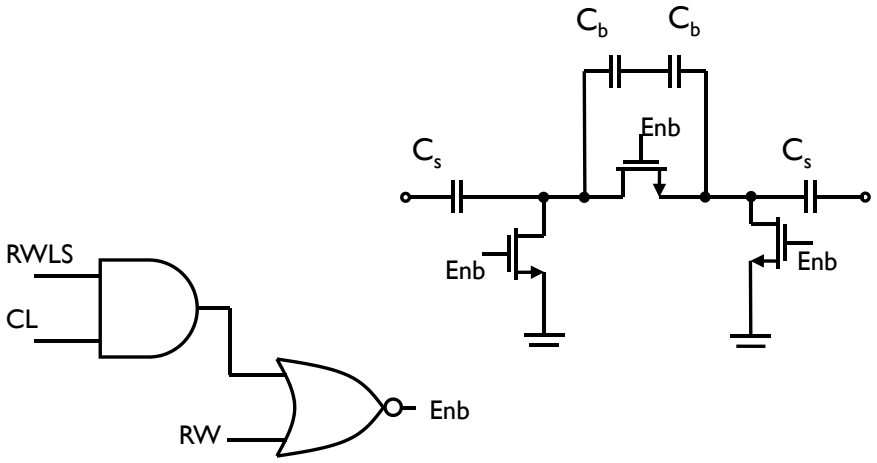
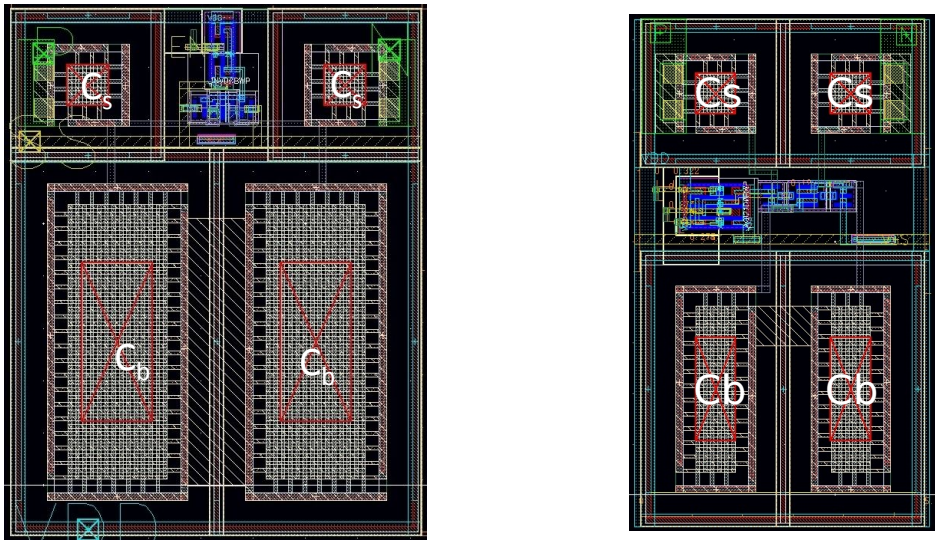


Figure 4-13: Circuit schematic of the MSB cell of tracking bank.



(a) LSB cell **(b)** MSB cell

Figure 4-14: Physical layout of the unit cell of tracking bank

The post-layout estimations of the capacitance step and quality factor of MSB cell and LSB cell are shown in Figure B-7 and Figure B-8; and Figure B-5 and Figure B-6, respectively. The quality factor of both banks is greater than 50 in ON and OFF states. The LSB cell has a step of 18 aF and the MSB cell has a step of 145 fF which is approximately eight times the LSB step.

Figure 4-16 shows the capacitance of the tracking bank as a function of its 9-bit control word. The tracking bank has significant constant minimum capacitance of 73 fF. It covers a range of 9 fF with an average step of 18 aF.

The operating range and design of the quality factors of the three capacitor banks is summarized in Table 4-1. The total off-state capacitance from all three banks is around 260 fF.

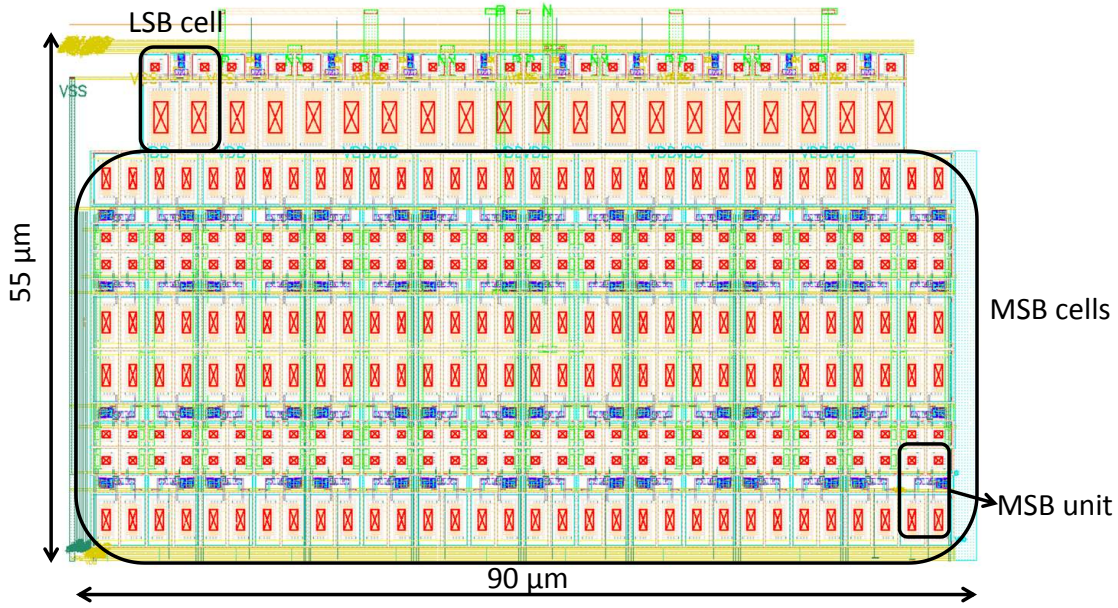


Figure 4-15: Physical layout of the tracking bank.

Table 4-1: Summary of capacitor banks.

Bank type	Bits	C_{min} (fF)	Range (fF)	ΔC	Q_{min}
PVT	5	140.7	273	8.8 fF	48
Acquisition	6	45	76.2	1.2 fF	58
Tracking segmented	9 (6+3)	73	9.5	18 aF	66

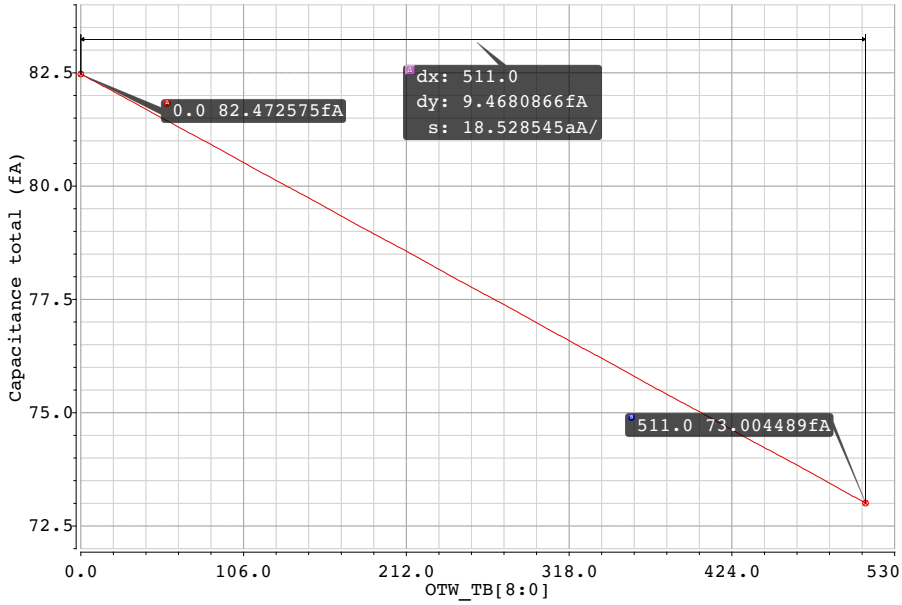


Figure 4-16: Capacitance range of the tracking bank.

4-1-4 Layout

The physical layout of the individual banks and the active part have already been presented. These blocks are arranged to minimize the routing parasitics from the inductor to capacitor banks and the active part as shown in Figure 4-17. Long

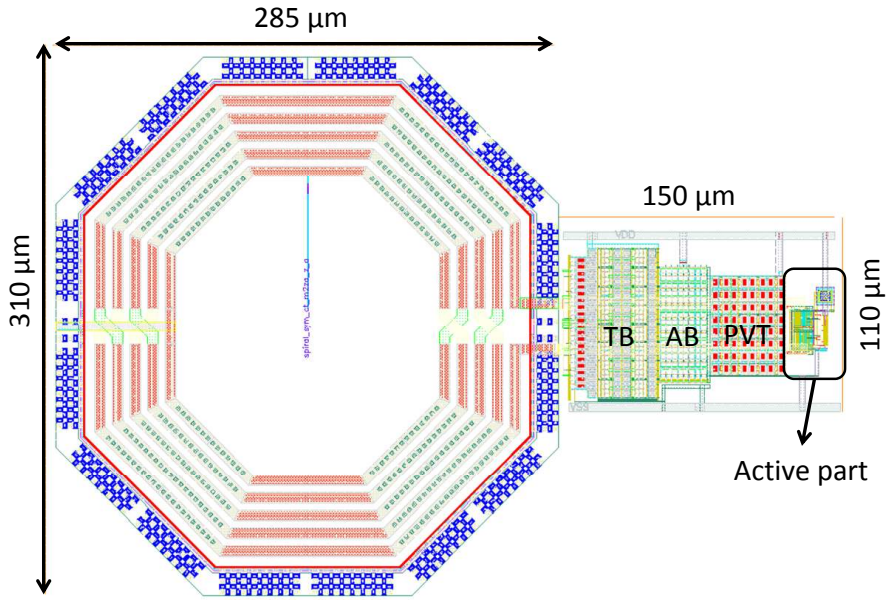


Figure 4-17: Layout of the DCO.

metal lines not only increase the fixed capacitance of the tank but also decrease the quality factor by increasing the series resistance. Top metals with large width are used for connecting the inductor with rest of the DCO to minimize the quality factor degradation. The capacitor banks and the active part are placed symmetrically about the inductor.

Post-layout simulations

The parasitics from the layout are extracted using Calibre and the post-layout simulations are performed to estimate the current consumption, swing, and phase-noise of the DCO. The effect of bond wire connections from supply to the chip are modeled by a 2 nH inductor. The capacitor banks are modeled using Verilog-A by using the capacitance step, range, and quality factor obtained from separate post-layout simulations. The rest of the DCO is simulated using the Calibre extracted views. This speeds-up the simulation significantly without sacrificing the accuracy.

Figure 4-18 shows the output of the DCO obtained from the transient simulation. The single-ended output swing is around 335 mV for a current consumption of 148 μA. The common-mode voltage is around 370 mV and is also shown in the figure. Figure 4-19 shows the phase-noise of the DCO estimated using the periodic steady-state (PSS) analysis. A phase-noise of -113 dBc/Hz at 1 MHz offset from the operating frequency is achieved. The major noise contributors are the inductance and the cross-coupled pair. The current consumption can be traded for phase-noise using the 5-bit bias control word.

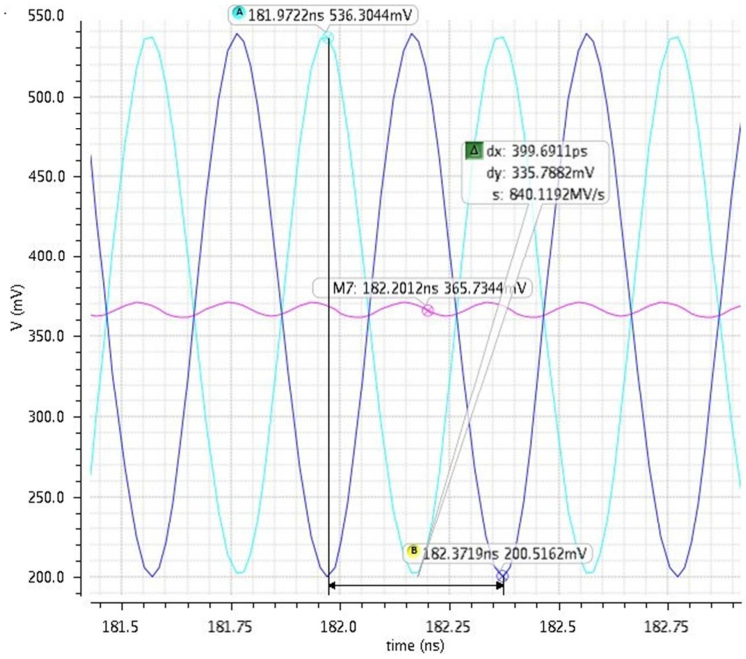


Figure 4-18: Transient simulation of the DCO.

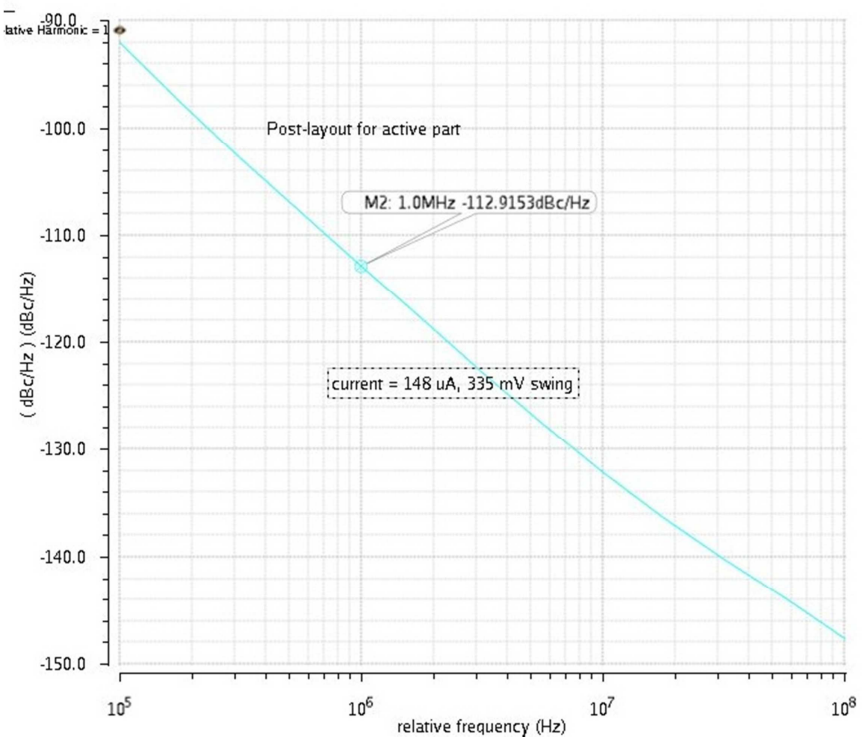


Figure 4-19: Phase-noise simulation of the DCO.

4-2 DCO buffer

The DCO needs to drive the PA and divide-by-2 pre-scaler as shown in Figure 2-5. A buffer is typically used to isolate the DCO from the effects of varying load, referred as load-pulling. Also, it allows the DCO to operate in the current-limited regime with a low output swing to reduce the power consumption. The implementation of this DCO buffer is discussed in this section.

4-2-1 Self-biased inverter

Since inverter based buffers are power efficient, a self-biased inverter, shown in Figure 4-20, is typically used as the DCO buffer. The DC-level of the DCO output is

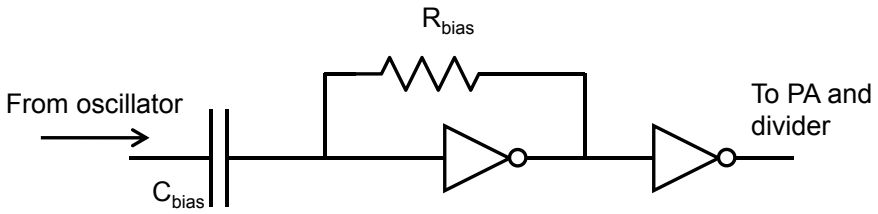


Figure 4-20: Self-biased inverter as the DCO buffer.

prone to PVT variations. Consequently, the output of the oscillator is ac-coupled to the buffer by means of a large coupling capacitor. Thus, the variations on the DC-level of the oscillator output do not impact the buffer output. The DC-level at the input of the buffer is fixed by a large resistor R_{bias}. Since the resistor is connected to the gate of the NMOS and PMOS of the inverter, no DC voltage drop occurs across it. This results in a same DC-voltage at the input and output nodes of the inverter i.e., it lies on the line "$V_{in} = V_{out}$". As shown in the Figure 4-21, the DC-level of

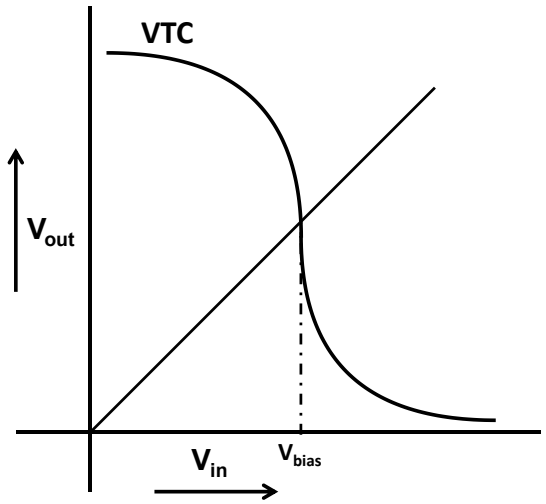


Figure 4-21: Biasing the self-biased inverter.

the signal is the point of intersection of the VTC and $V_{in} = V_{out}$ curves. The VTC

of the inverter is fixed such that this point of intersection corresponds to the point of maximum sensitivity on the voltage transfer characteristic (VTC) of the inverter and hence enables operation with low input-swing.

However, this approach uses passive components—C_{bias} and R_{bias}—that do not scale with technology. Hence it occupies a large area leading to higher system costs. Moreover, the large resistor R_{bias} used for biasing is a source of significant thermal noise which couples into the oscillator output, thereby degrading the phase noise performance. Thus the oscillator needs to expend more current to maintain the required phase noise levels. Furthermore, the large decoupling capacitor introduces significant parasitics, thereby loading the output node of the oscillator. This increases the power consumption as well as reduces the maximum operable frequency of the oscillator. Thus a self-biased inverter based buffer poses a serious challenge in scaling down the power consumption.

4-2-2 DC coupled buffer with tunable VTC

To address the aforementioned problems of self-biased inverter, a DC-coupled buffer with tunable VTC is proposed. In this approach, the bulky passives are avoided by DC coupling the oscillator output to the buffer. The variation in the DC-level of the DCO output now affects the operation of the buffer. Thus, to ensure the DCO buffer always operates at its maximum sensitivity point, VTC of the inverter is tuned as shown in Figure 4-22. In this way, the DC-level of the oscillator output always corresponds to the maximum sensitivity point on the VTC of the buffer.

To summarize, in an ac-coupled self-biased inverter (Figure 4-20), the VTC is *fixed* at its optimum point, such that the input DC-level set by R_{bias} ensures operation at maximum sensitivity. On the other hand, in the presented approach, the DC-level of the input buffer is fixed (by the DC-level of DCO output), while the VTC is *changed* to ensure the operation at maximum sensitivity point across PVT variations. In this way, an area efficient low-power buffer which operates with low-input swing over PVT variations is realized.

Figure 4-22 illustrates the idea of tuning the VTC for operation across process variations. The VTC of the inverter is varied by digitally controlling the device ratio of

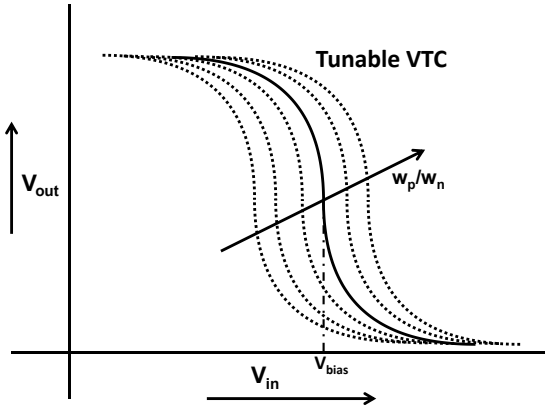


Figure 4-22: Tuning the VTC of the inverter.

PMOS to NMOS (w_p/w_n) of the inverter to cover process variation.

Figure 4-23 shows the schematic of the proposed buffer with digitally tunable VTC. The variation in DC-level of the oscillator output is estimated from Spectre simulations across process corners. The number and width of the switchable PMOS/NMOS devices is then chosen to cover this variation with extra margin. Four PMOS and NMOS devices with binary weighted widths serve as the additional devices and are controlled by the 5-bit digital word.

When the input code is "00000" i.e., none of the additional devices are connected, a default PMOS-NMOS pair is connected. This pair is sized to drive the subsequent load while operating at the maximum sensitivity point for an input DC-level corresponding to the typical corner. The MSB of the 5-bit buffer control determines whether PMOS or NMOS are turned on while the 4 LSB bits determine the number of those devices that are turned on.

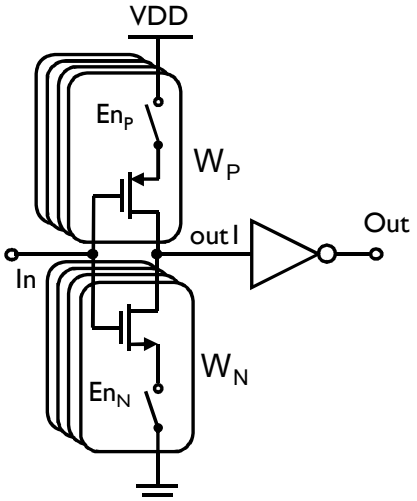


Figure 4-23: Schematic of the proposed VTC-tunable buffer.

The subsequent buffer of Figure 4-23 is used to drive the large output load. In its absence, the default pair of the tunable-buffer should be very large. This, in turn warrants large sizes for additional devices to alter the w_p/w_n ratio significantly, leading to a larger overall size of the buffer. Thus, the first buffer brings the low-input swing to rail-to-rail level and drives the relatively small subsequent buffer which drives the actual load (PA and divider).

By monitoring the duty-cycle of the oscillator output signal, the control code that tunes the VTC to the desired position is generated. At the optimal biasing point, the duty cycle of the oscillator output is 50%. If the input bias level is larger than the optimum level, then the digital control enables more PMOS devices such that the VTC shifts to the right. Similarly, if the input bias is smaller than the optimal point, more NMOS devices are enabled to shift the VTC towards left.

From post-layout simulations across various process corners, variation of the DC-level of the DCO output signal is estimated. At 1 V supply, the DC-level varies from 330 mV to 400 mV. The buffer is then designed to cover this range by adding NMOS and PMOS devices in parallel to the first inverter. These are connected to GND and

VDD lines via switches which are controlled by the 5-bit buffer control word. The MSB of the control word decides if additional PMOS or NMOS devices are required while the 4 LSB bits determine the number of devices connected in parallel to the first stage.

Figure 4-24 shows the transient simulation of the buffer across various process corners. By setting an appropriate control word, the buffer can withstand the DC-level variations on its input signal. The phase noise simulation from PSS analysis is also shown. The buffer consumes 56 μA and has a phase noise of -123 dBc/Hz at 1 MHz offset which 10 dB lower than that of the DCO.

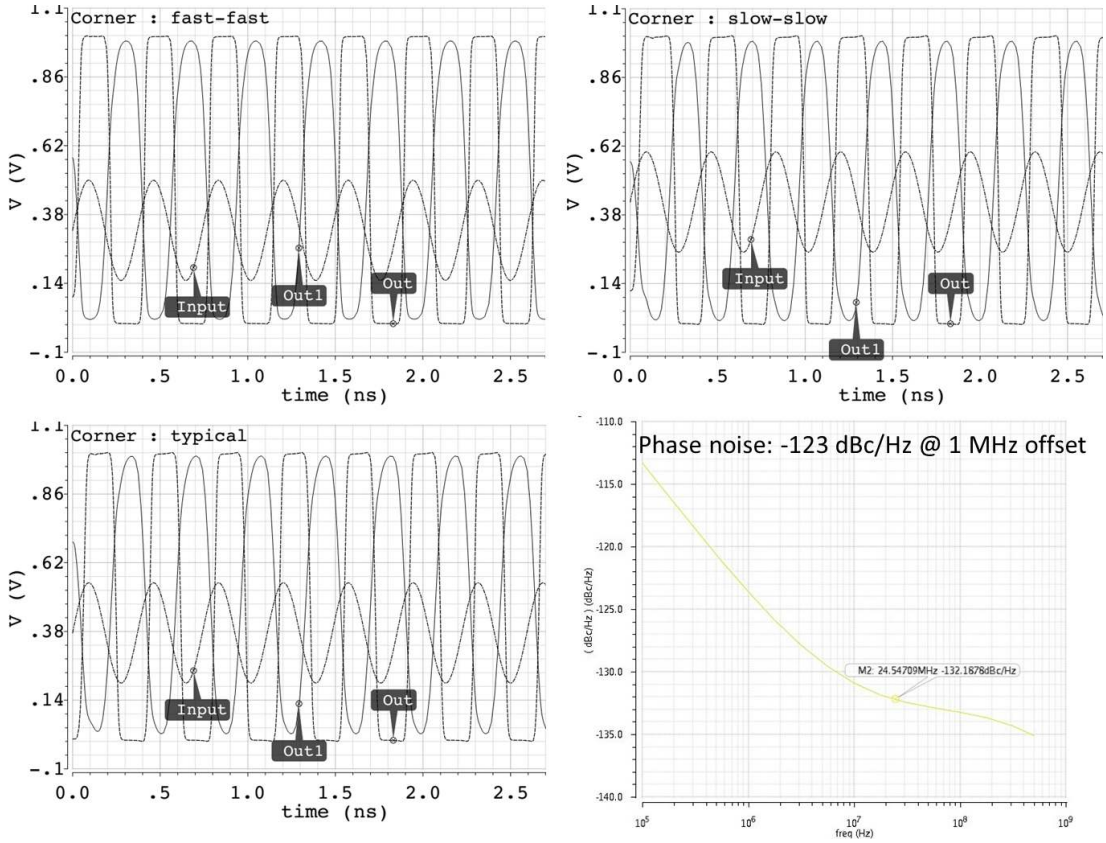


Figure 4-24: Operation of the buffer across process corners and phase-noise.

Since the proposed structure avoids passive components, it occupies a very small area of 5 μm x 10 μm and is aligned with the technology scaling. In comparison, a self-biased inverter would need a coupling capacitor of the order of 0.5 pF and a biasing resistor of the order of 200 kΩ. The coupling capacitance should be large enough to not attenuate the sinusoidal output of the DCO and the value of 0.5 pF is obtained from Spectre simulations. Using crtmom capacitor and ppoly resistors from TSMC library, the area of these two is estimated at 20 μm x 20 μm or eight times that of the DC-coupled buffer. Moreover, the power consumption is reduced by avoiding driving the bulky parasitics associated with the decoupling capacitors. Also, noise feed-through from the bias resistor is absent, thereby allowing lower current consumption for a given phase noise requirement compared to the traditional approach.

4-3 Divider

As discussed in Section 2-1, a divide-by-2 block is used after the DCO buffer and the divided down output clock is used for phase detection. This helps in reducing the operating frequency of the phase incrementer and hence saves power. A sense-amplifier based divider [5] is ideal when the low-input swing operation is needed. In this work, since the DCO buffer boosts the output to rail-to-rail voltage, a dynamic CMOS divider is used for its high-speed, low-power operation. The dynamic dividers are fast since they avoid the positive feedback transistors required to store the output until the next significant edge of the clock. Instead, the device and routing parasitic capacitances are used for storing the value. Since the operation frequency is quite high, the nodes are refreshed before the capacitor discharges, thereby retaining the value. Transmission gate based and tri-state inverter based dividers are compared to chose the optimum architecture.

4-3-1 Transmission-gate based dynamic divider

A transmission-gate based dynamic CMOS divider is shown in Figure 4-25 [53].

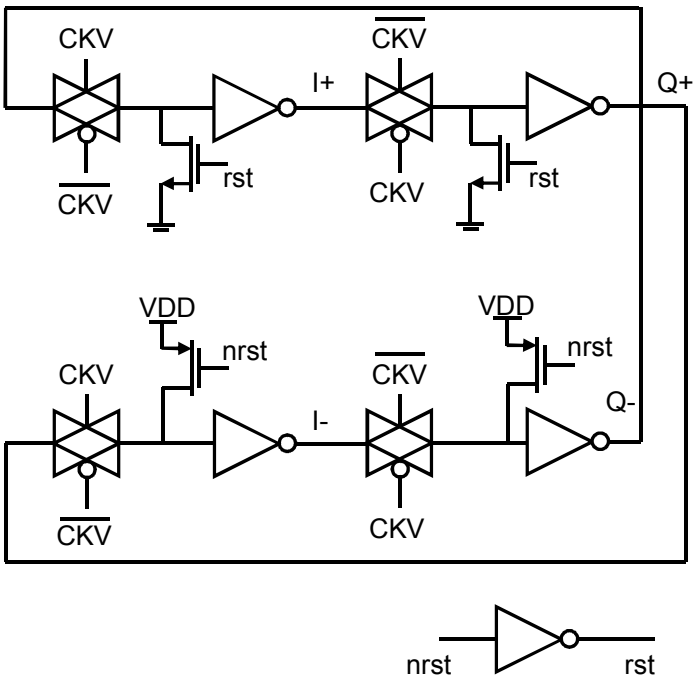


Figure 4-25: Transmission-gate based dynamic CMOS divider.

The transmission-gate inverter combination acts as a simple D-latch, with parasitics at the output of transmission gate acting as the storage element. The cascade of two-latches forms a master-slave flip-flop. Two such D-flip flops are cross-coupled to obtain a divider-by-2 with quadrature output phases. The outputs of each transmission gate should be initialized properly for this configuration to provide the required divide-by-2 signals. This is accomplished by the pull-up and pull-down MOS transistors which are activated only at the reset.

4-3-2 Tri-state inverter based dynamic divider

Figure 4-26 shows the simplified circuit schematic of the tri-state inverter based dynamic divider. A tri-state inverter replaces the transmission-gate inverter combi-

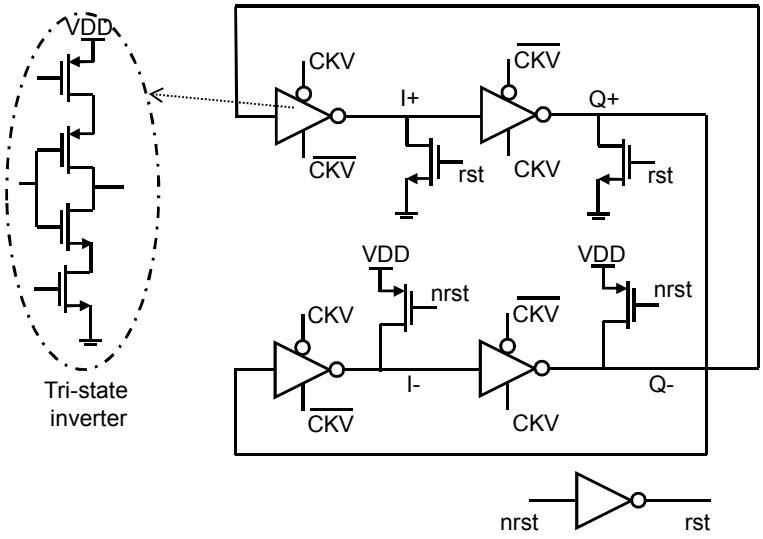


Figure 4-26: Tri-state inverter based dynamic CMOS divider.

nation of the previous configuration (see Figure 4-25). This structure is compared with the transmission-gate based architecture for power consumption and phase noise performance. It was observed that the transmission-gate based architecture has a slightly better phase-noise performance for a given power consumption and hence is adopted in this work.

4-3-3 Post-layout simulations

Initially, it was planned to use the quadrature phases of the divided-down DCO output to reduce the detection range of DTC. However, this could not be implemented and hence quadrature phases are no longer needed. Consequently, the divider of Figure 4-25 is simplified to just one D-flip-flop with its inverted output and input connected as shown in Figure 4-27.

The layout parasitics are extracted using Calibre and the post-layout simulations to estimate the phase noise and power consumption of divider-by-2 are performed. Figure 4-28 shows the transient simulation of the divider output at 1 V and 0.9 V supply voltage. The two output phases, Q and $\bar{Q}$, have a delay of around 40 ps at both 1 V and 0.9 V. This is due to the presence of the extra inverter in one-path (see Figure 4-27). The estimated current consumption is roughly around 30 μA for both supply voltages.

Figure 4-29 shows the phase-noise of the divider at 1 V and 0.9 V supply voltages estimated from PSS analysis. The phase-noise of the DCO is lower by 6 dB at the output of the divider by virtue of division by 2. However, if the phase-noise from the divider is large, it can limit the output phase noise. At 1 V supply, the phase noise at 1 MHz offset is -140 dBc/Hz which is 17 dB lower than that of the buffer and hence does not affect the phase noise of the output signal.

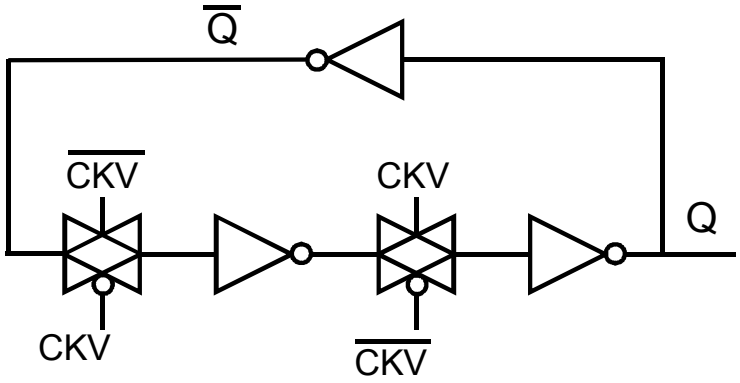


Figure 4-27: Simplified schematic of the implemented divider.

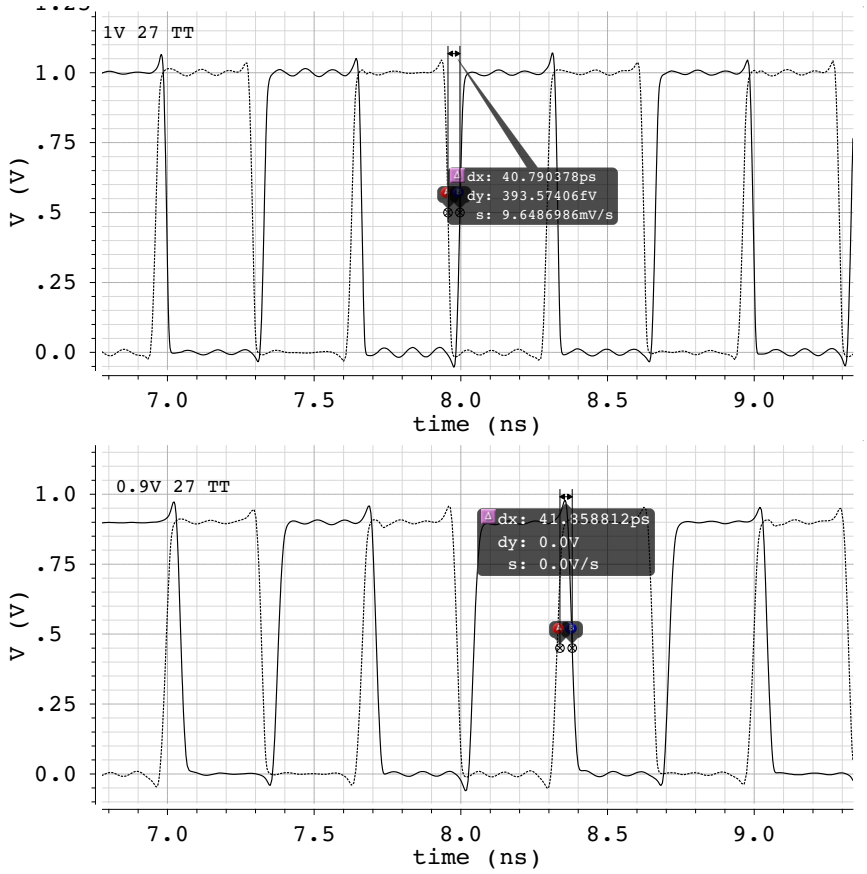


Figure 4-28: Transient simulation of the divider at 1 V and 0.9 V supply.

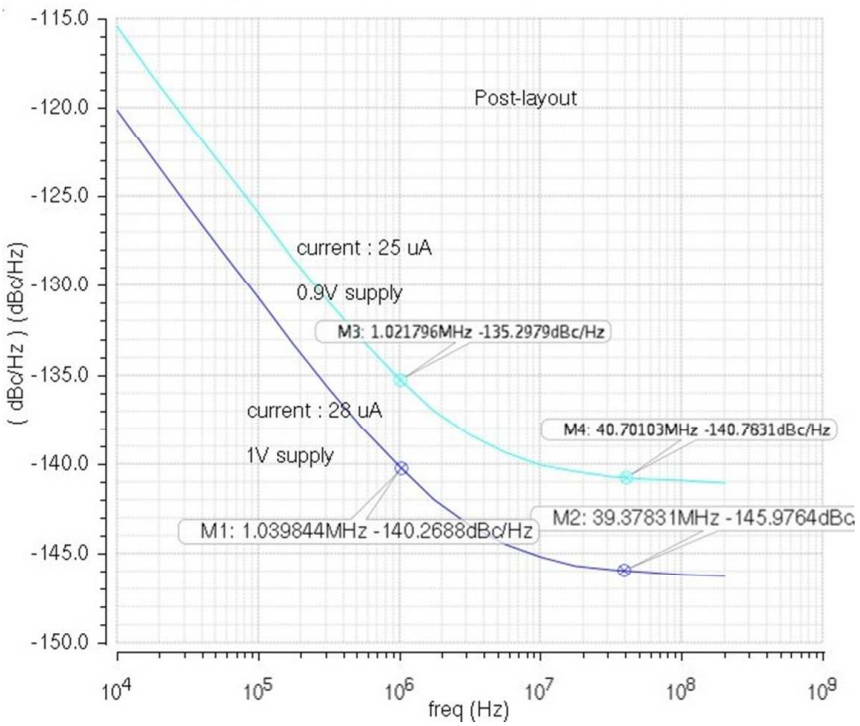


Figure 4-29: Phase noise of the divider at 1 V and 0.9 V supply.

4-4 Phase incrementer

A phase-incrementer performs the integer phase detection of the variable clock by counting the number of CKVD2 clock cycles. As discussed in Section 3-1-2, a 7-bit integer counter is required. The phase incrementer could be implemented either as synchronous or asynchronous logic. In a synchronous counter, all the flip-flops are clocked by the input clock. This leads to a large power consumption but the outputs are synchronized and the delay is small. Also, the CKVD2 clock is loaded by all the flip-flops which might increase the power consumption of the divider that drives the phase incrementer. For this reason, an asynchronous counter is used to realize the phase incrementer.

In an asynchronous counter, the output of each stage triggers the succeeding stage. Thus operating frequency of each succeeding stage is reduced by 2, resulting in lower overall power consumption. However, in this implementation, the outputs of each stage are available after the preceding stage with some delay. Figure 4-30 shows the timing diagram of the implemented synchronous counter. After the rising edge of

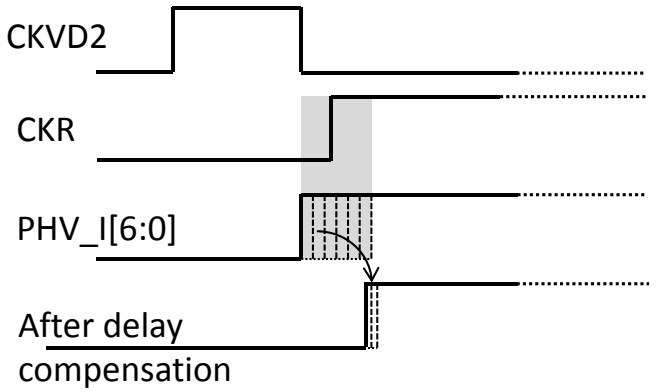


Figure 4-30: Timing waveforms of CKVD2, CKR, and phase incrementer output.

the CKVD2 signal, the output of the counter takes a long time before all the bits are settled. If the rising edge CKR that samples the output of the phase incrementer falls within this window when the output phase is still changing (highlighted area in Figure 4-30), incorrect value of output phase is fed to the phase detector block of the low-speed digital logic. This results in an incorrect estimation of phase error and will break the lock.

To address this issue, the delay between CKR and CKVD2 rising edges are estimated using post-layout simulations in all corners. Additional delay elements (buffers) are added at the outputs of each stage of the phase incrementer to synchronize them and ensure that the output of the phase incrementer begins to change only after the rising edge of CKR. Figure 4-31 shows the circuit schematic of the phase incrementer. The last two stages are implemented in a synchronous fashion since the delay is too large. The D-flip-flop within the asynchronous counter is implemented by a transmission gate based structure similar to that discussed in Section 4-3-1 while a library cell is used for the sampling flip-flops.

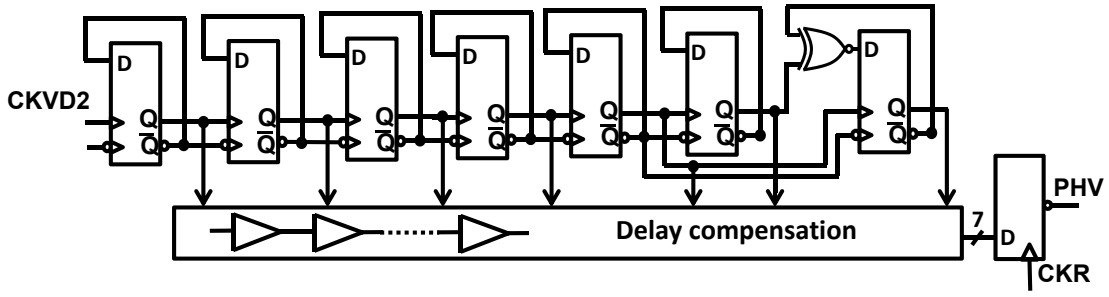


Figure 4-31: Circuit schematic of the phase incrementer.

Chapter 5

Measurements

The ultra-low-power ADPLL implemented in this work is fabricated in TSMC LP 40 nm CMOS process. The measurement set-up, stand-alone DCO measurements and ADPLL measurements are presented in this chapter.

5-1 ADPLL test plan

The chip micrograph of the low-power ADPLL is shown in Figure 5-1. The chip is pad-limited and occupies an active core area of only 0.2 mm^2. It has separate power supplies for DCO, TDC/DTC, low-speed logic, high-speed blocks, and the power amplifier. This reduces the noise coupling from digital logic into DCO or TDC via supply. They also have separate grounds. The metal casing available with the package is connected to all the GND pins, expect the noisy digital-GND, of the IC via down-bonds to keep the inductance minimum. The separate power supplies helps evaluate the current consumption of each of these blocks separately.

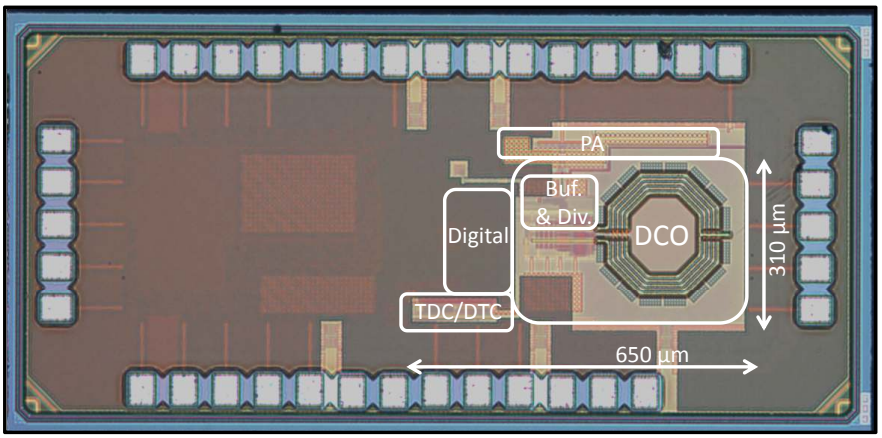


Figure 5-1: Chip-micrograph of the ADPLL prototype.

The chip is placed in a QFN32 package. The packaged chip is mounted on a PCB which also contains LDO regulators to provide required supply voltages. Also, an

FPGA interface with required voltage level-shifters is mounted on the PCB. An external 32 MHz clock generated by a signal generator is used as the reference clock for this ADPLL. It is provided via an SMA port of the PCB. Another SMA port provides the output of the PA which is read into the spectrum analyzer to measure the phase noise. The photographs of the test PCB and test bench setup are shown in Figure A-2 and Figure A-3, respectively.

Figure 5-2 shows the block diagram of test-setup. A Xilinx FPGA board acts as an

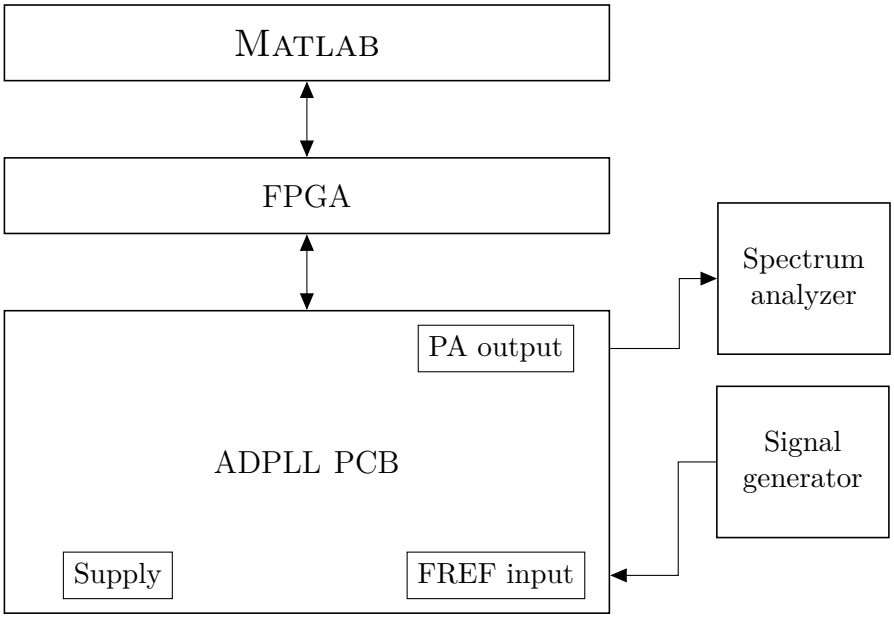


Figure 5-2: Block diagram of ADPLL test setup.

interface to the PC and the SPI implemented on the chip. A MATLAB program updates the SPI registers via the interface implemented in FPGA. The modulation data required to test the direct frequency modulation capability of the ADPLL is also implemented in the FPGA. Since SPI limits the frequency at which the data can change, the 10-bit modulation data is read by the ADPLL directly through bond pads.

5-2 Measurement results

The open-loop DCO and closed-loop ADPLL measurement results are presented in this section.

5-2-1 Open-loop DCO measurements

The performance of the DCO, namely phase noise, frequency range, and power consumption, are characterized in this section. The loop is opened to characterize the free-running DCO performance. This is done by setting the bank_en signal to "000", as discussed in Section 3-1-5. mem_dco signal of each bank is set to control the tuning word of the corresponding bank of DCO. The mem_dco signals represent signed data and its MSB is flipped internally in the control block of the low-speed logic before it is applied to the DCO.

Tuning range

The DCO is designed to cover 2.25 GHz – 2.85 GHz using three banks—5-bit PVT, 6-bit acquisition, and 9-bit tracking—to meet the required 2.4 GHz – 2.7 GHz frequency range. Figure 5-3 shows the measured frequency of the DCO with varying PVT and acquisition code at 1 V supply.

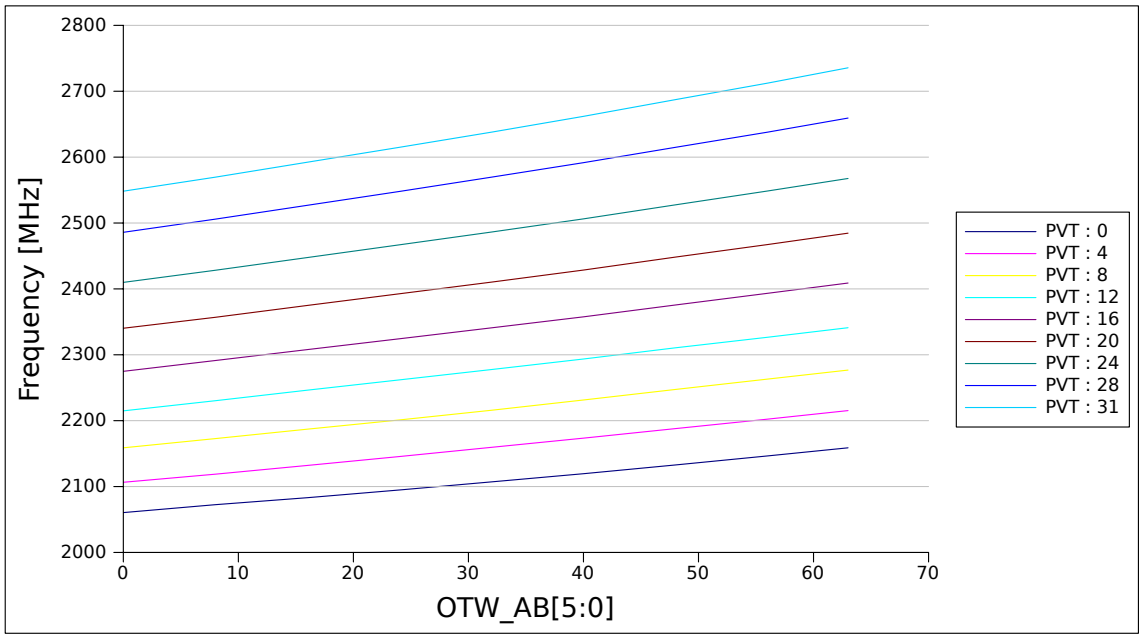


Figure 5-3: Measured tuning range of the DCO at 1 V supply.

The DCO covers the frequency range of 2.1 GHz – 2.7 GHz. Although the target frequency bands are covered, the frequency is shifted by around 150 MHz compared to the post-layout simulations. One reason could be the change in inductance value due to process variations. As discussed in Section 4-1-1, the inductance value can change by 10% over different corners leading to a frequency shift. Another possibility could be the under estimation of the parasitic capacitance by the extraction tool.

The frequency step of the PVT bank is around 11.6 MHz at the lowest operating frequency (2.1 GHz) and is 25.7 MHz at the highest operating frequency. At the typical running frequency, 2.4 GHz, the frequency step of the PVT bank is 17 MHz which is close to the simulated value. The frequency step of acquisition bank is 1.4 MHz, 2.3 MHz, and 3.1 MHz at the lowest, typical, and highest operating frequencies, respectively. The PVT bank covers a frequency range of 530 MHz while the acquisition bank has a range of around 150 MHz when operating around 2.4 GHz frequency.

The tuning range of the tracking bank is shown in Figure 5-4. The control words of the PVT and acquisition banks are tuned to change the operating frequency to around 2.44 GHz. The control word for the tracking bank is then swept to obtain its frequency range. It can be seen that the tracking range covers a range of nearly 20 MHz. The frequency step is around 40 kHz, slightly lower than that expected from post-layout simulations. The possible reason is the larger parasitics in off-state than expected from simulations. However, since the range is sufficiently larger than the step of acquisition bank, it is not an issue.

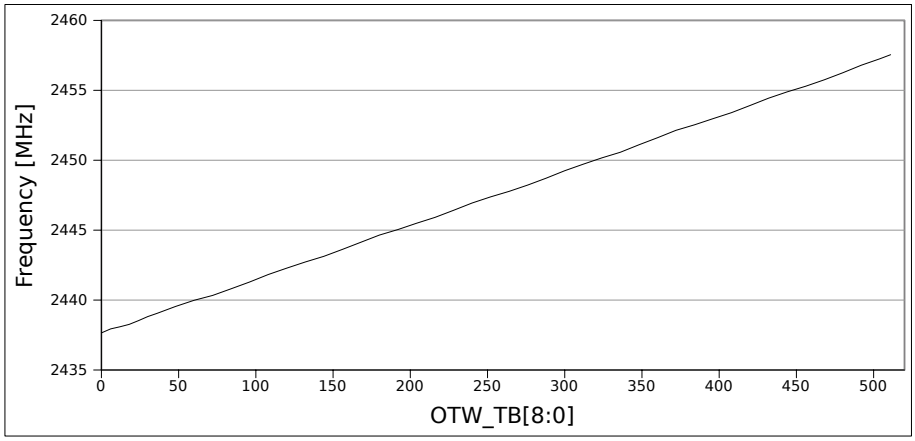


Figure 5-4: Measured tuning range of the tracking bank of the DCO.

Table 5-1 summarizes the measurement results of the frequency range of the three banks of the digitally controlled oscillator (DCO) at the typical operating frequency of around 2.4 GHz. We can see that the range of the finer bank is around 7 times that of its immediate coarse bank ensuring sufficient overlap between the banks.

Table 5-1: Measurement results of capacitor banks at around 2.4 GHz

	PVT	Acquisition	Tracking
No. of bits	5	6	9
Frequency step	17 MHz	2.3 MHz	40 kHz
Frequency range	530 MHz	150 MHz	20 MHz

Phase noise and power consumption

The phase noise and power consumption of the oscillator are measured at 2.4 GHz operating frequency. The DCO bias current is changed by tuning the 5-bit digital control word as discussed in Section 4-1-2. The measured phase noise of the DCOis shown in Figure 5-5. The DCO has a phase noise of -115.3 at 1 MHz offset for a combined (DCO+buffers) power consumption of 450 μA when operating at 2.46 GHz.

The current can be decreased to trade-off performance by tuning the bias control word. Figure 5-6 shows the plot of the measured phase noise of DCO as a function of current consumed of DCO and buffer. Since the DCO buffer and DCO share the same power supply, the power consumed by the DCO alone could not be measured. A FoM given by Eq. (5-1) is typically used to evaluate the performance of an oscillator. It takes into account the power consumption and phase noise to enable a fair comparison.

$$FoM = \mathcal{L}(\Delta f) + 10\log_{10}\left(\frac{\Delta f}{f_0}\right)^2 + 10\log_{10}\left(\frac{Power}{1\ mW}\right) \tag{5-1}$$

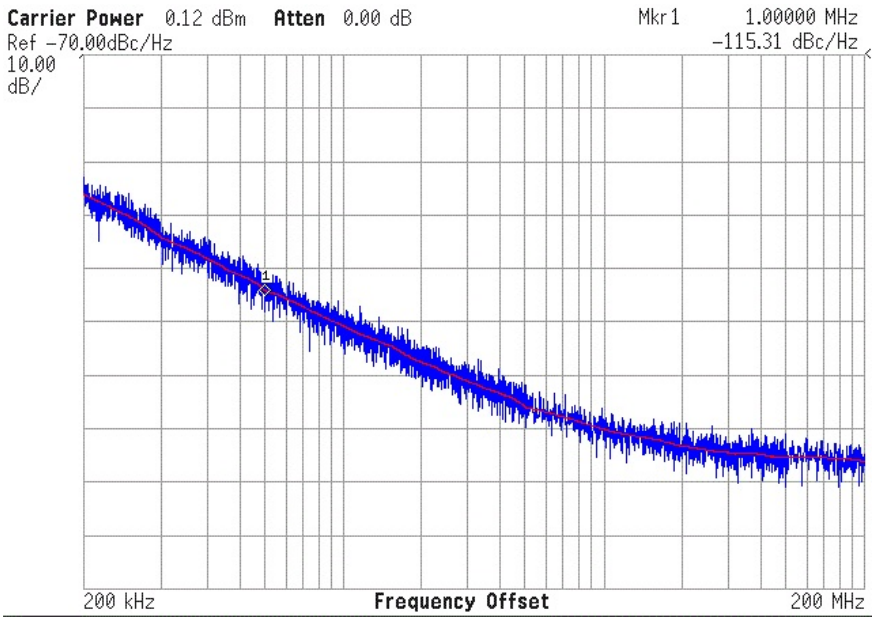


Figure 5-5: Measured phase noise of the DCO.

Since the power consumption of DCO alone could not be measured, the post layout estimation of the buffer power is used to estimate the DCO power from the combined power consumption. Post-layout simulations show that each buffer consumes around 60 μA and hence the power consumption of DCO is 330 μA for a phase noise of -115.3 dBc/Hz resulting in an FoM of -187.7 dB.

For a current consumption of around 360 μA, the phase noise at 1 MHz offset is around -112 dBc/Hz—the target value. However, the current consumption is around 90 μA higher than that expected from the post-layout simulations. The possible reason is that the quality factor of the inductor is less than the expected 14. The reduced quality factor degrades the phase noise and hence the DCO needs more current to maintain the required performance. This is also evident from the observation that a higher value of bias control word is needed to start-up the DCO oscillation than that estimated from simulations. The tank loss is higher and hence needs more current to replenish it to sustain the oscillation.

The phase-noise improves with the bias current until the current consumption is 400 μA. The DCO operates in the so-called current-limited range and its output swing increases with bias current. The increased swing results in a better phase noise performance. However, once the current is too large, the DCO enters the voltage-limited regime. In this mode, the output swing of the DCO is limited by the available voltage headroom. Hence increasing the current has negligible returns in terms of the phase noise improvement.

5-2-2 Loop-locking behavior

The settling time of the PLL is measured using a signal analyzer and is shown in Figure 5-7. From the figure it can be seen that, initially, the frequency deviates by a large value. In this region, the phase prediction is not accurate; consequently, the

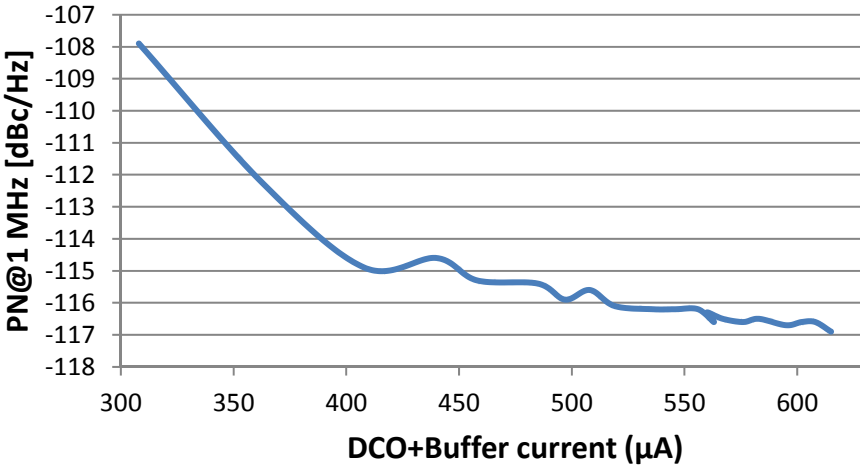


Figure 5-6: Measured phase noise of the DCO versus current.

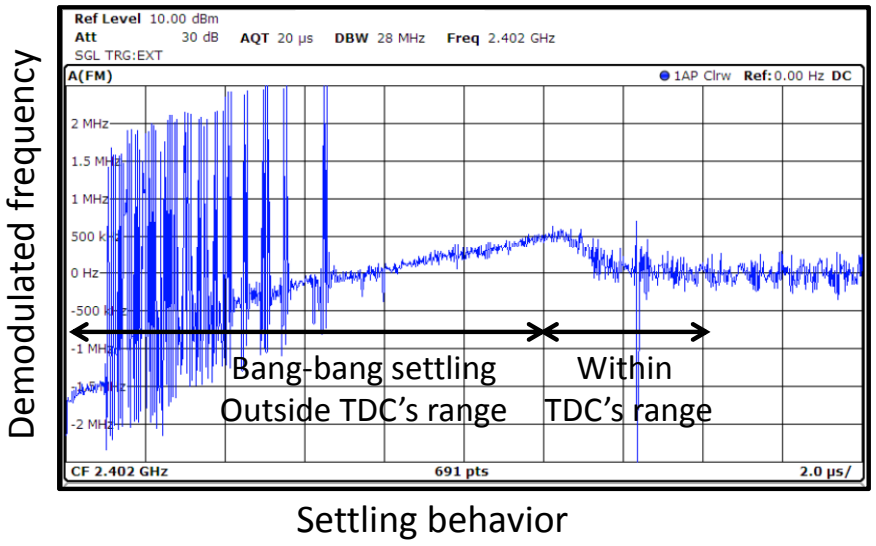


Figure 5-7: Measured settling behavior of the ADPLL.

phase error is out of narrow range of the TDC. The TDC hence acts like a bang-bang phase detector during this time period. Eventually, the output frequency becomes closer to the target value reducing the phase error to within the TDC's range. Hence the frequency deviation and phase noise is small. The ADPLL settles to the target value within 16 μs.

5-2-3 Spectral purity and phase noise

Figure 5-8 shows the spectrum of the ADPLL output clock when the frequency command word (FCW) set to 37.5 corresponding to a target frequency of 2.4 GHz. The reference spur is quite low at -70 dBc. The fractional spur occurring at 0.5*FREF, which is 16 MHz is also below 65 dBc as it corresponds to a channel with large fractional FCW.

The magnitude of the fractional spurs is measured for all the channels that com-

prise the *Bluetooth Smart*, 2.4 GHz – 2.48 GHz with a 2 MHz spacing. They are plotted in Figure 5-9. The worst-case fractional spur is -38 dBc for the near-integer

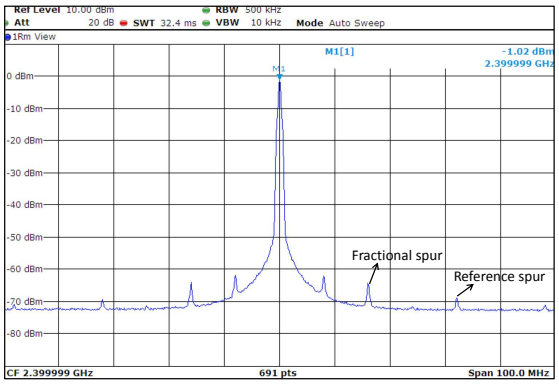


Figure 5-8: Output spectrum of the ADPLL for FCW = 37.5.

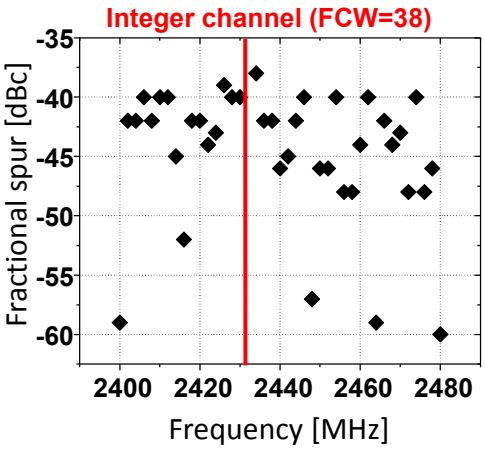


Figure 5-9: Fractional spur over *Bluetooth Smart* channels.

channels. Although this value meets the target requirements (< -30 dBc) it is high. The possible reasons could be the mismatch between the individual delay elements in the DTC [22]. Several spur-reduction techniques are proposed in the literature to address this issue. The FREF dithering employed in [22, 23] may be adapted to this architecture in future designs to improve the fractional spur performance.

The phase-noise plot of the frequency synthesizer measured using an Agilent 4440A spectrum analyzer is shown in Figure 5-10 for FCW = 38.125.

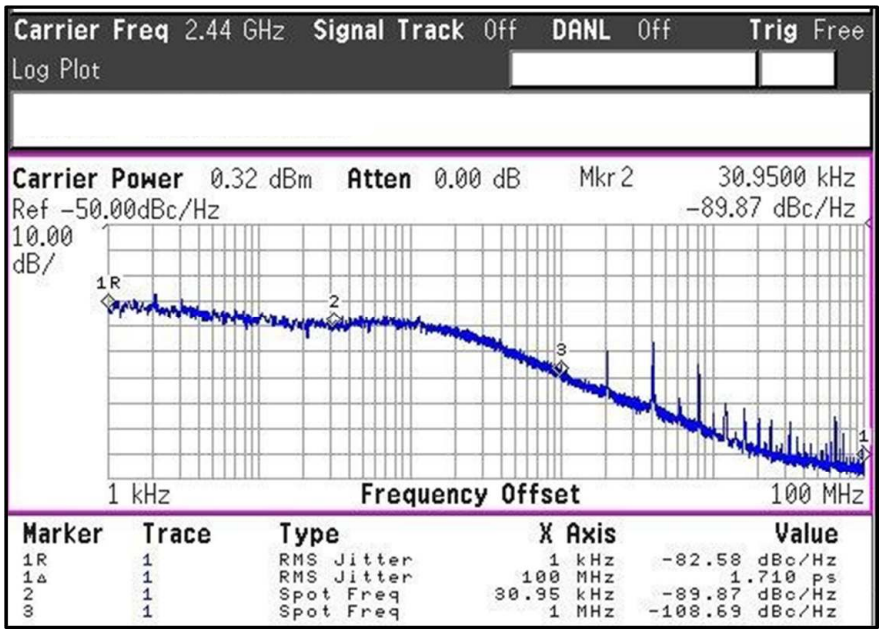


Figure 5-10: Measured phase noise of the synthesizer for FCW = 38.125.

The loop band-width is set to around 200 kHz by setting the proportional and integral constants of the tracking bank, $\alpha = 2^{-5}$ and $\rho = 2^{-11}$. The integrated phase

noise between 1 kHz – 100 MHz is 1.7 ps when FCW = 38.125. The phase noise at very low frequency offsets is dominated by the reference signal. The in-band phase noise is -90 dBc/Hz and is limited by either the accuracy of DTC gain calibration or the TDC resolution. The out-of band phase noise at 1 MHz offset is -108.7 dBc/Hz.

The phase noise of the synthesizer for far-off integer channel is shown in Figure 5-11 corresponding to an FCW of 37.5. The integrated jitter is 1.38 ps indicating the improvement possible by applying spur-reduction techniques.

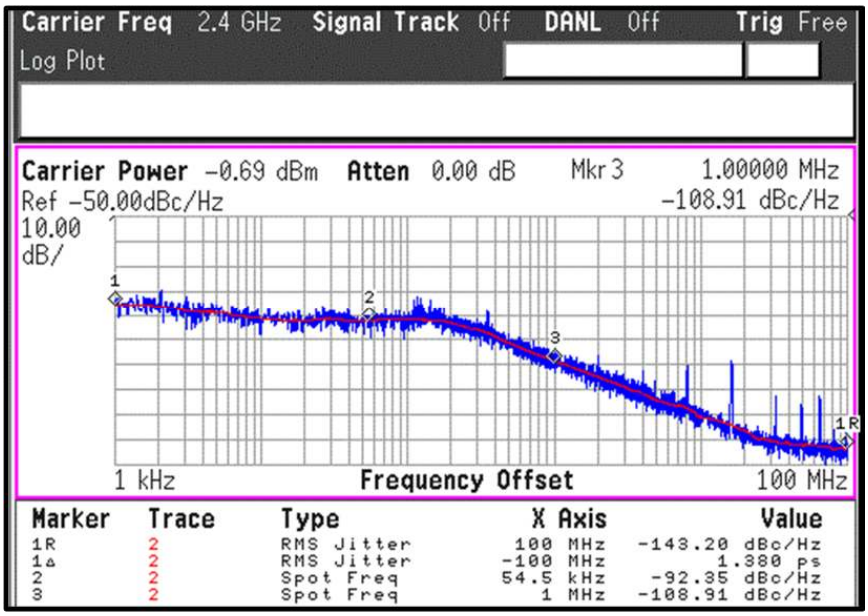


Figure 5-11: Measured phase noise of the synthesizer for FCW = 37.5.

5-2-4 Direct frequency modulation

The implemented ADPLL is capable of direct frequency modulation. It performs 2-point modulation to eliminate the impact of loop-bandwidth on the maximum data-rate achievable. The ADPLL is configured to provide modulation data for the two targeted standards—*ZigBee* and *Bluetooth Smart*. The modulation data is generated by the FPGA included in the test setup (see Figure 5-2).

Figure 5-12 and Figure 5-13 show the 2 Mcps HS-OQPSK modulation provided by the ADPLL. The ADPLL provides the modulation with 2.3% EVM while fulling the spectral mask requirements.

The 1 Mbps GFSK modulation provided by the ADPLL are shown in Figure 5-14 and Figure 5-15. The ADPLL provides the modulation with an FSK error of 5.2% (spec. $< 20\%$) and fulfills the spectral mask requirements.

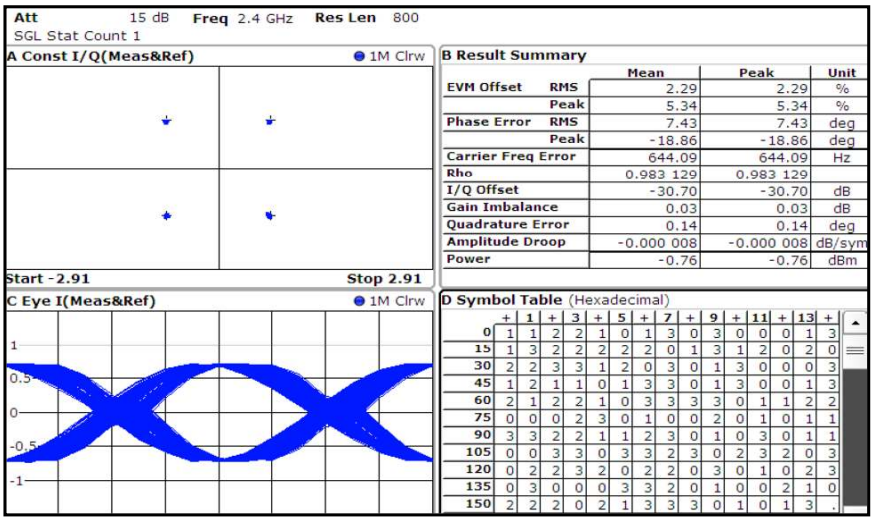


Figure 5-12: Measured *ZigBee* modulation accuracy.

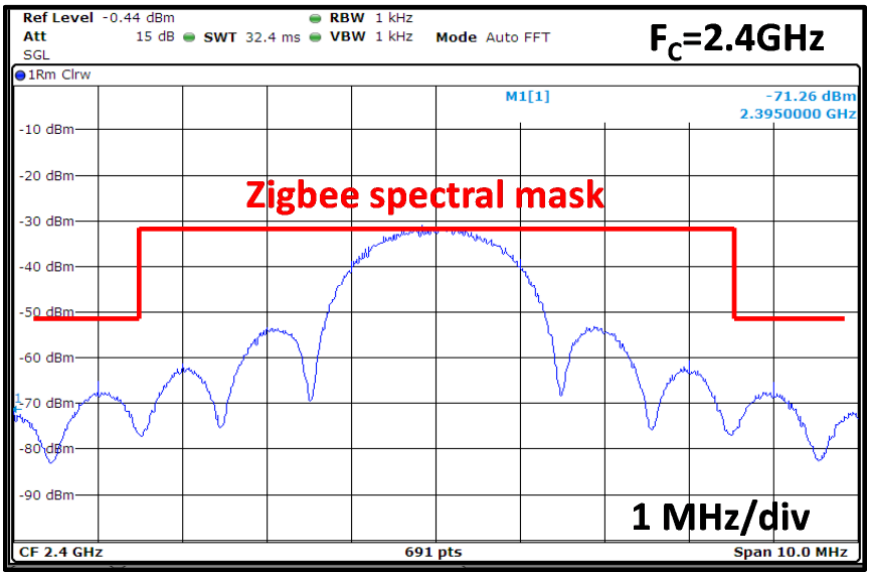


Figure 5-13: Measured *ZigBee* spectrum.

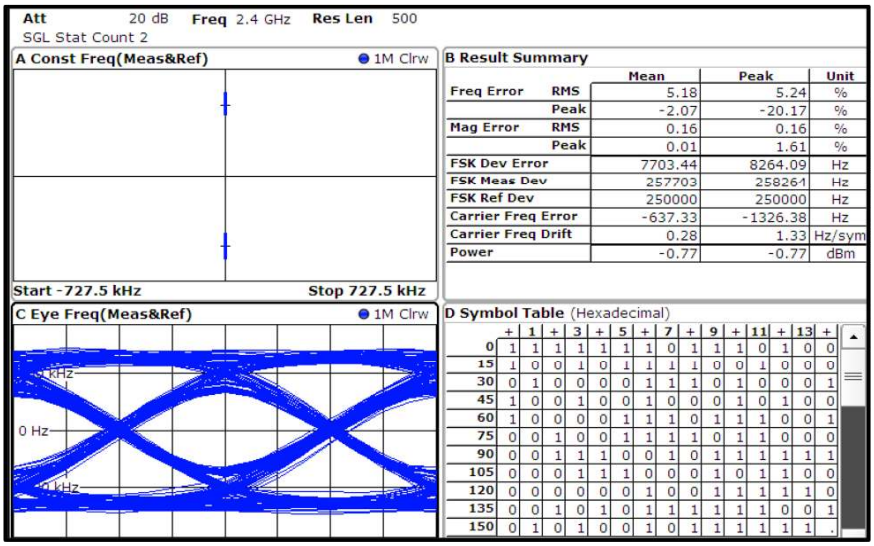


Figure 5-14: Measured *Bluetooth Smart* modulation accuracy.

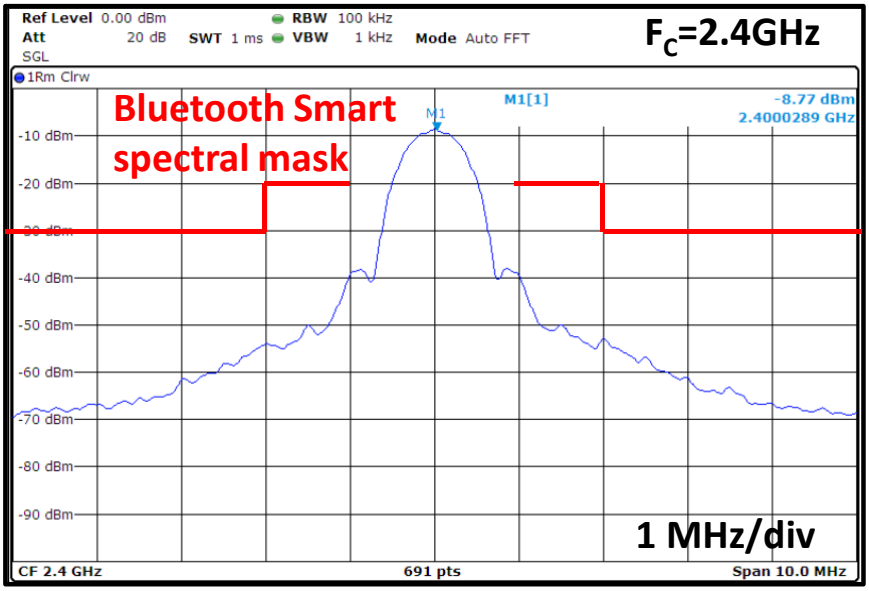


Figure 5-15: Measured *Bluetooth Smart* spectrum.

5-2-5 Power consumption

The power break-down of the ADPLL is shown in Table 5-2. The power consumption

Table 5-2: Measured power break-down of the ADPLL.

	Power consumption (μW)
DCO+buffer	450
low-speed digital	270
Divider + Phase incrementer	100
TDC+DTC	40
Total	860

of the DCO and buffer is dominant. The high frequency circuits divider and phase incrementer consume only 100 μW, thanks to the asynchronous counter architecture and dynamic logic for the divider. Most importantly, the TDC and DTC, which perform the fractional phase detection consume only 40 μW compared to the typical mW-level power consumption in time-to-digital converters (TDCs). The phase prediction and snapshotting techniques enabled the narrow-range low-frequency operation of TDC as discussed in Section 2-1. This proves the effectiveness of the DTC-assisted snapshot TDC in reducing the power consumption drastically compared to the conventional stand-alone TDC. The supply voltage and power consumption can be further reduced to 0.9 V and 750 μW by trading-off jitter performance (2.2 ps).

5-2-6 Performance summary and comparison

The performance of the ADPLL designed in this work is summarized and compared to the state-of-the-art PLLs in the literature and is shown in Table 5-3. The total power consumption of the ADPLL is 860 μW at 1 V supply for an integrated jitter of 1.7 ps operating at 2.4 GHz. This work achieves the lowest power consumption among the multi-GHz ADPLLs published in the literature with a state-of-the-art FoM of -236 dB.

Figure 5-16 compares the presented ADPLL with the state-of-the art PLLs in literature in terms of jitter variance and power consumption. The ADPLLs are denoted in black. The CP-PLLs with lowest power and highest performance are also depicted (in blue). The jitter^2-power FoM is also shown along with the desired direction i.e., to achieve low power and low jitter simultaneously. It is evident that this work presents the first sub-mW fractional-N ADPLL and achieves a power consumption at least 5 times lower than that of the ADPLLs with similar FoM. Also, this ADPLL occupies 3.75 times smaller area compared to its best performing analog counterpart [5] for similar applications, indicating its huge potential for cost savings. The sub-mW power consumption of this ADPLL shows the feasibility of leveraging ADPLLs in ultra-low-power applications.

[1] $FoM = 10log_{10}\Big(\big(\sigma^2_{jitter}\big) * \big(\frac{P}{1mW}\big)\Big)$ [30]

Table 5-3: Comparison with the state-of-the-art.

	This work	[5] ISSCC'12	[12] ISSCC'04	[43] ISSCC'11	[42] ISSCC'11	[45] ISSCC'13
Architecture	ADPLL DTC+TDC	CP-PLL Analog	ADPLL TDC-based	ADPLL Bang-Bang	ADPLL Bang-bang	ADPLL TDC-based
Application	Bluetooth Smart(V4)/ ZigBee	Bluetooth Smart(V4) ZiBbee	Bluetooth (V1)	NA	4G Cellular	Bluetooth EDR(V2)
Technology	40 nm	90 nm	130 nm	65 nm	40 nm	40 nm
Reference	32 MHz	24 MHz	13 MHz	48 MHz	40 MHz	26 MHz
Output (GHz)	2.1–2.7	1.7–2.48	2.4	4.9–6.9	2.9–4.0	2.4
RMS Jitter (ps)	**1.71**	2.66	1.02	0.61	0.56	0.98
Power (mW)	**0.86**	1.1	30	24	4.5	4.55
Supply	**1 V**	1.2 V	1.5 V	1.2 V	1.2 V	1.3 V
Reference spur	-70 dBc	-62 dBc	-80 dBc	-67 dBc	-72 dBc	-80 dBc
Core area(mm^2)	0.2	0.75	0.8	0.91	0.22	0.075
PN @ 1 MHz offset	-109	-111	-86@10 kHz	-116	-110	-113
FM capability	Two-point	Two-point	Two-point	No	Two-point	Two-point
FoM[1] (dB)	-236	-231	-225	-230	-238.3	-233.6

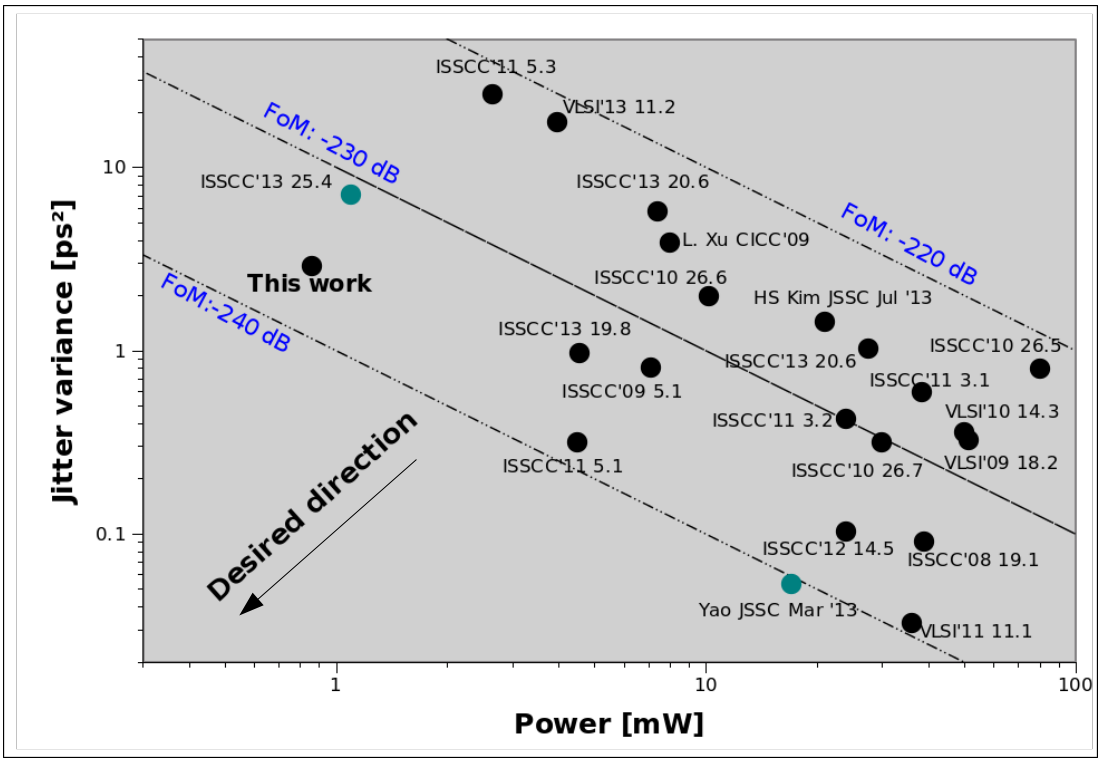


Figure 5-16: Jitter variance-power plot of state-of-the-art PLLs.

Chapter 6

Conclusions

In this thesis, the theory and implementation of an ultra-low-power all-digital phase-locked loop (ADPLL) for frequency synthesis and modulation in ultra-low-power transceivers is presented. In Chapter 2, it has been shown that TDC remains the main bottleneck in reducing the power consumption of ADPLLs to sub-mW levels. To this end, several low-power techniques are presented.

The power consumption is reduced at architecture level by using a combination of digital-to-time converter (DTC) and time-to-digital converter (TDC) with snapshotting to replace the traditional power-hungry TDC for fractional phase detection. This reduces the required detection range as well as the operating frequency of the TDC resulting in huge power savings. At the circuit-level, the power consumption of the DCO is reduced by using a low-power DCO buffer with tunable voltage transfer characteristic (VTC). This avoids bulky decoupling capacitors and biasing resistors, thereby reducing the power and area of the DCO plus buffers.

The RTL design of the low-speed digital logic and the circuit design and physical layout of the high speed blocks—DCO, DCO buffer, divider, and phase incrementer—are discussed in detail. To evaluate the effectiveness of the low-power techniques presented, a prototype is implemented in a TSMC LP 40 nm CMOS process. The synthesizer has a measured integrated jitter of 1.7 ps for a power consumption of 860 μW leading to a jitter^2-power FoM of -236 dB. This is at least 5 dB better than that achieved by the published PLLs in literature with similar power consumption [5]. The achieved FoM is slightly lower than the best FoM reported for ADPLLs, but has at least 5x lower consumption. In addition, the prototype ADPLL also supports high data-rate frequency modulation, and is demonstrated for *ZigBee* and *Bluetooth Smart* standards.

Figure 6-1 shows the jitter^2-power FoM of the state-of-the-art fractional-N ADPLLs plotted as a function of power consumption. This work presents the first RF ADPLL with a sub-mW power consumption with an FoM close to the state-of-the-art. From Figure 6-1, it can be observed that high performance ADPLLs tend to have a better FoM compared to low-power ADPLLs. The presented techniques enable ADPLLs to break the sub-mW barrier and thus to be employed in the emerging ultra-low-power WPAN applications.

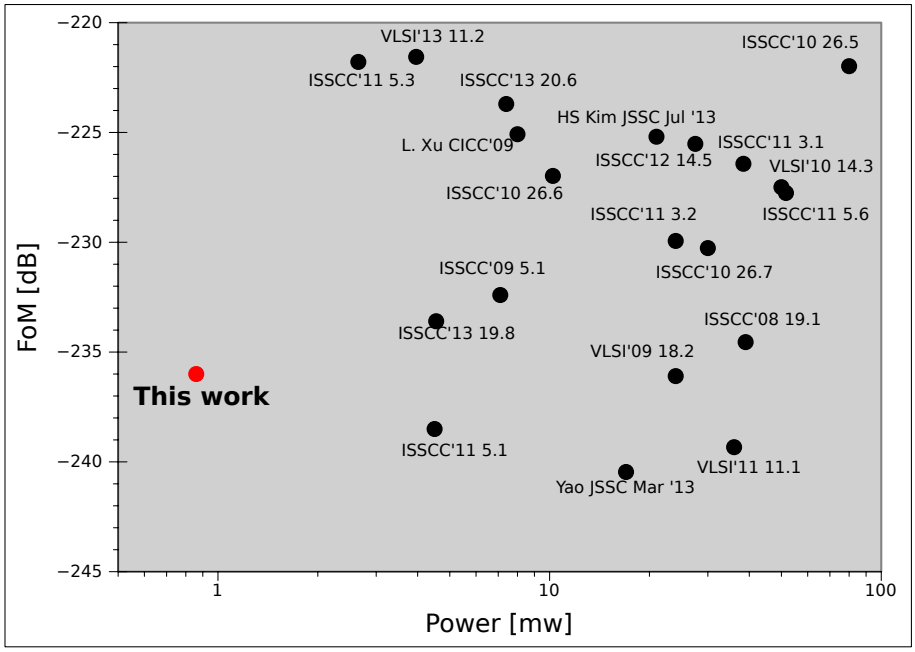


Figure 6-1: FoM versus power consumption of state-of-the-art PLLs.

6-1 Future work

- Although the power consumption of the TDC is reduced drastically, DCO consumes considerable power. While the power consumption is limited mainly by the Q-factor of the on-chip inductor, some improvements are possible. The bias current of the DCO in this implementation is varied by changing the top resistor. However, the cross-coupled transistors remain unchanged and hence are optimized for only a limited range of bias current. This can be solved by controlling the number of cross-coupled pair resistor cells attached to the LC-tank instead. Each such cell carries a fixed current and hence the power consumption and phase noise are optimum irrespective of the bias current. Also, swing regulation of the DCO should be implemented to ensure constant swing at all frequencies of operation.

- The automatic calibration of DC-coupled DCO buffer should be implemented. One way would be to measure the output duty cycle to control the buffer control word so as to maintain the 50% duty-cycle.

- Although the fractional spurs satisfy the requirements of the targeted application, they are still quite high. One possible way to mitigate the spurs is to increase the size of the delay cells of DTC to improve matching. Since the DTC consumes only 2.5% of the total power, increased size would have little impact on the overall power consumption. Also, FREF dithering proposed in [22, 23] may be adapted if needed. In addition, multiple phases of DCO could be exploited to further reduce the DTC range. This reduces the INL and the reduced power can be traded to further improve the matching performance.

Appendix A

Chip pinout, bonding diagram, and measurement setup

Table A-1 lists the pinout configuration of the implemented ADPLL prototype. The bonding diagram with the QFN32 package is shown in Figure A-1. The package has a metal layer underneath to which numerous down-bonds are made from IC, thereby minimizing the length of grounding bond wires.

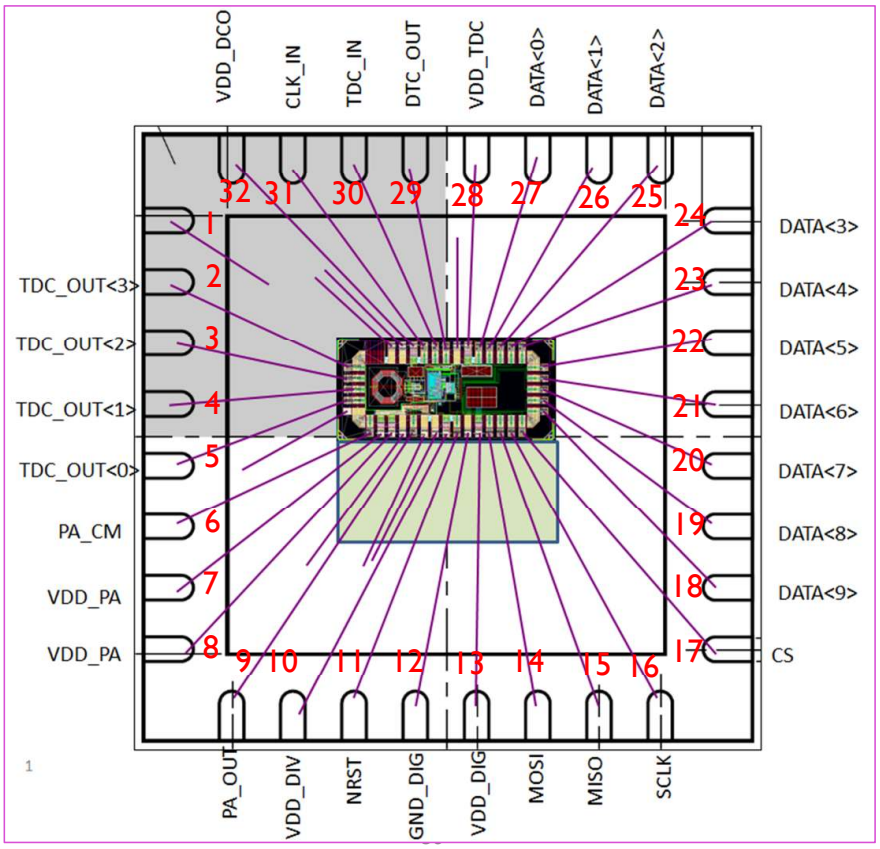


Figure A-1: Bonding diagram of the ADPLL IC.

Table A-1: Chip pinout of the ADPLL prototype.

Pin number	Name	Description
1	—	Grounded
2	TDC_OUT<3>	4-bit TDC output
3	TDC_OUT<2>	
4	TDC_OUT<1>	
5	TDC_OUT<0>	
6	PA_CM	Buffer calibration
7	VDD_PA	1 V PA supply
8	VDD_PA	1 V PA supply
9	PA_OUT	Output clock
10	VDD_DIV	1 V divider supply
11	NRST	Asynchronous reset acive low
12	GND_DIG	Digital ground
13	VDD_DIG	1 V digital supply
14	MOSI	SPI Control signal
15	MISO	SPI Control signal
16	SCLK	SPI clock
17	CS	SPI select
18	TX_data<9>	10-bit modulation data
19	TX_data<8>	
20	TX_data<7>	
21	TX_data<6>	
22	TX_data<5>	
23	TX_data<4>	
24	TX_data<3>	
25	TX_data<2>	
26	TX_data<1>	
27	TX_data<0>	
28	VDD_TDC	1 V TDC supply
29	DTC_OUT	DTC output for testing
30	TDC_IN	TDC input for testing
31	CLK_IN	32 MHz reference
32	VDD_DCO	1 V DCO supply

Measurement setup

The test PCB on which the packaged ADPLL prototype is mounted is shown in Figure A-2. The output clock of the ADPLL is taken out via an SMA connector. External signal generator is used to supply the reference clock and is connected to the test PCB via another SMA connector. The SPI registers of the ADPLL IC are controlled by a MATLAB program via FPGA.

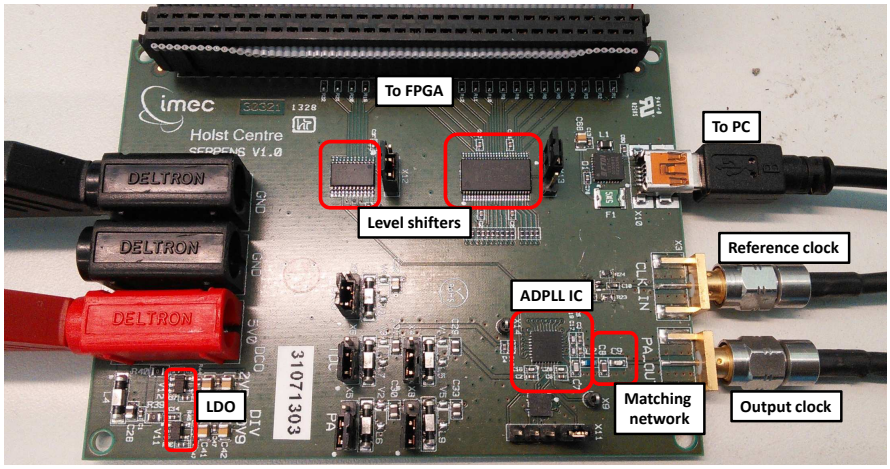


Figure A-2: Photograph of the test PCB.

The test bench built to characterize the implemented ADPLL prototype is shown in Figure A-3.

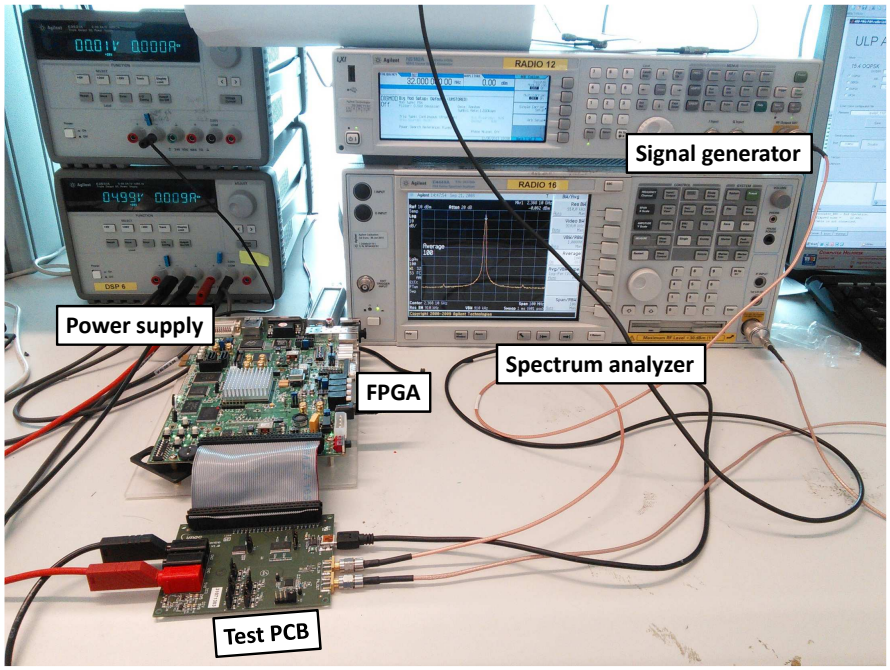


Figure A-3: Photograph of the test bench setup.

Appendix B

Simulations

B-1 Simulations of the capacitor banks

The post-layout estimations of the quality factor and capacitance step of the capacitor banks are shown here:

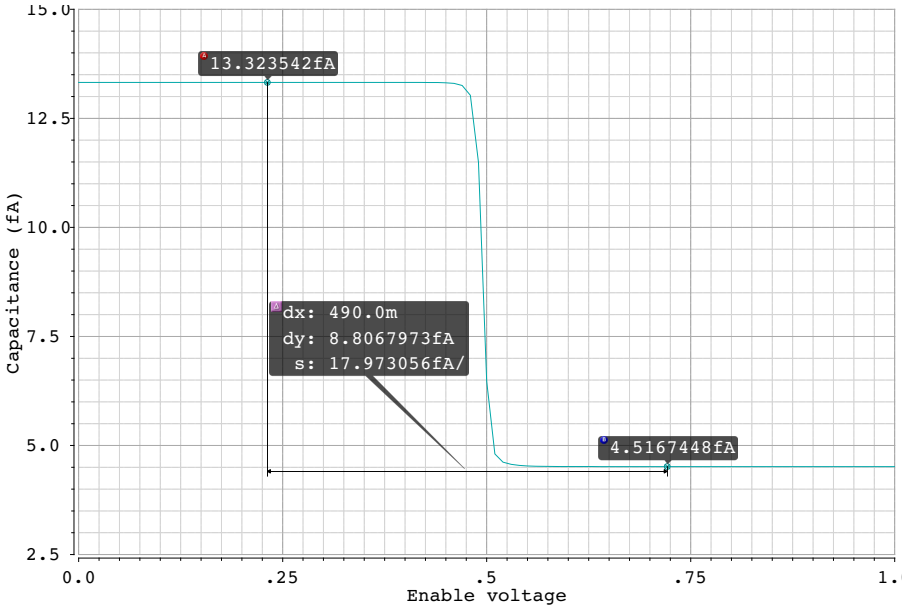


Figure B-1: C–V curve of the unit cell of the PVT bank.

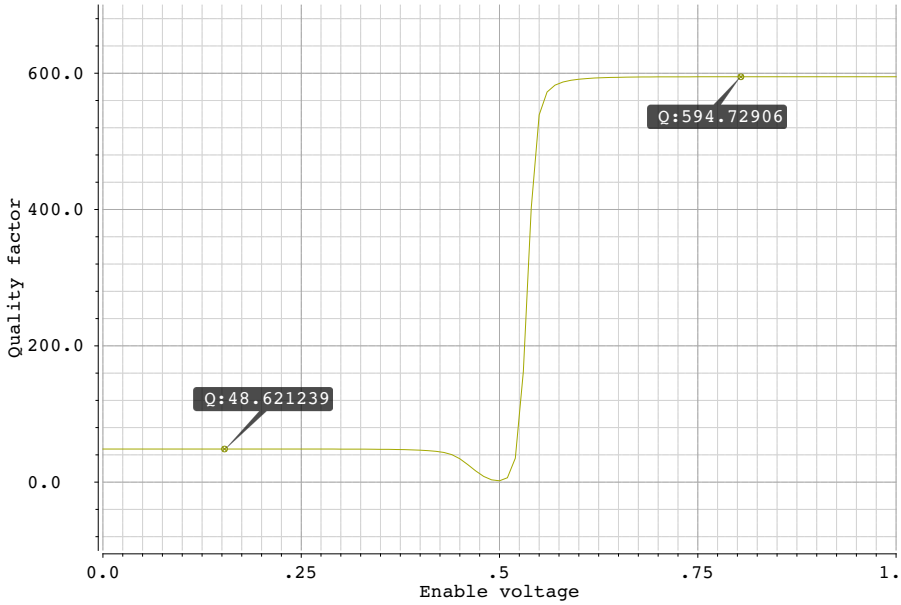


Figure B-2: Quality factor of the unit cell of the PVT bank.

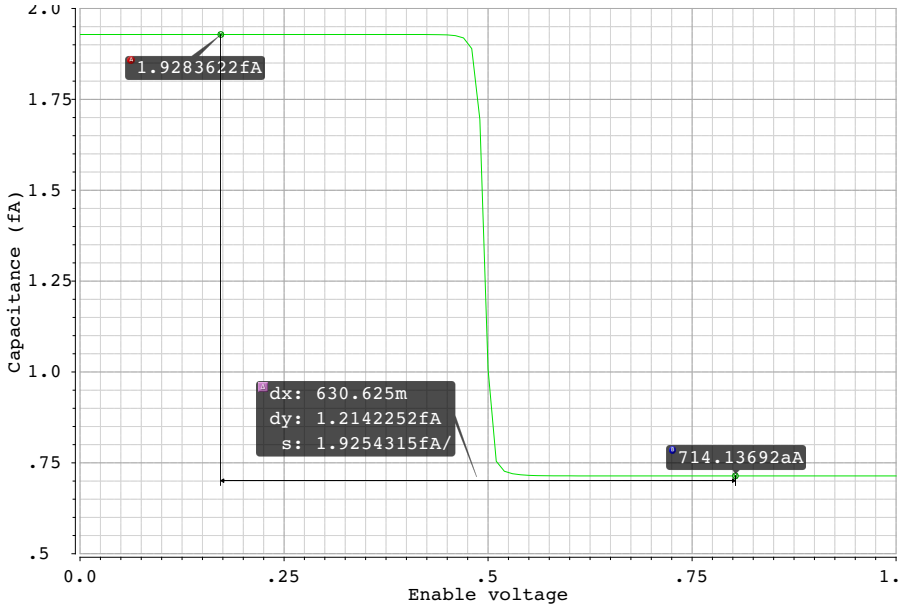


Figure B-3: C–V curve of the unit cell of the acquisition bank.

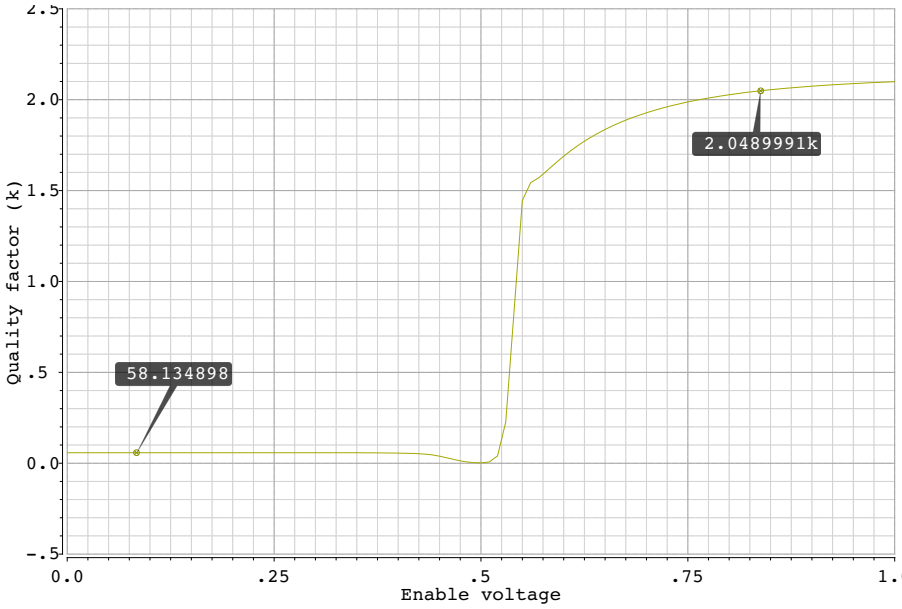


Figure B-4: Quality factor of the unit cell of the acquisition bank.

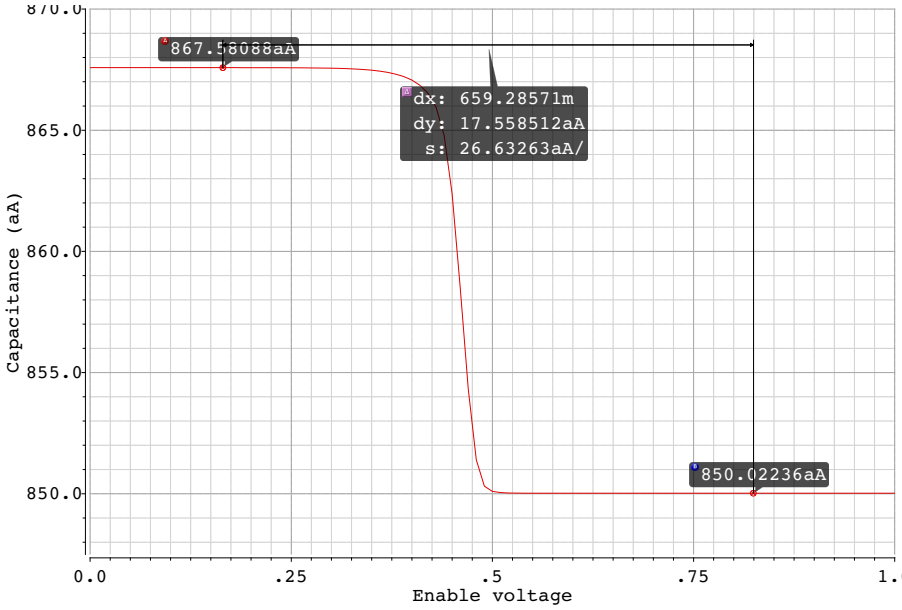


Figure B-5: C–V curve of the LSB cell of the tracking bank.

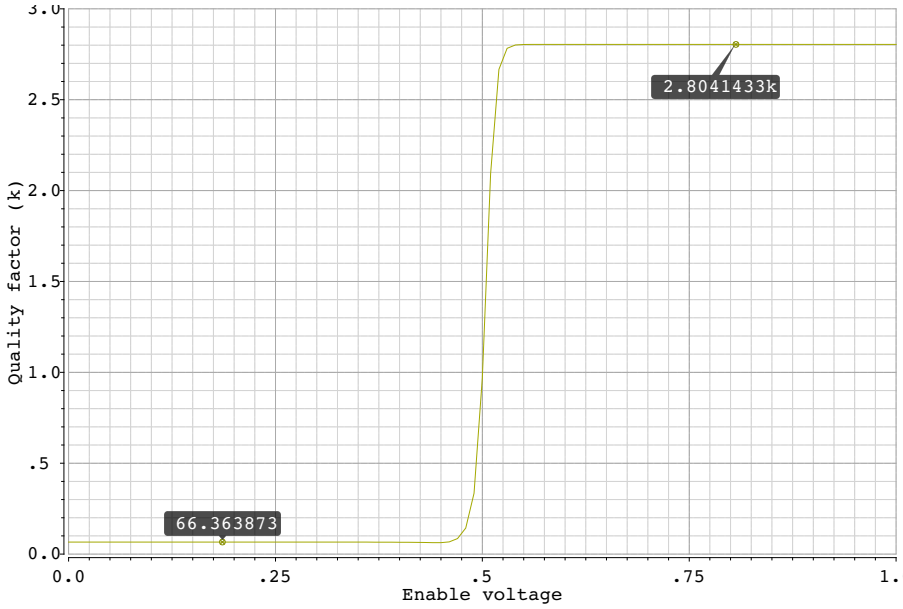


Figure B-6: Quality factor of the LSB cell of the tracking bank.

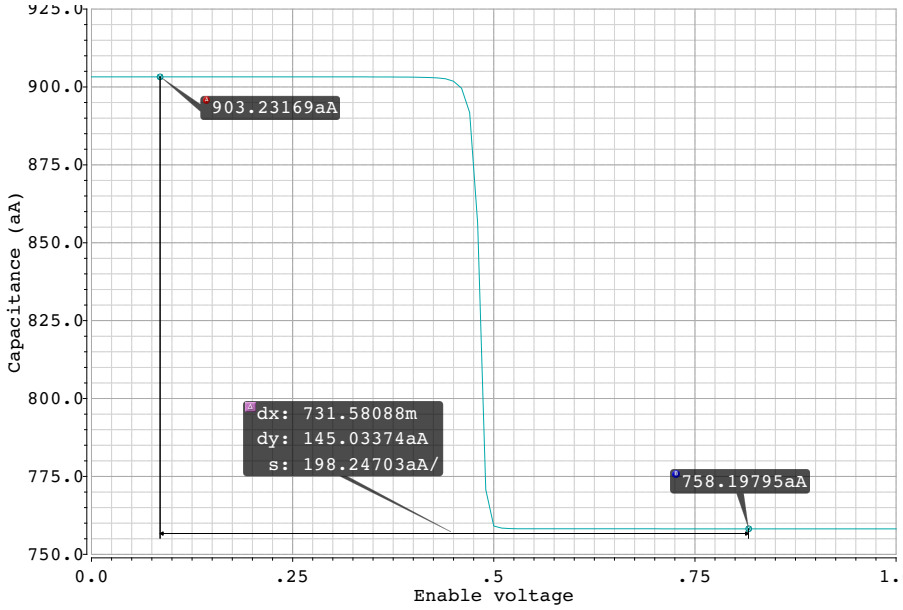


Figure B-7: C–V curve of the MSB cell of the tracking bank.

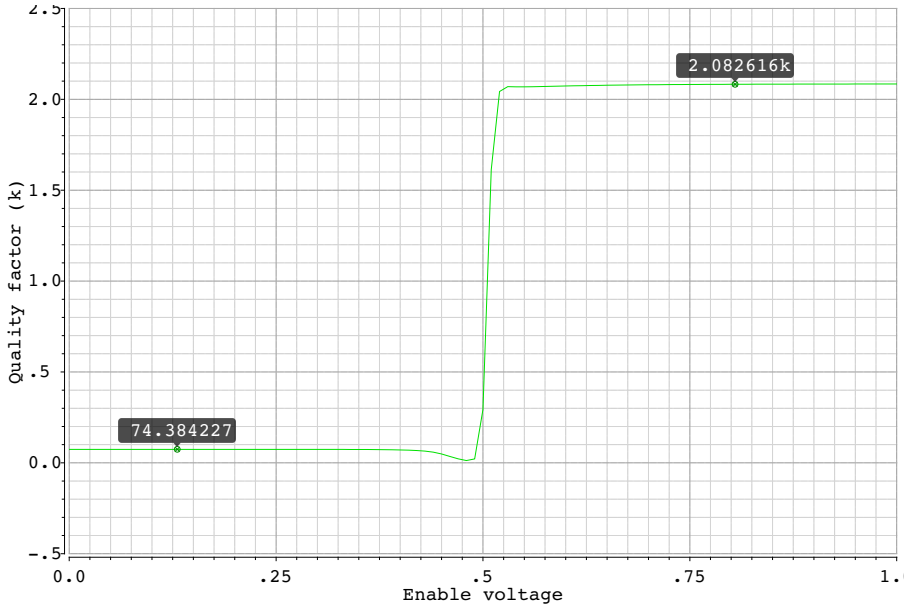


Figure B-8: Quality factor of the MSB cell of the tracking bank.

Bibliography

[1] V. K. Chillara, Y.-H. Liu, B. Wang, A. Ba, M. Vidojkovic, K. Philips, H. de Groot, and R. B. Staszewski., "An 860μW 2.1-to-2.7GHz All-Digital PLL-Based Frequency Modulator with a DTC-Assisted Snapshot TDC for WPAN (Bluetooth Smart and ZigBee) Applications." Solid-State Circuits Conference Digest of Technical Papers (ISSCC), Feb. 2014. to be published.

[2] C.-H. Park, S. Byun, Y. Song, S. Wang, C. Conroy, and B. Kim, "A low power CMOS Bluetooth transceiver with a digital offset canceling DLL-based GFSK demodulator," in *Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC. 2003 IEEE International*, pp. 96–481 vol.1, 2003.

[3] W. Kluge, F. Poegel, H. Roller, M. Lange, T. Ferchland, L. Dathe, and D. Eggert, "A Fully Integrated 2.4-GHz IEEE 802.15.4-Compliant Transceiver for ZigBee trade; Applications," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 12, pp. 2767–2775, 2006.

[4] H. Ishikuro, M. Hamada, K.-I. Agawa, S. Kousai, H. Kobayashi, D. M. Nguyen, and F. Hatori, "A single-chip CMOS Bluetooth transceiver with 1.5MHz IF and direct modulation transmitter," in *Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC. 2003 IEEE International*, pp. 94–480 vol.1, 2003.

[5] Y.-H. Liu, X. Huang, M. Vidojkovic, A. Ba, P. Harpe, G. Dolmans, and H. de Groot, "A 1.9nJ/b 2.4GHz multistandard (Bluetooth Low Energy/Zigbee/IEEE802.15.6) transceiver for personal/body-area networks," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International*, pp. 446–447, 2013.

[6] S. Kashmiri, M. A. P. Pertijs, and K. Makinwa, "A Thermal-Diffusivity-Based Frequency Reference in Standard CMOS With an Absolute Inaccuracy of 0.155 C to 125 C," *Solid-State Circuits, IEEE Journal of*, vol. 45, no. 12, pp. 2510–2520, 2010.

[7] M. Lutz, A. Partridge, P. Gupta, N. Buchan, E. Klaassen, J. McDonald, and K. Petersen, "MEMS Oscillators for High Volume Commercial Applications," in

Solid-State Sensors, Actuators and Microsystems Conference, 2007. TRANSDUCERS 2007. International, pp. 49–52, 2007.

[8] C. S. Lam, "A review of the recent development of MEMS and crystal oscillators and their impacts on the frequency control products industry," in *Ultrasonics Symposium, 2008. IUS 2008. IEEE*, pp. 694–704, 2008.

[9] F. M. Gardner, "Charge-Pump Phase-Lock Loops," *Communications, IEEE Transactions on*, vol. 28, no. 11, pp. 1849–1858, 1980.

[10] R. Staszewski, J. Wallberg, C.-M. Hung, G. Feygin, M. Entezari, and D. Leipold, "LMS-based calibration of an RF digitally controlled oscillator for mobile phones," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 53, no. 3, pp. 225–229, 2006.

[11] R.B.Staszewski, *Digital deep-submicron CMOS frequency synthesis for RF wireless applications*. Phd thesis, University of Texas, Dallas, 2002.

[12] R. Staszewski, K. Muhammad, D. Leipold, C.-M. Hung, Y.-C. Ho, J. Wallberg, C. Fernando, K. Maggio, R. Staszewski, T. Jung, J. Koh, S. John, I. Y. Deng, V. Sarda, O. Moreira-Tamayo, V. Mayega, R. Katz, O. Friedman, O. Eliezer, E. de Obaldia, and P. Balsara, "All-digital TX frequency synthesizer and discrete-time receiver for Bluetooth radio in 130-nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 39, no. 12, pp. 2278–2291, 2004.

[13] R. Staszewski, D. Leipold, O. Eliezer, M. Entezari, K. Muhammad, I. Bashir, C. M. Hung, J. Wallberg, R. Staszewski, P. Cruise, S. Rezeq, S. Vemulapalli, K. Waheed, N. Barton, M. C. Lee, C. Fernando, K. Maggio, T. Jung, I. Elahi, S. Larson, T. Murphy, G. Feygin, I. Deng, T. Mayhugh, Y. C. Ho, K. M. Low, C. Lin, J. Jaehnig, J. Kerr, J. Mehta, S. Glock, T. Almholt, and S. Bhatara, "A $24mm^2$ Quad-Band Single-Chip GSM Radio with Transmitter Calibration in 90nm Digital CMOS," in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pp. 208–607, 2008.

[14] J. Mehta, R. Staszewski, O. Eliezer, S. Rezeq, K. Waheed, M. Entezari, G. Feygin, S. Vemulapalli, V. Zoicas, C.-M. Hung, N. Barton, I. Bashir, K. Maggio, M. Frechette, M.-C. Lee, J. Wallberg, P. Cruise, and N. Yanduru, "A $0.8mm^2$ all-digital SAW-less polar transmitter in 65nm EDGE SoC," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pp. 58–59, 2010.

[15] H.-H. Chang, P.-Y. Wang, J.-H. Zhan, and B.-Y. Hsieh, "A Fractional Spur-Free ADPLL with Loop-Gain Calibration and Phase-Noise Cancellation for GSM/GPRS/EDGE," in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pp. 200–606, 2008.

[16] L. Fanori, A. Liscidini, and R. Castello, "3.3GHz DCO with a frequency resolution of 150Hz for All-digital PLL," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pp. 48–49, 2010.

[17] R. Staszewski, C.-M. Hung, D. Leipold, and P. Balsara, "A first multigigahertz digitally controlled oscillator for wireless applications," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 51, no. 11, pp. 2154–2164, 2003.

[18] R. Staszewski, S. Vemulapalli, P. Vallur, J. Wallberg, and P. Balsara, "1.3 V 20 ps time-to-digital converter for frequency synthesis in 90-nm CMOS," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 53, no. 3, pp. 220–224, 2006.

[19] M. Lee, M. Heidari, and A. Abidi, "A low noise, wideband digital phase-locked loop based on a new time-to-digital converter with subpicosecond resolution," in *VLSI Circuits, 2008 IEEE Symposium on*, pp. 112–113, 2008.

[20] C.-M. Hsu, M. Straayer, and M. Perrott, "A Low-Noise, Wide-BW 3.6GHz Digital $\Delta\Sigma$ Fractional-N Frequency Synthesizer with a Noise-Shaping Time-to-Digital Converter and Quantization Noise Cancellation," in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pp. 340–617, 2008.

[21] A. Liscidini, L. Vercesi, and R. Castello, "Time to digital converter based on a 2-dimensions Vernier architecture," in *Custom Integrated Circuits Conference, 2009. CICC '09. IEEE*, pp. 45–48, 2009.

[22] E. Temporiti, C. Weltin-Wu, D. Baldi, R. Tonietto, and F. Svelto, "A 3 GHz Fractional All-Digital PLL With a 1.8 MHz Bandwidth Implementing Spur Reduction Techniques," *Solid-State Circuits, IEEE Journal of*, vol. 44, no. 3, pp. 824–834, 2009.

[23] R. Staszewski, K. Waheed, F. Dulger, and O. Eliezer, "Spur-Free Multirate All-Digital PLL for Mobile Phones in 65 nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 46, no. 12, pp. 2904–2919, 2011.

[24] R. Staszewski, "State-of-the-art and future directions of high-performance all-digital frequency synthesis in nanometer CMOS," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pp. 229–232, 2010.

[25] M. Vidojkovic, Y.-H. Liu, X. Huang, K. Imamura, G. Dolmans, and H. De Groot, "A fully integrated 1.7–2.5GHz 1mW fractional-N PLL for WBAN and WSN applications," in *Radio Frequency Integrated Circuits Symposium (RFIC), 2012 IEEE*, pp. 185–188, 2012.

[26] D. Park, W. Lee, S. Jeon, and S. Cho, "A 2.5-GHz 860 μW charge-recycling fractional-N frequency synthesizer in 130nm CMOS," in *VLSI Circuits, 2008 IEEE Symposium on*, pp. 88–89, 2008.

[27] G. Devita, A. C. W. Wong, N. Kasparidis, P. Corbishley, A. Burdett, and P. Paddan, "A 0.9mW PLL integrated in an ultra-low-power SoC for WPAN and WBAN applications," in *ESSCIRC, 2010 Proceedings of the*, pp. 158–161, 2010.

[28] A. Italia and G. Palmisano, "A 1.2-mW CMOS frequency synthesizer with fully-integrated LC VCO for 400-MHz medical implantable transceivers," in *Radio Frequency Integrated Circuits Symposium, 2009. RFIC 2009. IEEE*, pp. 333–336, 2009.

[29] W.-H. Chen, W.-F. Loke, G. Thompson, and B. Jung, "A 0.5-V, 440μW frequency synthesizer for implantable medical devices," in *Custom Integrated Circuits Conference (CICC), 2011 IEEE*, pp. 1–4, 2011.

[30] X. Gao, E. Klumperink, M. Bohsali, and B. Nauta, "A 2.2GHz 7.6mW sub-sampling PLL with −126dBc/Hz in-band phase noise and 0.15psrms jitter in 0.18μm CMOS," in *Solid-State Circuits Conference - Digest of Technical Papers, 2009. ISSCC 2009. IEEE International*, pp. 392–393,393a, 2009.

[31] R. Staszewski, C.-M. Hung, K. Maggio, J. Wallberg, D. Leipold, and P. Balsara, "All-digital phase-domain TX frequency synthesizer for Bluetooth radios in 0.13 μ m CMOS," in *Solid-State Circuits Conference, 2004. Digest of Technical Papers. ISSCC. 2004 IEEE International*, pp. 272–527 Vol.1, 2004.

[32] R. B. Staszewski and P. T. Balsara, *All-Digital Frequency Synthesizer in Deep-Submicron CMOS*. Wiley-Interscience, 2006.

[33] E. Temporiti, G. Albasini, I. Bietti, R. Castello, and M. Colombo, "A 700-khz bandwidth sigma; delta; fractional synthesizer with spurs compensation and linearization techniques for wcdma applications," *Solid-State Circuits, IEEE Journal of*, vol. 39, no. 9, pp. 1446–1454, 2004.

[34] L. Xu, S. Lindfors, K. Stadius, and J. Ryynanen, "A 2.4-GHz low-power all-digital phase-locked loop," in *Custom Integrated Circuits Conference, 2009. CICC '09. IEEE*, pp. 331–334, 2009.

[35] W. R. Bennett, "Spectra of quantized signals," *Bell System Technical Journal*, vol. 27, no. 3, pp. 446–472, 1948.

[36] P. Dudek, S. Szczepanski, and J. Hatfield, "A high-resolution cmos time-to-digital converter utilizing a vernier delay line," *Solid-State Circuits, IEEE Journal of*, vol. 35, no. 2, pp. 240–247, 2000.

[37] M. Perrott, I. Tewksbury, T.L., and C. Sodini, "A 27-mw cmos fractional-n synthesizer using digital compensation for 2.5-mb/s gfsk modulation," *Solid-State Circuits, IEEE Journal of*, vol. 32, no. 12, pp. 2048–2060, 1997.

[38] R. A. Meyers and P. H. Waters, "Synthesiser review for pan-european digital cellular radio," in *VLSI Implementations for Second Generation Digital Cordless and Mobile Telecommunication Systems, IEE Colloquium on*, pp. 8/1–810, 1990.

[39] M. Bopp, M. Alles, M. Arens, D. Eichel, S. Gerlach, R. Gotzfried, F. Gruson, M. Kocks, G. Krimmer, R. Reimann, B. Roos, M. Siegle, and J. Zieschang, "A dect transceiver chip set using sige technology," in *Solid-State Circuits Conference, 1999. Digest of Technical Papers. ISSCC. 1999 IEEE International*, pp. 68–69, 1999.

[40] M. Ferriss and M. Flynn, "A 14mW Fractional-N PLL Modulator with an Enhanced Digital Phase Detector and Frequency Switching Scheme," in *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pp. 352–608, 2007.

[41] W. Khalil, S. Shashidharan, T. Copani, S. Chakraborty, S. Kiaei, and B. Bakkaloglu, "A 700-μA 405-MHz All-Digital Fractional-N Frequency-Locked Loop for ISM Band Applications," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 59, no. 5, pp. 1319–1326, 2011.

[42] 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-fs_{rms} Integrated Jitter at 4.5-mW Power," *J. Solid-State Circuits*, vol. 46, no. 12, pp. 2745–2758, 2011.

[43] N. Pavlovic and J. Bergervoet, "A 5.3GHz digital-to-time-converter-based fractional-N all-digital PLL," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, pp. 54–56, 2011.

[44] J. Zhuang and R. Staszewski, "A low-power all-digital PLL architecture based on phase prediction," in *Electronics, Circuits and Systems (ICECS), 2012 19th IEEE International Conference on*, pp. 797–800, 2012.

[45] J. wei Lai, C.-H. Wang, K. Kao, A. Lin, Y.-H. Cho, L. Cho, M.-H. Hung, X.-Y. Shih, C.-M. Lin, S.-H. Yan, Y.-H. Chung, P. Liang, G.-K. Dehng, H.-S. Li, G. Chien, and R. Staszewski, "A 0.27mm^2 13.5dBm 2.4GHz all-digital polar transmitter using 34Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2013 IEEE International, pp. 342–343, 2013.

[46] T. Tokairin, M. Okada, M. Kitsunezuka, T. Maeda, and M. Fukaishi, "A 2.1-to-2.8GHz all-digital frequency synthesizer with a time-windowed TDC," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pp. 470–471, 2010.

[47] M. Park, M. Perrott, and R. Staszewski, "An Amplitude Resolution Improvement of an RF-DAC Employing Pulsewidth Modulation," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 58, no. 11, pp. 2590–2603, 2011.

[48] J. Zhuang and R. Staszewski, "Gain estimation of a digital-to-time converter for phase-prediction all-digital PLL," in *Circuit Theory and Design (ECCTD), 2013 21st European Conference on*, 2013.

[49] K. Waheed and R. Staszewski, "Time-domain behavioral modeling of a multi-gigahertz digital rf oscillator using vhdl," in *Circuits and Systems, 2005. 48th Midwest Symposium on*, pp. 1669–1672 Vol. 2, 2005.

[50] A. Ismail and A. Abidi, "CMOS differential LC oscillator with suppressed up-converted flicker noise," in *Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC. 2003 IEEE International*, pp. 98–99 vol.1, 2003.

[51] P. Andreani and S. Mattisson, "On the use of MOS varactors in RF VCOs," *Solid-State Circuits, IEEE Journal of*, vol. 35, no. 6, pp. 905–910, 2000.

[52] W. Jiang, A. Tavakol, P. Effendrik, M. van de Gevel, F. Verwaal, and R. Staszewski, "Design of ADPLL system for WiMAX applications in 40-nm CMOS," in *Electronics, Circuits and Systems (ICECS), 2012 19th IEEE International Conference on*, pp. 73–76, 2012.

[53] C.-M. Hung, R. Staszewski, N. Barton, M.-C. Lee, and D. Leipold, "A digitally controlled oscillator system for SAW-less transmitters in cellular handsets," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 5, pp. 1160–1170, 2006.

Glossary

List of Acronyms

ADC	analog-to-digital converter
ADPLL	all-digital phase-locked loop
BBPLL	bang-bang phase-locked loop
CKR	re-timed reference
CKV	variable clock
CMOS	complementary metal-oxide-semiconductor
CP-PLL	charge-pump phase-locked loop
DCO	digitally controlled oscillator
DIBL	drain-induced barrier lowering
DTC	digital-to-time converter
FCW	frequency command word
FoM	figure-of-merit
FPGA	field programmable gate array
FSM	finite state machine
IF	intermediate frequency
LDO	low-dropout
LNA	low-noise amplifier
LO	local oscillator
MICS	medical implant communication service
OTW	oscillator tuning word

PCB	printed circuit board
PGA	programmable gain amplifier
PLL	phase-locked loop
PVT	process, voltage, and temperature
PA	power amplifier
RF	radio frequency
SNR	signal-to-noise ratio
TDC	time-to-digital converter
VCO	voltage-controlled oscillator
VTC	voltage transfer characteristic
WBAN	wireless body area network
WPAN	wireless personal area network