

國立臺灣大學電機資訊學院電子工程學研究所

碩士論文

Graduate Institute of Electronics Engineering
College of Electrical Engineering & Computer Science
National Taiwan University
Master Thesis



一個具有時間數位轉換器和數位延遲轉換器線性化技
術應用於頻率鍵移調變之小數型全數位鎖相迴路
A Fractional-N All-Digital Phase-Locked Loop using
TDC and DTC Linearization Technology for FSK
Application

張庭豪

Ting-Hao Chang

指導教授：林宗賢 博士

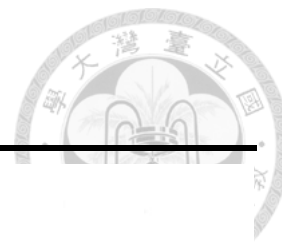
Advisor: Tsung-Hsien Lin, Ph.D.

中華民國 112 年 7 月

July, 2023



審定書



國立臺灣大學碩士學位論文 口試委員會審定書

MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

一個具有時間數位轉換器和數位延遲轉換器線性化技術應用
於頻率鍵移調變之小數型全數位鎖相迴路

A Fractional-N All-Digital Phase-Locked Loop using TDC and
DTC Linearization Technology for FSK Application

本論文係張庭豪 (R09943054) 在國立臺灣大學電子工程研究所完成之碩士學位論文，於民國 112 年 7 月 28 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Department / Institute of Electronics Engineering on 28 July 2023 have examined a Master's thesis entitled above presented by Ting-Hao Chang (R09943054) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

(指導教授 Advisor)

系主任/所長 Director:





To My Dear Family
謹獻給我的家人



摘要

本論文實現一個基於小數型全數位鎖相迴路兩點調製架構，在所提出的兩點調製器中，利用前饋消除技術消除兩信號注入點間的路徑延遲擴展。此架構採用計數器為基礎的數位鎖相迴路以達到極短的鎖定時間，時間數位轉換器運用亞時間數位轉換器對應環形振盪器多相位輸出降低單位數位轉換器所需偵測範圍，時間數位轉換器採用雙內插法克服時間數位轉換器因跳頻造成的增益誤差，並仍保有良好的線性度。以數位時間轉換器在回授路徑添加額外的噪聲抖動，隨機化時間數位轉換器的輸入信號，以降低鎖相迴路輸出之小數突波。

此鎖相迴路系統操作於1.2V，共花費6.71 mA電流，輸入參考時脈信號頻率為40 MHz，可輸出信號頻率為23.5至24.5 GHz，採用TSMC 90奈米製程設計，在2.4000390625 GHz下，所量測到的小數突波為-39 dBc。在2.4 GHz下，所量測的相位雜訊於1 MHz頻率偏移下為-99 dBc/Hz，由1 kHz積分到100 MHz的均方根抖動為2.24 ps，FoM_{Jitter}為-223dB。

關鍵字:全數位鎖相迴路、兩點調製、時間數位轉換器、數位時間轉換器、雙插值法



Abstract

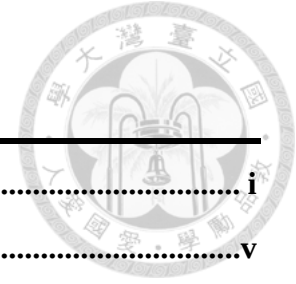
The thesis implement a Two-Point Modulation (TPM) architecture based on a fractional-N all digital phase-locked loop. In the proposed two-point modulator, the feedforward cancellation technique is used to eliminate the path delay spread between two signal injection points. This architecture uses a counter-based digital phase-locked loop to achieve extremely short lock times. The Time-to-Digital Converter (TDC) utilizes a set of sub-TDCs that correspond to the multi-phase output of the ring oscillator. This approach helps in reducing the detection range required for each individual sub-TDC. The converter adopts double interpolation method to overcome the TDC conversion gain error caused by frequency hopping, and still maintains good linearity. Noise dithering at the feedback path by a Digital-to-Time Converter (DTC) randomizes the TDC input signal pattern so that the fractional spur of the PLL output is reduced.

The TPM architecture is implemented in the design a 2.35-2.45 GHz ADPLL. Fabricated in the TSMC 90-nm CMOS technology, the whole system dissipates 6.71 mA from a 1.2 V supply and the active area is 0.108 mm². At 2.4000390625 GHz, the fractional spur is -39 dBc. At 2.4 GHz, the phase noise measured at 1 MHz offset is -99 dBc/Hz. RMS jitter integrated from 1 kHz to 100 MHz is 2.24 ps and figure-of-merit (FOM) is -223 dB.

Key word: All-Digital Phase-Locked Loop, Two-Point Modulation, Time-to-Digital Converter, Digital-to-Time Converter, Double Interpolation



Table of Contents



審定書	i
摘要	v
<i>Abstract</i>	vii
<i>List of Figures</i>	xi
<i>List of Tables</i>	xv
Chapter 1 Introduction	1
1.1 Motivation	1
1.2 Thesis Overview	3
Chapter 2 Introduction to All-Digital PLL & Two Point Modulation Technology	5
2.1 Basics of All-Digital-Phase-Locked Loops	5
2.2 Quantization Noise Analysis of Fractional-N ADPLL.....	7
2.2.1 TDC Quantization Noise	8
2.2.2 DCO Quantization Noise	9
2.3 Frequency Shift Keying (FSK) Modulation using Two Point Modulation (TPM) Technology	10
2.3.1 DCO Gain Variation for TPM architecture	11
2.3.2 Adaptive Gain Compensation of DCO	12
2.3.3 Path Delay Spread for TPM architecture.....	13
2.3.4 Proposed Delay Spread compensation to ADPLL	14
Chapter 3 TDC and DTC Linearization Technology	17
3.1 TDC Conversion Gain Variation during Frequency Hopping	17
3.2 Dual-Referenced Interpolating TDC	19
3.3 Quadruple-Referenced Interpolating TDC	23
3.4 Noise Dithering using Inverse DTC	26
3.4.1 DTC Based on Time-Mode Arithmetic Units	28
3.4.2 Inverse DTC Based on a Pulse Generator	29
3.4.3 Immunity to Channel-Length Modulation.....	30
Chapter 4 Implementation of a 2.4 GHz Fractional-N All-Digital PLL	33
4.1 Analog Design	34
4.1.1 Digital-Controlled Oscillator	34
4.1.2 Synchronous Counter and Sampling Circuit	35
4.1.3 Time-to-Digital Converter	36

4.1.4 Sense-Amplifier-Based Flip-Flop.....	39
4.1.5 Digital-to-Time Converter.....	40
4.2 RTL Design.....	42
4.2.1 Digital Loop Filter.....	42
4.2.2 Frequency Hopping Control Logic.....	44
4.2.3 Fast Locking Function.....	45
4.2.4 Arbiter.....	47
4.2.5 Calibration for DTC.....	48
4.2.6 Other Block.....	48
Chapter 5 Measurement Results.....	51
5.1 Measurement Setup.....	51
5.2 Chip Pin Configuration and Chip Photo.....	52
5.3 Measurement Results.....	53
5.3.1 Spectrum & Phase Noise for Integer-N channel.....	53
5.3.2 Spectrum & Phase Noise of a Near-Integer Fractional-N Channel.....	55
5.3.3 Loop-Bandwidth Drifts vs. DCO Nonlinearity.....	59
5.3.4 Frequency Hopping Measurement with TPM Architecture.....	60
5.3.5 BFSK Modulation Measurement with TPM Architecture.....	61
5.4 Summary.....	62
Chapter 6 Conclusions and Future Works.....	65
6.1 Conclusions.....	65
6.2 Directions for Improvement of this Chip.....	66
6.2.1 Loop-Bandwidth Tuning Capability.....	66
6.2.2 Calibration Loop Optimization.....	66
6.2.3 Enhancement of DCO Linearity.....	67
6.2.4 Other.....	67
6.3 Directions for Improvement of this Chip.....	68
6.2.1 Motivation.....	68
6.2.2 DTC-Assisted ADPLL.....	68
6.2.3 Rewritten of DTC Predicted Delay.....	69
6.2.4 Time-Register using Time-Mode Arithmetic Units.....	71
6.2.5 Proposed Time-Registers-Based ADPLL.....	72
6.2.6 Issues Still to be Resolved.....	73
6.2.7 Possible Solutions.....	74
References.....	77

List of Figures

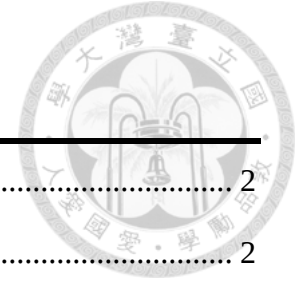


Fig. 1-1 Modulated waveform of FSK	2
Fig. 1-2 Block diagram of FSK modulator.....	2
Fig. 2-1 Block diagram of a typical CPPLL	5
Fig. 2-2 Block diagram of a divider-based ADPLL	6
Fig. 2-3 Block diagram of a counter-based ADPLL	7
Fig. 2-4 s-domain model of a counter-based ADPLL	8
Fig. 2-5 Improving DCO resolution with $\Delta\Sigma$ noise shaping technique.....	9
Fig. 2-6 Block diagram of the ADPLL with one-point modulation	10
Fig. 2-7 Block diagram of the ADPLL with two-point modulation.....	11
Fig. 2-8 Example of TPM with DCO gain variation	12
Fig. 2-9 Adaptive gain compensation of DCO gain variation.....	13
Fig. 2-10 Example of TPM with injection timing mismatch between two injection points	14
Fig. 2-11 Block diagram of conventional delay spread compensation.....	15
Fig. 2-12 Block Diagram of proposed delay spread compensation.....	16
Fig. 2-13 Example waveform of forward cancellation technology	16
Fig. 3-1 Analysis of a classical delay-chain TDC in (a) the conceptual brief and (b) the linear model	18
Fig. 3-2 Time Diagram with a frequency 2.4 and 3.4 times that of the reference frequency	19
Fig. 3-3 Block Diagram of the dual-referenced interpolating TDC (DI-TDC).....	20
Fig. 3-4 Block diagram and time diagram of quadrant detector.....	20
Fig. 3-5 (a) Conversion error by delay change with the conventional TDC (b) Concept of error compensation by the QI-TDC	21

Fig. 3-6 Time diagram of DI-TDC with (a) $T_{TDC,res} > T_{DCO}/12$ (b) $T_{TDC,res} < T_{DCO}/12$	22
Fig. 3-7 Block diagram of Quadruple-Referenced Interpolating TDC (QI-TDC)	24
Fig. 3-8 Comparison of time diagram of (a) DI-TDC and (b) QI-TDC	25
Fig. 3-9 Simulation result for QI-TDC and DI-TDC	25
Fig. 3-10 Performance of ADPLL with TDC nonlinearity	26
Fig. 3-11 Block diagram of noise dithering and dither-noise-cancellation loop	27
Fig. 3-12 (a) Block diagram of the prior art of DTC based on time-mode-arithmetic units and (b) its time diagram	28
Fig. 3-13 Time diagram with non-ideal switch effect	29
Fig. 3-14 Block diagram of proposed DTC based and (b) its time diagram	30
Fig. 3-15 Difference between any two SWD ON-time corresponding to different codes is an accurate integer multiple of $T_{DCO}/8$	30
Fig. 4-1 Block diagram of proposed ADPLL for FSK application	33
Fig. 4-2 (a) Schematic of four-stage differential RDCO (b) Schematic of delay cell	34
Fig. 4-3 Schematic of 5-bit synchronous counter and sampling circuit	36
Fig. 4-4 Schematic of delay chain of TDC	37
Fig. 4-5 (a) Schematic of inverter-based delay element (b) Schematic of C-DAC-based delay element	37
Fig. 4-6 Operation of C-DAC based delay element	39
Fig. 4-7 Schematic of sense-amplifier-based flip-flop	40
Fig. 4-8 Schematic of inverse DTC	41
Fig. 4-9 Schematic of pulse generator (PG)	42
Fig. 4-10 Topology between the CPPLL and the ADPLL	43
Fig. 4-11 Simulation result of ADPLL by Matlab	44
Fig. 4-12 Simulated phase noise by Matlab	44
Fig. 4-13 Finite State Machine (FSM) for DLF input control	46

Fig. 4-14 FSM transitions to PRS state if ferr (ΔPHE) changes sign	46
Fig. 4-15 Time diagram of fast-locking operation	46
Fig. 4-16 Time diagram of arbiter	47
Fig. 4-17 Block diagram of calibration block for DTC gain calibration	48
Fig. 4-18 Overall architecture of ADPLL marked with different clock domain	49
Fig. 5-1 Measurement setup	51
Fig. 5-2 Pin configuration and power domain diagram	52
Fig. 5-3 Chip photo	52
Fig. 5-4 Spectrum at 2.4 GHz	54
Fig. 5-5 Phase noise at 2.4 GHz	54
Fig. 5-6 Measured PLL spectrum for a near-integer fractional-N channel with $FCW = 60 + 2^{-10}$	55
Fig. 5-7 Measured phase noise with dithering OFF for a near-integer fractional-N channel with $FCW = 60 + 2^{-10}$	56
Fig. 5-8 Overlay analysis of measured phase noise with dithering and calibration ON or not	57
Fig. 5-9 (a) Maximum fractional spur level and (b) integrated jitter from 1 kHz to 100 MHz during scaled fractional-mode operation	58
Fig. 5-10 Measured PLL spectrum for a near-integer fractional-N channel with $FCW = 60 + 2^{-6}$	58
Fig. 5-11 Code-to-frequency curve of the DCO	59
Fig. 5-12 Phase noise with a reference frequency ranging from 39.5 to 40.2 MHz	60
Fig. 5-13 Measured time diagram when (a) TPM OFF (b) TPM OFF	60
Fig. 5-14 Summary of BFSK modulation measurements	61
Fig. 6-1 Principle of pre-distortion circuit	67
Fig. 6-2 Block diagram of DTC-assisted ADPLL	68

Fig. 6-3 Phase prediction scheme for $FCW = 2.25$	69
Fig. 6-4 Phase prediction by Eq.6.3 for $FCW = 2.25$	70
Fig. 6-5 (a) Block diagram of time register and (b) its time diagram.....	71
Fig. 6-6 Block diagram of proposed ADPLL	72
Fig. 6-7 Equivalent circuit of proposed design	73
Fig. 6-7 Simulation result of behavior model.....	74

List of Tables

Table 3-1 System specification.....	23
Table 4-1 Specification of the ADPLL design example	35
Table 4-2 Design parameters of the ADPLL design example.....	43
Table 5-1 Measured current consumption	53
Table 5-2 Measured performance summary for the proposed ADPLL	62
Table 5-3 Comparison table	63



Chapter 1 Introduction



1.1 Motivation

The advancement of wireless communication has contributed to the flourishing development of the internet of things (IoT) and wireless sensor network (WSN). Tens of billions of smart sensor nodes will be deployed for environment monitoring and information communicating in emerging applications such as smart factories, smart homes, wearable health devices, smart cars, smart grids and others. Most of the sensors in these applications do not require extremely high-speed data transmission, cost and power consumption are the main design considerations. Frequency shift keying (FSK) modulation provides a simple and stable data transmission method for the low-speed problem.

FSK is a digital modulation technique where the frequency of the carrier signal varies based on changes in the digital signal. In FSK, the output waveform has a higher frequency for a binary High input and a lower frequency for a binary Low input. Figure 1-1 illustrates the diagrammatic representation of the FSK modulated waveform along with its corresponding input.

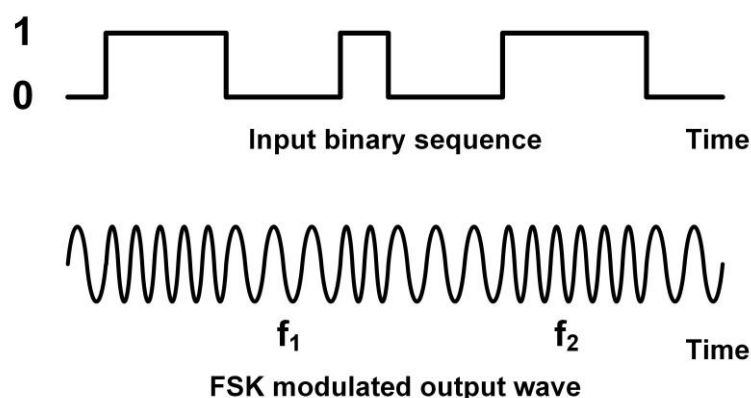


Fig. 1-1 Modulated waveform of FSK

A typical FSK transmitter is shown in Figure 1-2. The scheme involves two frequency synthesizers that generate signals with different frequencies. These frequency synthesizers are connected to a switch along with an internal clock. The binary input sequence is then applied to the transmitter, which selects the frequencies based on the binary input.

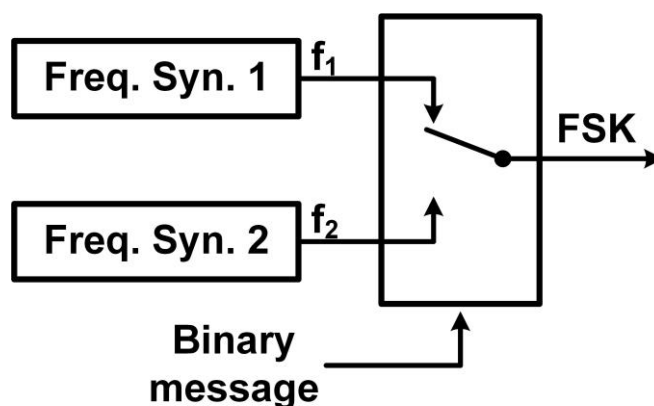


Fig. 1-2 Block diagram of FSK modulator

The frequency synthesizers are significant components in FSK scheme. For different communication channels, the output frequency is demanded to be programmable, which is usually generated by phase-locked loop (PLL). The PLL is widely used in system-on-chips (SoCs) owing to its negative feedback system that ensures the phase of the desired output signal locks to the reference input. The integer-N PLL can only generate clocks

whose frequencies are integer multiples of the reference frequency. In order to improve the resolution of the output frequency, fractional-N PLL use a delta-sigma modulator (DSM) to modulate divisor to synthesize effective fractional frequency throughout averaging.

Thanks to its programmable frequency capability, the PLL can function as an independent FSK architecture. As a result, the FSK scheme depicted in Fig 1-2 can eliminate the need for a separate frequency synthesizer and the subsequent signal selector by converting the binary message into the frequency command word, FCW, or divisor. The all-digital phase-locked loop (ADPLL) is a preferred solution for FSK application. Compared to analog PLL, the all-digital phase-locked loop (ADPLL) benefits from digital intensive nature such as low area, low power, high integration and high reconfigurability. Furthermore, since most of the main modules are digitally controlled, such as digital calibration, fractional division and two-point modulation (TPM) can be easily realized. In this thesis, an ADPLL for FSK application is presented.

1.2 Thesis Overview

This thesis is organized as follows. The introduction of ADPLL are described in Chapter 2. The principle of TPM which is usually used in PLL-based modulator is also presented here. Chapter 3 introduces the time-to-digital converter (TDC) and digital-to-time converter (DTC) linearization technology. Chapter 4 introduces the design considerations and the implementations of the building blocks. The measurement results of this work are demonstrated in Chapter 5. Finally, the conclusion and the future works are presented in Chapter 6.



Chapter 2 Introduction to All-Digital PLL & Two Point Modulation Technology



2.1 Basics of All-Digital-Phase-Locked Loops

Before introducing ADPLLs, the fundamental knowledge of analog PLLs are first introduced for better understanding of the following digital counterpart. The analog PLLs, such as the charge pump PLL (CPPLL) shown in Figure 2-1. REF is the input reference frequency and FBV is the output frequency of voltage-controlled oscillator (VCO) divided by N. The phases of REF and FBV are compared by phase/frequency detector (PFD) which estimates the time difference between their closest edges, and then charge pump (CP) charge/discharge the loop filter (LF) according to the UP/DN pulse. V_{CONT} then increases/decrease output frequency of VCO, attempting to minimize phase difference between REF and FBV. After the loop is locked, the output frequency becomes N times of REF.

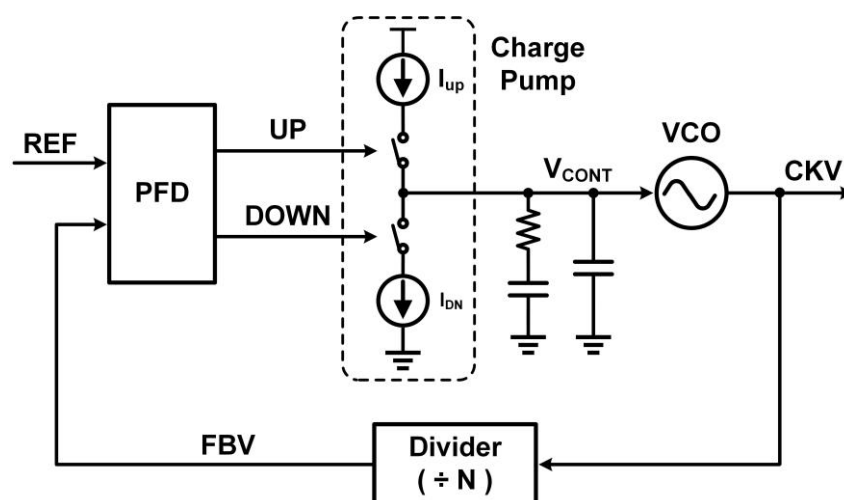


Fig. 2-1 Block diagram of a typical CPPLL

Since PLL is a discrete system, maintaining loop stability requires setting the loop bandwidth to less than 1/10 of the reference frequency. However, achieving this often necessitates a large capacitance in the loop filter, resulting in increased chip area consumption. To address this issue, various PLL structures have been proposed, such as the active loop filter [1] and time-based loop filter [2]. In digital PLLs [3], the passive loop filter, which occupies a significant area, is replaced by a digital loop filter (DLF). With advancements in CMOS technology, digital PLLs offer the promise of lower power consumption and smaller chip area requirements. In a classic ADPLL architecture, as shown in Figure 2-2, the VCO is replaced by a DCO, the PFD with CP is replaced by a TDC, and the passive loop filter is replaced by a digital one.

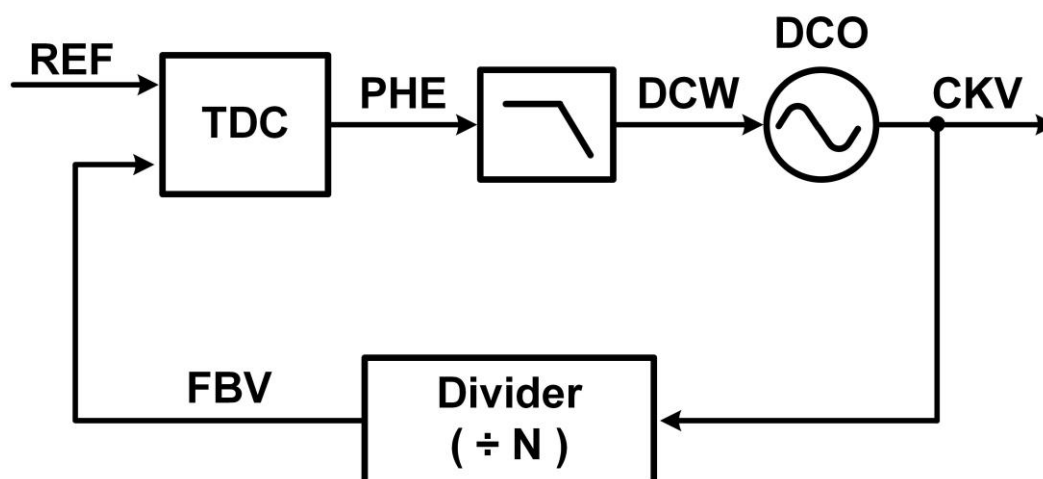


Fig. 2-2 Block diagram of a divider-based ADPLL

Another prominent ADPLL architecture is the counter-based ADPLL, depicted in Figure 2-3. The DCO phase, PHV, is obtained by the counter (representing the integer part) and TDC (representing the fractional part). The REF phase, PHR, is obtained by accumulating the FCW at the reference rate. Then, the PHR and the DCO phase, PHV, are compared using a simple subtractor to generate the phase error, PHE. The PHE inputs the DLF to generate DCW to adjust DCO frequency.

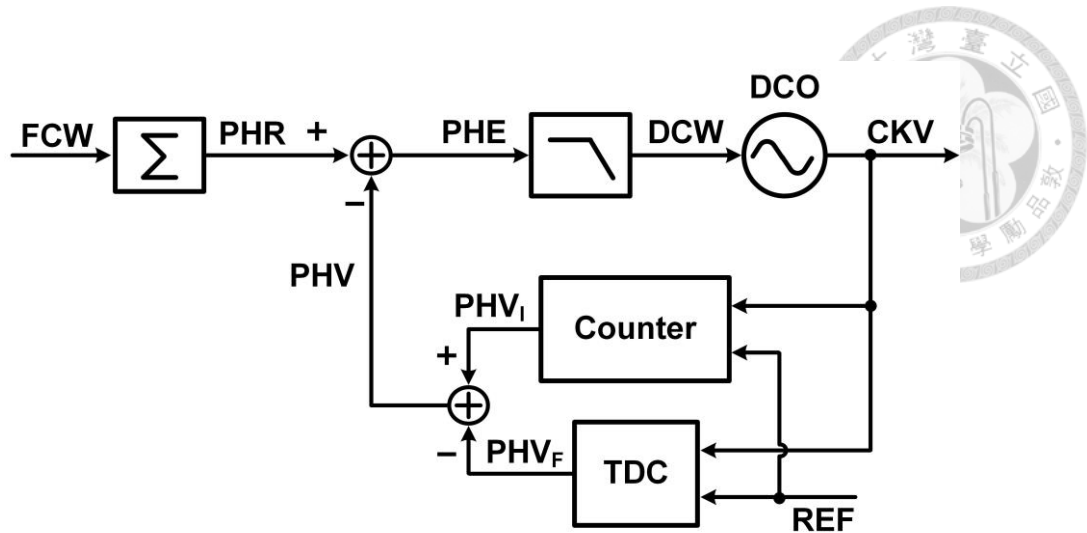


Fig. 2-3 Block diagram of a counter-based ADPLL

Digital solution promises many advantages over analog counterpart not only they removed the need for large capacitors. But also they are far superior to analog counterpart in terms of its programmability and stability to PVT variations. Nevertheless, the digital frequency synthesizer will face new issues such as finite resolution and quantization error. In the next section, Analysis of PLL quantization error will be illustrated.

2.2 Quantization Noise Analysis of Fractional-N ADPLL

The s-domain linear model [4] of the counter-based ADPLL is depicted in Figure 2-4. The reference phase ϕ_R is multiplied by the frequency division ratio N , as it is normalized to the DCO period for phase detection. The phase detector is simplified as a subtractor, followed by a multiplication factor of TDC conversion gain, to represent the digital phase error in radians of the DCO clock. The type-II loop filter consists of a proportional path and an integral path. The DCO phase ϕ_V represents the integration of the DCO frequency. Since phase detection is performed through edge counting, the open-loop transfer function does not involve N . In contrast to the divider-based architecture, where the feedback path is divided by N , the loop bandwidth remains independent of N . The open-loop transfer function is written as

$$\begin{aligned}
 H_{open}(s) &= \frac{T_{DCO}}{2\pi T_{TDC,res}} \times \left(\alpha + \frac{\beta}{1-z^{-1}} \right) \times \frac{2\pi K_{DCO}}{s} \\
 &\approx \frac{T_{DCO}}{2\pi T_{TDC,res}} \times \left(\alpha + \beta \times \frac{f_R}{s} \right) \times \frac{2\pi K_{DCO}}{s}
 \end{aligned} \tag{2.1}$$

where f_R denotes the reference frequency, $T_{TDC,res}$ denotes the resolution of TDC and T_{DCO} is the period of DCO.

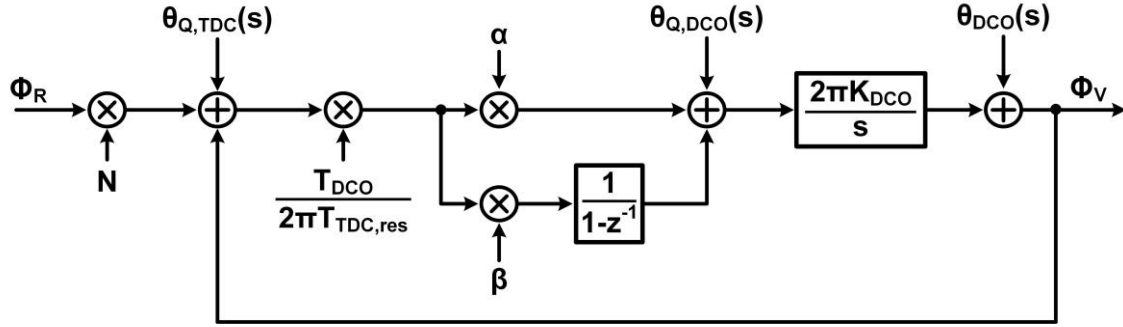


Fig. 2-4 s-domain model of a counter-based ADPLL

2.2.1 TDC Quantization Noise

Since the Time-to-Digital Converter (TDC) has a resolution limitation that prevents it from detecting phase errors smaller than the resolution, the TDC noise [5,7] at the input can be assumed to be uniform random variable between $[-0.5T_{TDC,res}, 0.5T_{TDC,res}]$ with a RMS of $\frac{T_{TDC,res}}{\sqrt{12}}$. When converted to phase domain. Phase noise is expressed as

$$\theta_{TDC}(s) = \frac{2\pi T_{TDC,res}}{T_{DCO}\sqrt{12}} \tag{2.1}$$

The finite resolution of the TDC introduces quantization noise that limits the in-band noise of the ADPLL. To enhance the performance of the ADPLL, a TDC with fine resolution is desired. High-resolution TDCs have been extensively studied for decades [5-6], including technologies such as double sampling and Vernier TDCs, among others. In general, the TDC detection range must encompass at least one period of the DCO for the application of a counter-based Fractional-N ADPLL. This ensures that the fractional input can be converted to a digital code without any loss.

2.2.2 DCO Quantization Noise

The DCO can only synthesize a discrete set of frequencies. In general case, the target frequency is never equal to anyone of these frequencies. The DCO frequency varies between two or more discrete frequencies and the average frequency over the long run is equal to the target one. Unfortunately, the resolution of DCO would affect the equivalent resolution of TDC. The frequency error due to finite resolution of DCO is Δf . And the accumulated delay after reference signal period, T_{REF} , is equal to

$$\Delta t \cong \frac{\Delta f}{f_{DCO}} T_{REF} \quad (2.3)$$

If Δt is greater than $T_{TDC, res}$, the accumulated delay introduced by the DCO exceeds the resolution of the TDC. As a result, certain bits of the TDC become non-functional, reducing the effective resolution of the TDC. To address this issue, the Delta-Sigma ($\Delta\Sigma$) noise shaping technique is introduced to increase the effective resolution of the DCO. As shown in Figure 2-5, N_f bits input modulator scale DCO resolution by 2^{-N_f} .

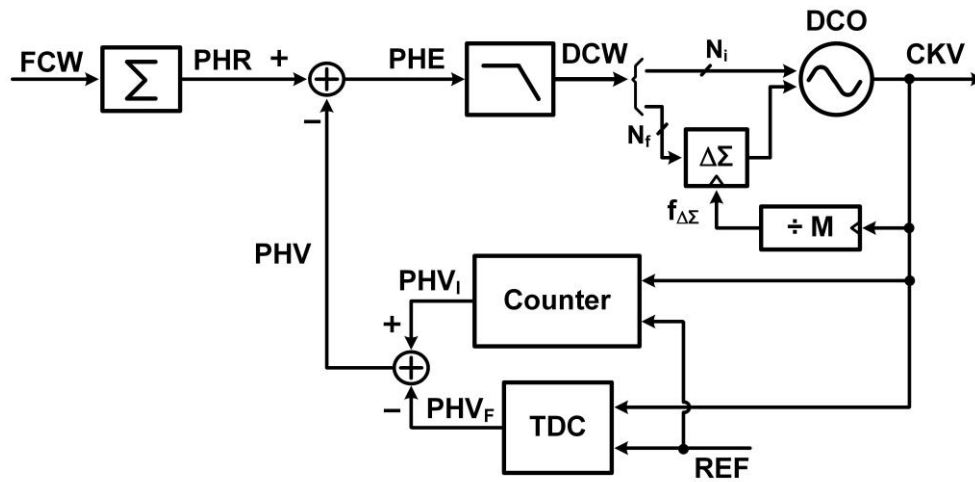


Fig. 2-5 Improving DCO resolution with $\Delta\Sigma$ noise shaping technique

2.3 Frequency Shift Keying (FSK) Modulation using Two Point Modulation (TPM) Technology

Figure 2-6 illustrates the ADPLL architecture for FSK modulation. A frequency modulation (FM) modulator can be implemented using a counter-based ADPLL by directly tuning the frequency command word, FCW. This method does not consume additional power since a PLL is already a necessary component in the transmitter for generating the local oscillation. However, the modulation signal, FCW, cannot be transferred losslessly to the ADPLL output due to the limitation of the PLL bandwidth. As shown in Fig. 2-6, the transfer function from the modulation signal to the ADPLL output is a low-pass filter. Therefore, modulation signals with frequencies higher than the bandwidth cannot be transferred losslessly to the output side.

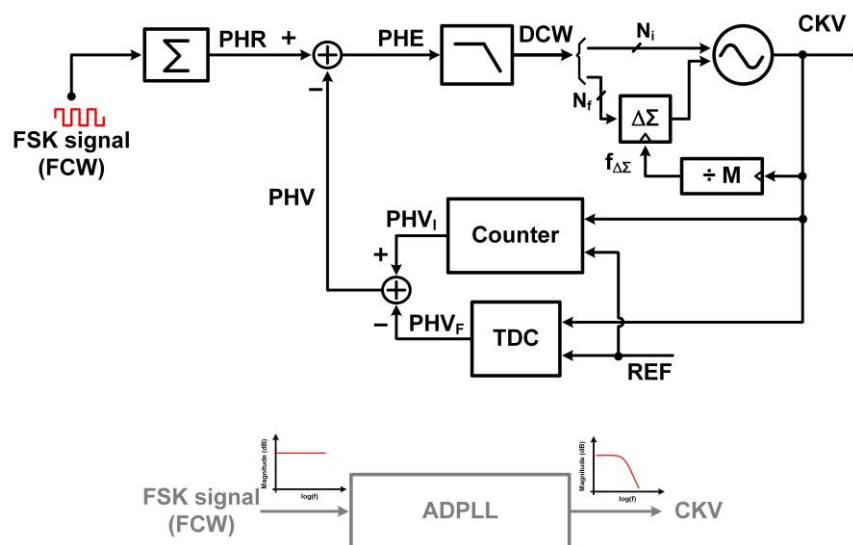


Fig. 2-6 Block diagram of the ADPLL with one-point modulation

To overcome the bandwidth limitation on the modulation signal, a two-point modulation (TPM) scheme was introduced in [8], as depicted in Figure 2-7. In this scheme, in addition to changing the FCW, the modulation signal is directly injected into the DCO. The second injection point provides a high-pass signal at the ADPLL output. When there

is no mismatch between the two injection points, the two injection signals combine to form an all-pass signal at the ADPLL output, overcoming the bandwidth limitation of the PLL. The overall system functions as a directly modulated open-loop DCO. However, due to factors such as MOS characteristics changing with process, voltage, and temperature (PVT) variation, proportional delay in the circuit, settling time of the DCO, limitations in the manufacturing process, and more, there can be gain and injection timing mismatches between the two injection points. These mismatches contribute to a poor bit error rate (BER) in communication. The impact of various non-ideal effects on the circuit will be explained in the following section.

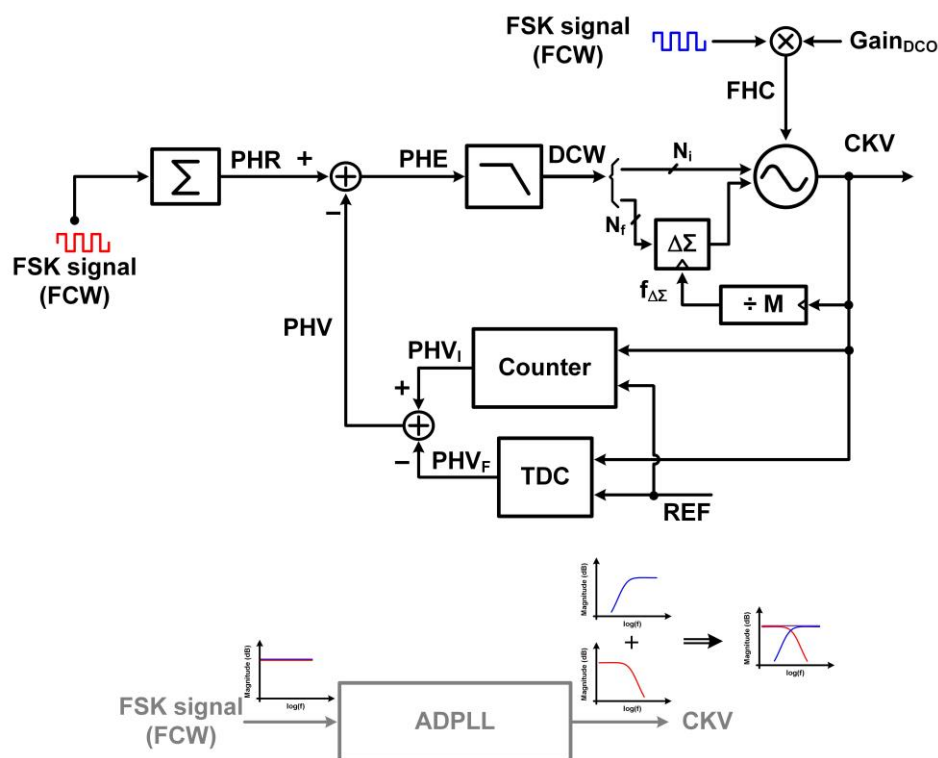


Fig. 2-7 Block diagram of the ADPLL with two-point modulation

2.3.1 DCO Gain Variation for TPM architecture

ADPLL provides a convenient method to adjust the frequency of the DCO by changing the DCO control code. However, it still faces the challenge of DCO gain variation caused by PVT variation. The waveform of frequency hopping operation with

DCO gain error is illustrated in Figure 2-8. For the convenience of discussion, let's consider the topology of PHR as a continuous-time signal, while PHV is illustrated by the equivalent signal CKV. As shown in Figure 2-8, when the target frequency transitions from f_1 to a higher frequency f_2 , any variation in the actual DCO gain compared to the ideal gain due to PVT variation would impact the accurate modulation of the DCO frequency. Specifically, if the actual DCO gain is higher (or lower) than the ideal gain, the high-pass signal injection cannot precisely modulate the DCO frequency, resulting in a DCO frequency that is higher (or lower) than the intended value. Consequently, the DCO frequency still needs to be gradually adjusted to the target frequency through negative feedback. The integration of phase error with frequency error, as illustrated in Figure 2-8, can cause misalignment between PHV and PHR. In summary, DCO gain variation introduces frequency errors within the ADPLL system.

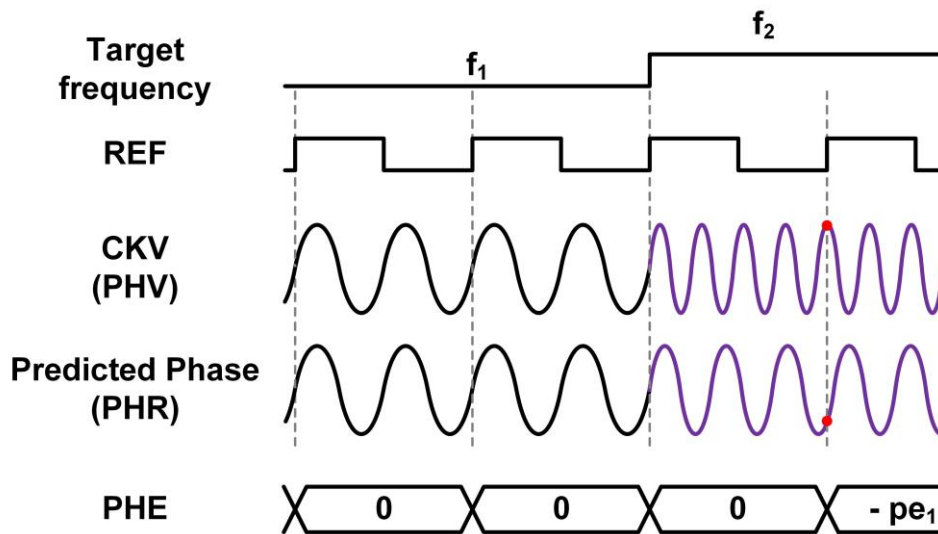


Fig. 2-8 Example of TPM with DCO gain variation

2.3.2 Adaptive Gain Compensation of DCO

The technique of adaptive gain compensation for the DCO is introduced in [8]. This approach involves a digital calibration block that dynamically corrects the multiplexer factor, Gain_{DCO} , to compensate for DCO gain changes resulting from PVT variation. The

simplified architecture is shown in Figure 2-9. The value obtained by multiplying the phase error, PHE, and the ΔFCW extracted by the differentiator is accumulated by an integrator. Additionally, to ensure that the PHE is generated by frequency hopping operation, a two-reference period delay is added after the differentiator. One is because the phase error takes time to accumulate. The other is to eliminate the influence of path delay spread. After the calibration is completed, ΔFCW becomes uncorrelated with PHE. As a result, frequency hopping does not introduce additional phase error to the PLL.

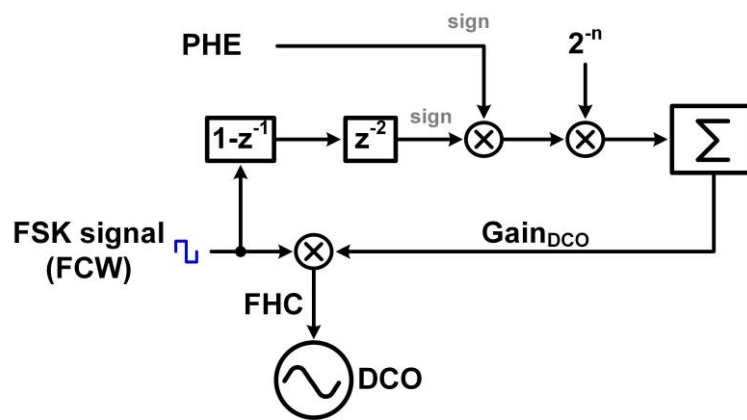


Fig. 2-9 Adaptive gain compensation of DCO gain variation

2.3.3 Path Delay Spread for TPM architecture

The TPM architecture is sensitive to the delay spread between the two injection paths. Due to the settling time of the DCO and the proportional delay of the TPM control logic, the high-pass injection signal is typically delayed compared to the low-pass signal. As a result, it introduces unwanted phase error to the FSK modulation system. The waveform depicting the path delay spread is shown in Figure 2-10. The timing of DCO frequency alteration is Δt later than the rising edge of REF. Within this time interval Δt , the actual frequency deviates from the target frequency, inducing a phase error that will be measured at the next rising edge of REF. In summary, the path delay spread only contributes to phase error, not frequency error, within the ADPLL.

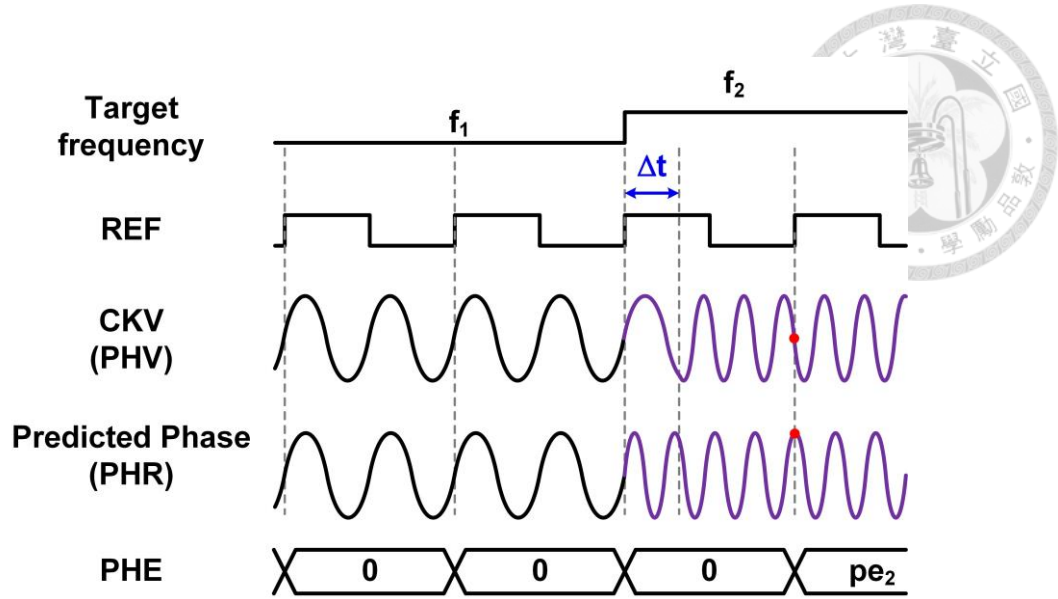


Fig. 2-10 Example of TPM with injection timing mismatch between two injection points

2.3.4 Proposed Delay Spread compensation to ADPLL

The conventional delay spread compensation architecture [8] consists of a DTC and a calibration block, as shown in Figure 2-11. As discussed earlier, the phase error contributed by the delay spread, PE_D , is equal to the product of the frequency hopping distance f_D and the time difference Δt .

$$PE_D \cong \frac{2\pi f_D \Delta t}{f_{DCO}} \quad (2.4)$$

The value of the time difference can be estimated through the calibration loop. Consequently, the phase error resulting from delay spread can be predicted and then compensated for by the DTC. However, this method is subject to long settling time and requires additional stability considerations for the calibration loop.

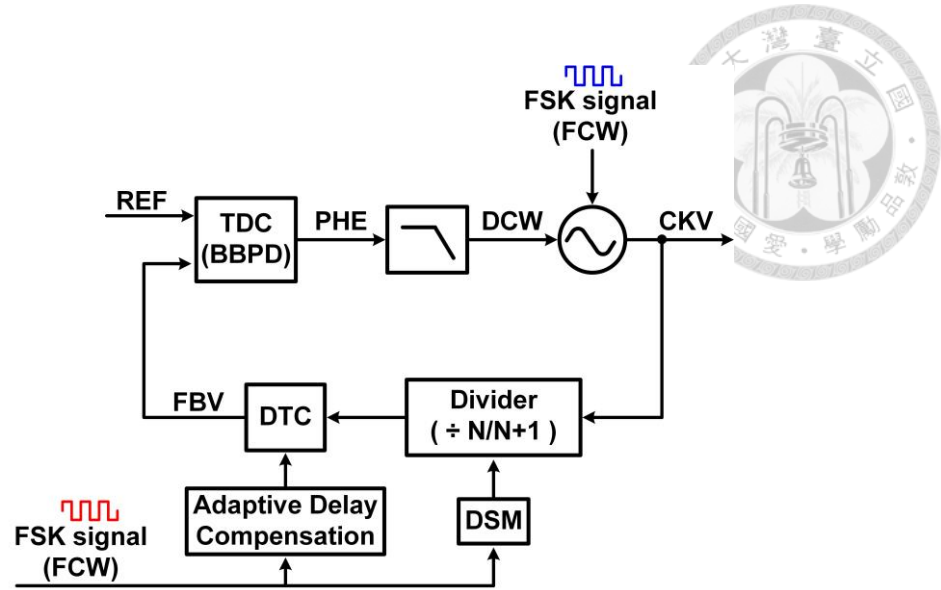


Fig. 2-11 Block diagram of conventional delay spread compensation

In this work, we propose a calibration-free scheme to compensate for the delay using forward cancellation technology. The proposed ADPLL scheme, as depicted in Figure 2-12, reuses the FCW accumulator to record the phase error caused by delay spread. Additional switches are employed to ensure that the phase error is not transferred to the DLF at the next rising edge of the REF signal after frequency hopping. At the next REF rising edge, the ADPLL interrupts the feedback path and replaces the PHR code with the PHV code. Consequently, the phase error information is recorded in the FCW accumulator, where it becomes a constant phase error within the ADPLL system and is subsequently ignored. The example waveform of forward cancellation technology is shown in Figure 2-13, where SWR represents the signal used to control the switches. If FCW changing is detected, SWR is set to HIGH. Simultaneously, the DLF input is set to 0, and PHR is replaced with PHV.

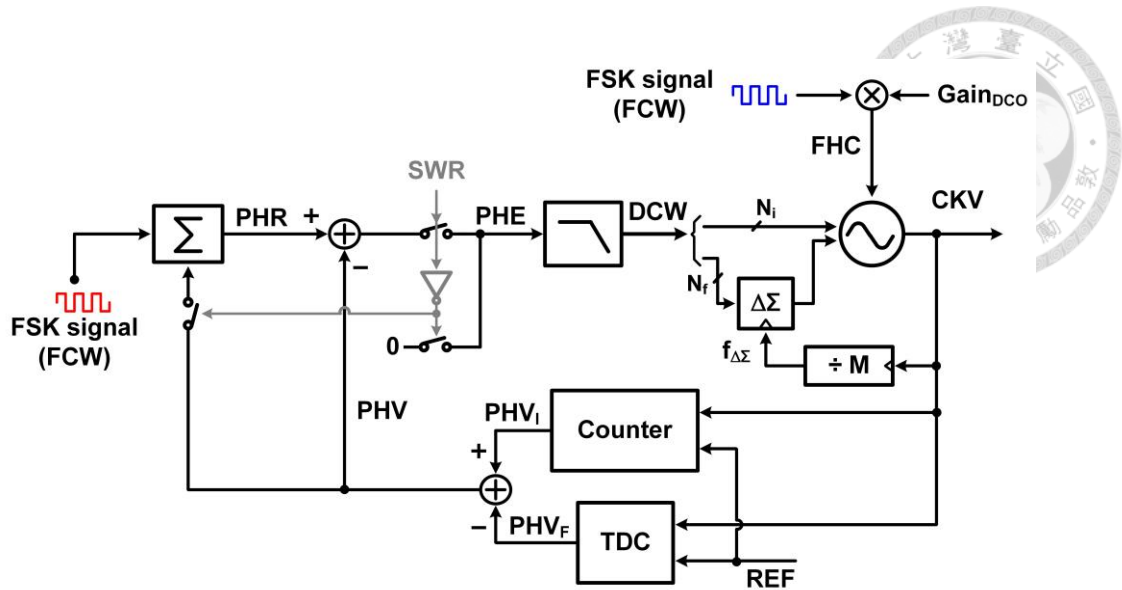


Fig. 2-12 Block Diagram of proposed delay spread compensation

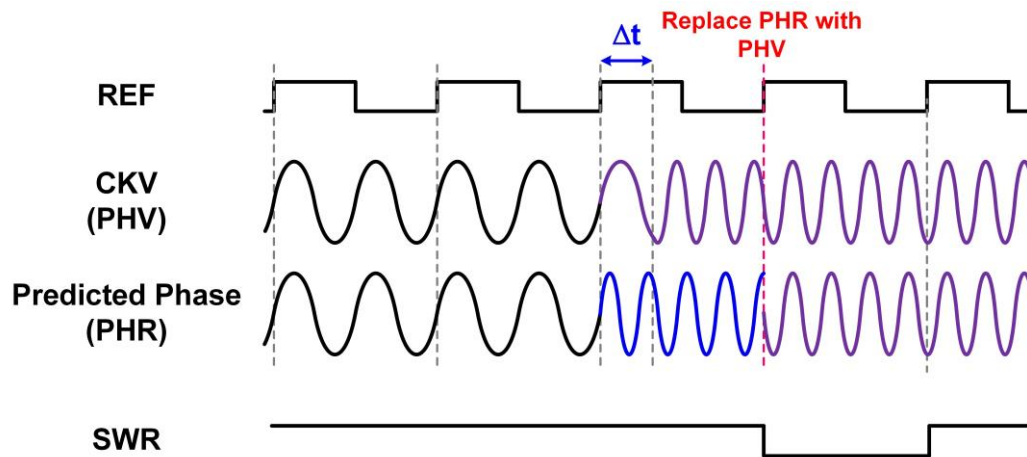


Fig. 2-13 Example waveform of forward cancellation technology

This method can cancel the phase error caused by delay spread without the need for a calibration circuit, requiring only the sacrifice of the information of one cycle. Moreover, this method relies on a precise TDC that is already present in the counter-based ADPLL. All the additional blocks are digital circuits that can be implemented using EDA tools. Therefore, the forward cancellation technology for delay spread is easy to implement and has minimal power consumption and chip area requirements.

Chapter 3 TDC and DTC Linearization Technology



3.1 TDC Conversion Gain Variation during Frequency Hopping

The simplified architecture of TDC in Figure 3-1(a) consists of a delay chain and number of arbiters. The measured time difference is represented by STOP and START signals (i.e., by the delay between their rising edges). As the START signal propagate through delay chains, the time difference between START and STOP is decreased in each stage by $T_{TDC,res}$, the delay time of delay element. In each stage, START and STOP signals are fed into an arbiter (D-Flip-Flop) that decides which of the two input signals came first. The measured time difference is encoded as a thermometer code at the register output.

The phase difference is converted to digital domain by TDC. The linear model of a TDC is shown in Fig. 3-1(b), which the transfer function is

$$K_{TDC} = \frac{T_{DCO}}{2\pi T_{TDC,res}} \quad (3.1)$$

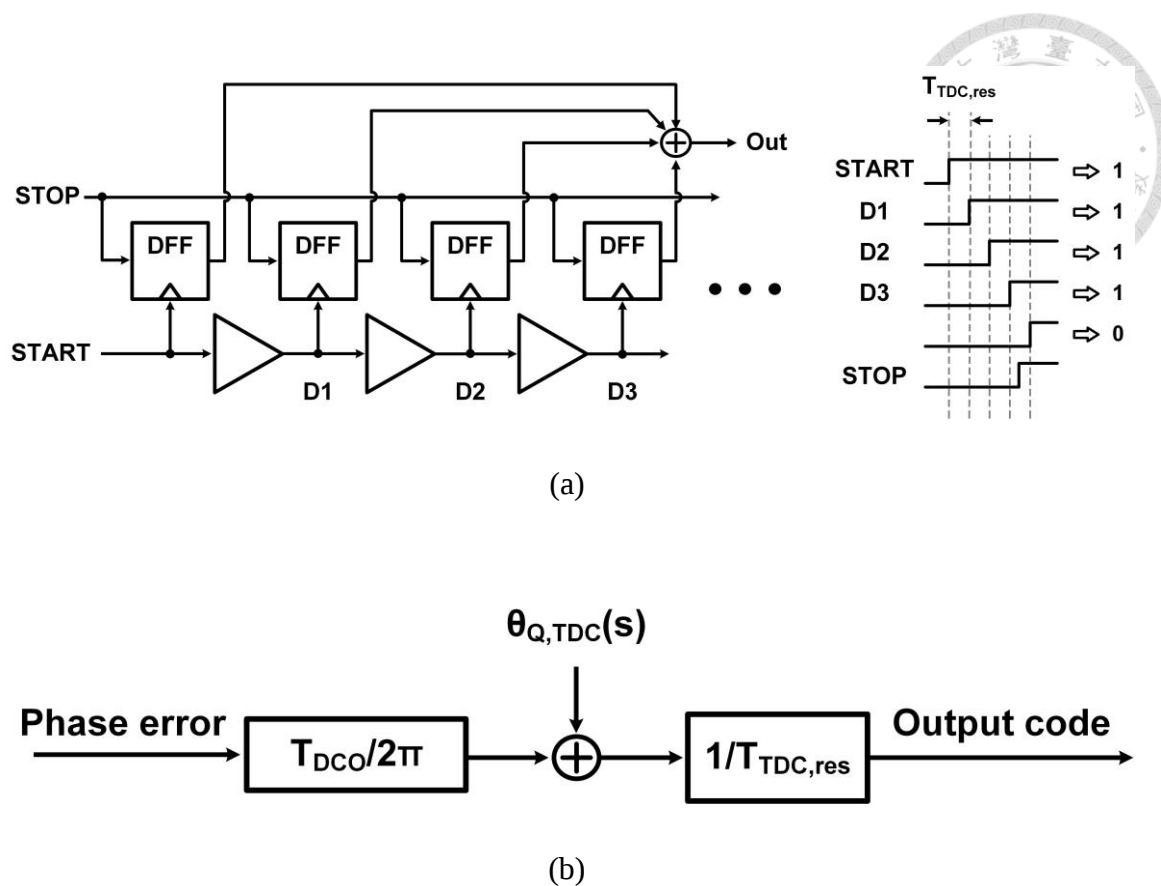


Fig. 3-1 Analysis of a classical delay-chain TDC in (a) the conceptual brief and (b) the linear model

The TDC conversion gain needs to be highly accurate for the Counter-based ADPLL, as illustrated in Figure 3-2. The figure shows two waveforms representing different DCO frequencies. One waveform has a frequency 2.4 times that of the reference frequency, while the other waveform has a frequency 3.4 times that of the reference frequency. Although the fractional phase offset is same with each other: 0.4, 0.8, 0.2, 0.6, 0.0, the absolute time intervals are different. The time interval is narrower in the higher DCO frequency. Hence, it is impossible to convert the same fractional phase of different frequencies to same digital code by using the TDC in Fig. 3-1.

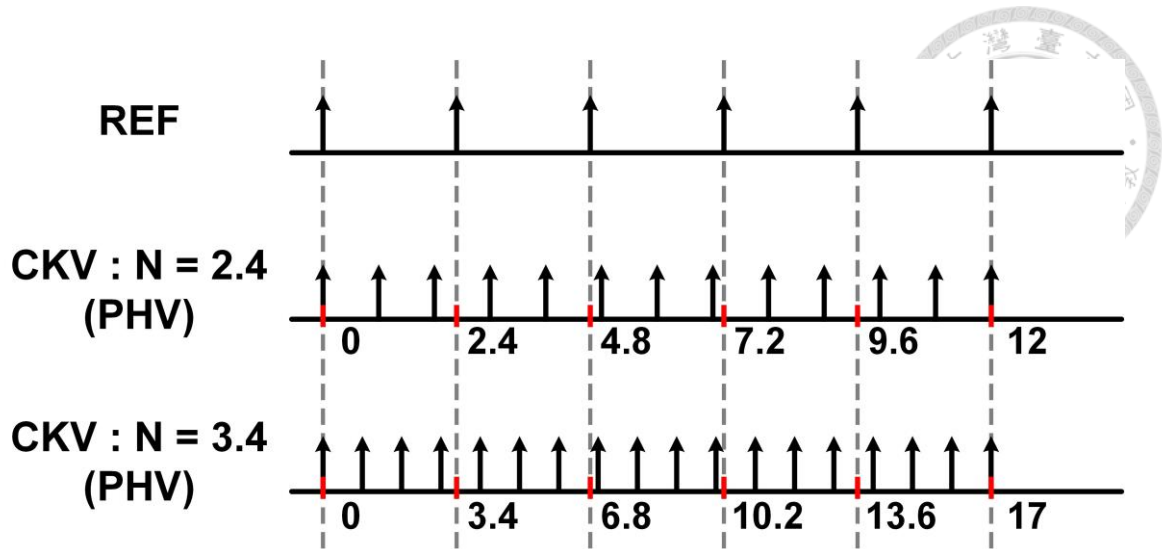


Fig. 3-2 Time Diagram with a frequency 2.4 and 3.4 times that of the reference frequency

In general PLL applications, the TDC gain can be adjusted in either the analog or digital domain through a calibration loop [9-10] when the output frequency is fixed. However, the calibration loop may fail due to its long settling time in FSK application. The allowable settling time should be shorter than the modulation period, which is impossible because the bandwidth of the calibration loop (several kilohertz) is typically smaller than the modulation period (ranging from DC to hundreds of kilohertz) under normal circumstances. To overcome the TDC gain variation issue, the prior art of output frequency insensitive TDC [11] will be introduced in the next section.

3.2 Dual-Referenced Interpolating TDC

The Dual-Referenced Interpolating TDC (DI-TDC) [11] architecture is shown in Figure 3-2. The DI-TDC consists four sub-TDC, a quadrant detector and an adder logic. The ring-DCO multi-phase ($\Phi_0 \sim \Phi_3$: 0, 0.5, 1, 1.5π) outputs are fed into their respective sub-TDCs. The input trigger signals of the sub-TDC, $\text{frd}\langle 7:0 \rangle$, are generated by a delay chain where each element has a delay of approximately $T_{\text{DCO}}/32$. The sub-TDC converts the time difference between the rising edge of REF and the rising edge of corresponding

DCO phase into code. Then, the adder sums the two output codes from the sub-TDCs based on the quadrant of the fractional phase. This yields the complete TDC output code. The quadrant detection can be determined by sampling the four phases using the midpoint of the TDC trigger signal, $frd<4>$. Take the waveform in Figure 3-4 as an example, the sampled signal of $\Phi 0$ is 1, and the sampled signal of $\Phi 1$ is 0. Hence, the fractional phase is in first quadrant, and the selector selects $Q0$ and $Q1$ as the inputs for the adder.

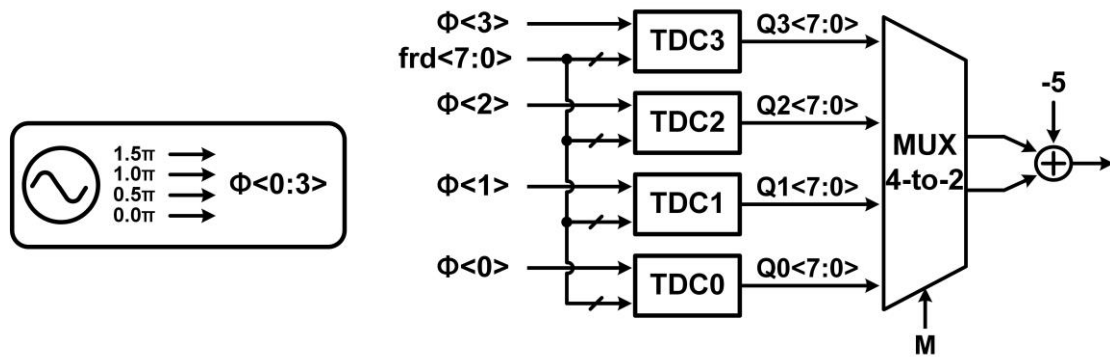


Fig. 3-3 Block Diagram of the dual-referenced interpolating TDC (DI-TDC)

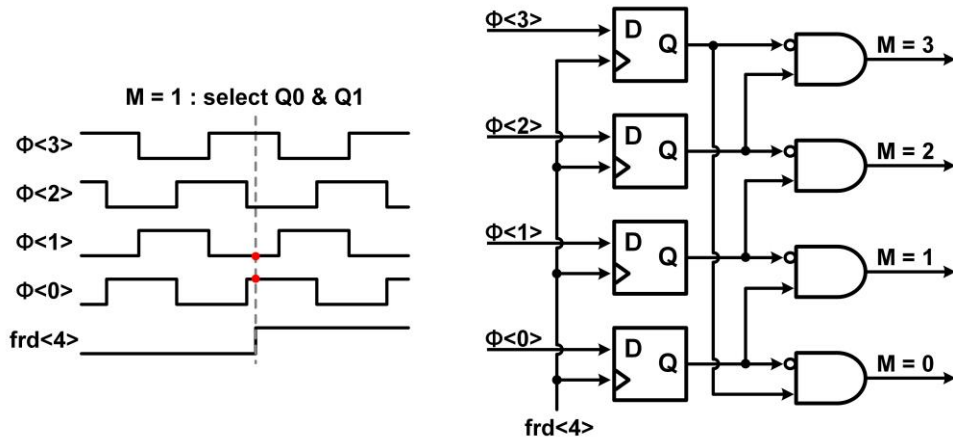


Fig. 3-4 Block diagram and time diagram of quadrant detector

Figure 3-5(a) illustrates a case where a TDC with a conversion error leads to unwanted phase error in the DLF of the ADPLL. The use of conventional TDC with sub-TDCs can help reduce this unwanted conversion error. This is because the detection range of each sub-TDC is reduced to one-quarter of the original range. Similarly, the integral

conversion error is also reduced to one-quarter of its original value. According to Fig. 3-5(a), the gain error still results in a discontinuous phase to conversion error curve. Near the boundary of the sub-TDC detection range, a large phase error is injected into the DLF, which can affect the stability of the PLL and lead to poor fractional spurs. To further reduce the integral conversion error of the TDC and eliminate the discontinuous points, the final TDC output code is generated by using two adjacent sub-TDC outputs. By summing the two TDC outputs, the conversion error is significantly reduced, and the discontinuous points are eliminated. This is because most of the conversion error is canceled out by errors of opposite polarities, as shown in Fig. 3-5(b).

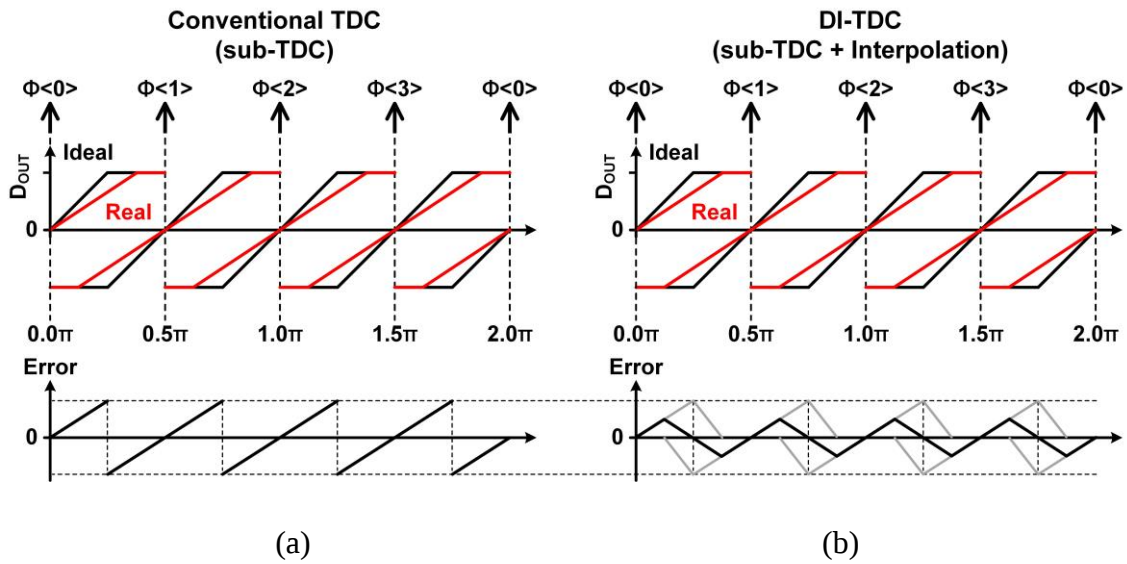


Fig. 3-5 (a) Conversion error by delay change with the conventional TDC (b) Concept of error compensation by the DI-TDC

The detailed operation, along with a time diagram, is described in Figure 3-6. Fig. 3-6(a) illustrates the case where $T_{\text{TDC, res}}$ is greater than $T_{\text{DCO}}/12$. In this scenario, the transfer function consists of two line segments with different slopes. One slope is larger than the ideal TDC gain, while the other is smaller. By combining these two line segments, the average gain is equal to the ideal value. On the other hand, Figure 3-6(b) depicts the case where $T_{\text{TDC, res}}$ is less than $T_{\text{DCO}}/12$. Similarly, by combining a line segment with a

slope of zero and another line segment with a slope larger than the ideal value, the average gain is equal to the ideal value.

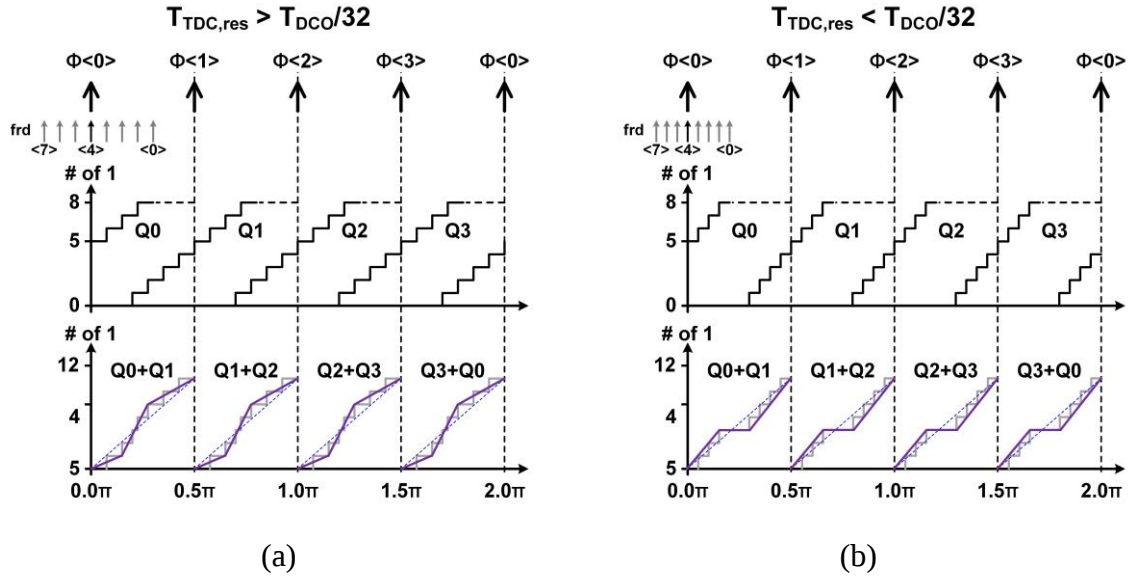
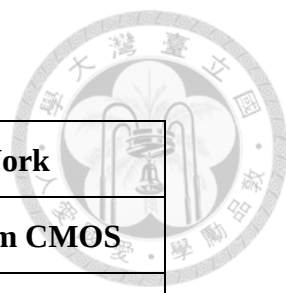


Fig. 3-6 Time diagram of DI-TDC with (a) $T_{TDC,res} > T_{DCO}/12$ (b) $T_{TDC,res} < T_{DCO}/12$

The DI-TDC offers excellent linearity and high immunity to frequency and PVT variations. Although there is conversion error in each sub-TDC, it is guaranteed to have no conversion error when the time difference is equal to integer multiples of one-eighth of T_{DCO} . Moreover, the average conversion error is zero without the need for any calibration scheme. This makes the DI-TDC an excellent solution for FSK modulation systems. However, the nonlinearity of the DI-TDC is still not sufficiently small to suppress the fractional spur of the ADPLL output to meet the specifications outlined in Table 3-1. To address this, we propose the quadruple-referenced interpolating TDC (QI-TDC), which will be described in the next section.

Table 3.1 System specification



	This Work
Technology	TSMC 90-nm CMOS
Supply Voltage	1.2 V
Reference Frequency	40 MHz
Target Frequency	2.35~2.45 GHz
Phase noise @ 1MHz (dBc)	< -95
Fractional spur (dBc)	< -45
Power (mW)	< 10
Core area (mm²)	< 0.2

3.3 Quadruple-Referenced Interpolating TDC

The architecture of the Quadruple-Referenced Interpolating TDC (QI-TDC) is shown in Figure 3-7. In comparison to the conventional DI-TDC, the number of sub-TDCs has been expanded from four to eight. However, the detection range of each sub-TDC remains approximately $T_{DCO}/4$, which is the same as the DI-TDC. The outputs of all the sub-TDCs are utilized to generate the final TDC output simultaneously, whereas the DI-TDC only utilizes two of them. Based on the quadrant detection, the outputs of four sub-TDCs are summed with their complements from the other four sub-TDCs that correspond to the opposite quadrants.

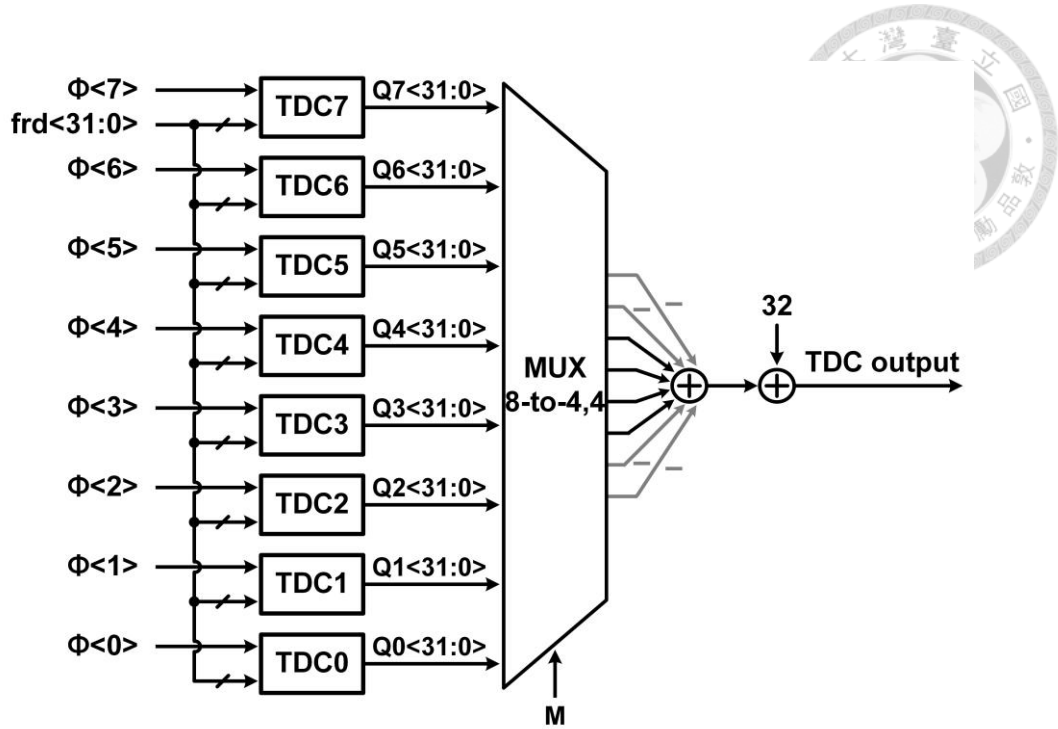


Fig. 3-7 Block diagram of Quadruple-Referenced Interpolating TDC (QI-TDC)

To briefly explain the principle of QI-TDC, we first ignore the four complement outputs of the sub-TDCs. Figure 3-8 illustrates the case of a TDC with a conversion error. Since the default detection range of the sub-TDC in QI-TDC is still the same as in DI-TDC, the phase-to-error curve of a sub-TDC remains the same as in DI-TDC. As mentioned in the previous section, it is guaranteed to have no conversion error when the time difference is equal to integer multiples of one-eighth of T_{DCO} . As depicted in Figure 3-8(a), the phase-to-error curve exhibits characteristics of a quarter T_{DCO} period, an average of 0, and a duty cycle of 50%. By shifting the phase-to-error curve with one-eighth of T_{DCO} , most of the conversion errors are canceled out by errors of opposite polarities, as shown in Figure 3-8(b). Consequently, the linearity of QI-TDC is improved compared to DI-TDC through double interpolation. The simulation result for QI-TDC and DI-TDC is displayed in Figure 3-9.

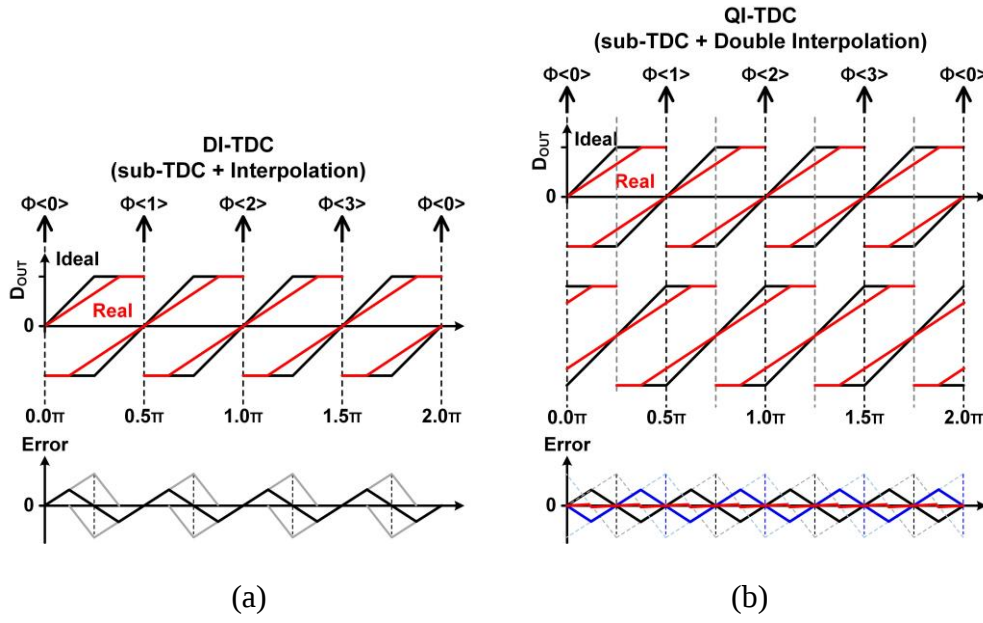


Fig. 3-8 Comparison of time diagram of (a) DI-TDC and (b) QI-TDC

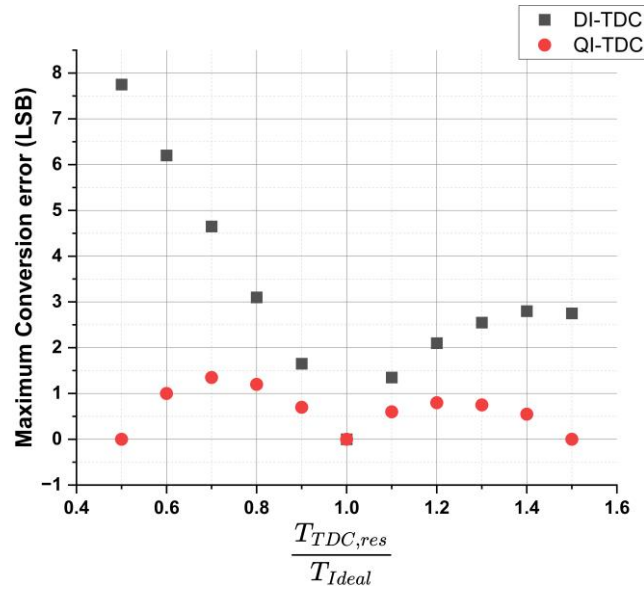


Fig. 3-9 Simulation result for QI-TDC and DI-TDC

The utilization of the complement outputs from the four sub-TDCs enhances the resolution of the TDC by a factor of two. In the digital domain, by summing the output codes of the eight sub-TDCs, the output of the QI-TDC is increased by two bits. Hence, with the assistance of the two pseudo bits, the QI-TDC achieves a four-fold improvement

in resolution. However, in order to balance linearity and resolution, one bit is truncated to improve linearity. As a result, the proposed QI-TDC achieves better linearity and finer resolution compared to the DI-TDC, while still retaining the advantages of DI-TDC in terms of immunity to PVT and frequency variation.

3.4 Noise Dithering using Inverse DTC

The mechanism of fractional spurs is shown in Figure 3-10. When the TDC input is periodic, the non-linearity of the TDC introduces a corresponding periodicity in the measured phase error. This periodicity in the phase error modulates the frequency of the DCO, resulting in a FSK modulation-like behavior. Consequently, the target frequency undergoes changes with a specific period. This interaction between the periodic phase error and the DCO frequency gives rise to multiple fractional spurs.

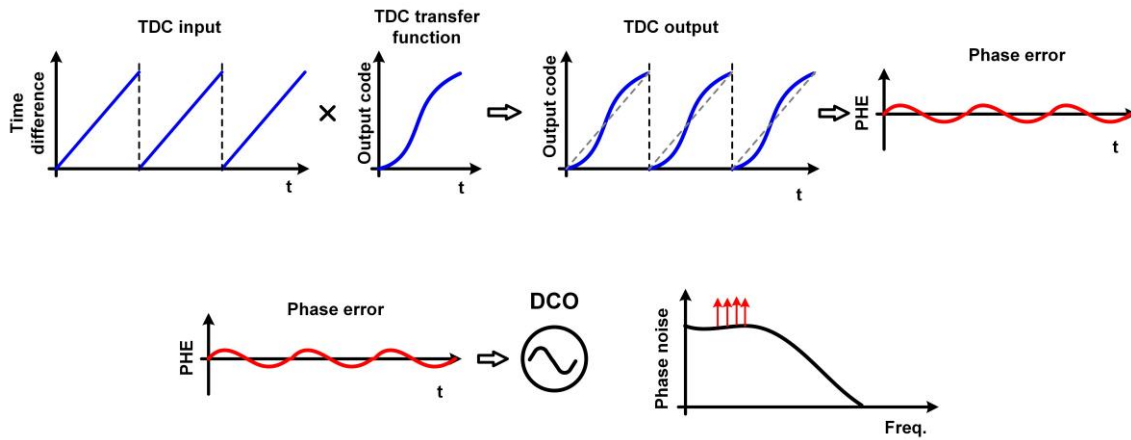


Fig. 3-10 Performance of ADPLL with TDC nonlinearity

Therefore, to further reduce fractional spurs in the ADPLL output, noise dithering is an effective solution. By introducing dithering [12] in the input reference path, the periodic fractional phase offset pattern is randomized, resulting in a reduction of the magnitude of fractional spurs. However, the introduction of additional noise input will increase the phase noise floor. To mitigate the negative impact of dithering on the noise

floor, a background-dither-noise-cancellation loop is employed, as shown in Figure 3-11. This loop cancels out the dithering signal before it is injected into the DLF, ensuring that the additional noise does not affect the phase noise floor. The phase error contributed by noise dithering and cancellation loop is

$$\phi_{err} = K_{TDC} \times K_{DTC} \times D_{dit} - D_{dit} \times K_{dit} \quad (3.2)$$

for perfect cancellation, the multiplication factor, K_{dit} , should be equal to the multiplication of TDC's gain, K_{TDC} , and DTC's gain, K_{DTC} . With the help with QI-TDC, the phase-to-code conversion gain of the TDC is independent of the ADPLL output frequency. However, the conventional DTCs [12, 14] still suffer from gain variation during frequency hopping since the conversion between time domain and phase domain needs to be divided by the value of T_{DCO} as stated in the equation

$$P_{diff} = \frac{2\pi T_{diff}}{T_{DCO}} \quad (3.3)$$

where P_{diff} is the phase difference whose unit is rad. and T_{diff} is the time difference (sec.)

To tackle this issue, we propose an inverse DTC which its gain is independent of ADPLL output frequency so that the dithering signal can be perfectly cancelled by cancellation loop.

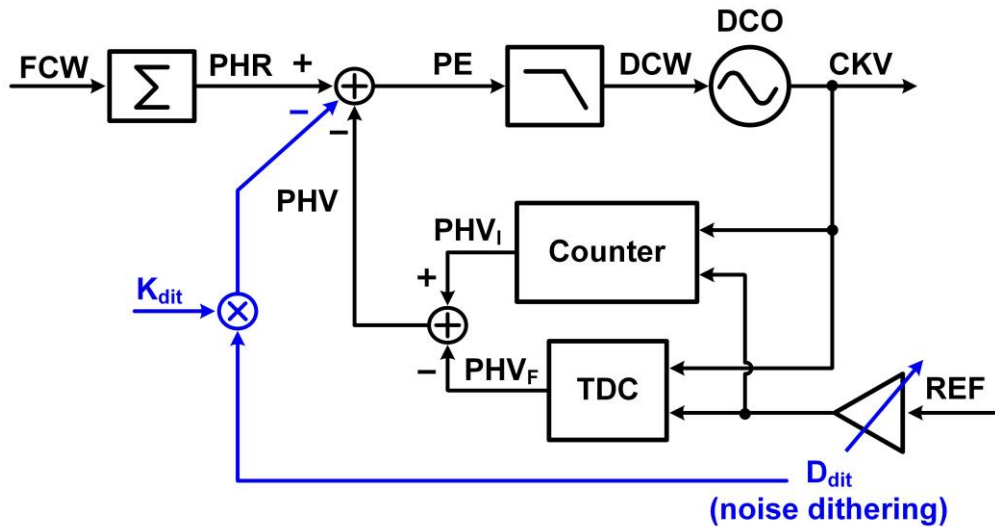


Fig. 3-11 Block diagram of noise dithering and dither-noise-cancellation loop

3.4.1 DTC Based on Time-Mode Arithmetic Units

The inverse DTC is an improved circuit based on the prior art [13], which is shown in Figure 3-12(a). It utilizes time-mode arithmetic units to overcome gain variation with frequency variation. The capacitor is initially pre-charged to V_{DD} . Then, SWD is switched to HIGH, turning on the switch to provide a current, I_1 , for pre-discharging the capacitor for one T_{DCO} using the tunable resistor array. Next, REF triggers the second discharge operation with discharge current, I_2 , and the comparator output changes to HIGH when the voltage of the capacitor crosses the threshold voltage, V_{TH} . The detailed time diagram is displayed in Fig. 3-12(b). Thanks to the first discharge operation, the output delay of the DTC can be interpreted as a scaling of T_{DCO} , where the scaling factor is the ratio between the discharge current during the first and second discharge operations, I_1 and I_2 . The equation of the DTC output delay is

$$Out_{DTC} = \frac{I_2}{I_1} T_{DCO} + T_{offset,DTC} \quad (3.4)$$

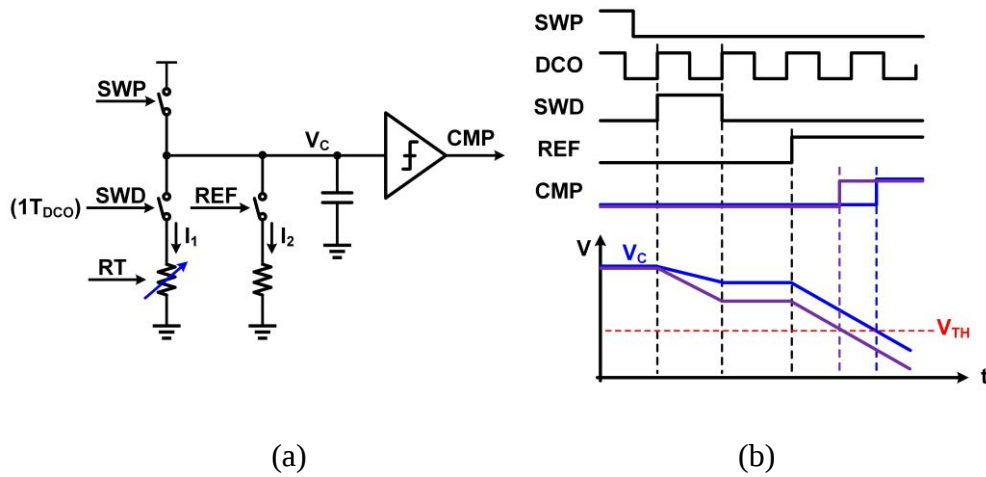


Fig. 3-12 (a) Block diagram of the prior art of DTC based on time-mode-arithmetic units and (b) its time diagram

The constant delay offset, $T_{offset,DTC}$, is ignored in the ADPLL, so the output delay will change with the frequency, which is suitable for ADPLL for FSK application. However, the DCO-referenced DTC may degrade into the conventional DTC due to the

non-ideal switch effect. This effect is depicted in Figure 3-13. The ON time of the switch is sensitive to the MOS threshold voltage, which means that the pre-discharge time may not be equal to one T_{DCO}. As the DCO period decreases, the non-ideal switch effect becomes more significant because at high ADPLL output frequencies, the waveform of SWD is closer to a triangle than a square.

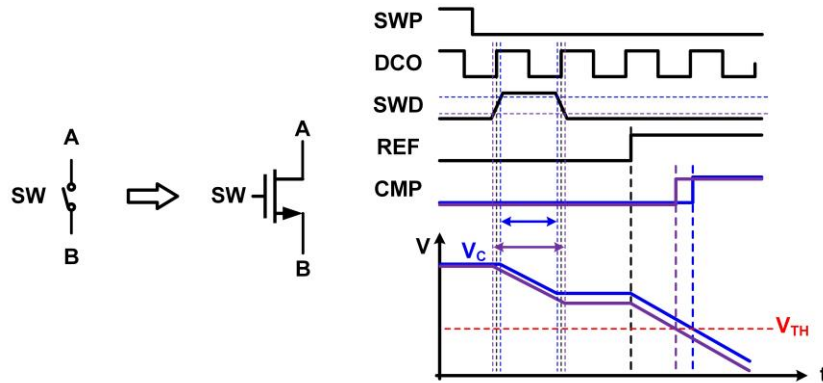


Fig. 3-13 Time diagram with non-ideal switch effect

3.4.2 Inverse DTC Based on a Pulse Generator

To address this issue, the proposed architecture is shown in Figure 3-14. By controlling the pre-discharge time to generate different output delays, the time diagram in Figure 3-12 illustrates the operation. Initially, the capacitor is pre-charged to VDD. Then, SWD is switched to HIGH, turning on the switch to provide a current, I_S , for pre-discharging the capacitor for $N T_{DCO}$. Next, REF triggers the second discharge operation with a discharge current, MI_S , and the comparator output transitions to HIGH when the capacitive voltage crosses V_{TH} . The resolution of the pre-discharge time is $T_{DCO}/8$, with the assistance of the multiple outputs of the ring DCO. Although the pre-discharge times may not be exact integer multiples of $T_{DCO}/8$ due to non-ideal switch effect, the difference between any two discharge times corresponding to different codes is an accurate integer multiple of $T_{DCO}/8$, as shown in Figure 3-15. In other words, the output delay caused by

the non-ideal switch effect can be treated as a constant offset that can be disregarded in PLLs. The equation of the DTC output delay is

$$Out_{DTC} = \frac{NT_{DCO}}{M} + T_{offset,DTC} \quad (3.5)$$

The delay output only depends on the length of pre-discharge time, NT_{DCO} , and the current ratio of two discharges.

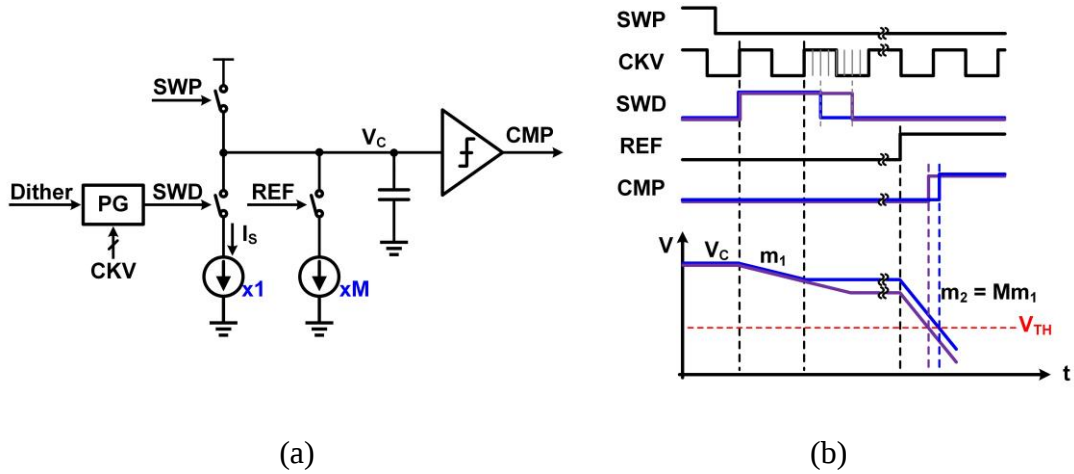


Fig. 3-14 Block diagram of proposed DTC based and (b) its time diagram

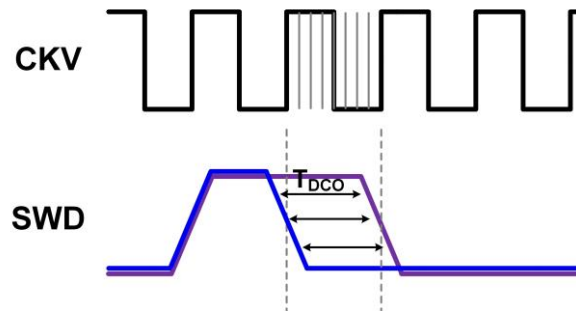


Fig. 3-15 Difference between any two SWD ON-time corresponding to different codes is an accurate integer multiple of $T_{DCO}/8$

3.4.3 Immunity to Channel-Length Modulation

The operation of the inverse DTC can be divided into two steps: (1) Sampling the DCO period and storing it as a capacitive voltage drop. (2) Outputting the stored data in a time-delayed form. The first step can be seen as a time-to-voltage conversion, while the

second step is an inverse conversion from time to voltage. Therefore, the inverse DTC can be considered as a combination of a time-to-voltage converter (TVC) and a voltage-to-time converter (VTC). The detailed derivation of this operation is explained in [16].

The transfer function of VTC is

$$dt = \frac{C(V)}{I_S(V)} dV \rightarrow T_{pch} = \int_{V_{pch}}^{V_{DD}} \frac{C(V)}{I_S(V)} dV \quad (3.6)$$

where V_{pch} is the capacitive voltage after sampling and V_{pch} is the pre-discharge time.

And the transfer function of TVC is expressed as

$$dV = \frac{C(V)}{M \cdot I_S(V)} dt \rightarrow T_{DTC} = \int_{V_{TH}}^{V_{pch}} \frac{C(V)}{M \cdot I_S(V)} dV = \frac{1}{M} \int_{V_{TH}}^{V_{pch}} \frac{C(V)}{I_S(V)} dV \quad (3.7)$$

where T_{DTC} is the time difference between REF and CMP. If only consider channel length effect, the current ratio remains fixed. Hence the ratio M can be moved outside the integral.

Thus from Eq.(3.8) and Eq.(3.7),

$$M \cdot T_{DTC} + T_{pch} = \int_{V_{TH}}^{V_{DD}} \frac{C(V)}{I_S(V)} dV \quad (3.8)$$

thus from Eq.(3.8),

$$T_{DTC} = \int_{V_{TH}}^{V_{DD}} \frac{C(V)}{I_S(V)} dV - \frac{T_{pch}}{M} \quad (3.9)$$

The valid DTC output delay is proportional to pre-changing time and the linearity of the inverse DTC is immune to channel length modulation. Hence, it relaxes the DTC's current source linearity requirement and design limit of DTC's current source. The detailed circuit implementation would be described in the next section.



Chapter 4 Implementation of a 2.4 GHz Fractional-N All-Digital PLL



In this chapter, we present the circuit designs of the proposed fractional-N ADPLL, which include the counter, the TDC, the DTC, the DCO, and the RTL blocks. The overall architecture of the ADPLL is illustrated in Figure 4-1. It can be divided into four main parts: the TPM architecture, the fast-locking logic, the DTC dithering, and the basic ADPLL, which comprises the counter, TDC, DCO, DLF, and other components. The custom circuits are explained in the first half of this chapter, while the RTL blocks are discussed in the second half.

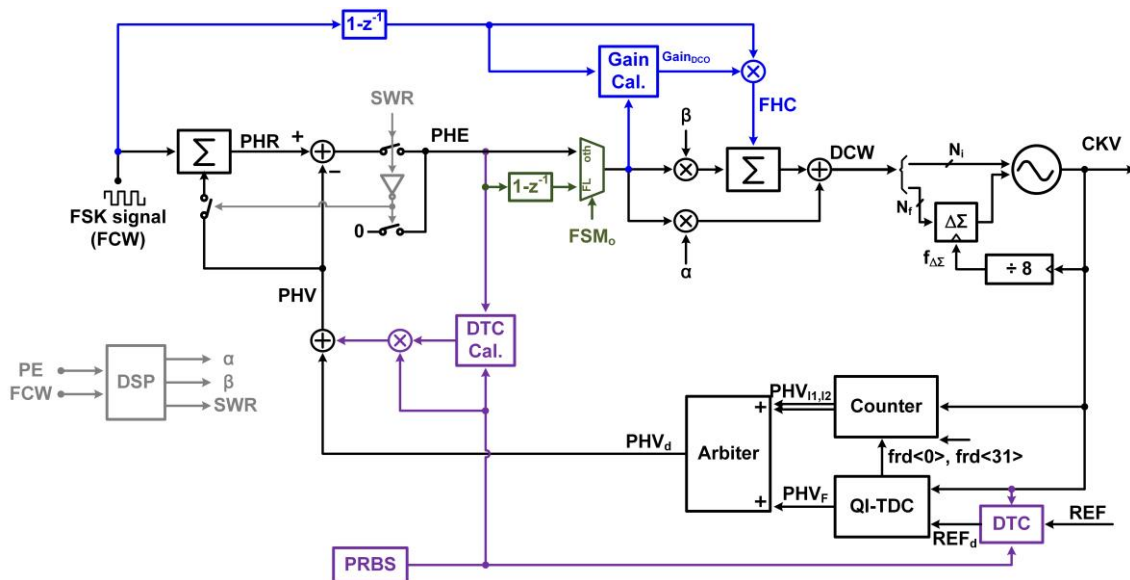


Fig. 4-1 Block diagram of proposed ADPLL for FSK application



4.1 Analog Design

4.1.1 Digital-Controlled Oscillator

The DCO is the core element of the ADPLL, responsible for converting the digital control word, DCW, into a clock output. The four-stage ring differential oscillator in Figure 4-2 is implemented to fulfill the multi-phase requirement. To enhance the digitalization degree of the DCO, the current source array consisting of a reference current source and current mirrors is removed. Instead, a simple MOS switch array is used to adjust the resistance value. The delay element is composed of two inverters and a latch to realize the full-differential clock in Fig 4-2(b). Differential delay cells can overcome coupling noise or common-mode noise at the expense of power.

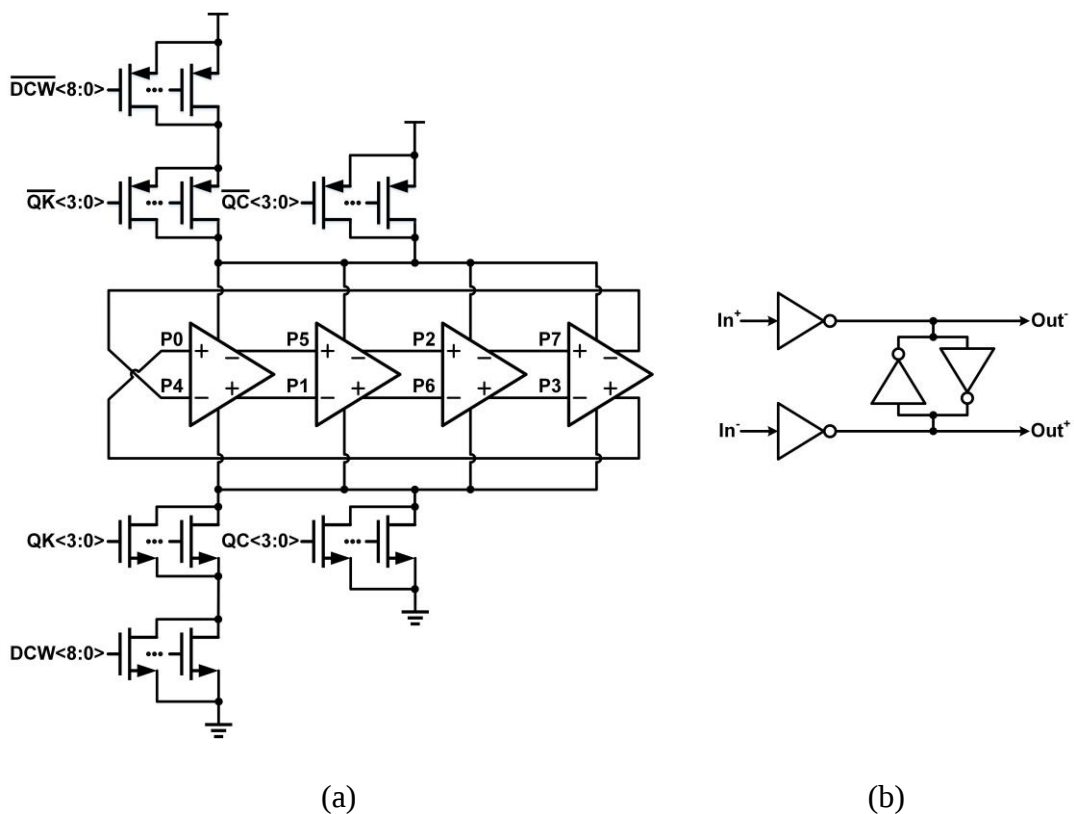


Fig. 4-2 (a) Schematic of four-stage differential RDCO (b) Schematic of delay cell

To compensate the frequency drift due to process variation, except 9 bits DCO code, $DCW<8:0>$ for ADPLL input, there are two sets of codes defined by external signal for

frequency tuning. The two sets of code, QK<3:0> and QC<4:0> adjust K_{DCO} and center frequency respectively. All input signals of the DCO, excluding the supply voltage, are digital, allowing for preliminary system trimming in a straightforward manner. However, due to the limitations in the resolution of hand-tuned codes and the linearity of the DCO, the measurement results are not as satisfactory as expected. Further details on this matter will be provided in Chapter 5. The post-simulation results are displayed in Table 4.1.

Table 4-1 Specification of the ADPLL design example

Corner	SS	TT	FF
Temperature (°C)	125	27	-40
Frequency range (GHz)	0.95~2.55	1.28~3.32	1.69~4.22
Power @ 2.4 GHz (mW)	4.58	4.45	4.5
Noise @ 1 MHz offset (dBc/Hz)	-96.8	-96.7	-96.7

4.1.2 Synchronous Counter and Sampling Circuit

The number of counter bits depends on the required detection range and is not necessarily greater or equal to the ratio, N , of the target frequency to the reference frequency. For instance, let's consider a 5-bit counter. Assuming the target frequency is 2.4 GHz and the reference frequency is 40 MHz, we have $N = 60$. With this counter, it is possible to lock the target frequency at

$$f_{out} = (N + z \times 2^5) \times f_R \quad (4.1)$$

where z represents the set of integers. f_R indicates the reference frequency. Hence, the target frequency could be locked at 1.12, 2.4, 3.68 GHz, and so on. In other word, the detection range of the counter, DR_C , can be written as

$$DR_C = \pm 2^{N_c-1} \times f_R \quad (4.2)$$

where N_c denotes the number of counter bits. In this work, we use a 5-bit counter for integer phase detection to achieve a lock range of [1.76, 3.04] GHz.

Figure 4-3 depicts the schematic of the DCO phase counter and its sampling circuit. The phase counter is responsible for generating the DCO phase by counting the number of DCO rising edges. To obtain accurate integer phase information, a synchronous counter is implemented. In this configuration, the external clock signal is connected to the clock input of each D-flip-flop within the counter, ensuring that all individual output bits change state simultaneously in response to the common clock signal. This synchronization eliminates any ripple effect and propagation delay. To extract the integer phase information precisely at the moment of REF arrival, sense-based flip-flops are used to sample the counter's outputs with the REF signal. However, there is a possibility of metastability if the counter output is still unsettled at the moment of REF arrival. To mitigate this issue, the counter's output is sampled using two signals that arrive 0.25π apart from each other. This approach ensures that at least one of the sampled signals does not experience metastability. The arbiter shown in Figure 4-1 is responsible for selecting the sampled signal that does not exhibit metastability, based on the fractional phase.

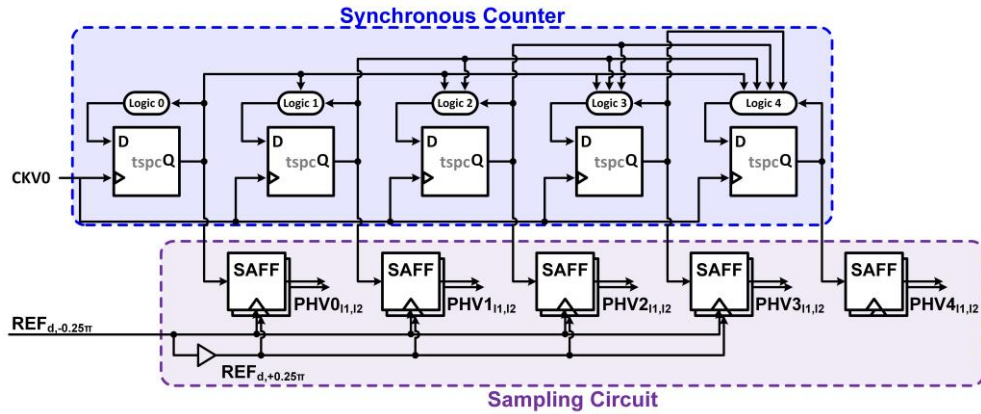


Fig. 4-3 Schematic of 5-bit synchronous counter and sampling circuit

4.1.3 Time-to-Digital Converter

The TDC is a crucial block of the ADPLL as it detects the fractional phase offset between REF_d and CKV . To minimize quantization noise, a finer resolution for the TDC is preferred. Nevertheless, the gate delay of the basic time unit is limited by the process

technology, the inverter-based buffer delay, which is around 10 ps in 90nm process. In this work, a parallel architecture shown in Figure 4-4 is implemented to enhance the resolution of the TDC. The delay chain comprises four sets of eight-stage delay chains, with the inputs of the first stage spaced approximately 3.25 ps apart.

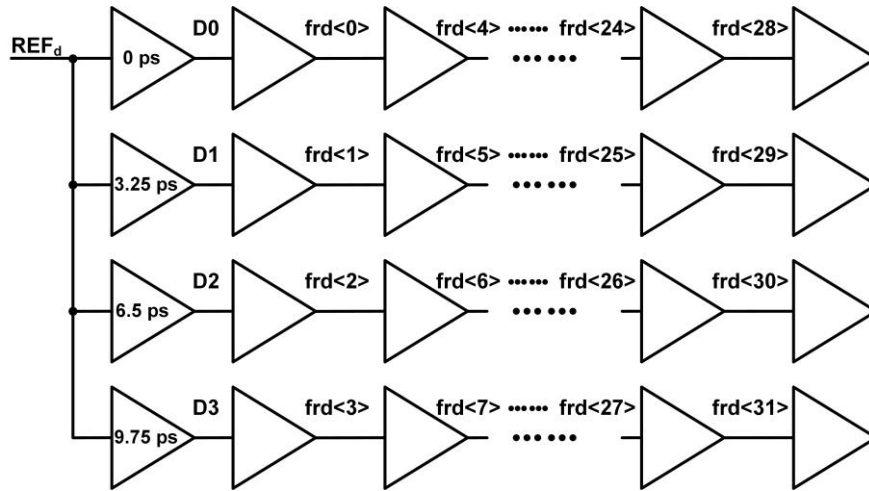


Fig. 4-4 Schematic of delay chain of TDC

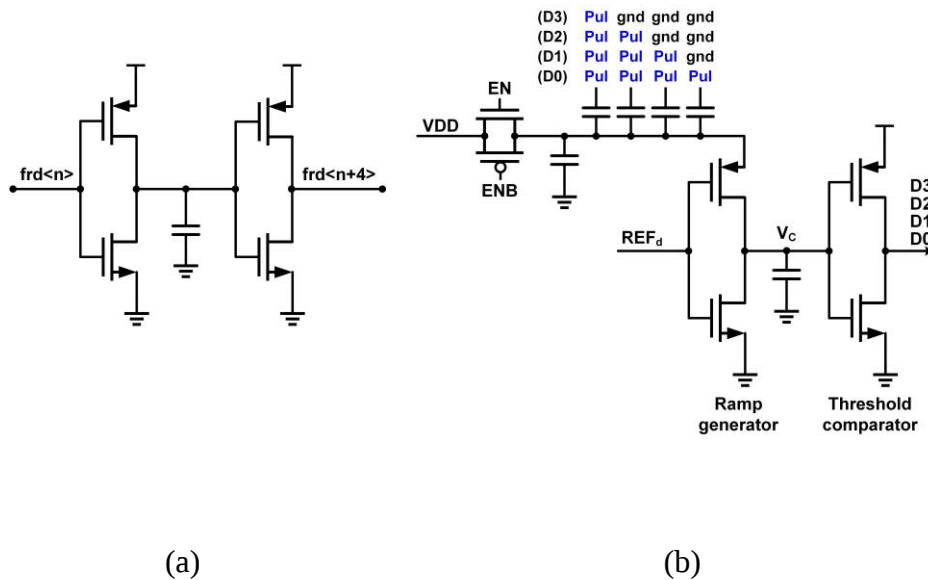


Fig. 4-5 (a) Schematic of inverter-based delay element (b) Schematic of C-DAC-based delay element

The first stage of the delay chain utilizes a C-DAC-based scheme [14], as depicted in Figure 4-5(b), while the subsequent stages employ inverter-based delay elements, as shown in Figure 4-5(a). In the ideal case, the overall delay chain covers $1/4$ of T_{DCO} with a 32 divided time difference. With the assistance of the eight sub TDC of the QI-TDC, the detection range of the TDC covers the entire T_{DCO} . As a result, the overall TDC resolution is approximately $T_{DCO}/256$. The resolution of the delay chain is $T_{DCO}/128$, and with the assistance of phase interpolation, the resolution is doubled to $T_{DCO}/256$. The impact of delay variation caused by PVT variation is not a major concern in circuit design because the QI-TDC is immune to PVT variation through phase interpolation. Finally, the `frd<31:0>` samples these DCO signals, `CKV<7:0>`, to generate the thermometer code.

The C-DAC-based delay cell consists of a capacitor bank, an inverter-based ramp generator, and a threshold comparator. The operation of the C-DAC cell is carried out in two phases: pre-charge ($\phi 1$) and charge-sharing ($\phi 2$), as shown in Figure 4-6. During the $\phi 1$ phase, the capacitive voltage V_C is pre-charged to V_{DD} , and the signal at the other end of the capacitor $C1$ is either `gnd` or `Pul` (V_{DD}). During the charge-sharing phase $\phi 2$, the transmission gate is turned off, the capacitor is floating, and the `Pul` signal is set to 0, leading to charge redistribution. The drop in V_C is proportional to the number of capacitors $C1$ connected to the `Pul` signal. Thus, the operation of the C-DAC cell is completed. Subsequently, the delay cell functions as a constant-slope DTC, providing different output delays based on the capacitive voltage V_C .

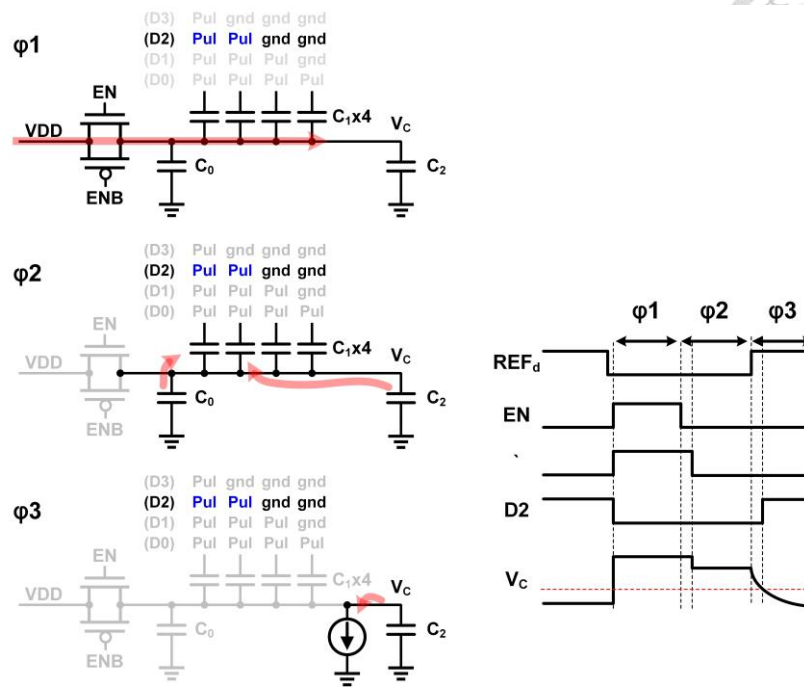


Fig. 4-6 Operation of C-DAC based delay element

4.1.4 Sense-Amplifier-Based Flip-Flop

The sampling circuit plays a crucial role in the TDC as it quantizes the detected signal precisely when the clock signal arrives. The sense-amplifier-based flip-flop (SAFF) [17] is the preferred solution in the choice of sampling element architecture. Compared to the TSPC flip-flop, the SAFF exhibits better power performance and smaller setup times. The SAFF consists of a sense-amplifier (SA) stage and a slave latch, as shown in Figure 4-7. The SA stage captures the data, D, DB, immediately after the rising edge of CLK, and the slave latch is used to maintain the output during the negative half cycle of CLK. The dead zone of the SAFF is approximately 2 ps, which is smaller than the TDC resolution (~ 3.25 ps). Therefore, the SAFFs meet the circuit requirements for the TDC in this work.

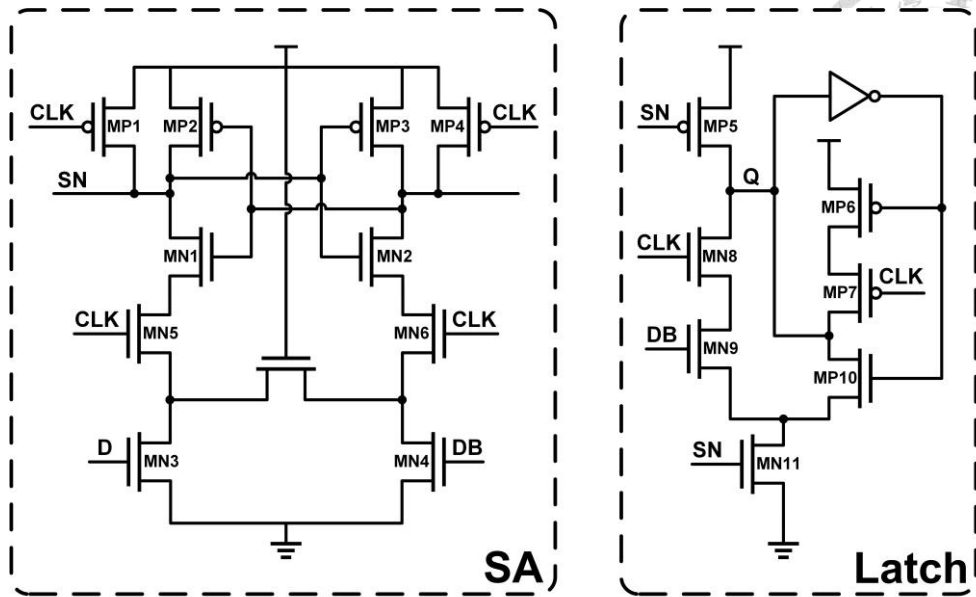


Fig. 4-7 Schematic of sense-amplifier-based flip flop

4.1.5 Digital-to-Time Converter

The DTC is used to generate a dithering signal to randomize the input periodic pattern of the TDC. Figure 4-8 illustrates the DTC architecture. The DTC range is set to 0.5π in order to cover the nonlinearity's period that arises from the $T_{DCO}/8$ repetition. The current ratio between the first and second discharge current is 16, and the resolution of the first discharge time is $T_{DCO}/8$. Therefore, the DTC resolution is $T_{DCO}/128$, and the range of the first discharge time is from 1 to 5 T_{DCO} . As mentioned in the previous chapter, the current flow variation versus capacitive voltage does not play an important role in the inverse DTC. Because of this feature, various resistive current source architectures can be used in the inverse DTC, connecting current source units in parallel makes it easy to achieve proportional current. Therefore, we use diode-connected PMOS as the current source due to its simple structure and fast settling time.

The pulse generator (PG) in Figure 4-9 is utilized to generate discharge times of varying lengths. The PG consists of two chains of 4 flip-flops (FFs) clocked by the DCO signal and two MUXes for selecting the desired signal. The outputs of the FFs are

multiplexed based on sel_2 , resulting in a signal called Q_{int} with a rising edge that can be shifted over time with a T_{DCO} resolution. The Q_{int} signal is retimed by an FF to generate the fractional part of the pulse, SWD. The retimer is clocked by the selected DCO phase, allowing the STOP signal to be shifted over time with a $T_{\text{DCO}}/8$ resolution. However, the retiming process is susceptible to metastability issues due to the proximity of the CLK input and D input of the retimer. To address this, the multiplexer output, which is generated by an auxiliary chain of FFs, is selected. The auxiliary chain of FFs is clocked by the CKV4 signal, π apart from the clock of the main chain, providing a time margin of $(+0.25T_{\text{DCO}}, -0.25T_{\text{DCO}})$ for the retimer. This ensures sufficient reliability for the use of a 2.4 GHz clock.

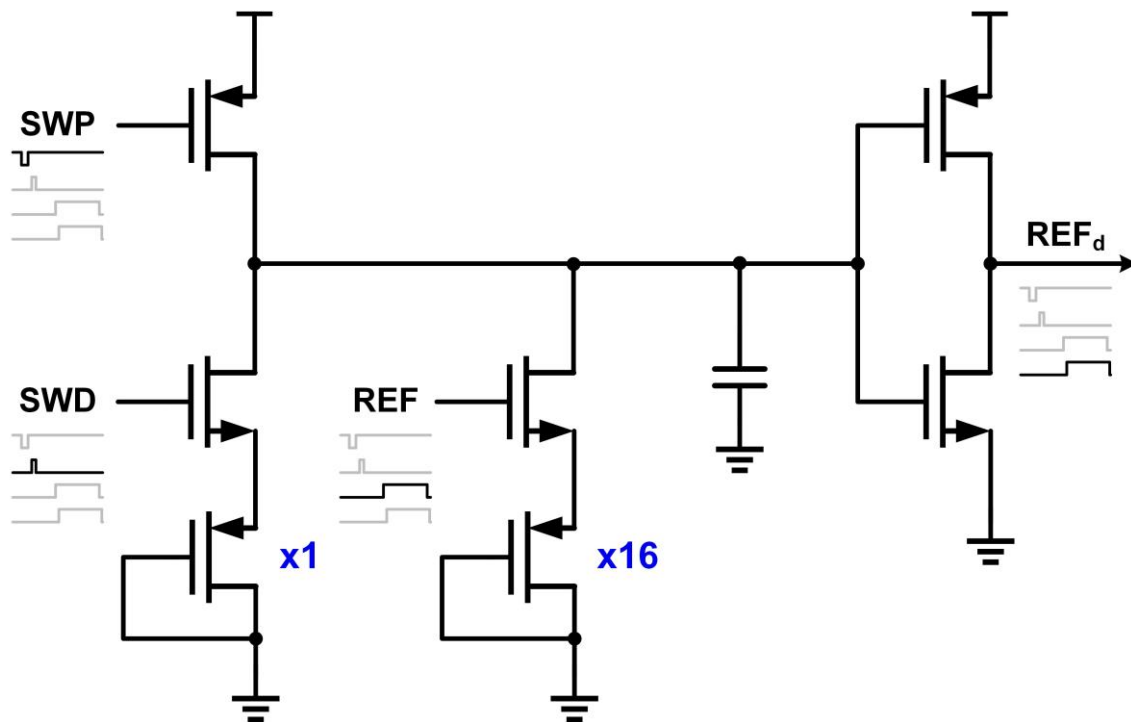


Fig. 4-8 Schematic of inverse DTC

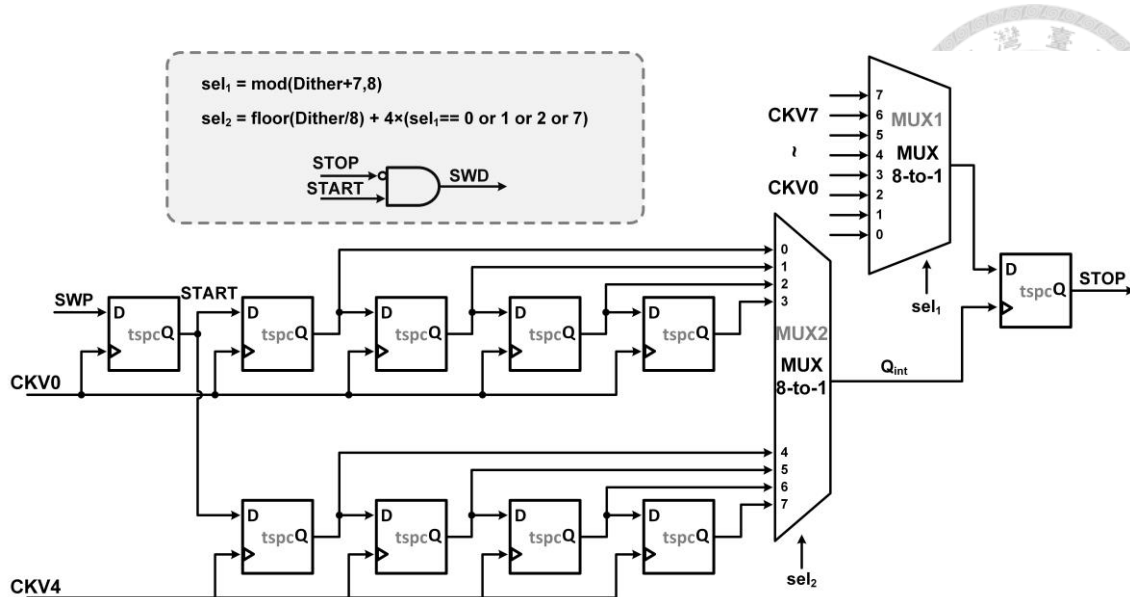


Fig. 4-9 Schematic of pulse generator (PG)

4.2 RTL Design

4.2.1 Digital Loop Filter

The loop filter determines the performance of the ADPLL. It is required to filter out the phase noise from both the reference path and the DCO. Thus, the bandwidth and the loop parameters must be carefully designed. From section 2.2, we are familiar with the transfer function of the ADPLL. The loop parameters, namely the proportional gain, β , and the integral constant, α , can be calculated by examining the topology between the CPPLL and the ADPLLs [18]. By comparing the type-II DLF with the analog loop filter shown in Figure 4-9, we can determine the values of α and β as

$$\begin{aligned}\alpha &= R - \frac{T_R}{2C} \\ \beta &= \frac{T_R}{C}\end{aligned}\tag{4.3}$$

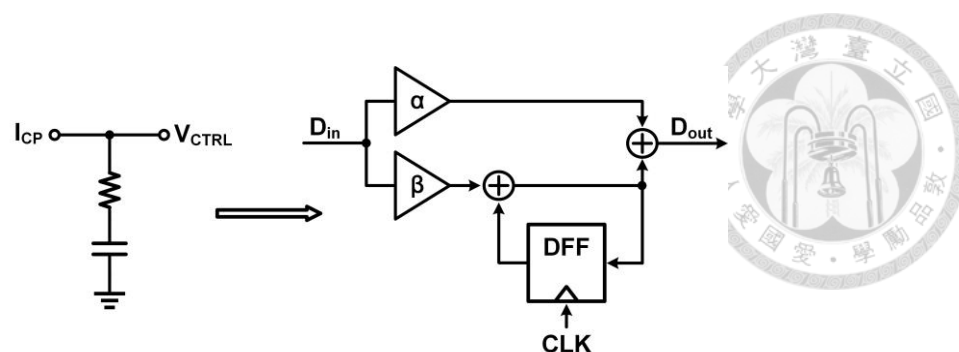


Fig. 4-10 Topology between the CPPLL and the ADPLL

Table 4-2 Design parameters of the ADPLL design example

Reference frequency (f_R)	40 MHz
Phase margin (PM)	70°
Unity gain bandwidth (f_{UGBW})	3 MHz
Frequency division ratio (N)	60
DCO gain (K_{DCO})	195 kHz/LSB
TDC gain (K_{TDC})	$256/2\pi$ (code/rad)

The design example of Table 4-2 is applied to calculate the values of the proportional and integral gain. For simple hardware implementation, the coefficients of the loop filter are approximated as the closest values of power of 2 as shown as follow.

$$\begin{aligned}\alpha &= 2^{-2} \\ \beta &= 2^{-5}\end{aligned}\tag{4.4}$$

However, the approximation may affect the effective loop bandwidth and phase margin of the ADPLL. Figure 4-11 shows the bode plot of the ADPLL. The effective phase margin is 71° and the effective -3dB closed loop bandwidth is 2.64 MHz. Calculated phase noise of the ADPLL according to the linear phase-domain model is shown in Figure 4-12. The calculated noise only considers the TDC quantization noise with $T_{DCO}/256$

resolution, DCO noise simulated by Cadence and phase noise of reference signal. And the nonlinearity of TDC is not taken into account.

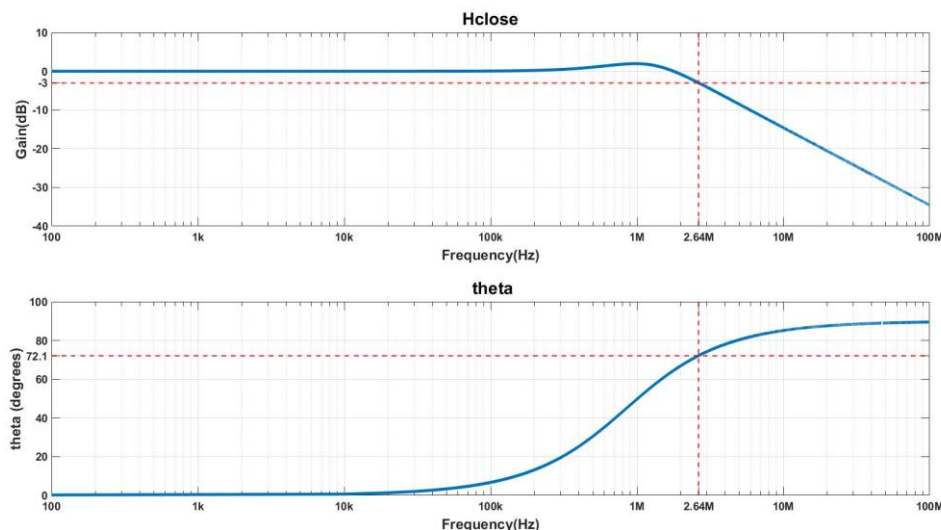


Fig. 4-11 Simulation result of ADPLL by Matlab

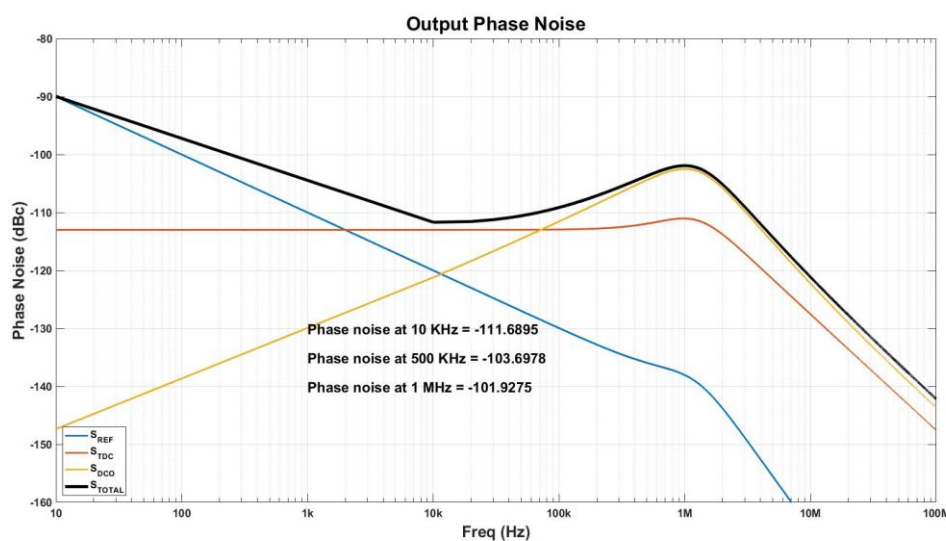


Fig. 4-12 Simulated phase noise by Matlab

4.2.2 Frequency Hopping Control Logic

The Frequency Hopping Control Logic (FHCL) is utilized to implement the TPM operation. It detects any changes in the FCW and calculates the necessary difference in the DCO Digital Control Word, DCW, based on the FCW difference, and the finite state

machine (FSM) in Figure 4-13 transitions to the High-Pass Signal Injection (HPI) state. Then, the calculated difference value is then added to the integrator in the DLF. At the same time, the FSM transitions to the Phase Reset (PRS) state. In the next REF cycle, the DLF input switches to 0 to prevent the entry of phase error caused by delay path spread into the DLF, and the feedback phase information is stored in the integrator of FCW to cancel the phase error caused by delay spread. Finally, the FSM returns to the Phase Locked (PL) state, completing the TPM operation.

The high-pass signal is directly injected into the integrator, eliminating the need for an additional tune bank to achieve TPM operation. In this work, the high-pass signal injection for TPM, the signal injection for fast locking, and the PLL-detected error input are all integrated into the DLF, and the choice of input depends on the state of the FSM.

4.2.3 Fast Locking Function

The Fast Locking function is utilized to reduce the settling time of the PLL. When the absolute value of the phase error exceeds 2π , the FSM transitions to the Fast Locking (FL) state. In the FL mode, the integral gain β is increased, the proportional gain is set to 0, and a differentiator is added before the DLF. This configuration allows the loop to adjust the DCW based on the frequency error. The FSM remains in the FL state until the frequency error changes sign, indicating that the DCO frequency is close to the target frequency as shown in Figure 4-14. At this point, the FSM transitions to the PRS state to reset the FCW integrator. This concept is similar to the method of eliminating delay spread by realigning the PHV and PHR. It enables the main loop to re-enable from very small phase and frequency errors as shown in Figure 4-15.

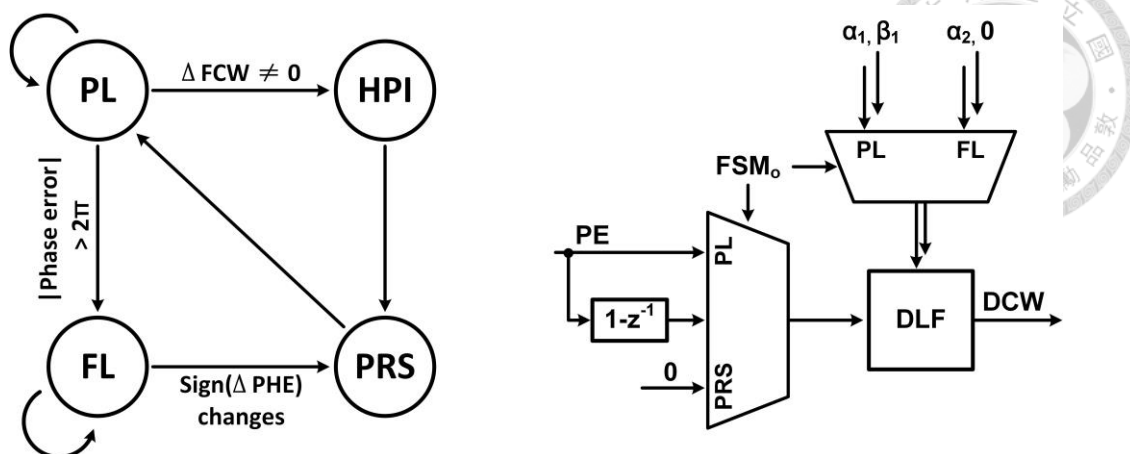


Fig. 4-13 Finite State Machine (FSM) for DLF input control

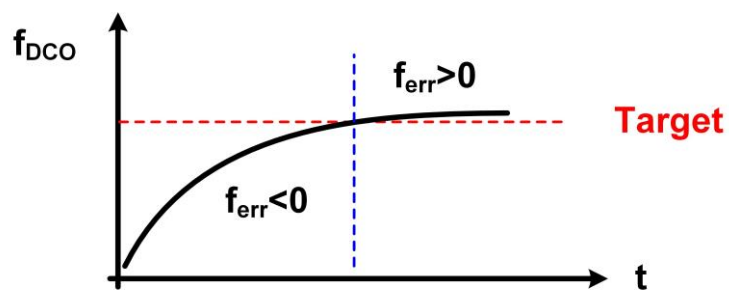
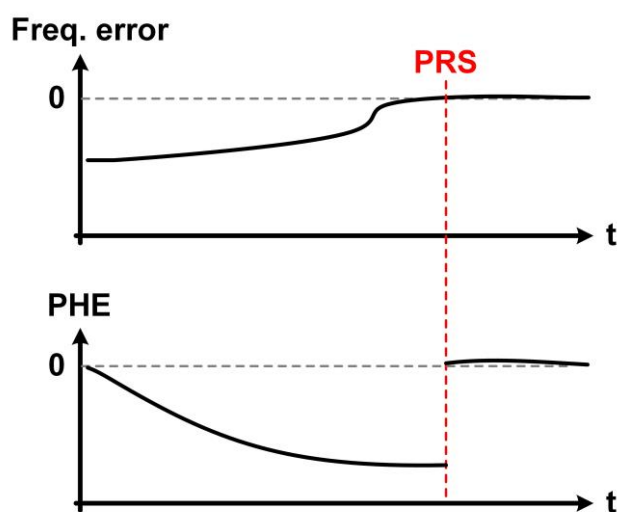
Fig. 4-14 FSM transitions to PRS state if f_{err} (ΔPHE) changes sign

Fig. 4-15 Time diagram of fast-locking operation

4.2.4 Arbiter

According to the previous discussion, the sampling circuit may encounter metastability issues when the sampled signal is unsettled. To address this issue, the sampling circuit samples the counter output twice to ensure that at least one set of data is correct. The arbiter then selects the set of data with a lower probability of metastability based on PHV_F . Figure 4-16(a) provides an example to illustrate the principle of double sampling. In this case, the sampled voltage by $frd<0>$ is approximately $V_{DD}/2$, resulting in a quantized result in an unstable state. By introducing the additional sampling signal, $frd<31>$, which arrives 0.25π later than $frd<0>$, the sampled result is 0. The sampled result by $frd<31>$ is stable. Hence, by choosing the sampled result associated with a sampling signal that is further away from the rising edge of CKV , the occurrence of metastability is significantly reduced. To explain how the arbiter functions, an extreme case is shown in Figure 4-16(b). The rising edge of $frd<16>$ aligns with the rising edge of CKV_0 . Therefore, PHV_F can take on two possible values: 0 or the full code (255). If PHV_F is close to 0, the TDC confirms that CKV_0 leads $frd<16>$, and consequently, the arbiter selects PHV_{I2} as the detected integer phase. Similarly, if PHV_F is close to 255, the arbiter selects PHV_{I1} as the detected integer phase. To simplify the logic, PHV_{I1} and PHV_{I2} are multiplexed based on the most significant bit (MSB) output of the TDC.

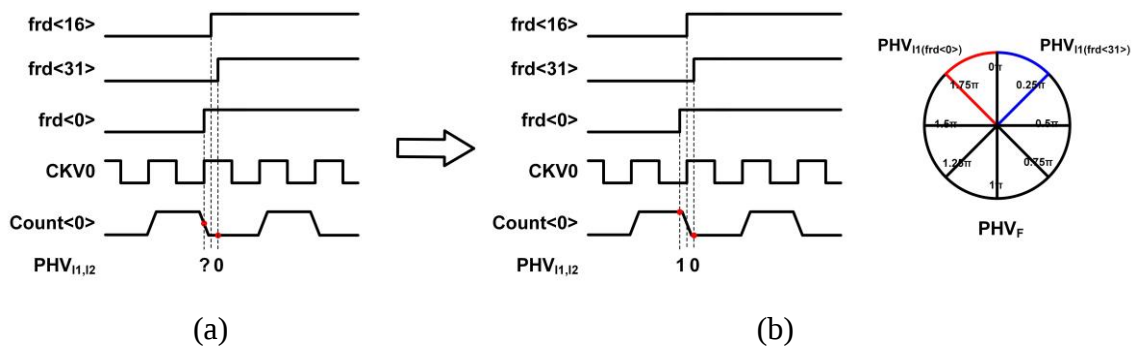


Fig. 4-16 Time diagram of arbiter

4.2.5 Calibration for DTC

Although the inverse DTC is immune to frequency and PVT variations, it can still be affected by circuit mismatch. Specifically, the mismatch between the current sources can lead to gain errors in the DTC. To address this issue, a calibration block is implemented to track the variation of the DTC gain. Since the gain error resulting from circuit mismatch is not influenced by the DCO frequency, the calibration block can operate independently of FSK modulation.

The calibration block, as shown in Figure 4-17, has a similar architecture to the calibration block used for DCO gain calibration. However, in this case, the two reference delays are removed because the dither input is injected into the reference path in the phase domain and cancelled in the digital domain during the same reference cycle.

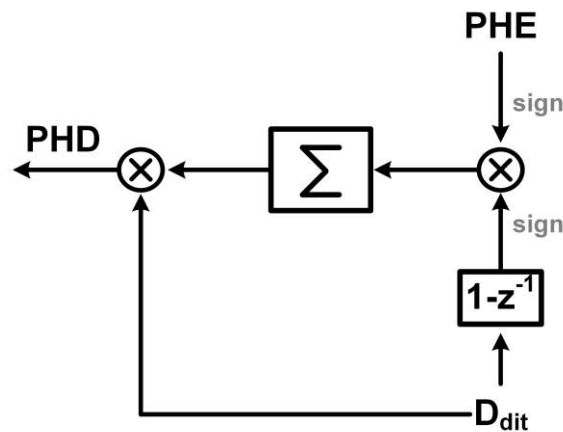


Fig. 4-17 Block diagram of calibration block for DTC gain calibration

4.2.6 Other Block

In addition to the previously mentioned circuit, the thermo-binary converter, the quadrant detector and the adder in the QI TDC, and the subtractor for calculate phase error, PHE, are also synthesized by EDA tool. For the convenience of synthesis, the digital blocks clocked by the same signal are integrated together. The digital blocks can be divided into two parts with clock. The DSM circuit to decrease DCO quantization

noise is clocked by $f_{CKV}/8$ and the other digital blocks are clocked in reference frequency. The counter for integer phase detection and the pulse generator in the inverse DTC is custom design since their propagation delay should be to maintain consistency between different bits. The digital blocks synthesized by tool is marked in the Figure 4-18.

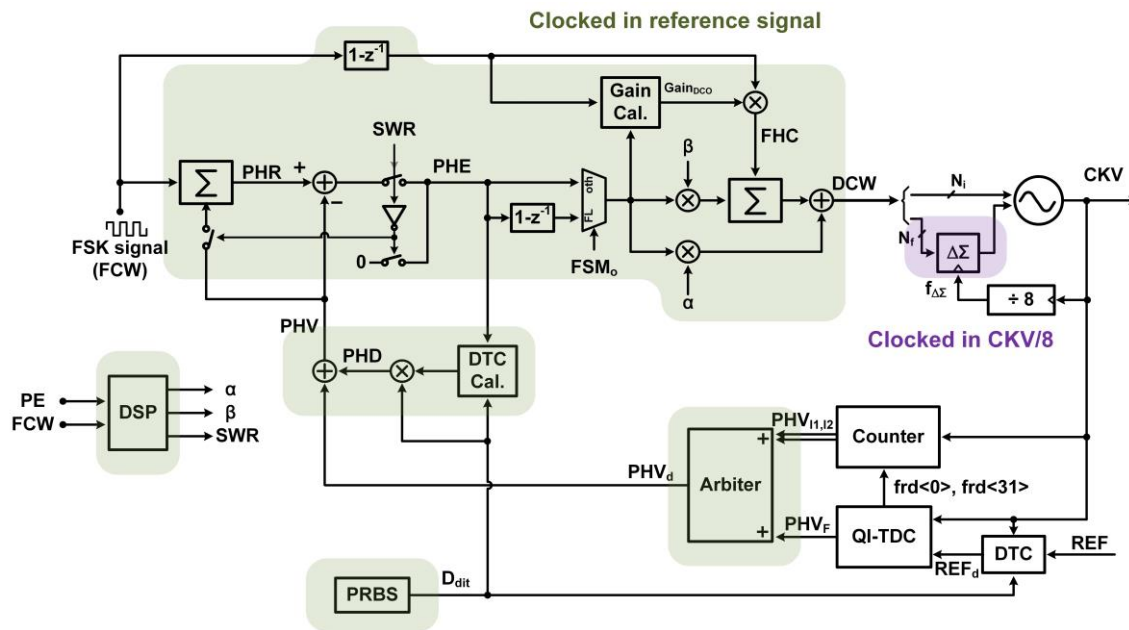


Fig. 4-18 Overall architecture of ADPLL marked with different clock domain



Chapter 5 Measurement Results



5.1 Measurement Setup

Figure 5-1 shows the testing environment to evaluate the phase noise and spectrum performance of the proposed ADPLL. Regulators (LT3020) are adopted to provide DC biases. The signal-end output RF signal is fed into signal source analyzer (R&S FUSP) to measure noise and spectrum of the ADPLL. The digital control signals including frequency command word (FCW), DCO frequency band selection, PLL mode selection and externally controlled calibration code are given by serial-input-parallel-output (SIPO) interface.

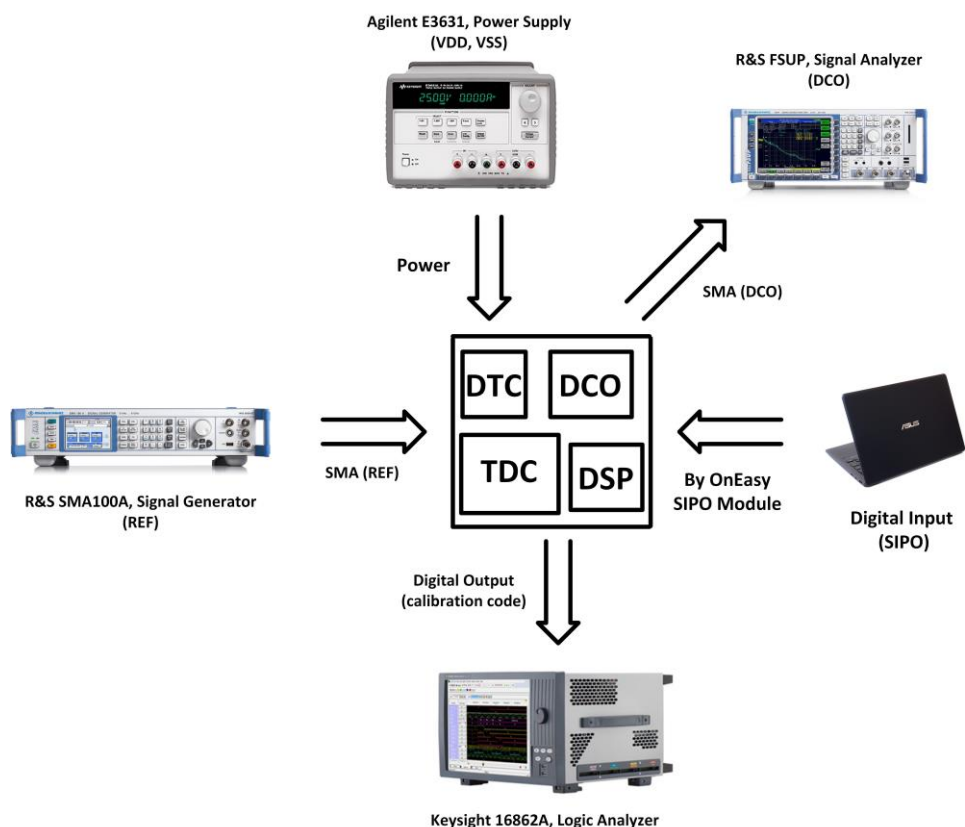


Fig. 5-1 Measurement setup

5.2 Chip Pin Configuration and Chip Photo

Figure 5-2 shows the pin configurations and the chip power-domain distribution. To alleviate unwanted noise coupling through power lines, there are six separate power domains in the chip: DCO, Digital, DTC&TDC, Other-block, Output Buffer and SPI. Figure 5-3 displays the die photograph of this chip, which was fabricated in TSMC 90 nm 1P9M CMOS mixed-signal process. The chip including PADs occupies an area of $1272\mu\text{m} \times 1374\mu\text{m}$. The active core area is only 0.108 mm^2 .

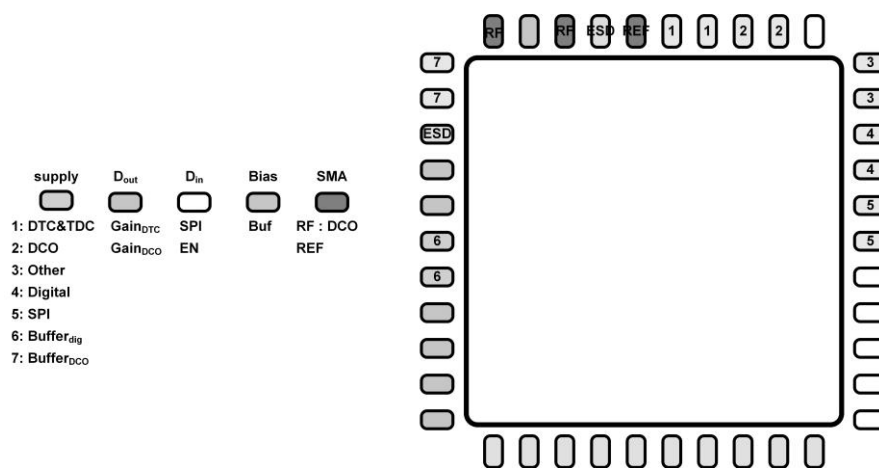


Fig. 5-2 Pin configuration and power domain diagram

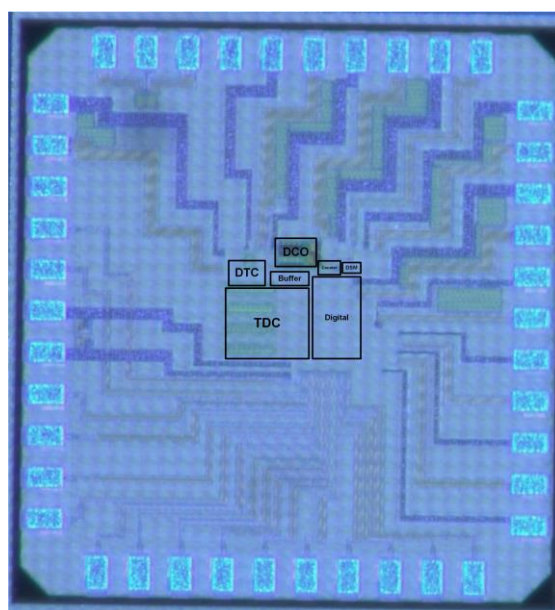


Fig. 5-3 Chip photo

5.3 Measurement Result

The proposed ADPLL draws 6.71 mA from a 1.2 V supply. The current consumption of each building block is illustrated in Table 5-1.

Table 5-1 Measured current consumption

	Post-sim (tt)	Measure
DCO	4.40 mA	4.39 mA
Other	0.77 mA	0.94 mA
Digital block	-	0.72 mA
DTC&TDC	0.64 mA	0.67 mA
Total	5.82 mA (w/o digital)	6.71 mA

5.3.1 Spectrum & Phase Noise for Integer-N channel

The spectrum of the DCO frequency at 2.4 GHz is shown in Figure 5-4. The reference spur measures approximately -39 dBc. The measured result is 6 dB worse than the post-simulation result. There are several possible reasons for this discrepancy: (1) The cross-coupling effect was not taken into account during the post-simulation process. (2) The isolation buffers between the DCO output and the TDC may not be functioning properly, leading to changes in the load capacitance of the DCO. (3) When the DCO resistor array switch is switched, the parasitic capacitance of the switch may introduce a significant current that modulates the DCO frequency.

The phase noise of the DCO frequency at 2.4 GHz is depicted in Figure 5-5. The measured phase noise at a 1 MHz offset is -99.8 dBc, which is similar to the result calculated using Matlab. However, the integrated jitter from 1 kHz to 100 MHz measures at 2.24 ps, significantly worse than the calculated result of 0.8 ps using Matlab. One possible reason for this discrepancy is that the noise level of the noise sources, such as

the TDC and REF, is severely underestimated, resulting in the in-band noise dominating the contribution to jitter.

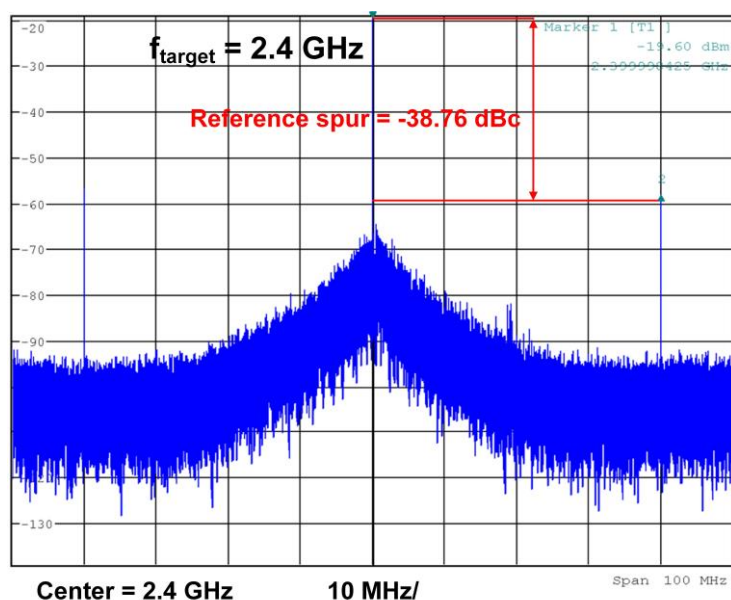


Fig. 5-4 Spectrum at 2.4 GHz

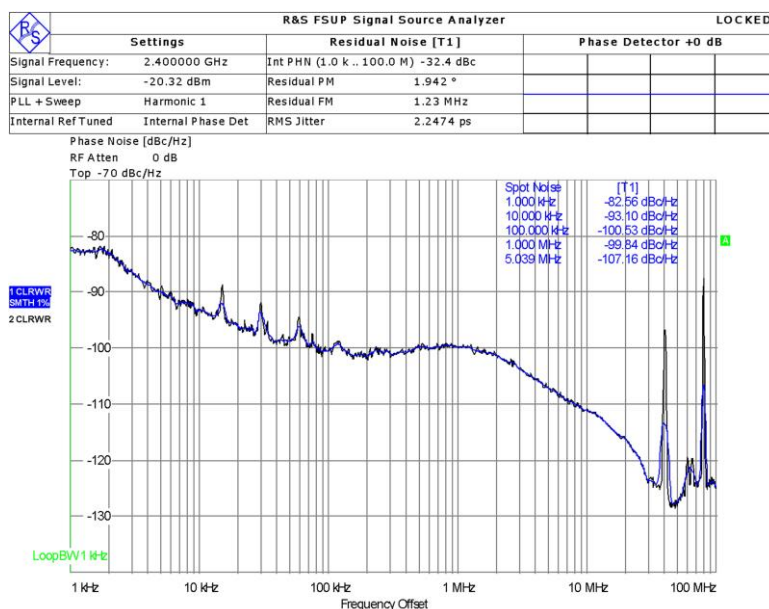


Fig. 5-5 Phase noise at 2.4 GHz

5.3.2 Spectrum & Phase Noise of a Near-Integer Fractional-N Channel

Figure 5-6(a) shows the measured PLL spectrum for a near-integer fractional-N channel with $FCW = 60 + 2^{-10}$, where the DTC dithering was turned OFF to assess the spur performance of the QI-TDC only. The worst-case fractional spur is -37.2 dBc at an offset of 39 kHz. When the DTC dithering is turned ON, an about 3 dB reduction of the worst-case fractional spur is measured in Fig. 5-6(b). However, the improvement is not as significant as anticipated. The likely reasons are that (1) the phase-spacing mismatches in the multi-phase DCO introduce non-linearity errors in the TDC, causing the TDC nonlinearity to deviate from a TDCO/8 periodic pattern. Consequently, the probability density function varies across different quadrants ($NT_{DCO}/8$). As a result, the narrow-range DTC has limited effectiveness in reducing fractional spurs. (2) The DTC resolution ($T_{DCO}/128$) is same as the resolution of TDC's delay chains. Hence, the idle tone increases when the TDC input approaches the quantization scale.

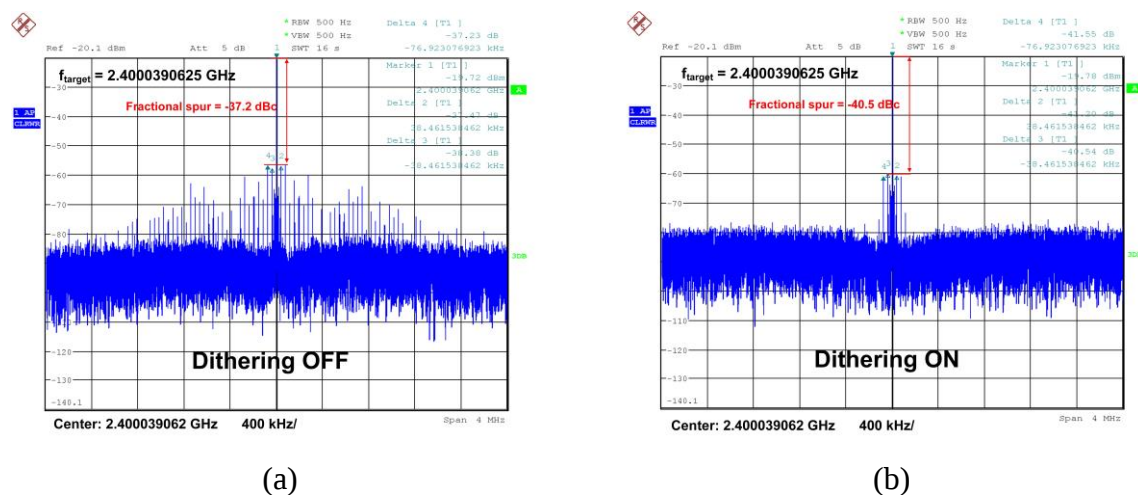


Fig. 5-6 Measured PLL spectrum for a near-integer fractional-N channel with FCW = $60 + 2^{-10}$

The measured phase noise where the DTC dithering is turned OFF is shown in Figure 5-7. At a frequency offset 1 MHz, the phase noise is -95.6 dBc/Hz. And, the integrated jitter from 1 kHz to 100 MHz is 3.07 ps. The Figure 5-8 shows an overlay analysis of the

phase noise with DTC dithering enabled or disabled, and with the DTC calibration loop turned on or off. However, the calibration loop failed to work as expected, as the calibration code did not converge. In order to compensate for the DTC gain error caused by element mismatch, the calibration code had to be manually adjusted through a trial and error process using an external signal. The calibration loop cannot converge due to the higher-than-expected system noise. The learning factor ($2^{-\eta}$) is set too large, which prevents convergence. The integrated jitter (1 kHz-to-100 MHz) where dithering ON and calibration ON (fail to converge) is 6.35 ps, and the integrated jitter where dithering ON and calibration OFF is 4.78 ps.

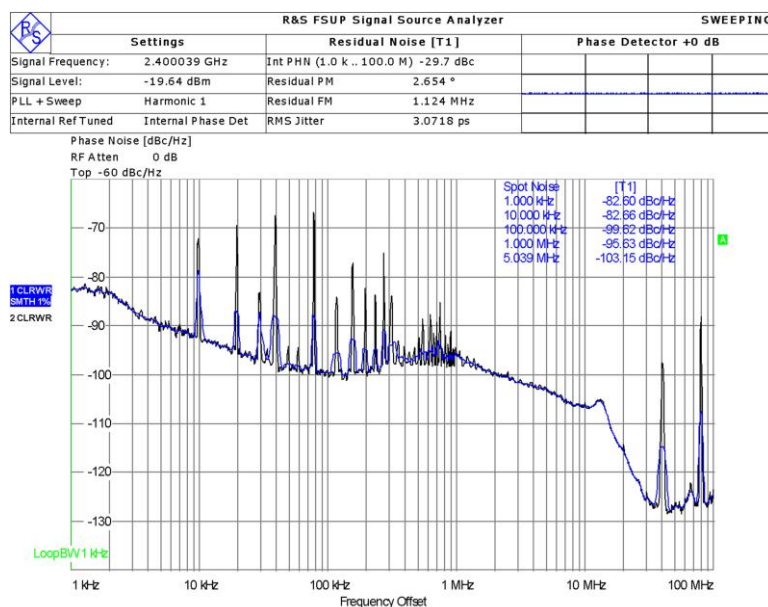


Fig. 5-7 Measured phase noise with dithering OFF for a near-integer fractional-N channel with $FCW = 60 + 2^{-10}$

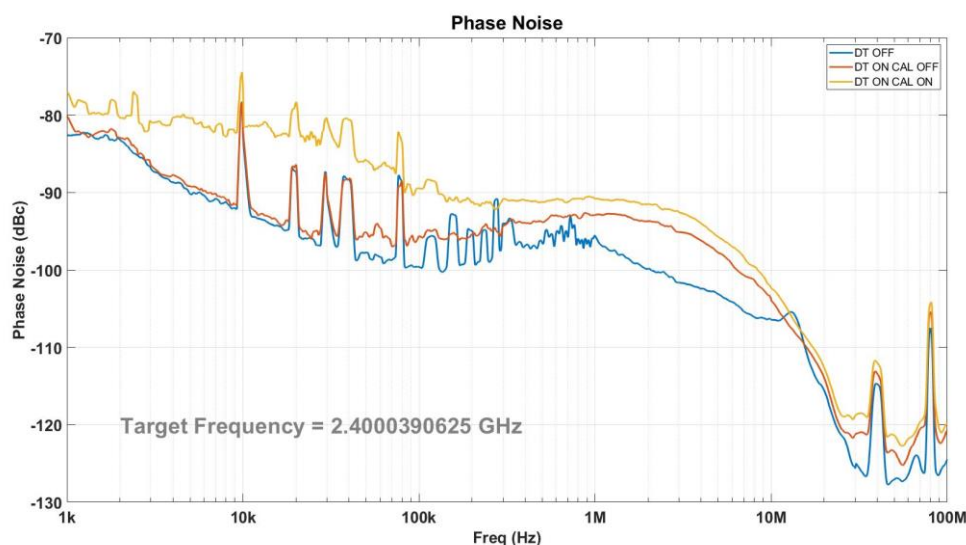
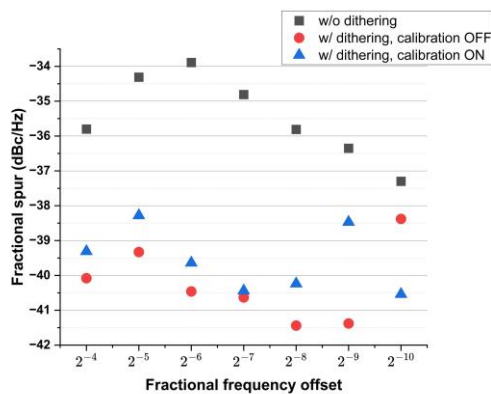


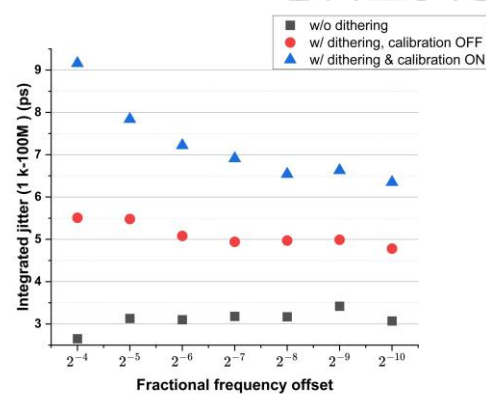
Fig. 5-8 Overlay analysis of measured phase noise with dithering and calibration ON or not

Figure 5-9 summarizes the maximum fractional spur level and integrated jitter from 1 kHz to 100 MHz during scaled fractional-mode operation. When dithering is turned off, the worst fractional spur occurs at $FCW = 60 + 2^{-6}$, rather than at the nearest-integer fractional-N channel. This is due to the repetitive nature of the partial QI-TDC nonlinearity. At $FCW = 60 + 2^{-6}$, the energy of the spur is distributed between 625 kHz, its double frequency, 1.25 MHz, and octave frequency, 5 MHz, offset, with the amplitude of the spur at the harmonic fractional frequencies offset being higher than at 625 kHz offset.

The presence of dither-noise helps reduce fractional spurs at any fractional-N channel. However, the effectiveness of dithering diminishes as the channel gets closer to the integer-N frequency, potentially due to the limited range of the DTC. Nevertheless, regardless of the frequency channel, dither-noise effectively reduces the amplitude of spurs at the harmonic fractional frequencies offset. Figure 5-10 illustrates the spectrums with and without dithering at $FCW = 60 + 2^{-6}$. The higher harmonic spur is eliminated by the dither-noise.

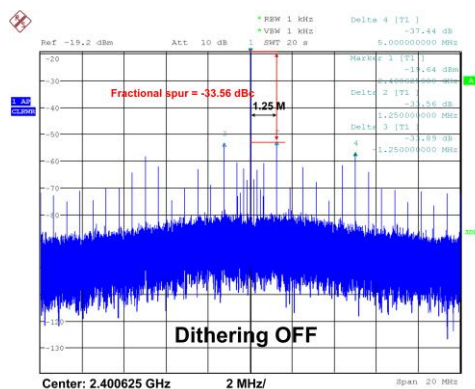


(a)

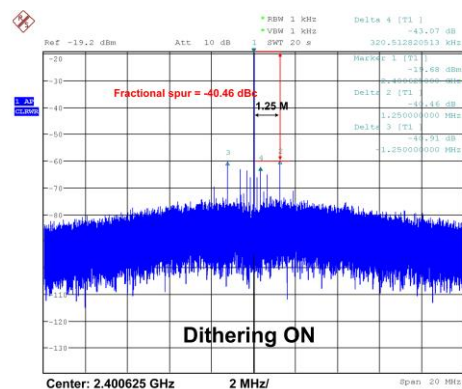


(b)

Fig. 5-9 (a) Maximum fractional spur level and (b) integrated jitter from 1 kHz to 100 MHz during scaled fractional-mode operation



(a)



(b)

Fig. 5-10 Measured PLL spectrum for a near-integer fractional-N channel with $FCW = 60 + 2^{-6}$

Since the DTC calibration code failed to converge, we turned off the calibration block and provided the calibration code with an external signal (SIPO). The summary of integrated jitter is shown in Fig 5-11(b). The jitter level is about 3 ps when dithering is off, and about 5 ps when dithering is on with external calibration code.

5.3.3 Loop-Bandwidth Drifts vs. DCO Nonlinearity

The DCO nonlinearity causes the bandwidth to drift with output frequency variation. The code-to-frequency curve of the DCO is shown in Figure 5-11. When the code input is small, the gain of the frequency-to-code conversion is large. Conversely, when the code input is large, the gain of the frequency-to-code conversion is small or even close to 0.

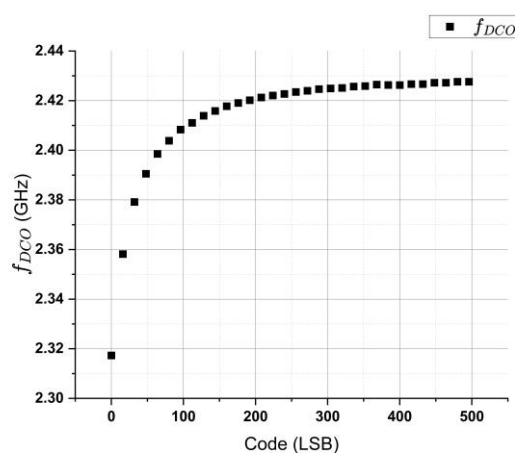


Fig. 5-11 Code-to-frequency curve of the DCO

This results in the ADPLL loop bandwidth meeting design specifications only in a small frequency range. Figure 5-12 illustrates the phase noise with a reference frequency ranging from 39.5 to 40.2 MHz, corresponding to output frequency range of 2.37 to 2.412 GHz. The loop bandwidth decreases as the output frequency increases. The ADPLL performance is highly dependent on the target frequency bands, making it unsuitable for FSK applications. Actually, the phase noise performance at 2.394 GHz (3.99×60 MHz) than at 2.4 GHz due to the DCO resolution of the hand-tuned code. When compared to the general method of adjusting the variable resistance of the current source, digital control is limited by its accuracy, and the resulting bandwidth after digital adjustment is still far from the optimal value.

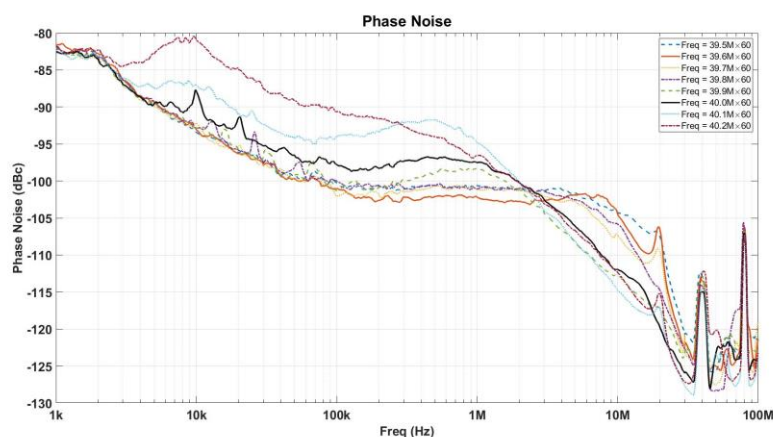


Fig. 5-12 Phase noise with a reference frequency ranging from 39.5 to 40.2 MHz

5.3.4 Frequency Hopping Measurement with TPM Architecture

Figure 5-13 shows the frequency hopping operation where f_1 is 2.39 GHz and f_2 is 2.41 GHz. And the frequency of the modulation signal is 400 kHz. The measured result which the modulation signal only was injected into the feedback path (OPM) is shown in Fig. 5-13(a). The frequency slowly change from f_1 to f_2 due to the loop-bandwidth limit. Compared to Fig. 5-13(b), the modulation signal was both injected into DCO and feedback path is shown in Fig. 5-13(b). Hence the output frequency can alter from f_1 to f_2 without delay. However, the frequency ripple at 2.39 GHz is much larger than at 2.41 GHz because the loop-bandwidth is larger than expected due to the DCO nonlinearity.

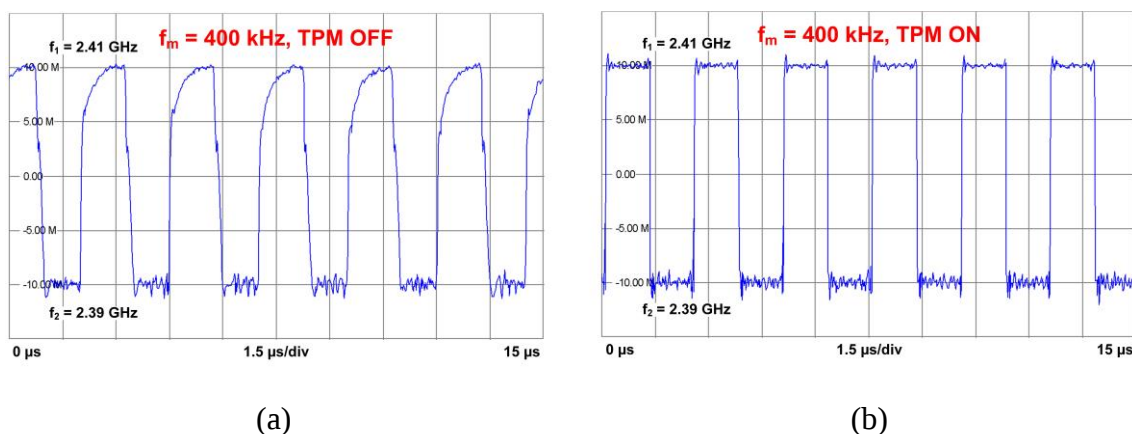


Fig. 5-13 Measured time diagram when (a) TPM OFF (b) TPM OFF

5.3.5 BFSK Modulation Measurement with TPM Architecture

Figure 5-14 presents the summary of BFSK modulation measurements with a data rate of 800 kbps and a deviation frequency of 1.25 MHz. The FSK signal pattern is a repeating sequence of 0101, which can be generated using a 400 kHz square waveform. The MXA signal analyzer (N9020A) is used to demodulate and measure the output of the ADPLL. The measured BFSK error is 5.36% RMS, and all the measured signals can be demodulated correctly.

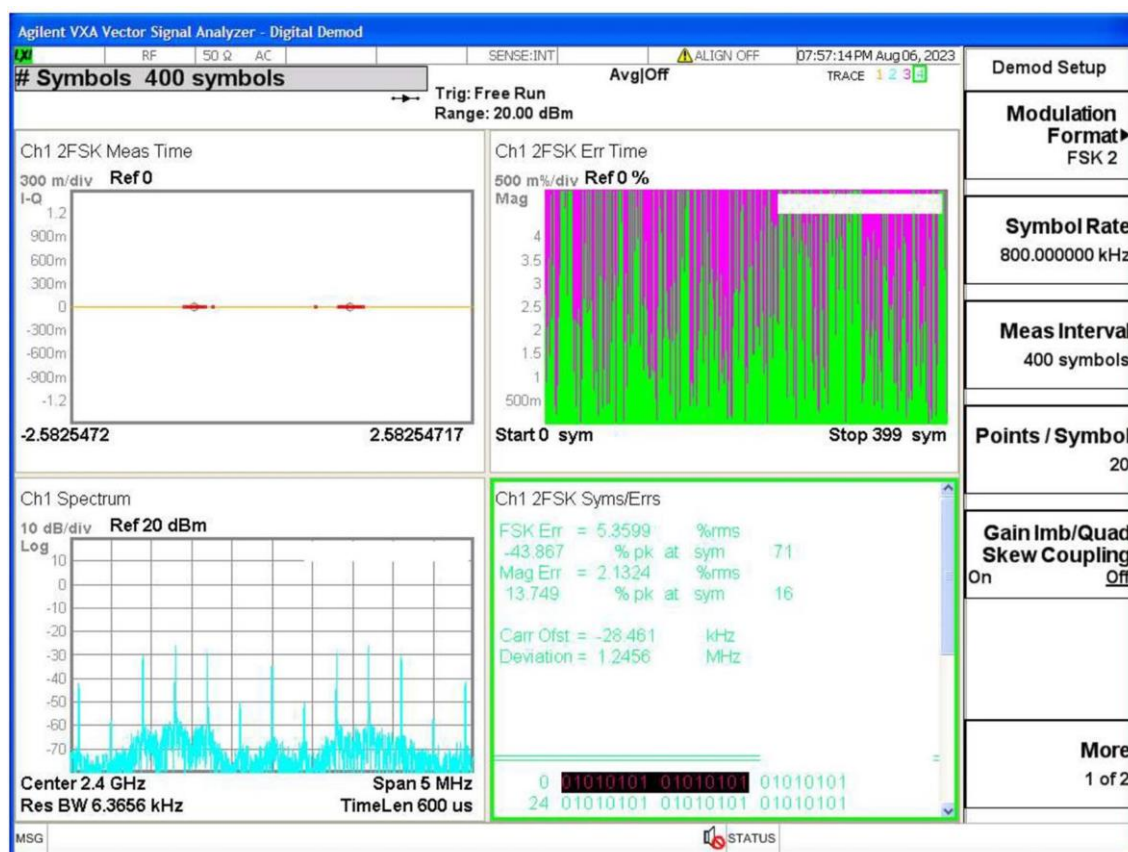


Fig. 5-14 Summary of BFSK modulation measurements

5.4 Summary

From the above measurement results, the effectiveness of the proposed technique is demonstrated. The only block that is not functioning properly is the calibration loop of the DTC. However, with the help of the QI-TDC, the fractional spur level is approximately -35 dBc. Furthermore, the DTC dithering can provide a 3-to-5 dB reduction to the fractional spur. Table 5-2 summaries the measured performance and Table 5-3 shows the comparison results with other works, where various techniques are employed to enable the TDCs and DTCs to maintain a stable and accurate gain across different frequencies and PVT variations.

Table 5-2 Measured performance summary for the proposed ADPLL

Process	TSMC 90-nm CMOS		
Supply / Current	1.2 V / 6.71 mA		
Output Frequency	2.35-2.45GHz		
Reference Frequency	40 MHz		
Mode	Integer-N	Frac-N Dithering ON	Frac-N Dithering OFF
Fractional Spur	-	-37.2 dBc	-40.5 dBc
Phase noise @ 1 MHz offset	-99 dBc/Hz	-95.5 dBc/Hz	-92.9 dBc/Hz
Integrated Jitter (1k-100M)	2.44 ps	3.07 ps	4.78 ps

Table 5-3 Comparison table

	[11]	[19]	[15]	[20]	This work
Technology	65	28	40	12	90
Technique	DI-TDC	DI-TDC	TAU+TDC	Embedded TDC, Vernier TDC	QI-TDC, Dithering
Osc.	Ring	Ring	LC	Ring	Ring
Supply (V)	1.0-1.4	0.4-1.0	1.1(DCO), 1	0.8	1.2
Power (mW)	10.8	6.95 @ 2.056G	3.48	9.7 @ 2.4G	8.06
Ref. Freq. (MHz)	64	4-80	40	40	40
Output Freq. (GHz)	2	0.103-2.056	2.68	1.8-2.7	2.4
Frac. Spur (dBc/Hz)	-	-23	-59	-	-39
Phase Noise (dBc/Hz)	-107 @ 4 MHz	-	-119.7 @ 1 MHz	-90 @ 40 kHz	-99 @ 1 MHz
Integ. Jitter (ps)	3.15 (10k-100M)	2.13@2.056G (10k-100M)	0.182 (10k-40M)	4.71 -	2.44 (1k-100M)
Active Area (mm ²)	0.047	0.0043	0.31	0.063	0.108
FOM ⁽¹⁾ (dB)	-219.7	-225	-249.4	-220.7	-223

$$FOM^{(1)} = 10 \cdot \log \left(\left(\frac{\sigma_t}{1s} \right)^2 \cdot \frac{Power}{1mW} \right)$$



Chapter 6 Conclusion and Future Work



6.1 Conclusions

In this thesis, a calibration-free TDC based fractional-N ADPLL using QI-TDC is presented. The ADPLL is designed to generate FSK clock signal. By adopting two-point modulation (TPM), settling time of ADPLL is greatly reduced. The TPM provides an all-pass transfer function from input modulation signal to output clock signal by adding an extra signal injection point at the DCO. Nevertheless, the injection timing mismatch between two injection points leads unwanted phase error and increases bit error rate (BER) in communication. To address this issue, a forward cancellation technology is presented. Since cancellation is done in digital domain, no matched analog components are required and the cost of area and power is ultra-low.

To increase the resolution of output frequency, fractional-N architecture is preferred. Fractional-N ADPLL can break the limit of output frequency resolution, which is reference frequency in integer-N PLL. However, the TDC in fractional-N ADPLL suffers from conversion error with frequency and PVT variation and degrades PLL performance. In this thesis, we proposed a quadruple-referenced interpolating TDC (QI-TDC) based on [11]. QI-TDC can provide accurate conversion gain and greatly linearity under frequency and PVT variations. Furthermore, to reduce the magnitude of fraction spurs, additional noise is added into PLL's feedback path to randomize the periodic pattern of TDC input. The dithering is realized by a DTC which is specially designed to track gain variation under frequency and PVT variations.

The proposed ADPLL is implemented in 90-nm CMOS technology. Based on measurement results, the fractional spur level is suppressed by as much as 16 dB than [19]. Additionally, DTC dithering provides about 3 dB reduction of fractional spur at $FCW = 60 + 2^{-10}$. The whole chip consumes 8.01 mW, while active area is 0.108 mm².

6.2 Directions for Improvement of this Chip

6.2.1 Loop-Bandwidth Tuning Capability

To optimize the phase noise performance of the ADPLL, a wide loop-bandwidth is typically set to suppress the poor phase noise characteristics of the ring oscillator. However, when evaluating the effect of TPM, the presence or absence of TPM does not show significant differences due to the wide loop-bandwidth. In order to clearly distinguish the effect of TPM, it is necessary to introduce the capability to externally adjust the loop-bandwidth of the ADPLL. By reducing the bandwidth of the ADPLL specifically for observing FSK operation, more representative data can be obtained.

6.2.2 Calibration Loop Optimization

To enhance the design and functionality of the DTC calibration loop, several improvements can be considered. One approach is to incorporate a filter to effectively filter out noise before feeding it into the calibration block. This can help improve the accuracy of the correction process by reducing the impact of unwanted noise sources. Another enhancement is to increase the tap count of the LMS (Least Mean Squares) algorithm used in the calibration loop. By increasing the number of taps, the LMS algorithm can have better convergence and tracking performance, ensuring accurate tracking of DTC gain variations over time. Alternatively, the correction circuit itself can be implemented off-chip. This requires the output of relevant signals, such as phase error and dithering signals, to be sent to an external device, such as a computer, for further processing. This off-chip implementation allows for more sophisticated algorithms and

processing capabilities to be applied, potentially improving the overall performance and accuracy of the DTC calibration process.

6.2.3 Enhancement of DCO Linearity

The nonlinear relationship between the DCO control word (DCW) and the output frequency introduces bandwidth drift in the PLL. To mitigate this issue, a pre-distortion algorithm can be employed to linearize the conversion characteristic of the DCO block. The pre-distortion circuit acts as an inverse model of the DCO characteristic, and when combined with the DTC, it creates an overall block that exhibits improved linearity, as depicted in Figure 6-1.

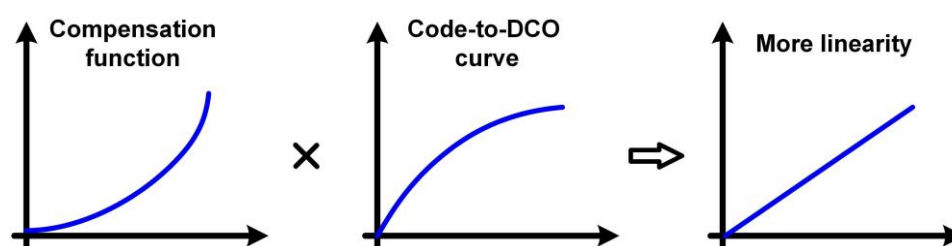


Fig. 6-1 Principle of pre-distortion circuit

6.2.4 Other

In addition to the aforementioned improvements, there is still ample room for enhancing the circuit design. The counter and TDC can be redesigned as digital synthesizable blocks, which would increase the integration level of the circuit. Additionally, this redesign has the potential to reduce the power and area costs associated with the design. To overcome the nonlinearity characteristics due to delay-time mismatches, multiple sampling by D flip-flops is a possible solution. The similar principle is depicted in [21] called stochastic TDC.

6.3 Proposed Fractional Phase Offset Compensation

6.3.1 Motivation

In this thesis, we present a calibration-free TDC-based fractional-N ADPLL using QI-TDC. The TDC occupies approximately half of the core area and contributes to 10% of the total power consumption. In the second schematic, we simplify the problem by disregarding FSK applications and instead concentrate on reducing power consumption, minimizing area, and improving the performance of the ADPLL, including addressing phase noise and fractional spur.

6.3.2 DTC-Assisted ADPLL

The principle of the DTC-assisted ADPLL in Figure 6-2 is to use the accumulated fractional FCW to determine the delay amount to the REF such that REF_d is aligned to the next CKV as the loop is locked [22]. Figure 6-3 depicts the phase prediction scheme for the case $FCW = 2.25$. Only the timestamps of the signal rising edges are shown for the simplicity and all signals are normalized to one CKV period. The required delay is equal to the time deviation between the REF edge and the CKV edge that shows a periodical pattern of 1, 0.75, 0.5 and 0.25 with repetition period of 4.

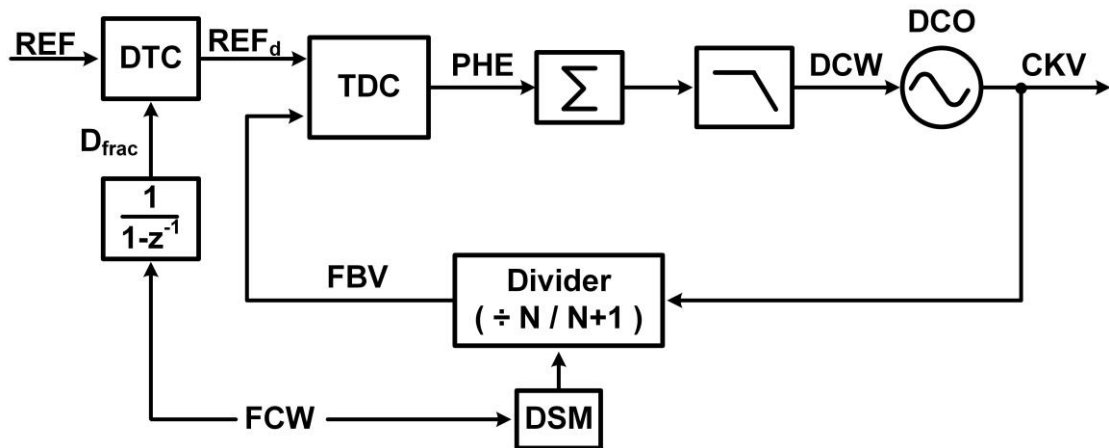


Fig. 6-2 Block diagram of DTC-assisted ADPLL

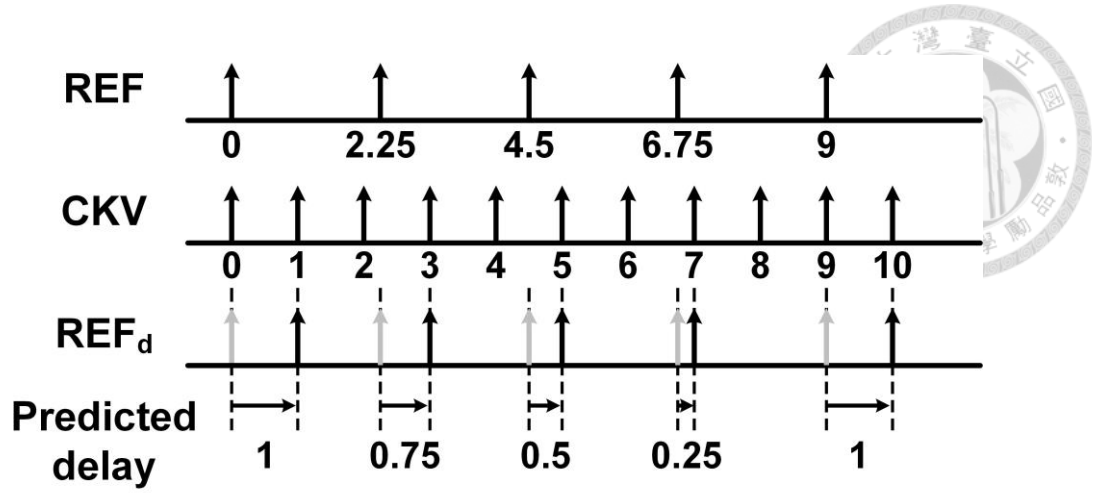


Fig. 6-3 Phase prediction scheme for FCW = 2.25

The predicted delay, DL_{pre} , provided by the DTC can be expressed as

$$DL_{pre} = (1 - D_{frac})T_{DCO} \quad (6.1)$$

with the help of DTC, the quantization noise error is reduced. So the TDC detection range can be narrow with almost zero time difference input. As a result, the TDC detection range can be narrowed down to almost zero time difference. The DTC can be implemented using a C-DAC, ramp generator, and threshold comparator, which consumes less power and occupies less area compared to a high-resolution TDC with a wide detection range. However, according to Eq. 6.1, the output range of the DTC should cover at least one T_{DCO} . Designing a wide output range with high linearity can be challenging. In general, the linearity of the DTC is inversely related to its dynamic range. Since a larger capacitor or resistor array of the DTC introduces larger parasitic effects, which in turn reduces the linearity of the DTC.

6.3.3 Rewritten of DTC Predicted Delay

The DTC control code, D_{frac} , is the accumulated fractional FCW, it can be written in a recursive form as

$$D_{frac}[n] = \text{mod}(D_{frac}[n-1] - FCW_F, 1) \quad (6.2)$$

where FCW_F is the fractional part of FCW. Based on Eq. 6.2, Eq. 6.1 can be rewritten as

$$DL_{pre}[n] = \text{mod}(DL_{pre}[n-1] - FCW_F \times T_{DCO}, T_{DCO}) \quad (6.3)$$

the next value of the predicted delay is obtained by adding a fractional phase unit, $FCW_F \times T_{DCO}$, to the previous value. This allows us to divide the predicted delay into two parts: one part remains the same as the previous cycle's value, while the other part is a constant value determined by the product of FCW_F and T_{DCO} . The time diagram illustrating this concept is shown in Figure 6-4. The predicted delay can be seen as the previous predicted delay (blue) subtracted by the constant delay (red). Additionally, when an overflow occurs in the predicted delay, an additional T_{DCO} needs to be added to obtain the correct value violating Eq. 6.3, as shown in Figure 6-4. However, this does not affect the concept that the equation is trying to illustrate, and in circuit implementation, it can be achieved by adjusting the input of the divider.

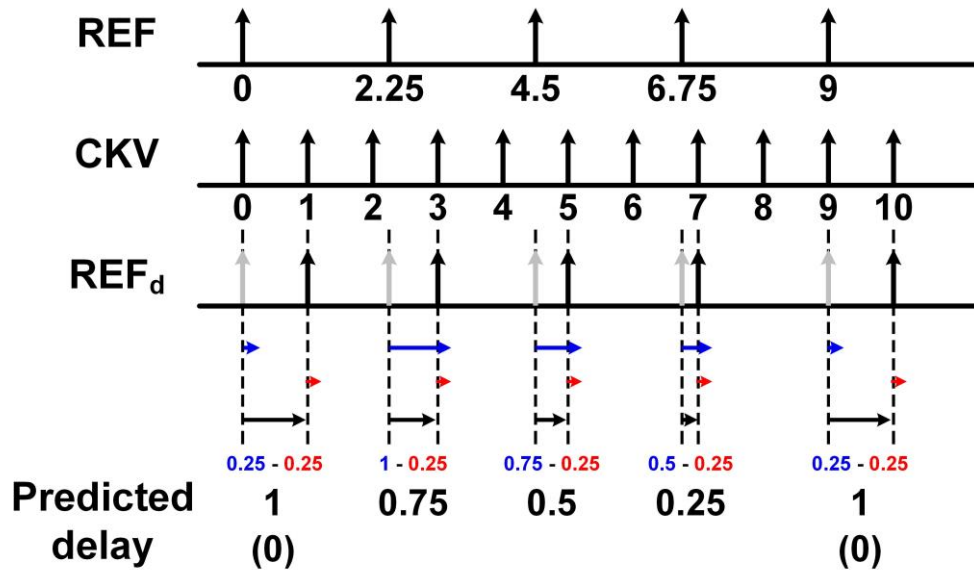


Fig. 6-4 Phase prediction by Eq. 6.3 for $FCW = 2.25$

6.3.4 Time-Register using Time-Mode Arithmetic Units

The time-mode arithmetic unit in [15] provides a way to store time information. The simplified scheme is shown in Figure 6-5(a). It consists of two charge pumps, a capacitor for storing time information and a threshold comparator. The time information SWD is stored as capacitive voltage by discharging for the ON time of SWD. If we want to extract the stored information in the capacitor, the TRI signal turns on the switch to discharge the capacitor and the CMP signal alters to 1 as the capacitive voltage crosses the threshold voltage, V_{TH} . Because the amount of charge discharged by the capacitor from VDD to V_{TH} is fixed and equal to the amount of charge discharged in the first stage, T_{sam} , plus the amount of charge discharged in the second stage, T_{out} . Output delay of the time register, T_2 , can be expressed as

$$T_{out} = T_{TR} - T_{sam} \quad (6.4)$$

where T_{TR} represents the time it takes for the capacitor to discharge from VDD to V_{TH} with a continuous current.

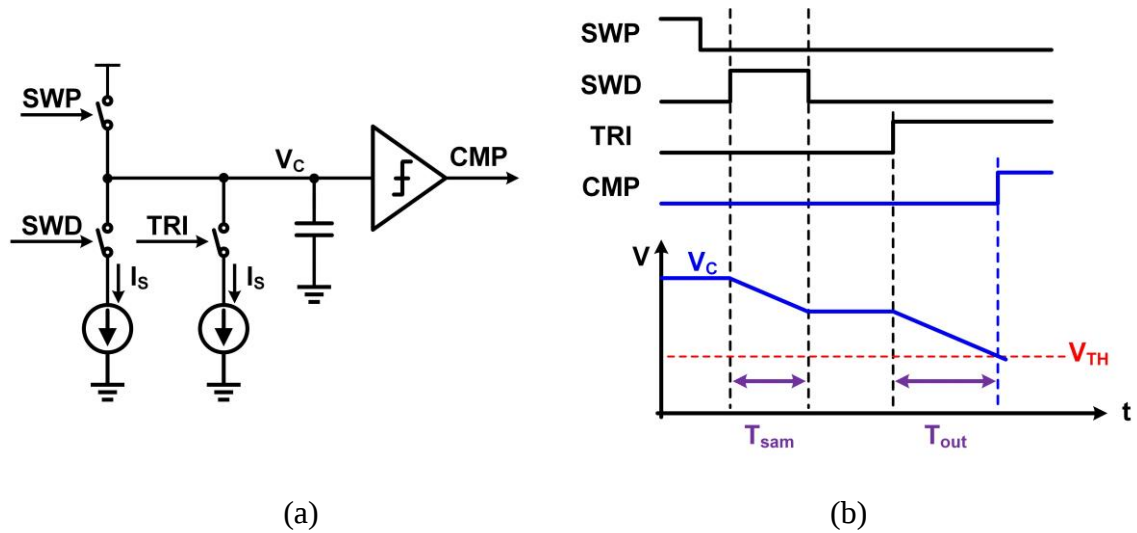


Fig. 6-5 (a) Block diagram of time register and (b) its time diagram

The linearity of the time-register has already been discussed in Section 3.4.3. The time-register exhibits excellent linearity and is immune to channel length modulation and PVT variation. Therefore, the first term, $DL_{pre}[n-1]$, in Eq. 6.3 can be obtained from the time-register. The second term can be obtained using a voltage/digital controlled delay line (VCDL/DCDL) with delay-locked loop (DLL).

6.3.5 Proposed Time-Registers-Based ADPLL

The time registers and the DLL replace the function of the DTC in the ADPLL shown in Fig. 6-2. In the proposed ADPLL, as shown in Figure 6-6, the architecture needs to be divided into two time registers for separate processing. The bottom time register is used to store the time difference, $DL_{pre}[n-1]$, between the FBV and the REF, while the upper time register, combined with the VCDL/DCDL, generates the predicted delay. This division is necessary because a single time register cannot simultaneously store the desired signal and output the signal.

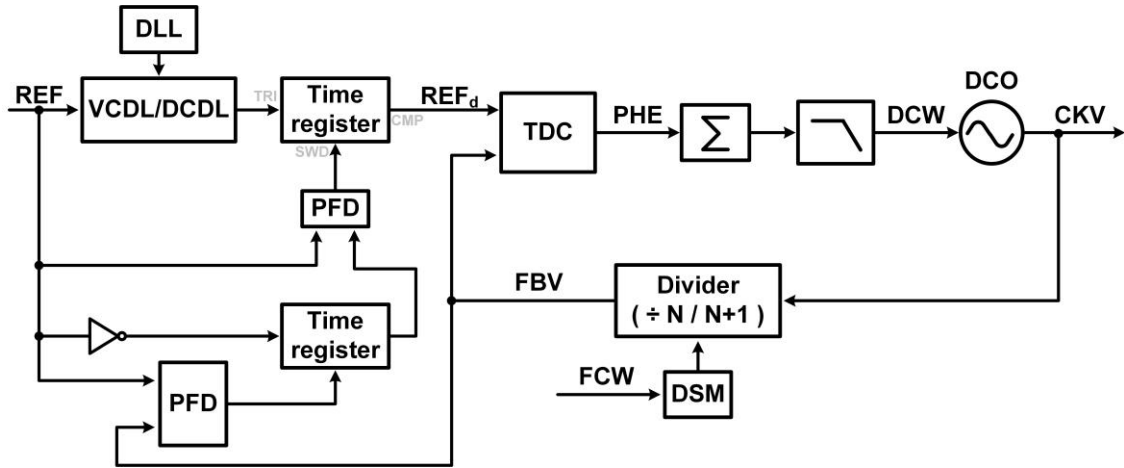


Fig. 6-6 Block diagram of proposed ADPLL

Since the VCDL/DCDL is locked at constant digital code or control voltage. The linearity of VCDL/DCDL will not affect the performance of the ADPLL. Additionally, it provides a pure analog way to achieve fractional phase compensation.

6.3.6 Issues Still to be Resolved

The time-register-based ADPLL is not a traditional phase-locked loop (PLL) but rather a frequency-locked loop (FLL). This is because the combination of the two time registers is equivalent to a differentiator, which focuses on tracking frequency rather than phase. As a result, the overall architecture can be simplified to the scheme shown in Figure 6-7. The time registers serve as a differentiator in the phase domain, while the VCDL/DCDL with DLL functions as the reference fractional frequency control word (FCW_F) and the TDC is used to extract the frequency difference between the target frequency and the CKV. Therefore, the proposed design cannot achieve phase locking.

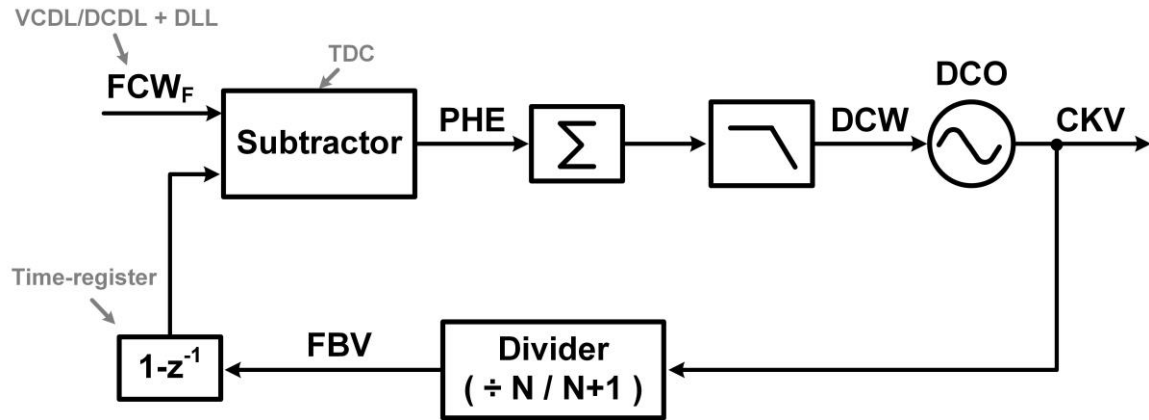


Fig. 6-7 Equivalent circuit of proposed design

The simulation result of behavior model is shown in Figure 6-8. The external noise is injected into the ADPLL at the time of the dotted line. Although the frequency can be locked by the loop. The CKV phase cannot be locked and the phase has large offset at the time of noise injection. Besides, the CKV drift slowly over time. The two are caused by different mechanisms.

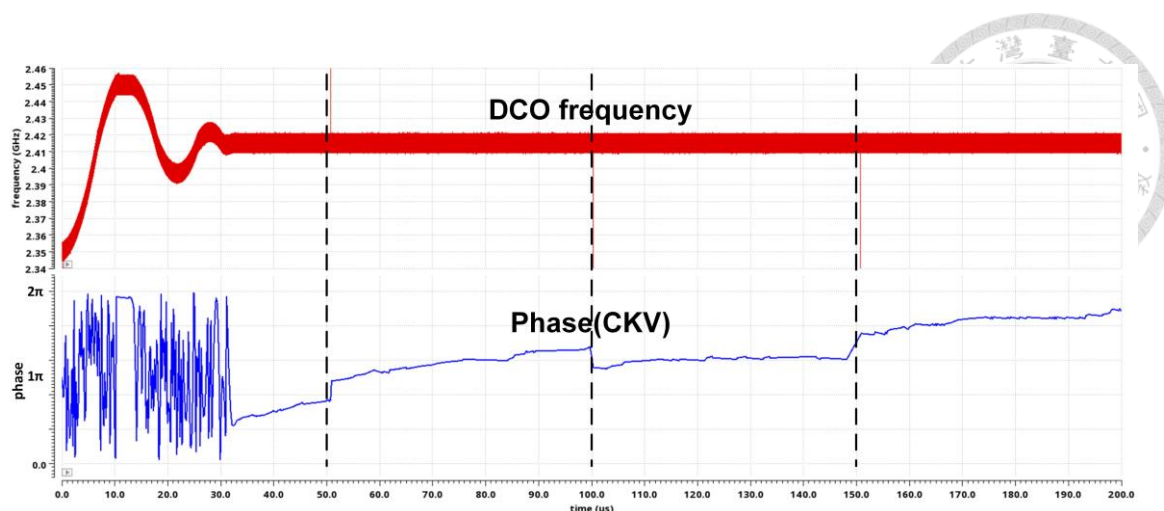


Fig. 6-8 Simulation result of behavior model

The first mechanism is related to the finite detection range of the TDC. The phase error caused by the injected noise exceeds the detection range of the TDC. As a result, part of the phase information is lost during the detection process, leading to a significant instantaneous phase shift.

The second mechanism is related to the finite resolution of the TDC. The frequency error which is smaller than the TDC resolution cannot be detected. However, the frequency error will cause the CKV phase to slowly shift. From the above discussion, it can be seen that without the help of other circuits, it is impossible to make an FLL function as a PLL.

6.3.7 Possible Solutions

Because the proposed architecture only has frequency-locking capability but not phase-locking. Maybe the time-registers can be applied in injection locked PLLs (ILPLL) [23] or multiply delay lock loops (MDLL) [24]. The former forces the oscillator phase to align with the reference signal by injecting a pulse. The latter zeros the phase error of the loop at each rising edge of the input clock. The time-register-based FLL can be applied to correct the frequency of ILPLL or MDLL.

Another method is designing a TDC with memory, such as gated ring oscillator TDC [25]. Unlike a conventional oscillator-based TDC, the GRO structure restricts the oscillator's transitions to occur only during a specific measurement period, effectively gating its operation. The objective is to maintain a frozen state of the ring oscillator between measurements. In other words, the time information is stored in the TDC in analog domain, the TDC can be seen as an integrator is connected in series with a simple TDC in Fig. 3-1. Hence the frequency error is turned to phase error and enter an oscillator-based TDC. From the point of view of the transfer function, this architecture is PLL.

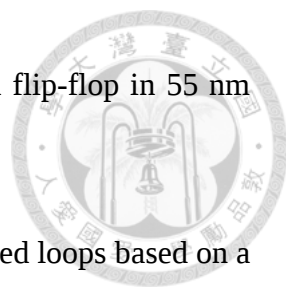


References



- [1] P. Larsson, "An offset-cancelled CMOS clock-recovery/demux with a half-rate linear phase detector for 2.5 Gb/s optical communication" *IEEE ISSCC*, pp. 74-75, Feb. 2001.
- [2] J. Zhu et al., "A 0.0021 mm² 1.82mW 2.2GHz PLL Using Time-based Integral Control in 65nm CMOS" *IEEE ISSCC*, pp. 338-340, Feb. 2016.
- [3] W. C. Lindsey et al., "A survey of digital phase-locked loops" in *Proceedings of the IEEE*, vol. 69, no. 4, pp. 410-431, Apr. 1981.
- [4] R. B. Staszewski et al., "Phase-Domain All-Digital Phase-Locked Loop" *IEEE TCAS-II*, vol. 52, no. 3, pp. 159-163, Mar. 2005.
- [5] J. Tangudu et al., "Quantization noise improvement of time to digital converter (TDC) for ADPLL" *IEEE ISCAS*, pp. 1020-1023, May, 2009.
- [6] J. Yu, et al., "A 12-Bit Vernier Ring Time-to-Digital Converter in 0.13 μ m CMOS Technology" *IEEE JSSC*, vol. 45, no. 4, pp. 830-842, Apr. 2010.
- [7] P. Madoglio, "Quantization effects in all-digital phase-locked loops" *IEEE TCAS-II*, vol. 43, no. 12, pp. 2776-2786, Dec. 2008.
- [8] G. Marzin et al., "A 20 Mb/s Phase Modulator Based on a 3.6 GHz Digital PLL With -36 dB EVM at 5 mW Power" *IEEE JSSC*, pp. 2974-2988, Dec. 2012.
- [9] C. M. Hsu et al., "A Low-Noise Wide-BW 3.6-GHz Digital $\Delta\Sigma$ Fractional-N Frequency Synthesizer With a Noise-Shaping Time-to-Digital Converter and

- Quantization Noise Cancellation" *IEEE JSSC*, vol. 43, no. 12, pp. 2776-2786, Dec 2008.
- [10] D. Liao et al., "An 802.11 a/b/g/n digital fractional-N PLL with automatic TDC linearity calibration for spur cancellation" *IEEE JSSC*, vol. 52, no. 5, pp. 391-400, May 2017.
- [11] S. Kim et al., "A 2 GHz synthesized fractional-N ADPLL with dual-referenced interpolating TDC" *IEEE JSSC*, vol. 43, no. 12, pp. 391-400, Feb 2016.
- [12] C. R. Ho et al., "A fractional-N digital PLL with background-dither-noise-cancellation loop achieving <-62.5 dBc worst-case near-carrier fractional spurs in 65nm CMOS" *IEEE ISSCC*, pp. 394-396, Feb. 2018.
- [13] Z. Gao et al., "A 2.6-to-4.1GHz Fractional-N Digital PLL Based on a Time-Mode Arithmetic Unit Achieving -249.4 dB FoM and -59 dBc Fractional Spurs" *IEEE ISSCC*, pp. 380-382, Feb. 2022.
- [14] P. Chen et al., "A $31\text{-}\mu$ W, 148-fs Step, 9-bit Capacitor-DAC-Based Constant-Slope Digital-to-Time Converter in 28-nm CMOS" *IEEE JSSC*, vol. 54, no. 11, pp. 3075-3085, Nov. 2019.
- [15] Z. Gao et al., "A 2.6-to-4.1GHz Fractional-N Digital PLL Based on a Time-Mode Arithmetic Unit Achieving -249.4 dB FoM and -59 dBc Fractional Spurs" *IEEE ISSCC*, pp. 380-382, Feb. 2022.
- [16] S. M. Dartizio et al., "A 76.7 fs-Integrated-Jitter and -71.9 dBc In-Band Fractional-Spur Bang-Bang Digital PLL Based on an Inverse-Constant-Slope DTC and FCW Subtractive Dithering" *IEEE ISSCC*, pp. 3-5, Feb. 2023.

- 
- [17] H. You et al., "A low-power high-speed sense-amplifier-based flip-flop in 55 nm MTCMOS" *Electronics*, May 2020.
- [18] V. Kratyuk et al., "A design procedure for all-digital phase-locked loops based on a charge-pump phase-locked-loop analogy" *IEEE TCAS-II*, vol. 54, no. 3, pp. 247-251, Mar. 2007.
- [19] M. Lee et al., "A 0.0043-mm² 0.3–1.2-V frequency-scalable synthesized fractional-N digital PLL with a speculative dual-referenced interpolating TDC" *IEEE JSSC*, vol. 56, no. 1, pp. 99-108, Jan. 2019.
- [20] K. Kwon et al., "Open-Source Fully-Synthesizable ADPLL for a Bluetooth Low-Energy Transmitter in 12nm FinFET Technology" *IEEE RFIC*, pp. 155-158, June 2022.
- [21] S. J. Kim et al., "A 0.6 V 1.17 ps PVT-tolerant and synthesizable time-to-digital converter using stochastic phase interpolation with 16× spatial redundancy in 14nm FinFET technology" *IEEE ISSCC*, pp. 1-3, Feb. 2015.
- [22] N. Pavlovic et al., "A 5.3 GHz digital-to-time-converter-based fractional-N all-digital PLL" *IEEE ISSCC*, pp. 394-396, Feb. 2018.
- [23] B. Razavi, "A study of injection locking and pulling in oscillators" *IEEE JSSC*, vol. 39, no. 9, pp. 1415-1424, Sep. 2004.
- [24] R. Farjad-Rad et al., "A low-power multiplying DLL for low-jitter multigigahertz clock generation in highly integrated digital chips" *IEEE JSSC*, vol. 37, no. 12, pp. 1804-1812, Dec. 2002.
- [25] Z. Matthew et al., "A Multi-Path Gated Ring Oscillator TDC With First-Order Noise Shaping" *IEEE JSSC*, vol. 44, no. 4, pp. 1089-1098, Apr. 2009.