**Master's Thesis**

# Design of ADPLL with Proportional and Integral Gain Co-Optimization Technique

## 비례 이득값과 적분 이득값의 동시 최적화 기술을 사용하는 ADPLL 의 설계

**by**

**Kyoungjoon Ha**

**August, 2023**

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

# Design of ADPLL with Proportional and Integral Gain Co-Optimization Technique

지도 교수 정 덕 균

이 논문을 공학석사 학위논문으로 제출함

**2023 년 8 월**

서울대학교 대학원

전기·정보공학부

하 경 준

하경준의 석사 학위논문을 인준함

**2023 년 8 월**

위 원 장 _____ (인)

부위원장 _____ (인)

위 원 _____ (인)

# Design of ADPLL with Proportional and Integral Gain Co-Optimization Technique

by

Kyoungjoon Ha

A Thesis Submitted to the Department of
Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for the Degree of
Master of Science
at
SEOUL NATIONAL UNIVERSITY

August, 2023

Committee in Charge:

Professor Su-Hwan Kim, Chairman

Professor Deog-Kyoon Jeong, Vice-Chairman

Professor Woo-Seok Choi

# Abstract

Noise performance of a PLL is an important factor to consider when designing a PLL. The unwanted variation in the timing clock edges can deteriorate system performance. For high jitter performance, proportional and integral gain of a PLL should be an optimum value. However, it is hard to choose the gains in the design step due to limited information about the noise environment. Therefore, the optimum values of proportional and integral gain must be tracked adaptively. This thesis proposes ADPLL which can optimize proportional and integral gain adaptively.

In this thesis, detailed analysis of the optimization technique is done. Based on the analysis, the optimization technique is validated for its effectiveness. Also, circuit implementations of PLL blocks are provided with detailed explanation.

The proposed PLL generates 3.2GHz clock with 100MHz reference clock. Power consumption is 7.6mW at supply voltage of 1V. Integrated RMS jitter of 857fs is achieved. The proposed PLL was fabricated in 28-nm CMOS process and occupies an effective area of $0.033\text{mm}^2$.

**Keywords :** All-Digital Phase-Locked Loop(ADPLL), Gain Optimization, Stochastic Resonance, Flicker Noise, Bang-bang Phase Frequency Detector(BBPFD), Digital Loop Filter(DLF)

**Student Number :** 2021-27376

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation

Phase-Locked Loops(PLLs) are widely used to generate clock signals for various applications such as wireline communication, CDR, and modern microprocessors. Jitter of the PLL is an important factor to consider when designing a PLL. The unwanted variation in the timing clock edges can deteriorate system performance. For example, jitter of a clock can cause bit errors in wireline transmitters. Also, it can cause timing violations in microprocessor. Therefore, it is important for PLL to have good jitter performance.

For optimal jitter performance, proportional and integral gains should be set as appropriate value. PLL with large proportional gain can filter out oscillator noise by tracking a reference clock. Conversely, a PLL with small loop gain can filter out reference clock noise [1]. Furthermore, PLL with large integral gain has undesirable

peaking at out-band. PLL with small integral gain cannot filter out low-frequency noise of the oscillator leading to high in-band noise. Thus, the optimal value of gains differs depending on the noise environment.

Conventionally, proportional and integral gains have been adjusted by the user after the design because noise environment is hard to predict. The designer would monitor the noise environment and choose the value of gains. However, this encounters the problem that the gains can be sub-optimal and the surrounding environment might vary in time. To things matter worse, gain of a bang-bang phase detector varies depending on jitter due to its non-linear property [2]. Unfortunately, the optimal value of loop gain is unknown priorly due to limited information of the noise environment.

Therefore, All-Digital Bang-Bang PLL that optimizes proportional and integral gains adaptively is proposed. The proposed PLL achieves -232.5dB of a figure-of-merit(FoM) with 857fs of RMS jitter and 7.6mW of power consumption.

## 1.2 Thesis Organization

This thesis is organized as follows. In Chapter 2, the backgrounds of a PLL and theoretical jitter analysis of a PLL is presented. The operation of each block and its schematics are explained.

In Chapter 3, the backgrounds of optimizing proportional and integral gains is presented. The method utilizes the statistical data of the bang-bang phase detector output. Also, it is mathematically analyzed for its effectiveness.

In Chapter 4, implementation of the PLL is presented. The schematics of bang-bang phase frequency detector, ring oscillator, digitally controlled resistor, and frequency divider are provided. Behavior and operation principles of the circuits are explained and implemented specs are given.

In Chapter 5, simulation results of the proposed PLL are presented. The phase noise and RMS jitter of the proposed are shown and compared to conventional PLL.

In Chapter 6, thesis is summarized and concluded.

# Chapter 2

# Backgrounds of All-Digital Phase-Locked Loop

## 2.1 Overview

Historically, ADPLL has received recognition because it can solve the problem of leakage current which was a huge trouble in CPPLL. ADPLL has developed rapidly thanks to technology development. Even though quantization noise is an inevitable problem, ADPLL can still perform in in various applications due to its benefits over CPPLL. Most of all, since internal signals in ADPLL is in digital form, these signals can be utilized and processed to achieve high performance. This characteristics of ADPLL leads to the huge potential of performance improvement and various architectures. In this thesis, digital output of phase detector is utilized to optimize jitter performance.

# 2.2 PLL Building Blocks

## 2.2.1 Overview

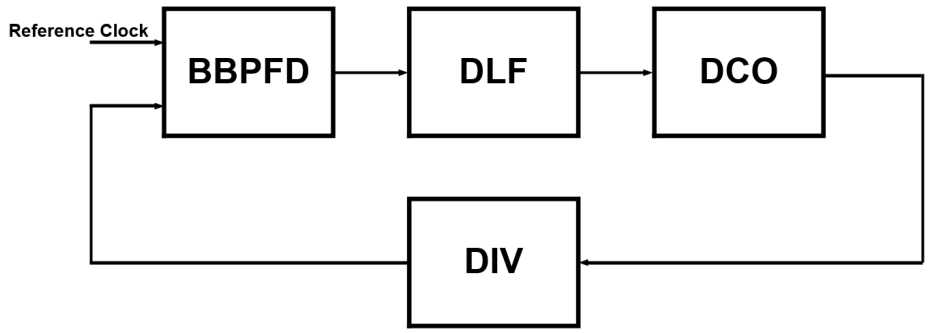



Figure 2.1 Basic block diagram of ADPLL

A basic block diagram of ADPLL is shown in Figure 2.1. ADPLL is composed of a phase frequency detector (PFD), digital loop filter (DLF), digitally controlled oscillator (DCO), and a frequency divider(DIV).

A phase frequency detector compares the phase and frequency of reference clock and feedback clock and generates an output accordingly. Specifically, the PFD remembers the clock edges it has been comparing and keeps comparing the following edges as long as phase error is less than $2\pi$. Digital loop filter is a synthesized digital block that processes the signal from BBPFD into a digital code that can control DCO. Digitally-controlled oscillator is an oscillator whose frequency is controlled by a digital code. It generates an output clock signal which is fed back to BBPFD through

frequency divider. Divider divides the frequency of the output clock signal by its preset dividing ratio. By changing the divide ratio one can choose the frequency of the output clock which is the multiplication of the frequency of the reference clock and the dividing ratio.

### 2.2.2 Bang-Bang Phase Frequency Detector

Phase frequency detector(PFD) is a circuit that compares the phase and frequency of two clock signals. Typical PFD consists of one AND gate and two D-flip flops as shown in Figure 2.2. When rising edge of reference clock(REFCLK) precedes that of divided clock(DIVCLK), UP signal is set HIGH. Soon after, when rising edge of divided clock arrives, DN signal is set HIGH. As UP and DN signals are set high, both flip flops are reset bringing the system back to initial state. This is the typical scenario how PFD will behave in locked situations. Figure 2.3 depicts this behavior.

On the other hand, in unlocked situations, two or more subsequent rising edges can arrive. However, only the first arriving edge will affect of the state of the system and other edges will be ignored. That is, PFD remembers the edges it has been comparing. This behavior is depicted in Figure 2.4. The fifth rising edge of reference clock is still compared with fifth rising edge of divided clock even sixth rising edge weighs in. While phase detector(PD) simply compares the phase of nearest two edges, PFD also compares the frequencies not only the phases. This is why PFD-based PLL has much faster locking speed than PD-based PLL.

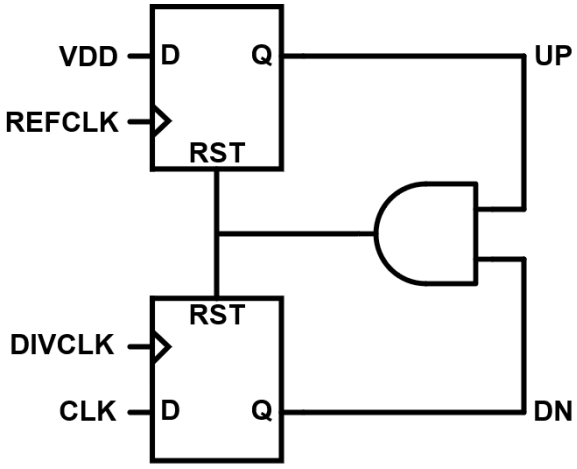



Figure 2.2 Schematic of Analog PFD

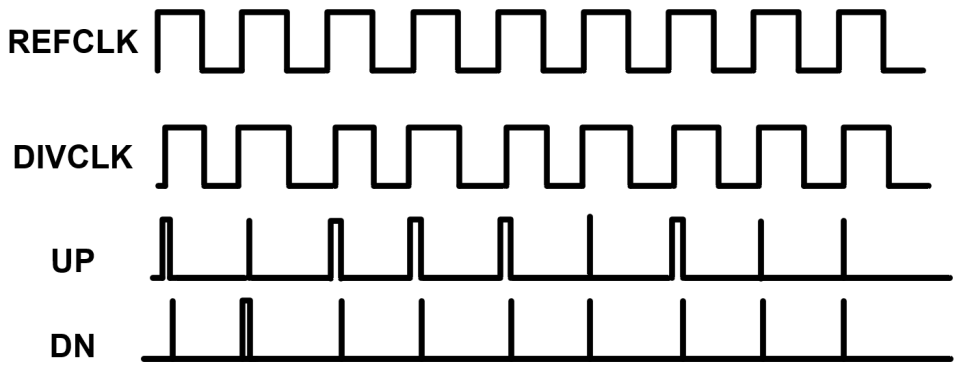



Figure 2.3 Behavior of PFD when PLL is locked

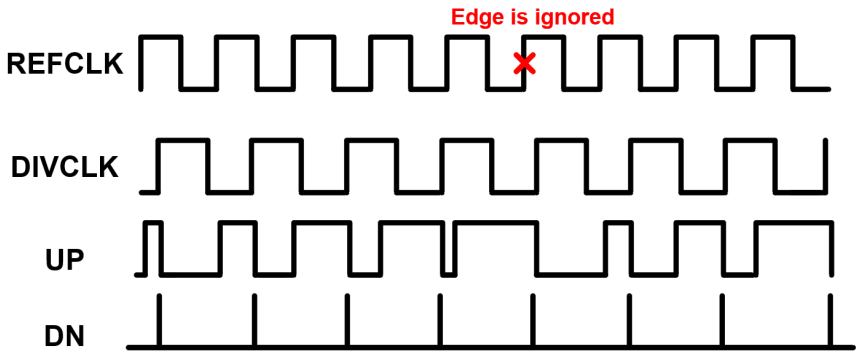



Figure 2.4 Behavior of PFD when cycle slipping occurs

Bang-Bang Phase Frequency Detector(BBPFD) is nothing but PFD that outputs signals in bang-bang form. While pulse width of outputs of PFD implies the amount of phase error, outputs of BBPFD contain only early-late information. Timing diagram of BBPFD operation is depicted in Figure 2.5. Note that output signals contain only early-late information because they are in bang-bang form.

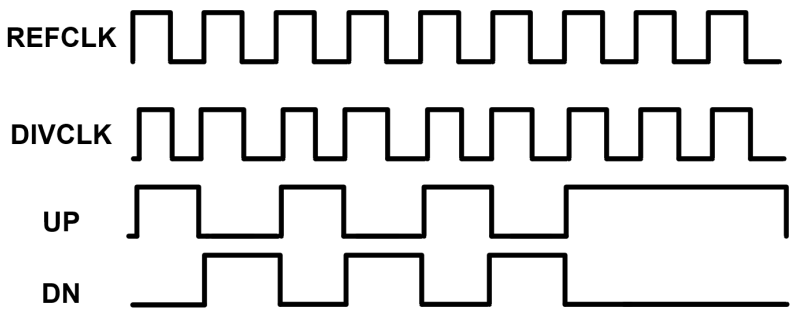



Figure 2.5 Behavior of BBPFD

### 2.2.3 Digitally Controlled Oscillator

Digitally Controlled Oscillator(DCO) is a clock generator whose frequency is controlled by a digital code. Usually, frequency is proportional to digital code. Frequency – Digital code plot is shown in Figure 2.6. DCO gain($K_{DCO}$) is defined as frequency change($\Delta f$) per one LSB change of the digital code. DCO is critical in PLL performance and there are various factors to consider when designing a DCO: frequency range, power consumption, and noise performance.

First, frequency range must be considered when designing a DCO. Frequency range must be set carefully set so that target frequency is included with enough margin. However, target frequency might deviate from frequency range due to PVT variations. Therefore, DCO gain must be set large with enough margin so that target frequency is inside the frequency range. But larger DCO gain leads to more deterministic jitter so DCO gain should be chosen carefully.

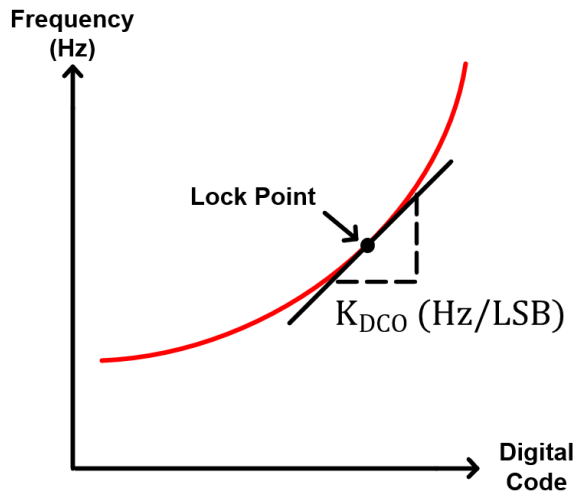



Figure 2.6 Digital Code – Frequency plot of DCO

Second, power consumption should also be considered when designing an oscillator. Power consumption of an oscillator accounts for most of the power consumption of PLL. Usually, power consumption and jitter are inversely proportional as depicted in Figure 2.7 [3]. That is, power and jitter performance have a trade-off relationship. However, this trade-off can be mitigated depending on the architecture of an oscillator. For example, ring oscillator has inferior jitter performance and tend to consume more power compared to LC oscillator. Even though, ring oscillator is widely used due to its capability of generating multi-phase clocks, large tuning range, and small chip area.

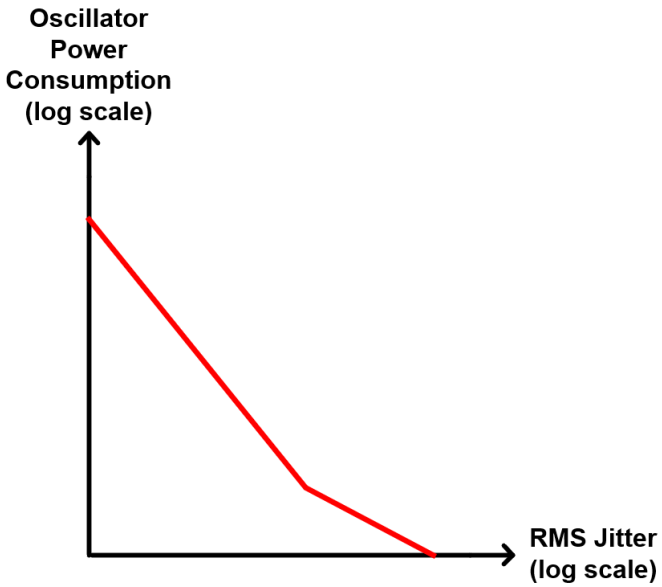



Figure 2.7 Power consumption versus Jitter

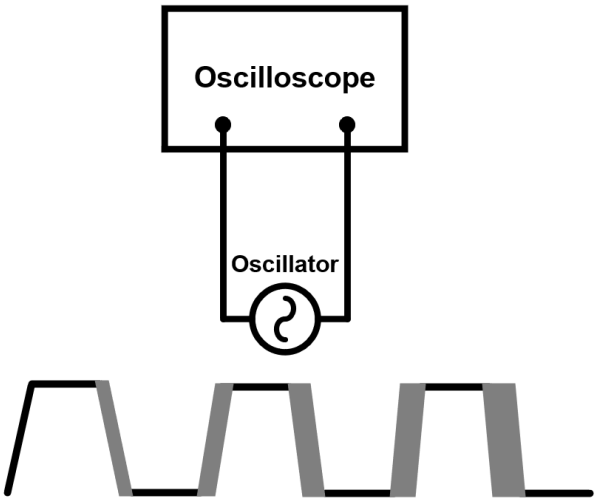



Figure 2.8 Accumulating jitter

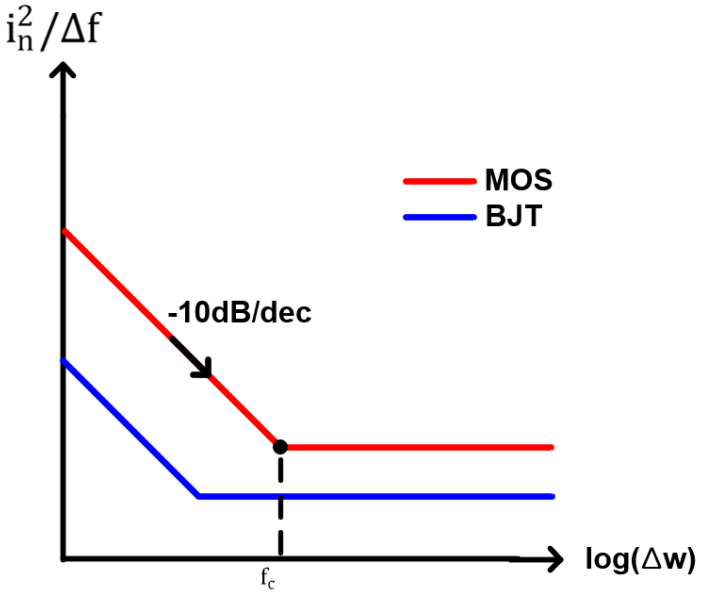



Figure 2.9 Flicker noise and thermal noise of MOSFETs and BJTs

Lastly, noise performance is also a critical factor that should be considered when designing an oscillator. Jitter has a property that it is accumulated throughout time as shown in Figure 2.8. Jitter of an oscillator can originate from various factors such as thermal noise in MOSFETs, shot noise in BJTs, and flicker noise of devices. Flicker noise and thermal noise of MOSFETs and BJTs are depicted in Figure 2.9. Flicker corner frequency denoted as $f_c$ is the frequency at which power spectral density of thermal noise and flicker noise is the same. Due to accumulating property of jitter, flicker noise and thermal noise of the devices are transformed into $1/f^3$ and $1/f^2$ region in the phase noise plot of an oscillator which is shown in Figure 2.10. $1/f^3$ noise is dominant in low frequency region and it appears as slow frequency drift in PLL output.

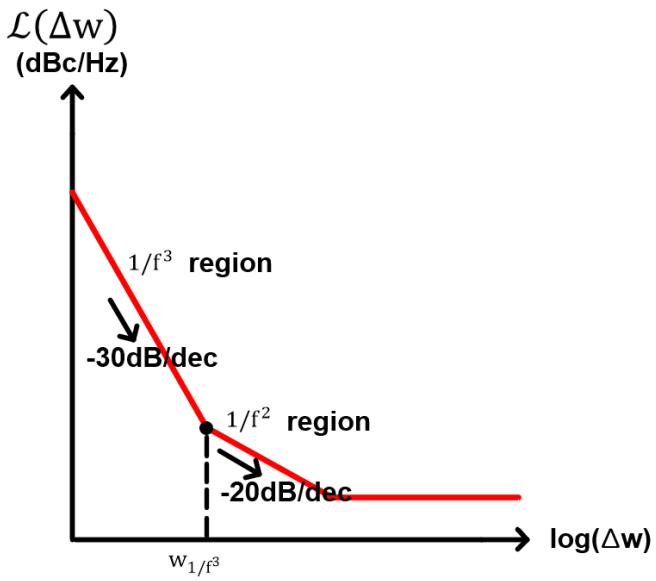



Figure 2.10 Phase noise plot of an oscillator

### 2.2.5 Digital Loop Filter

Loop Filter is a critical component in PLL that performs the shaping of the output of the phase detector. Digital loop filter(DLF) is a loop filter that is digitally synthesized to behave like loop filter in digital domain. Block diagram of typical DLF is shown in Figure 2.11. DLF consists of proportional path and integral path.

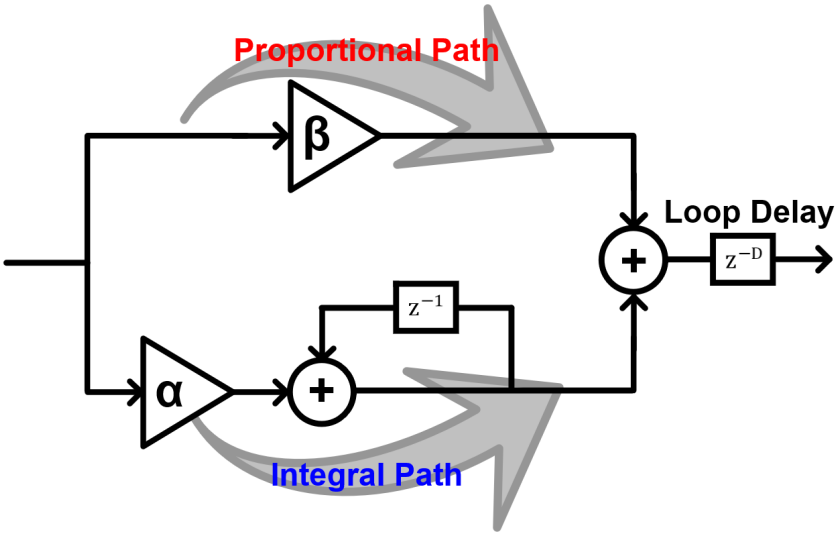



Figure 2.11 Block Diagram of Digital Loop Filter

Integral path integrates the phase error between the reference clock and divided clock every cycle. The phase error is accumulated in the form of digital code. By accumulating the instant phase errors, integral path tracks the frequency error. Integral gain α can be thought as a magnitude of sensitivity reacting to frequency mismatch of reference clock and divided clock.

On the other hand, proportional path tracks instant phase error by simply delivering phase error information to the oscillator. Proportional gain $\beta$ can be thought as a magnitude of sensitivity reacting to instant timing mismatch of reference clock and divided clock.

Loop filter delay due to pipelining introduced in the actual implementation is represented as $D$. Large loop delay means that calibration of phase error is delayed. Intuitively, large loop delay degrades the jitter performance of PLL. Actually, loop delay in BBPLL has been analyzed mathematically in time domain and concluded that it deteriorates jitter performance [4]. Specifically in first-order PLL, variance $\sigma_\tau$ and peak-to-peak jitter $\tau_{pp}$ relates with $D$ as follows.

$$\sigma_\tau^2 = \frac{(D+1)^2}{3} \tag{2.1}$$

$$\tau_{pp} = 2(D+1) \tag{2.2}$$

Therefore, loop filter delay must be minimized when designing a loop filter.

### 2.2.6 Frequency Divider

A frequency divider divides the frequency of the output clock of the PLL. Dividing ratio $N$ determines the PLL output frequency and satisfies the following equation where $f_{out}$, $f_{ref}$ represents the frequency of the output clock and reference clock of the PLL respectively.

$$f_{out} = N \cdot f_{ref} \tag{2.3}$$

Dividing ratio $N$ effects the overall loop dynamics of the PLL so it should be taken in to account when performing loop analysis.

Phase noise of the divider output is dominated by white noise and normally it is not critical to PLL jitter performance. The divider output phase noise is low-pass filtered through out the PLL path. However, retiming flipflop can be used to minimize the phase noise of the divider output [3].

# 2.3 Noise Analysis

Noise analysis of ADPLL now to be done was proposed in [5]. Since PLL is discrete-time sampling system, noise analysis can be performed in z-domain. Discrete-time model of the ADPLL is shown in Figure 2.12. $K_{BBPFD}$ is the gain of the BBPFD; $\alpha$ and $\beta$ are the integral and proportional gains respectively; N is the dividing ratio of the divider, and $K_T$ is the period gain of the DCO. Main sources of the noise are reference clock noise, quantization noise of BBPFD, quantization noise of DCO, and random noise of DCO which are denoted as $J_{REF}$, $J_{Q_{TDC}}$, $J_{Q_{DCO}}$, and $J_{DCO}$ respectively. Jitter of an output clock is denoted as $J_{PLL}$.

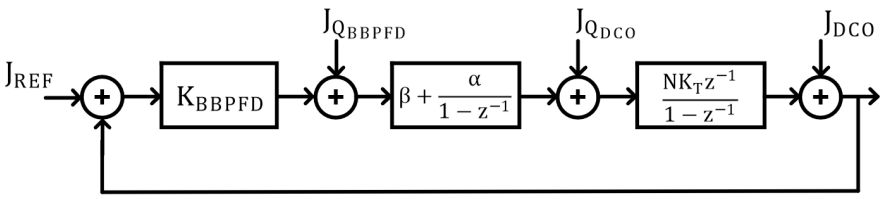



Figure 2.12 Discrete-time noise model of the ADPLL

Loop gain of the PLL $\epsilon$ is defined as $K_{BBPFD}N\beta K_T$ which is used often throughout noise analysis. Using Mason's Rule, the noise transfer function from $J_{DCO}$ to $J_{PLL}$ can be derived. Here, $J_{PLL,DCO}$ is defined as output jitter induced by random noise of the DCO.

$$\frac{J_{PLL,DCO}}{J_{DCO}} = \frac{(1-z^{-1})^2}{1+\left\{\epsilon\left(1+\frac{\alpha}{\beta}\right)-2\right\}z^{-1}+(1-\epsilon)z^{-2}} \tag{2.4}$$

Since in typical PLL $\alpha \ll \beta$, the above transfer function can be approximated.

$$\frac{J_{PLL,DCO}}{J_{DCO}} = \frac{1-z^{-1}}{1-(1-\epsilon)z^{-1}} \tag{2.5}$$

This derivation can similarly be applied to other noise as follows.

$$\frac{J_{PLL,REF}}{J_{REF}} = \frac{\epsilon z^{-1}}{1-(1-\epsilon)z^{-1}} \tag{2.6}$$

$$\frac{J_{PLL,Q_{BBPFD}}}{J_{Q_{TDC}}} = \left(\frac{1}{K_{TDC}}\right)\frac{\epsilon z^{-1}}{1-(1-\epsilon)z^{-1}} \tag{2.7}$$

$$\frac{J_{PLL,Q_{DCO}}}{J_{Q_{DCO}}} = \frac{NK_T z^{-1}}{1-(1-\epsilon)z^{-1}} \tag{2.8}$$

Here, $J_{PLL,REF}$, $J_{PLL,Q_{BBPFD}}$, and $J_{PLL,Q_{DCO}}$ represent output noise of PLL induced by reference clock noise, quantization noise of BBPFD, and quantization noise of DCO respectively. Therefore, total PLL output noise can be written as

$$J_{PLL} = \frac{1 - z^{-1}}{1 - (1 - \epsilon)z^{-1}} J_{DCO} + \frac{\epsilon z^{-1}}{1 - (1 - \epsilon)z^{-1}} J_{REF} + \left(\frac{1}{K_{TDC}}\right) \frac{\epsilon z^{-1}}{1 - (1 - \epsilon)z^{-1}} J_{Q_{TDC}} + \frac{N K_T z^{-1}}{1 - (1 - \epsilon)z^{-1}} J_{Q_{DCO}} \quad (2.9)$$

Random noise of the DCO can be modeled as a unit step function with random magnitude [6]. Intuitively, this is due to the accumulating nature of an oscillator. Furthermore, since the power spectral density of reference clock is uniform, jitter of the reference clock can be modeled as a pulse function with random magnitude [6]. Similarly, quantization noise also has a uniform power spectral density, thus can be modeled as a pulse function. Therefore, $J_{DCO}, J_{REF}, J_{Q_{TDC}}$, and $J_{Q_{DCO}}$ can be written as

$$J_{DCO} = \frac{\delta_{\tau_{DCO},k}}{1 - z^{-1}} \quad (2.10)$$

$$J_{REF} = \delta_{\tau_{REF},k} \quad (2.11)$$

$$J_{Q_{BBPFD}} = \delta_{Q_{BBPFD},k} \quad (2.12)$$

$$J_{Q_{DCO}} = \delta_{Q_{DCO},k} \quad (2.13)$$

Here, $\delta_{\tau_{DCO},k}$ and $\delta_{\tau_{REF},k}$ are magnitude of jitter caused by random noise of DCO and reference clock at kth reference cycle respectively. $\Delta_{Q_{BBPFD},k}$ and $\delta_{Q_{DCO},k}$ are magnitude of jitter caused by quantization noise of BBPFD and DCO at kth reference cycle respectively.

Substituting Equation (2.10) ~ (2.13) into Equation (2.5) ~ (2.8) and then taking inverse z transformation yields

$$J_{PLL,DCO}[nT] = \sum_{k=-\infty}^{n} \delta_{\tau_{DCO},k} \cdot (1-\epsilon)^{n-k} \cdot u[(n-k)T] \tag{2.14}$$

$$J_{PLL,REF}[nT] = \sum_{k=-\infty}^{n-1} \delta_{\tau_{REF},k} \cdot \epsilon(1-\epsilon)^{n-k-1} \cdot u[(n-k-1)T] \tag{2.15}$$

$$J_{PLL,Q_{BBPFD}}[nT] = \sum_{k=-\infty}^{n-1} \frac{\delta_{Q_{BBPFD},k}}{K_{BBPFD}} \cdot \epsilon(1-\epsilon)^{n-k-1} \cdot u[(n-k-1)T] \tag{2.16}$$

$$J_{PLL,Q_{DCO}}[nT] = \sum_{k=-\infty}^{n-1} \delta_{Q_{DCO},k} \cdot NK_T(1-\epsilon)^{n-k-1} \cdot u[(n-k-1)T] \tag{2.17}$$

where $J_{PLL,DCO}[nT]$, $J_{PLL,REF}[nT]$, $J_{PLL,Q_{BBPFD}}[nT]$, and $J_{PLL,Q_{DCO}}[nT]$ represent the accumulated jitter until $nT$ caused by $J_{DCO}, J_{REF}, J_{Q_{BBPFD}}$, and $J_{Q_{DCO}}$.

Taking expectations of square of Equation (2.14) ~ (2.17) yields

$$\mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,DCO}}[\mathrm{nT}]\right] = \frac{1}{\epsilon(2-\epsilon)}\delta^2_{\tau_{\mathrm{DCOrms}}} \tag{2.18}$$

$$\mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,REF}}[\mathrm{nT}]\right] = \frac{\epsilon}{2-\epsilon}\delta^2_{\tau_{\mathrm{REFrms}}} \tag{2.19}$$

$$\mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,Q_{BBPFD}}}[\mathrm{nT}]\right] = \frac{\epsilon}{\mathrm{K}^2_{\mathrm{BBPFD}}(2-\epsilon)}\delta^2_{\mathrm{Q_{BBPFD}rms}} \tag{2.20}$$

$$\mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,Q_{DCO}}}[\mathrm{nT}]\right] = \frac{\mathrm{N}^2\mathrm{K}^2_{\mathrm{T}}}{\epsilon(2-\epsilon)}\delta^2_{\mathrm{Q_{DCO}rms}} \tag{2.21}$$

where $\delta_{\tau_{\mathrm{REF}}\mathrm{rms}}$, $\delta_{\tau_{\mathrm{DCO}}\mathrm{rms}}$, $\delta_{\mathrm{Q_{BBPFD}rms}}$, and $\delta_{\mathrm{Q_{DCO}rms}}$ are the RMS values of $\delta_{\tau_{\mathrm{REF}},\mathrm{k}}$, $\delta_{\tau_{\mathrm{DCO}},\mathrm{k}}$, $\delta_{\mathrm{Q_{BBPFD}},\mathrm{k}}$, and $\delta_{\mathrm{Q_{DCO}},\mathrm{k}}$ respectively.

Finally, total output clock jitter power can be found by adding Equations (2.18) ~ (2.21).

$$\begin{aligned}\mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL}}[\mathrm{nT}]\right] &= \mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,DCO}}[\mathrm{nT}]\right] + \mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,REF}}[\mathrm{nT}]\right] \\ &\quad + \mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,Q_{BBPFD}}}[\mathrm{nT}]\right] + \mathrm{E}\left[\mathrm{J}^2_{\mathrm{PLL,Q_{DCO}}}[\mathrm{nT}]\right] \\ &= \frac{1}{\epsilon(2-\epsilon)}\xi_{\mathrm{outband}} + \frac{\epsilon}{2-\epsilon}\xi_{\mathrm{inband}}\end{aligned} \tag{2.22}$$

where $\xi_{\mathrm{outband}}$ and $\xi_{\mathrm{inband}}$ are

$$\xi_{\mathrm{outband}} = \delta^2_{\tau_{\mathrm{DCOrms}}} + \mathrm{N}^2\mathrm{K}^2_{\mathrm{T}}\delta^2_{\mathrm{Q_{DCO}rms}} \tag{2.23}$$

$$\xi_{\mathrm{inband}} = \delta^2_{\tau_{\mathrm{REFrms}}} + \frac{\delta^2_{\mathrm{Q_{BBPFD}rms}}}{\mathrm{K}^2_{\mathrm{BBPFD}}} \tag{2.24}$$

It can be observed from Equation (2.22) that if $\epsilon$ increases, $\xi_{\text{outband}}$ will be more filtered out. On the other hand, if $\epsilon$ decreases, $\xi_{\text{inband}}$ will be more filtered out. Since reference clock noise and PFD noise are low pass filtered and DCO noise is high pass filtered by the loop, increasing the bandwidth of the PLL will result in filtering out more DCO noise and vice versa. This idea corresponds with aforementioned observation made from Equation (2.22).

Taking derivative with respect to $\epsilon$ of Equation (2.22) yields,

$$\epsilon_{\text{OPT}} = \frac{2}{1+\sqrt{1+4\frac{\xi_{\text{inband}}}{\xi_{\text{outband}}}}} \tag{2.25}$$

where $\epsilon_{\text{OPT}}$ is optimum value of loop gain($\epsilon_{\text{OPT}} = K_{\text{BBPFD}} N \beta_{\text{OPT}} K_T$) However, $\epsilon_{\text{OPT}}$ is hard to be determined priorly for several reasons. First, since BBPFD is a non-linear device, the gain of BBPFD $K_{\text{BBPFD}}$ is dependent on surrounding noise environment. Second, $\xi_{\text{inband}}$ and $\xi_{\text{outband}}$ are hard to predict due to complexity of noise environment. At last, period gain of DCO $K_T$ is PVT-dependent. Therefore, $\beta_{\text{OPT}}$ cannot be determined priorly so it must be tracked dynamically.

# Chapter 3

# Proportional and Integral Gain Co-Optimization

## 3.1 Overview

In this chapter, the method of proportional and integral gain co-optimization technique will be described. To start with, stochastic resonance which is the foundation of optimizing proportional gain will be explained. Next, the optimum value of loop gain will be derived. Based on the derivation, the method of optimizing proportional gain will be introduced. At last, the need for optimizing integral gain and the method will be explained.

# 3.2 Stochastic resonance

Stochastic resonance is a physical phenomenon in which performance metric of a nonlinear system can be enhanced in the presence of noise [7]. Stochastic resonance also can be exploited in BBPLLs to improve jitter performance.

As mentioned in Chapter 2, there are four main noise sources in BBPLLs : 1) quantization error induced by BBPFD, 2) quantization error induced by DCO, 3) random noise of reference clock, and 4) random noise of DCO. When quantization noise dominates the random noise, BBPLL enters a limit-cycle regime (see [4]). Reference spur stands out in the overall spectrum. On the other hand, if random noise dominates the quantization noise BBPLL enters a random-noise regime. In this regime, spurious tones are reduced but low-frequency noise stands out in the spectrum.

Recall from Equation 2.1 that limit cycle jitter can be expressed as

$$J_{lc} = N \cdot \beta \cdot \frac{K_T}{\sqrt{3}} \tag{3.1}$$

where $J_{lc}$ represents the limit cycle jitter and loop filter delay is zero. In [7], random noise jitter is derived as

$$J_{rn} = \sqrt{\frac{\pi}{8}} \cdot \frac{K_W T_0}{\beta^2 K_T^2 - \pi K_F T_0^2 \ln(\delta)} \cdot \beta K_T \tag{3.2}$$

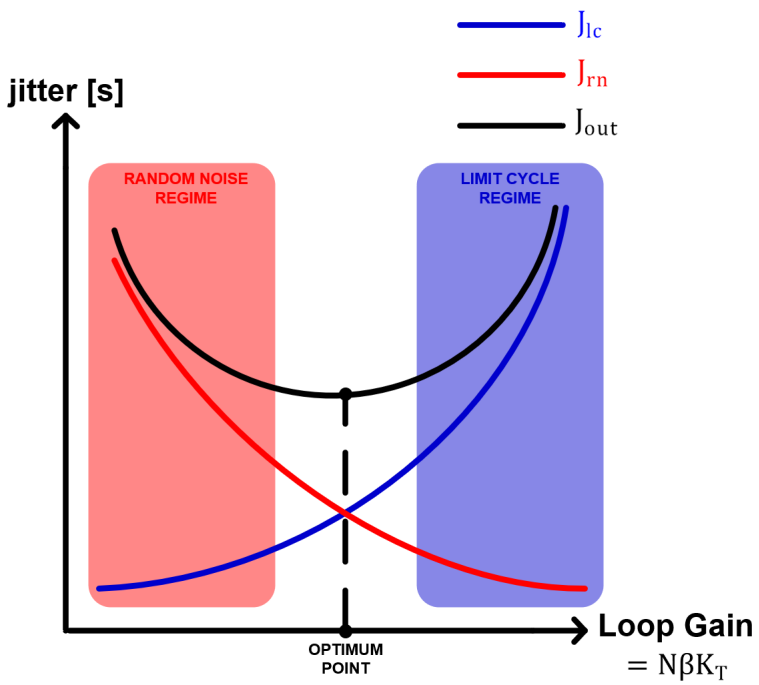



Figure 3.1 Plot of $J_{lc}$ and $J_{rn}$ and optimum point of $\beta$

where $J_{rn}$ represents the random noise jitter induced by DCO noise whose PSD is defined as $(K_W + K_F/f)/(2\pi f)^2$. Plot of $J_{lc}$ and $J_{rn}$ is shown in Figure 3.1 [7].

We can observe from Figure 3.1Figure 3.1 that too large loop gain will result in limit cycle regime and too small loop gain will result in random noise regime. There is an optimum point of loop gain where stochastic resonance takes place [7]. At this point, jitter induced by random noise and limit cycle is balanced and total jitter is minimized.

# 3.3 Optimizing Proportional Gain

In this chapter, the method of optimizing proportional gain will be mathematically derived. This derivation was introduced in [5].

From Figure 2.12, the transfer function from noise sources to BBPFD output is given as follows:

$$\frac{Y_{DCO}}{J_{DCO}} = \frac{K_{BBPFD}(1 - z^{-1})}{1 - (1 - \epsilon)z^{-1}} \tag{3.3}$$

$$\frac{Y_{REF}}{J_{REF}} = \frac{K_{BBPFD}(1 - z^{-1})}{1 - (1 - \epsilon)z^{-1}} \tag{3.4}$$

$$\frac{Y_{Q_{BBPFD}}}{J_{Q_{BBPFD}}} = \frac{1 - z^{-1}}{1 - (1 - \epsilon)z^{-1}} \tag{3.5}$$

$$\frac{Y_{Q_{DCO}}}{J_{Q_{DCO}}} = \frac{K_{BBPFD}NK_T z^{-1}}{1 - (1 - \epsilon)z^{-1}} \tag{3.6}$$

where $Y_{DCO}, Y_{REF}, Y_{Q_{BBPFD}}$, and $Y_{Q_{DCO}}$ represent the BBPFD output induced by $J_{DCO}, J_{REF}, J_{Q_{BBPFD}}$, and $J_{Q_{DCO}}$ respectively. Now, substituting Equation 2.10 ~ 2.13 into Equation 3.3 ~ 3.6 yields $Y_{DCO}, Y_{REF}, Y_{Q_{BBPFD}}$, and $Y_{Q_{DCO}}$ as follows:

$$Y_{DCO} = \frac{K_{BBPFD}\delta_{\tau_{DCO},k}}{1-(1-\epsilon)z^{-1}} \tag{3.7}$$

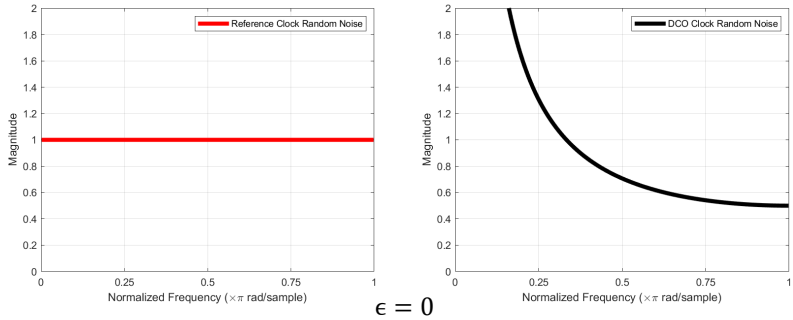
$$Y_{REF} = \frac{K_{BBPFD}\delta_{\tau_{REF},k}(1-z^{-1})}{1-(1-\epsilon)z^{-1}} \tag{3.8}$$

$$Y_{Q_{BBPFD}} = \frac{\delta_{Q_{BBPFD},k}(1-z^{-1})}{1-(1-\epsilon)z^{-1}} \tag{3.9}$$

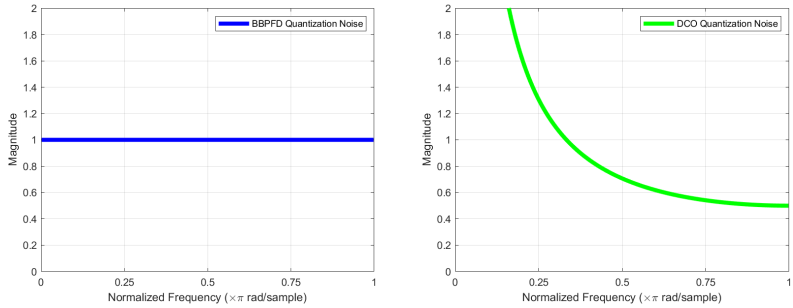
$$Y_{Q_{DCO}} = \frac{K_{BBPFD}NK_T\delta_{Q_{DCO},k}z^{-1}}{1-(1-\epsilon)z^{-1}} \tag{3.10}$$

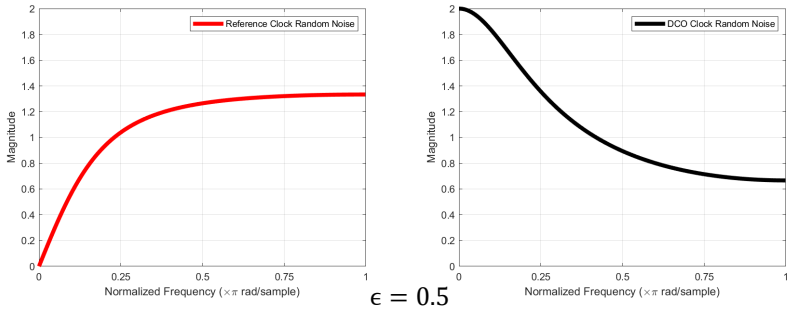
Frequency response of Equation 3.7 ~ 3.10 is plotted in Figure 3.2. Note that frequency response of Equation 3.7 ~ 3.10 are normalized to $K_{BBPFD}\delta_{\tau_{DCO},k}$, $K_{BBPFD}\delta_{\tau_{REF},k}$, $\delta_{Q_{BBPFD},k}$, and $K_{BBPFD}NK_T\delta_{Q_{DCO},k}$ respectively. It can be observed from Figure 3.2 that different frequency component is dominant depending on $\epsilon$. To be specific, when $\epsilon < 1$, low frequency component stands out in BBPFD output induced by DCO random noise. By contrast, high frequency component dominates when $\epsilon > 1$.

Furthermore, it can be observed that the frequency response has uniform density when $\epsilon$ is the optimum value. According to Equation 2.18 ~ 2.21, jitter induced by DCO random noise and DCO quantization noise is minimum when $\epsilon = 1$ while jitter induced by reference noise and BBPFD quantization noise is minimum when $\epsilon = 0$. Figure 3.2 (c) shows that BBPFD output induced by DCO random noise and DCO quantization noise shows a frequency response of uniform spectral density when $\epsilon = 1$. On the other hand, Figure 3.2 (a) indicates that BBPFD output induced by reference noise and BBPFD quantization noise shows a frequency response of uniform density when $\epsilon = 0$. Relying on this frequency distribution, loop gain can be optimized to minimize jitter.

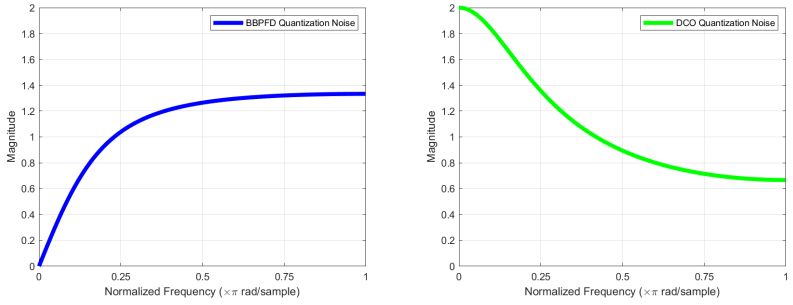



$\epsilon = 0$




$\epsilon = 0.5$



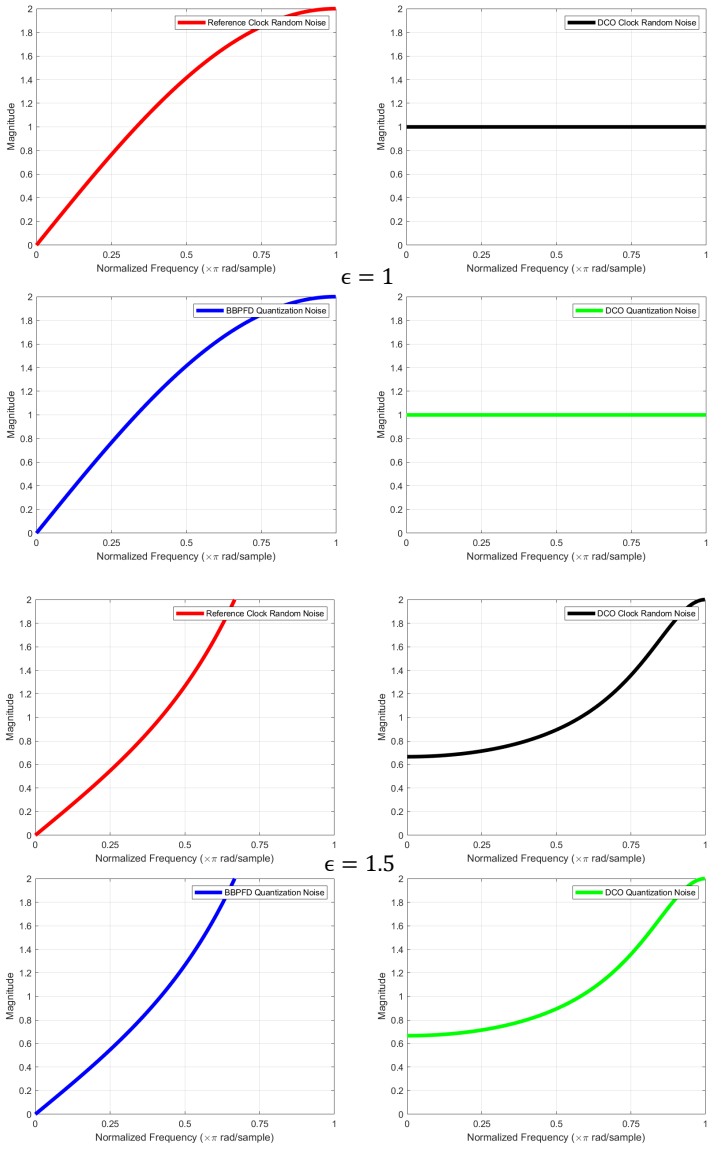



Figure 3.2 Plots of frequency response of BBPFD output induced by reference clock noise, DCO random noise, BBPFD quantization noise, and DCO quantization noise when (a) $\epsilon = 0$ (b) $\epsilon = 0.5$ (c) $\epsilon = 1$ (d) $\epsilon = 1.5$

Frequency can be discriminated by observing the autocorrelation of a signal. Autocorrelation of a signal $x(t)$ is defined as

$$\mathrm{Rxx(a)} = \mathrm{E}\left[\sum_{k=-\infty}^{n-a} (x[k]-\mu)(x[k+a]-\mu)\right] \tag{3.11}$$

where $\mu, \sigma,$ $n,$ and $a$ are the mean, standard deviation, the number of samples, and the order of autocorrelation respectively. In the case of our derivation, autocorrelation of BBPFD output $y(t)$ is as follows:

$$\mathrm{Ryy(a)} = \mathrm{E}\left[\sum_{k=-\infty}^{n} y[(k-a)T]\cdot y[kT]\right] \tag{3.12}$$

where $\mu = 0$ and $\sigma = 1$ is substituted into Equation 3.11 because BBPFD output is either 1(UP) or -1(DN) and the average number of 1 and -1 matches when PLL is locked.

Computing the first order autocorrelation of $Y_{DCO}$, $Y_{REF}$, $Y_{Q_{BBPFD}}$, and $Y_{Q_{DCO}}$ yields:

$$\mathrm{Ry_{DCO}y_{DCO}}(1) = \frac{K_{BBPFD}^2(1-\epsilon)}{\epsilon(2-\epsilon)}\delta_{\tau_{DCOrms}}^2 \tag{3.13}$$

$$\mathrm{Ry}_{\mathrm{REF}}\mathrm{y}_{\mathrm{REF}}(1) = \frac{-\mathrm{K}_{\mathrm{BBPFD}}^2\epsilon}{2-\epsilon}\delta_{\tau_{\mathrm{REFrms}}}^2 \tag{3.14}$$

$$\mathrm{Ry}_{\mathrm{Q_{BBPFD}}}\mathrm{y}_{\mathrm{Q_{BBPFD}}}(1) = \frac{-\epsilon}{2-\epsilon}\delta_{\mathrm{Q_{BBPFDrms}}}^2 \tag{3.15}$$

$$\mathrm{Ry}_{\mathrm{Q_{DCO}}}\mathrm{y}_{\mathrm{Q_{DCO}}}(1) = \frac{\mathrm{K}_{\mathrm{BBPFD}}^2\mathrm{N}^2\mathrm{K}_{\mathrm{T}}^2(1-\epsilon)}{\epsilon(2-\epsilon)}\delta_{\mathrm{Q_{DCOrms}}}^2 \tag{3.16}$$

Summing up Equation 3.13 ~ 3.16 yields:

$$\mathrm{Ryy}(1) = \frac{\mathrm{K}_{\mathrm{BBPFD}}^2[(1-\epsilon)\xi_{\mathrm{outband}} - \epsilon^2\xi_{\mathrm{inband}}]}{\epsilon(2-\epsilon)} \tag{3.17}$$

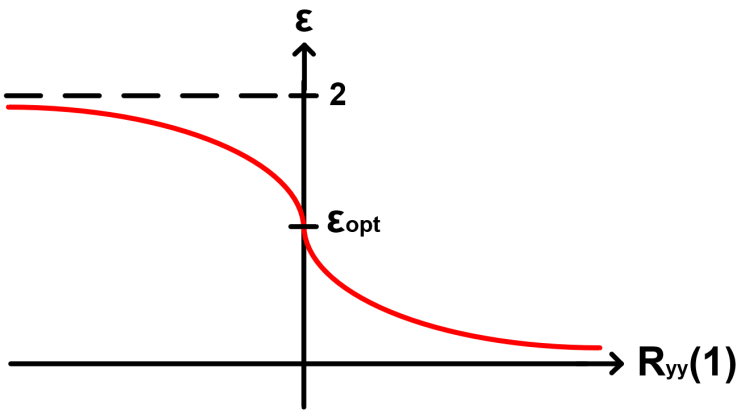



Figure 3.3 Plot of $\epsilon$ with respect to $\mathrm{Ryy}(1)$

Plot of $\epsilon$ with respect to $Ryy(1)$ is shown in Figure 3.3. $\epsilon$ is a monotonic-decreasing function with respect to $Ryy(1)$. Also, $\epsilon = \epsilon_{opt}$ when $Ryy(1) = 0$ because root of $(1-\epsilon)\xi_{outband} - \epsilon^2\xi_{inband} = 0$ is $2/(1+\sqrt{1+4\frac{\xi_{inband}}{\xi_{outband}}})$ which is the optimum value of $\epsilon$ as derived from Equation 2.25. That is, $\epsilon < \epsilon_{opt}$ when $R_{yy}(1) < 0$ and $\epsilon > \epsilon_{opt}$ when $R_{yy}(1) > 0$. Therefore, loop gain can be optimized by increasing $\beta$ when $R_{yy}(1) > 0$ and decreasing $\beta$ when $R_{yy}(1) < 0$.

To gain some intuition, $R_{yy}(1)$ can be rewritten in simplified form as follows:

$$R_{yy}(1) = E\left[\sum y[nT] \cdot y[(n+1)T]\right] \tag{3.18}$$

Since phase detector's output is proportional to the instant phase error, phase detector $y[nT]$ implies the magnitude of instant phase error. Thus, Equation 3.18 can be rewritten as:

$$\begin{aligned} R_{yy}(1) &= E\left[\sum y[nT] \cdot y[(n+1)T]\right] \\ &\propto E\left[\sum \theta_{error}[nT] \cdot y[(n+1)T]\right] \\ (in\ case\ of\ BBPFD) &\propto E\left[\sum \mathrm{sign}(\theta_{error}[nT]) \cdot y[(n+1)T]\right] \end{aligned} \tag{3.19}$$

From Equation 3.19, it can be shown that jitter is minimized when there is no correlation between the instant phase error and the subsequent BBPFD output.

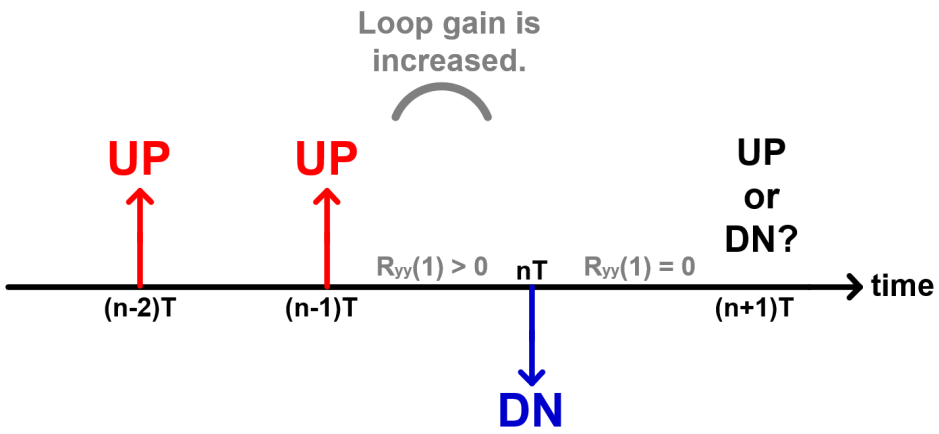



Figure 3.4 Brief behavior of loop gain optimization

Brief behavior of optimization technique is shown in Figure 3.4. Suppose BBPFD outputs two consecutive UPs at time $(n-2)T$ and $(n-1)T$. Since $Ryy(1) > 0$, loop gain will be increased. BBPFD output at time nT will tend to be DN due to the increased loop gain. However, since $Ryy(1) = 0$ at time nT, no further optimization will take place leaving more freedom for optimization. At time nT, PLL operates in stochastic resonance region making BBPFD output at time $(n+1)T$ completely random(i.e. probability of UP and DN is 0.5 and 0.5 each). Two questions arise at this point. Is probability of UP and DN at $(n+1)T$ really 0.5 and 0.5? If not, what optimization can be done at nT?

# 3.4 Optimizing Integral Gain

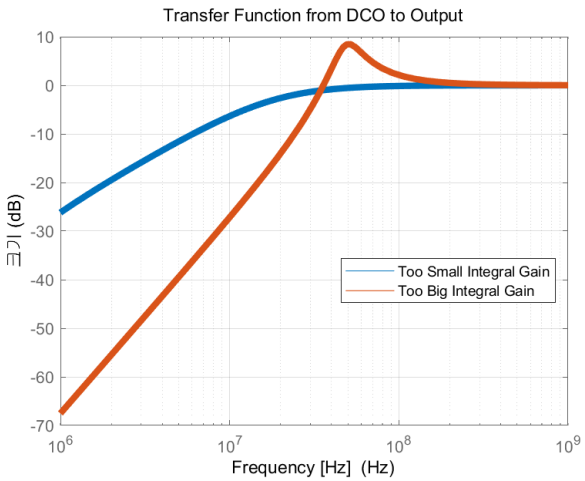



Figure 3.5 Transfer Function from DCO to Output

Transfer function from the oscillator to output can be expressed as follows:

$$H_{OSC}(s) = \frac{s^2}{s^2 + 2\zeta w_n s + w_n^2}$$

$$w_n = \sqrt{K_{pd}K_{vco}K_I},\ \zeta = 0.5K_P\sqrt{\frac{K_{pd}K_{vco}}{K_I}}.$$

Bode plot of the above transfer function is shown in Figure 3.5. PLL with small integral gain is an overdamped system leading to 40dB roll-off near the bandwidth. On contrast, PLL with

large integral gain is an underdamped system leading to phase noise amplification due to peaking. As shown in Figure 3.5, loop with large integral gain can effectively filter out low frequency noise of an oscillator but phase noise amplification takes place due to stability. Loop with small integral gain is stable enough but cannot effectively filter out low frequency noise of an oscillator. There will be an optimum value of integral gain between the two where overall jitter performance is optimal.

Low frequency noise of an oscillator appears as frequency drift in output [8, 9]. Frequency drift $f_D$ can be observed as temporal difference of BBPFD outputs.

$$\mathrm{sign}\left(\theta_{f_d}[nT]\right) \propto (y[nT] - y[(n-1)T]) \tag{3.20}$$

where $\theta_{f_d}[nT]$ is a phase error at time $nT$ due to frequency drift. As mentioned in Chapter 3.3, jitter is minimized when there is no correlation between instant phase error and subsequent BBPFD output. Thus, jitter is minimized when:

$$\begin{gathered} E[\sum \theta_{f_d}[nT] \cdot y[(n+1)T]] = 0 \\ (in\ case\ of\ BBPFD)\ E\left[\sum \mathrm{sign}(\theta_{error}[nT]) \cdot y[(n+1)T]\right] = 0 \end{gathered} \tag{3.21}$$

Substituting Equation 3.18 into Equation 3.21 yields that jitter is minimized when:

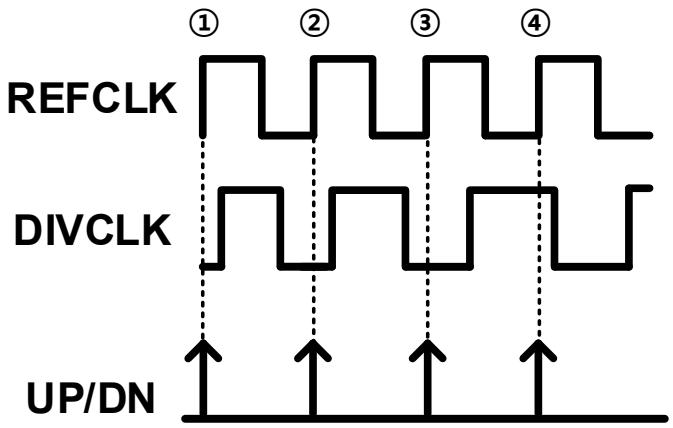
$$E\left[\sum (y[nT] - y[(n-1)T] \cdot y[(n+1)T])\right] = 0 \tag{3.22}$$

$E[\sum y[nT] \cdot y[(n+1)T]] = 0$ due to loop gain optimization technique described in Chapter 3.3. Thus, Equation 3.22 can be re-written as:

$$E[\sum y[(n-1)T] \cdot y[(n+1)T]] = R_{yy}(2) = 0 \quad (3.23)$$

It can be concluded that flicker noise can be minimized when $R_{yy}(2) = 0$ and this is valid only when loop gain optimization is in process. Now the answers to the aforementioned questions arise. From Figure 3.4, probability of UP and DN at $(n+1)T$ is not 0.5 due to frequency drift. Also, optimization can be done at time $nT$ so that $R_{yy}(2) = 0$.

Timing diagram when frequency drift occurs is shown in Figure 3.6 (a) and principle of detecting frequency drift is shown in Figure 3.6 (b).




(a)

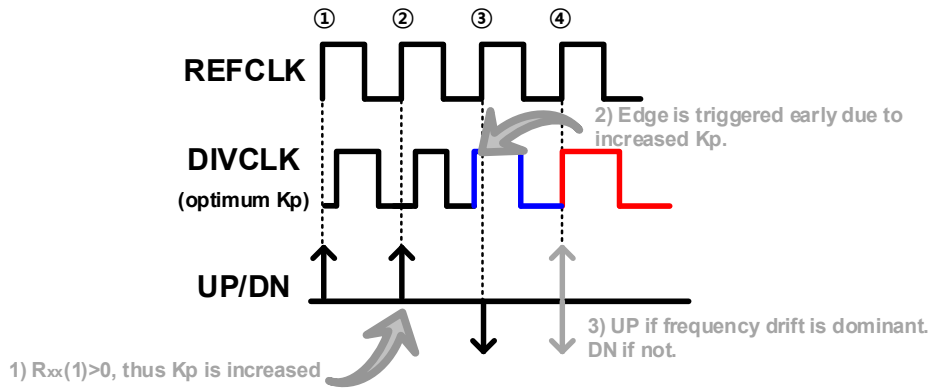


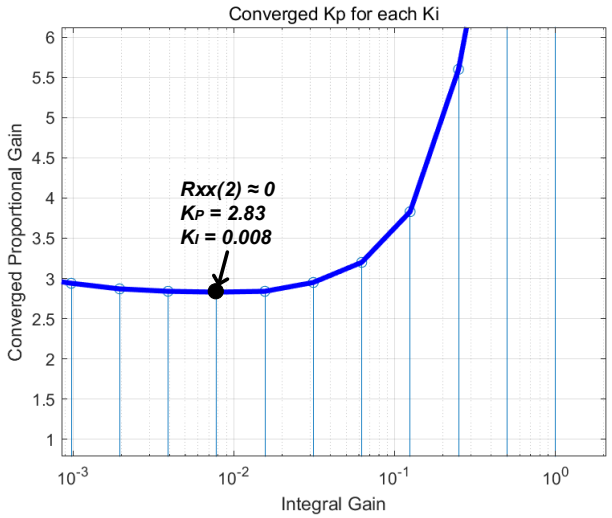

Figure 3.6 (a) Timing diagram when frequency drift occurs

(b) Detecting frequency drift by $R_{yy}(2)$

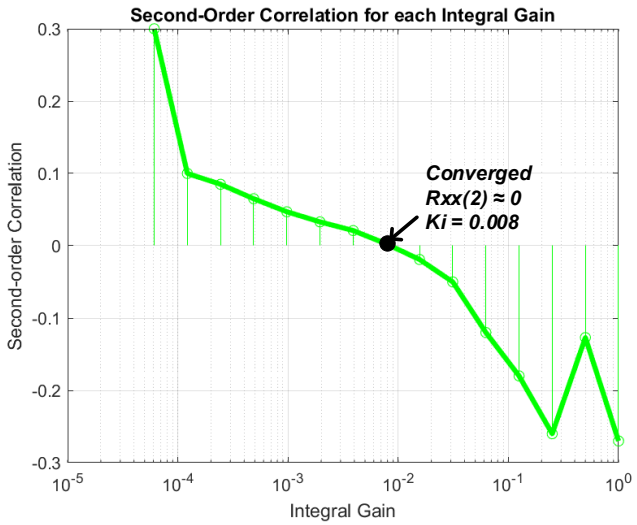
When frequency of output clock drifts to lower frequency, BBPFD will cast consecutive DNs as shown in Figure 3.6 (a). In this case, proportional gain optimization technique will increase the proportional gain at the second rising edge. So, it is likely that BBPFD will cast DN at the third edge making $R_{yy}(1) = 0$. At the fourth edge, REFCLK will precede DIVCLK if frequency drift is dominant compared to bandwidth set by proportional gain. In contrast, DIVCLK will precede REFCLK if frequency drift is not dominant. Therefore, it can be concluded that $R_{yy}(2) > 0$ if integral gain is too small and $R_{yy}(2) < 0$ if integral gain is too large.

Now, behavioral simulation was done to verify the above conclusion. The behavioral simulation was done by sweeping the value of integral gain while proportional gain is optimized adaptively. The converged value of proportional gain, average second-order correlation, and RMS jitter for each integral gain were measured. The simulation results are shown in Figure 3.7. From Figure 3.7 (b), it

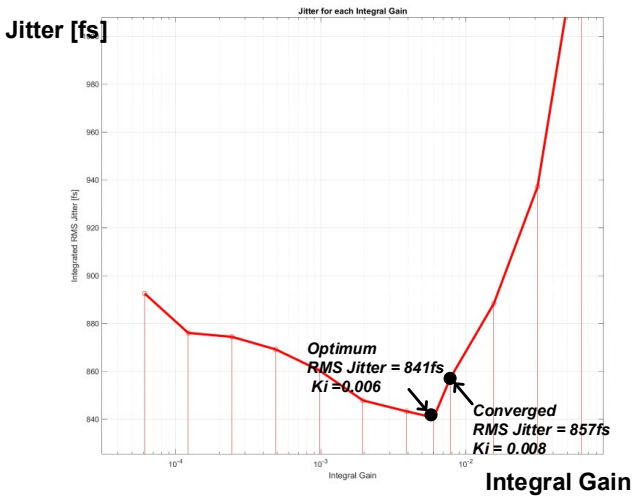
can be observed that $R_{yy}(2) = 0$ when integral gain is 0.008. In Figure 3.6 (c), jitter performance is the best at this point. It is verified that integral gain $\alpha > \alpha_{opt}$ when $R_{yy}(2) < 0$ and $\alpha < \alpha_{opt}$ when $R_{yy}(2) > 0$.




(a)




(b)




(c)

Figure 3.7 Plots of behavioral simulation results done by sweeping the integral gain

(a) Optimized value of proportional gain (b) Second-order autocorrelation (c) RMS Jitter

# 3.5 Implementation

## 3.5.1 Implementation of Gain Optimizer

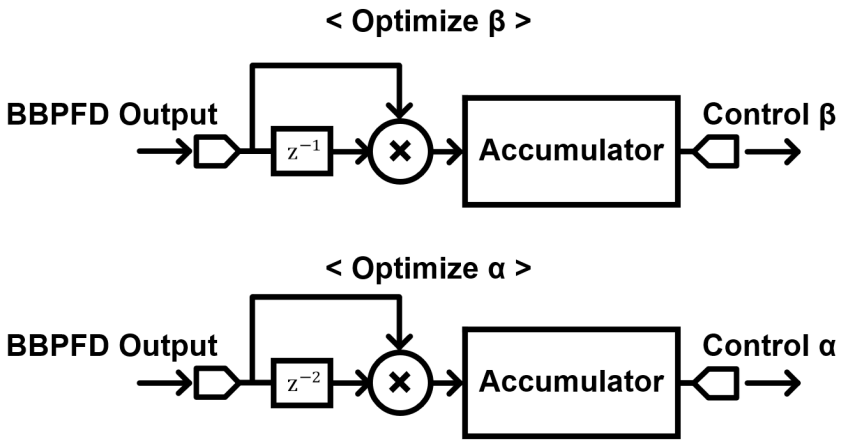



Figure 3.8 Implementation of gain optimizer

The conclusion of Chapter 3.3 was that proportional gain β can be optimized by increasing β when positive first-order autocorrelation is observed and vice versa. The conclusion of Chapter 3.4 was that integral gain α can be optimized by increasing α when positive second-order correlation is observed and vice versa.

Implementation of gain optimizer is shown in Figure 3.8. This implementation was priorly proposed in [9] and [10]. Averaging part when computing autocorrelation is not implemented since we only need polarity information. β and α are controlled in such a way that β and α are proportional to the accumulated value. That is, positive autocorrelation will increase the value stored in accumulator thus increasing

the gain. In contrast, negative autocorrelation will decrease the value stored in accumulator which will lead to decrease of the gain

.

### 3.5.2 Implementation of Gain Path

In most digital PLLs, proportional and integral gains are implemented only in exponential form(i.e. $2^n$) because bit-shifting is cheap in power. However, to maximize the effect of gain optimization technique, it is beneficial to implement gains in mantissa and exponential form. In other words, $\beta$ and $\alpha$ should be implemented as:

$$\beta = \beta_{\text{int}} \times 2^{\beta_{\text{exp}}} \tag{3.24}$$

$$\alpha = \alpha_{\text{int}} \times 2^{\alpha_{\text{exp}}} \tag{3.25}$$

where $\beta_{\text{int}}$ and $\alpha_{\text{int}}$ are positive integers and $\beta_{\text{exp}}$ and $\alpha_{\text{exp}}$ are negative integers.

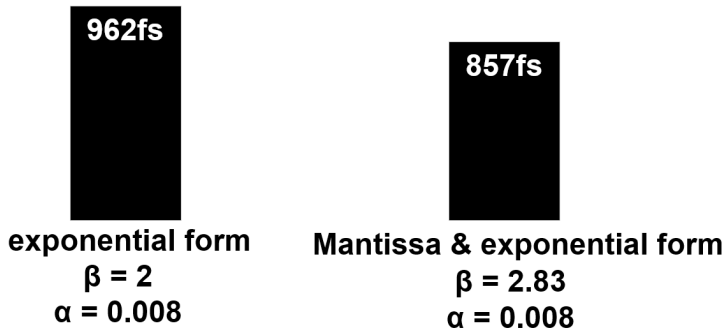



Figure 3.9 RMS jitter comparison between two gain forms

1) mantissa and exponential form and 2) exponential form

To validate the effectiveness of using mantissa and exponential form over exponential form, behavioral simulation is done and RMS jitter was compared. The results are shown in Figure 3.9. The RMS jitter was approximately 12% degraded when using only exponential form. Therefore, implementing gains as mantissa and exponential form is advantageous for maximizing the effectiveness of gain optimization technique.

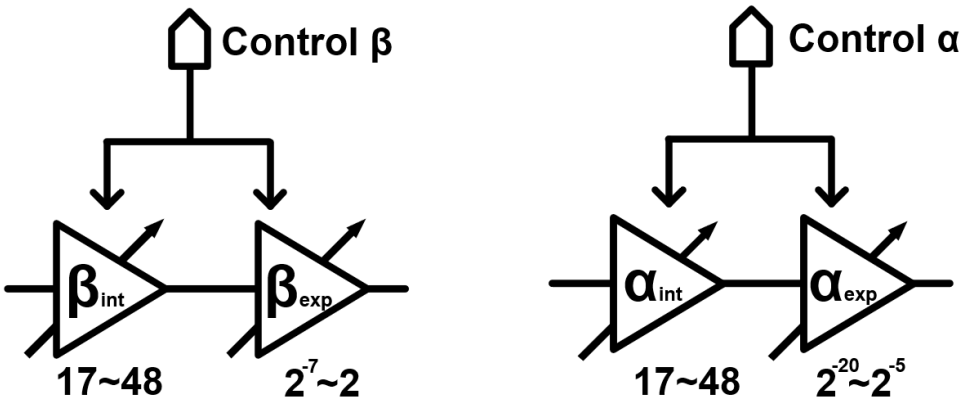



Figure 3.10 Implementation of gain paths

Implementation of the proportional and integral gain paths are shown in Figure 3.10. Gain stage denoted as $\beta_{int}$ and $\alpha_{int}$ implements the mantissa and gain stage denoted as $\beta_{exp}$ and $\alpha_{exp}$ implements the exponent. The gain optimizer controls $\beta_{int}$ and $\alpha_{int}$ to control $\beta$ and $\alpha$. However, if $\beta_{int}$ and $\alpha_{int}$ reaches its maximum(or minimum value), $\beta_{int}$ and $\alpha_{int}$ cannot increase(or decrease) anymore. In this occasion, $\beta_{int}$ and $\alpha_{int}$ are divided(or multiplied) by 2 and $\beta_{exp}$ is increased(decreased) by 1.

The range of value that $\beta_{int}$, $\alpha_{int}$, $\beta_{exp}$ and $\alpha_{exp}$ can have is $17 \sim 48$, $17 \sim 48$, $2^{-7} \sim 2$, and $2^{-20} \sim 2^{-5}$ respectively. The range of $\beta_{int}$ and $\alpha_{int}$ are designed as $17 \sim 48$ to minimize the number of events $\beta_{int}$ and $\alpha_{int}$ reaches the maximum or minimum value. For example, if $\beta_{int}$ reaches 17 and multiplied by 2, subsequent value of $\beta_{int}$ is 34. In this case, 35 ~ 48 act as a margin such that $\beta_{int}$ does not hit the maximum value easily.

# Chapter 4

# Design of ADPLL with P/I Gain Co-Optimization Technique

## 4.1 Design Considerations

In this chapter, an All-Digital PLL with P/I Gain co-optimization technique is proposed. The target frequency is 3.2 GHz with 100MHz reference clock. The objective is to observe better phase noise performance with PLL P/I gain co-optimization technique than without the optimization technique. The digital loop filter is designed so that the optimization can be turned or off. In the following section, the architecture of the PLL and specific implementations of the blocks are shown.

## 4.2 Proposed Architecture

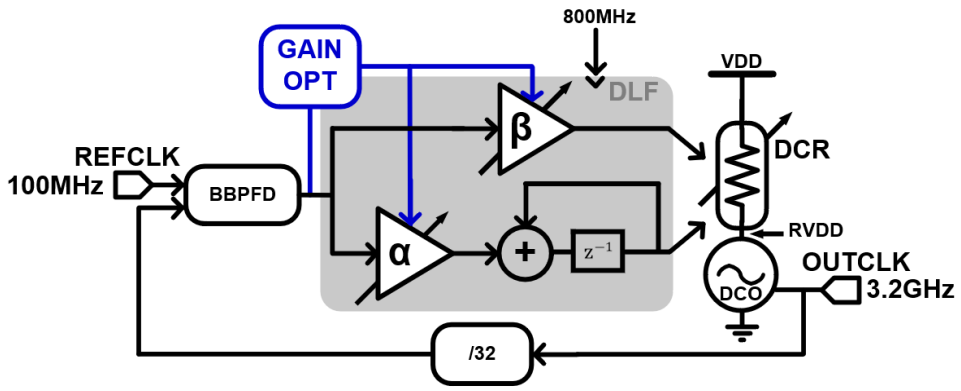



Figure 4.1 Block Diagram of the proposed PLL

The overall architecture of the proposed PLL is shown in Figure 4.1. The proposed PLL consists of BBPFD, DLF, Gain Optimizer, DCO, and divider. Signal paths including gain optimizer is marked as blue in the figure.

Gain optimizer utilizes the BBPFD outputs to optimize the proportional and integral gain simultaneously and dynamically. Its implementation is shown in Figure 3.7.

Topology of digital loop filter shown in Figure 4.1 is different from that of digital loop filter shown in Figure 2.11. To start with, the register($z^{-1}$) of the integral path is placed on the feedforward path instead of feedback path. Also, direct proportional path is used. That is, proportional path and integral path directly controls the DCO without the adder. The adder in topology shown in Figure 2.11 can provide glitches to the DCO which can be problematic [11]. Adding a retiming stage at the output can resolve the problem but will increase loop filter delay which can amplify jitter.

However, the loop filter shown in Figure 4.1 resolves the problem of glitches by removing the adder and directly controlling the DCO. Topology shown in Figure 4.1 perform nearly the same jitter performance with the conventional topology with zero delay (Figure 2.11) [11]. Furthermore, since direct proportional path and integral path control the same DCR in parallel, $d\beta/d\alpha$ can be maintained constant over process, voltage, and temperature (PVT) variations [12].

Digitally controlled resistor(DCR) is adopted as a method to control the frequency of the DCO [13]. DCR is controlled by 31-bit thermometer row code and column code. It is an array of MOSFETs that can be turned on or off by the digital codes so that it can change its overall resistance accordingly. Ring VDD denoted as RVDD is also varied which in turn controls the operating frequency of the oscillator.

# 4.3 Circuit Implementation

## 4.3.1 Bang-Bang Phase Frequency Detector

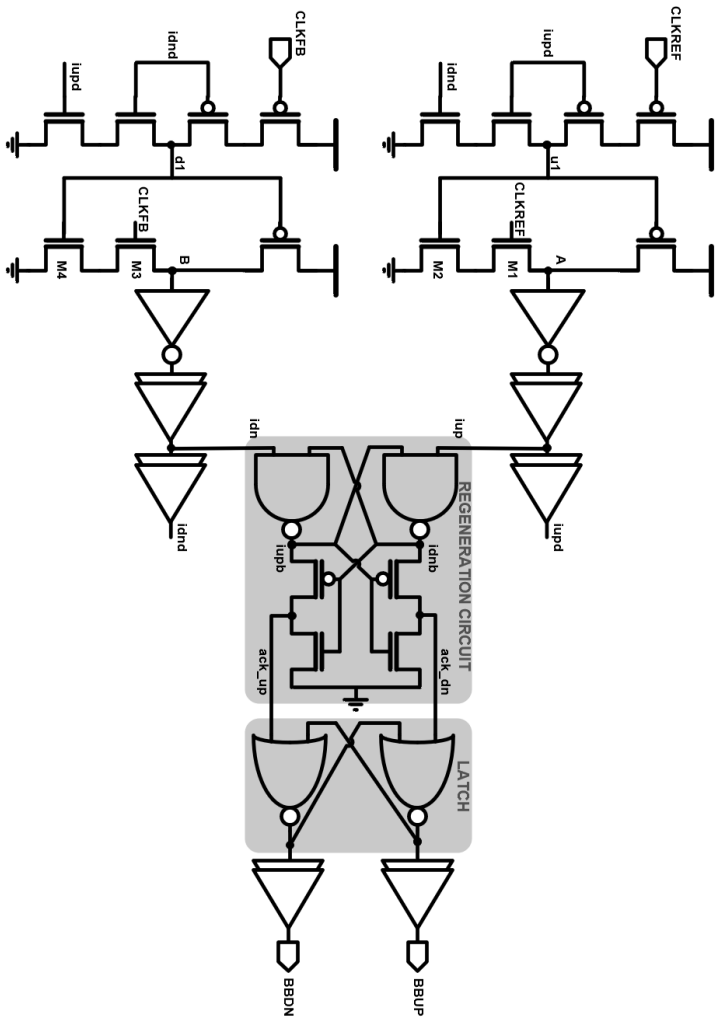



Figure 4.2 Schematic of BBPFD

Implementation of BBPFD is shown in Figure 4.2. The behavior of BBPFD is as follows.

● Initial phase : Initially when both clocks(CLKREF and CLKFB) are idle, node u1 and d1 are pre-charged. This is because node iupd and idnd are forced to be pulled down by the feedback path. Thus at initial state, node u1 and d1 are pre-charged and node A and B are floating but wandering around $0.5V_{DD} \sim V_{DD}$.

● Phase 1 : When rising edge of CLKREF occurs, node A will be pulled down setting node iup and iupd to $V_{DD}$. When rising edge of CLKFB occurs, node B will be pulled down setting node idn and idnd to $V_{DD}$.

● Phase 2 : The regeneration circuit will capture the rising edge that takes precedence. Node ack_up will be set to $V_{DD}$ if rising edge of CLKREF takes precedence and vice versa.

● Phase 3 : Node ack_up and ack_dn will be latched and latched value will be buffered to the output BBUP and BBDN.

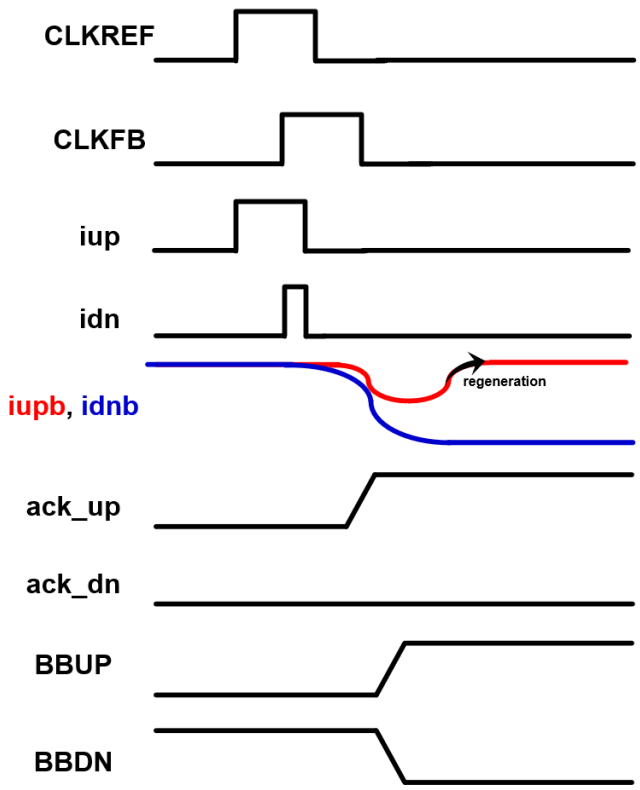



Figure 4.3 Timing Diagram of BBPFD

Timing diagram of BBPFD is shown in Figure 4.3. Regeneration circuit regenerates the node iupb because node idnb has been pulled down priorly. This regeneration effect is due to positive feedback. Its principle is similar to that of strongarm latch [14].

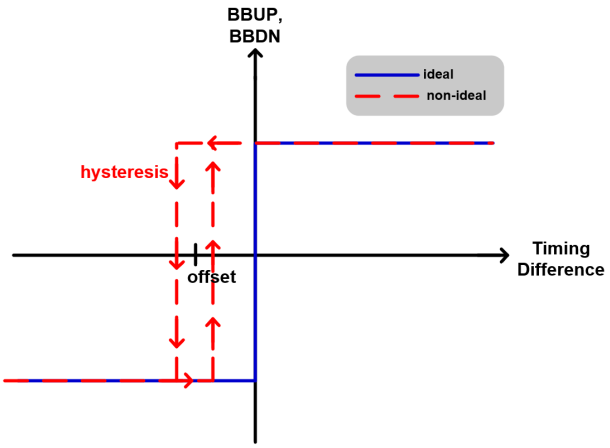



Figure 4.4 Plot of output voltage versus timing difference between CLKREF and CLKFB

Plot of output voltage versus timing difference between two clocks is shown in Figure 4.4. Ideally, BBPFD should output BBUP if CLKREF precedes CLKFB and vice versa. However, two non-ideal effects can occur when implementing BBPFD; offset and hysteresis.

First, offset is mostly due to mismatch of two clock paths in layout design. Thus, layout design should be done carefully so that two clock paths are symmetric. Offset in BBPFD will cause the PLL to be locked with static timing error. Fortunately, offset will not degrade jitter performance.

Second, hysteresis is a phenomenon where current outputs are dependent on its previous states [15]. There are two possible states in BBPFD; state when CLKREF precedes CLKFB and state when CLKFB precedes CLKREF. The BBPFD output depends in which state the BBPFD was at previous cycle. Hysteresis occurs due to inner node voltages of the previous step. Thus, hysteresis effect can be reduced by flushing out inner node voltage quickly. In schematics shown in Figure 4.2, history

information of previous cycle is stored in node A and B. By increasing the width of MOSFET M1 ~ M4, node A and B can be flushed out quickly thus reducing hysteresis effect.

The post-layout simulation of the implemented BBPFD showed the offset of 30 fs and hysteresis of 50 fs.

### 4.3.2 Ring Oscillator

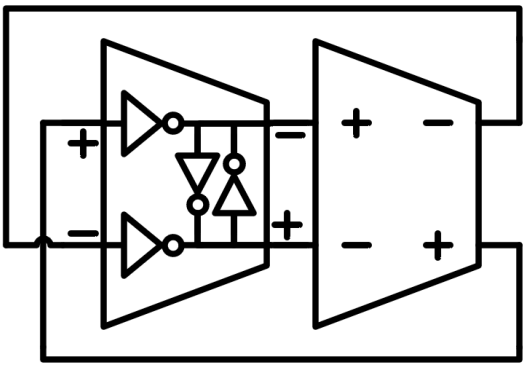


Figure 4.5 Schematics of an oscillator

Figure 4.5 shows an the ring oscillator with pseudo-differential inverter chain topology [16]. Slope of a transition edge should be steep for good jitter performance which is explained by impulse sensitivity function [17]. The transition edges are made steeper by positive feedback of an inverter latch.

Oscillator is implemented in two stages to minimize parasitic capacitance introduced in layout design [18]. Ring oscillator with lesser number of stages

introduce smaller layout design and fewer number of wires leading to smaller parasitic capacitance. Therefore, it is beneficial to design a ring oscillator in minimum number of stages regarding speed and power consumption. Post-layout simulation of the implemented ring oscillator was done. Table 4.1 shows the frequency range and power consumption. At corner TT, the oscillator consumes 6.69mW while running at 3.2GHz.

Table 4.1. Frequency range and power consumption of the ring oscillator

| Corner | Frequency range | Power Consumption |
|---|---|---|
| FF | 2.21GHz ~ 4.38GHz | 3.63mW ~ 10.56mW |
| TT | 1.9GHz ~ 4.05GHz | 3.06mW ~ 9.89mW |
| SS | 1.61GHz ~ 3.62GHz | 2.5mW ~ 7.55mW |

Figure 4.6 shows the phase noise plot of the free-running ring-oscillator. 100kHz and 1MHz phase noise are -61.6dBc/Hz and -90.5dBc/Hz respectively.

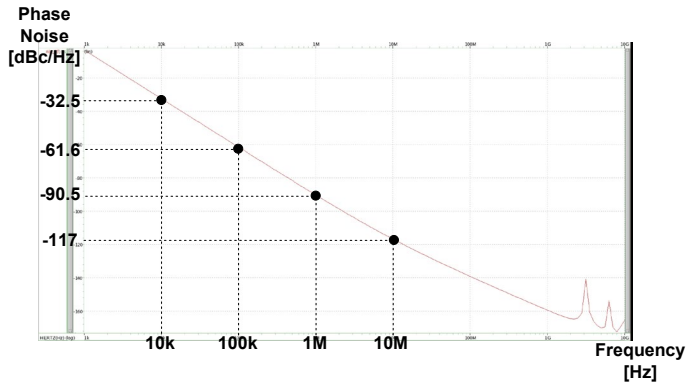



Figure 4.6 Phase noise plot of the free-running oscillator

### 4.3.3 Digitally Controlled Resistor

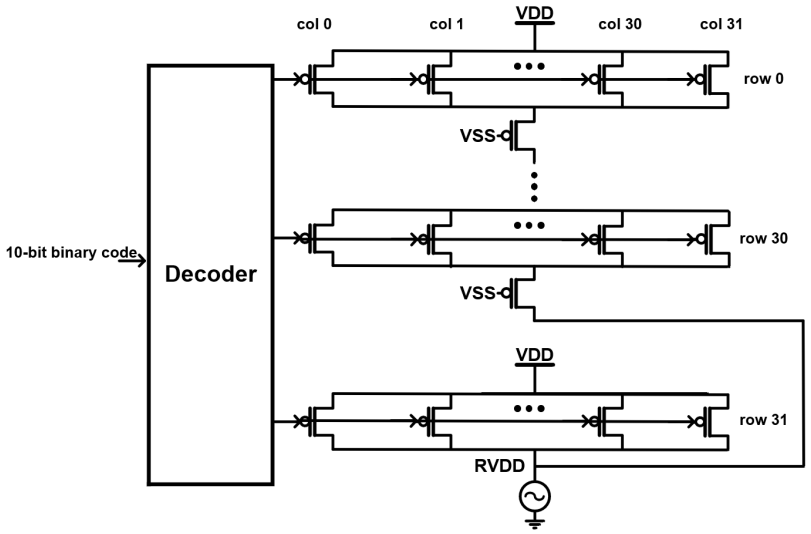
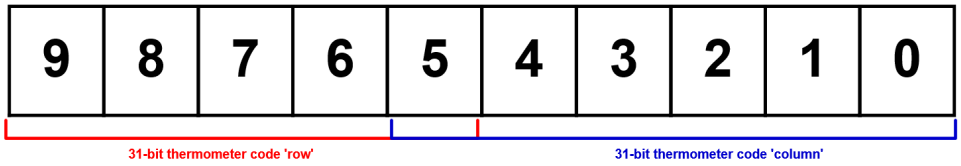



Figure 4.7 Basic concept of DCR

Digitally Controlled Resistor(DCR) is used to control the frequency of the oscillator [13]. Basic concept of DCR is depicted in Figure 4.7. Each MOSFET is turned off or on, so the overall resistance is changed and ring $V_{DD}$ is also changed accordingly.

The oscillator control code that loop filter outputs is 10-bit binary code. Binary codes cannot directly control the frequency; it needs to be transformed into thermometer form. However, 1023-bit is needed to transform 10-bit binary code into thermometer code which is too excessive. Therefore, 10-bit code is split into 5-bit and 6-bit with 1-bit overlapped as shown in Figure 4.8 (a). 5-bit of MSB can be transformed into 31-bit thermometer code which represents ‘row’ code. Similarly,




(a)

| 6-bit binary code | 31-bit thermometer code | |
|---|---|---|
| 6'b000000 | 31'h00000000 | |
| 6'b000001 | 31'h00000001 | |
| 6'b000010 | 31'h00000003 | |
| 6'b000011 | 31'h00000007 | |
| ... | ... | |
| 6'b011111 | 31'h7FFFFFFF | |
| 6'b100000 | 31'h7FFFFFFF | 0 is filled from LSB. |
| 6'b100001 | 31'h7FFFFFFE | |
| 6'b100010 | 31'h7FFFFFFC | |
| ... | ... | |

(b)

Figure 4.8 (a) Bit split of control code for decoding

(b) Transformation of 6-bit binary code into 31-bit thermometer code

6-bit of LSB can be converted into 31-bit thermometer code which represents ‘column’ code. 6-bit binary code can be converted into 31-bit thermometer code by filling out 0 again from LSB as shown in Figure 4.8 (b). 6’b011111 and 6’b100000 are both decoded to 31’h7FFFFFFF because the change is reflected into row code due to 1-bit overlap.

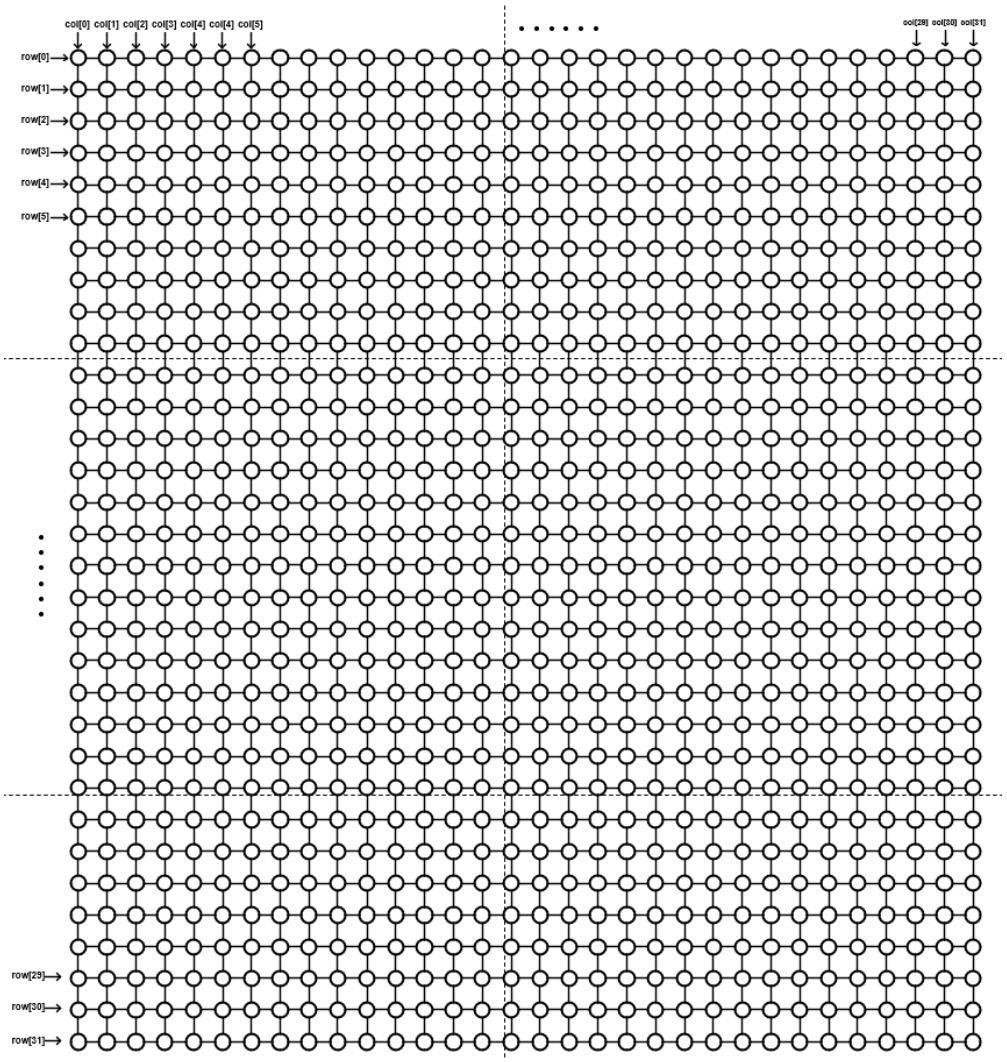



Figure 4.9 Digitally Controlled Resistor

So far, decoding method of 10-bit frequency control code has been explained. By using this method, only 62 bits are used and only single bit is changed according to binary code change. 31-bit row code and 31-bit column code control each unit cell as shown in Figure 4.9. Unit cells are designed so that unit cells in odd rows are turned

on when column code is 1 and even rows are turned on when column code is 0. This way, MOSFETs are turned off or on in sequential order thus, preventing glitches.

### 4.3.4 Frequency Divider

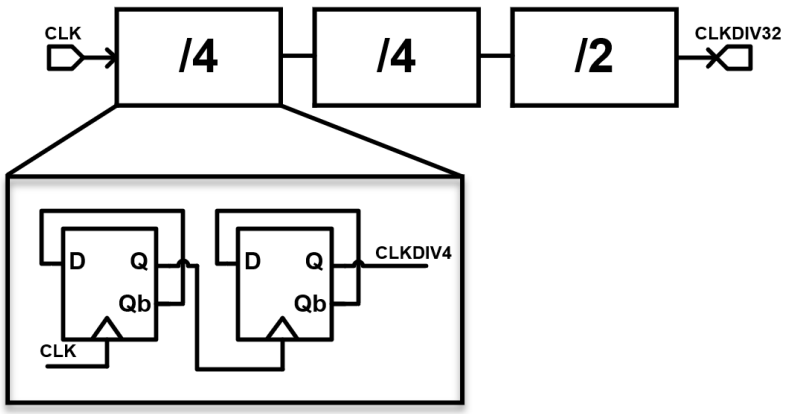



Figure 4.10 Schematics of divider

Schematic of implemented divider is shown in Figure 4.10. Dividing ratio of 32 is implemented by cascading 5 flip flops. Post-layout simulation of implemented divider was done to make sure divider operates well at the maximum frequency of an oscillator at corner FF. The maximum operation frequency is 10.2GHz. Power consumption at 3.2GHz is 178.6uW with supply voltage of 1V.

# Chapter 5

# Simulation Results

## 5.1 Transient Analysis

Post-layout simulation of the overall PLL was done by using FineSim. Frequency acquisition and locking behavior were observed from the simulation as shown in Figure 5.1.

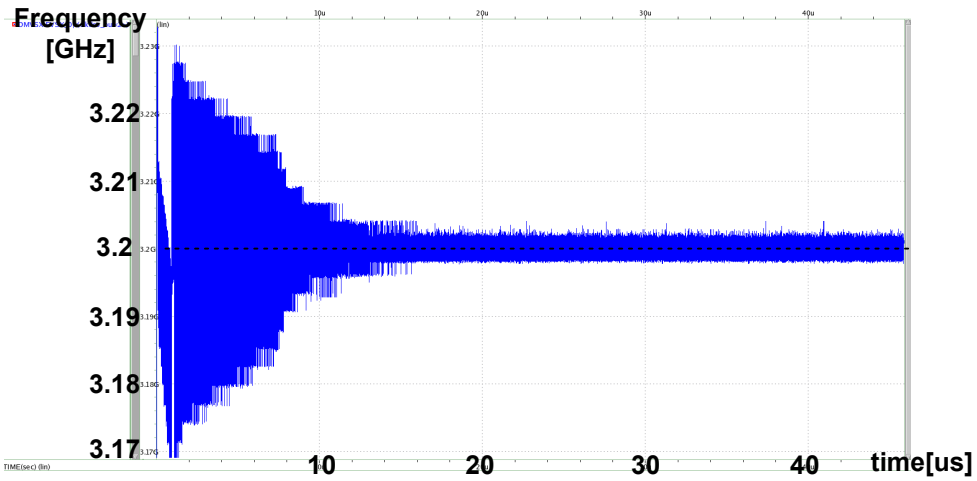



Figure 5.1 Frequency acquisition and locking of the PLL

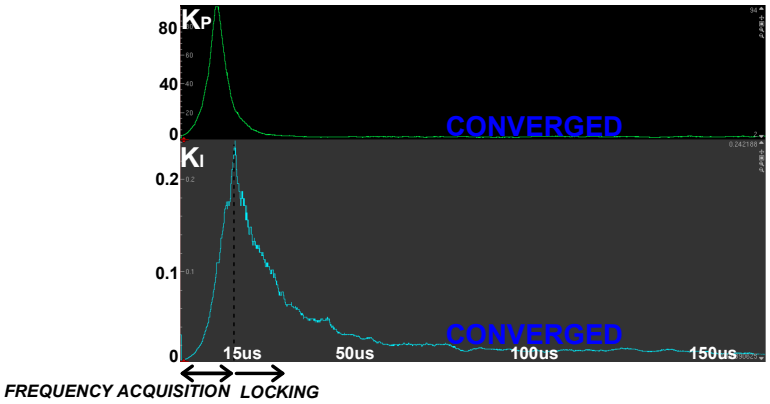



Figure 5.2 Behavioral simulation showing proportional and integral

Behavioral simulation was also done by using System Verilog. Figure 5.2 shows the proportional and integral gain with respect to time. During the frequency acquisition, integral gain rapidly increases due to consecutive outputs of BBPFD. Thanks to the increased integral gain, frequency acquisition step shown in Figure 5.1 is accelerated. After frequency acquisition and locking step, proportional and integral gains are converged to the optimal value as shown in Figure 5.2.

# 5.2 Phase Noise

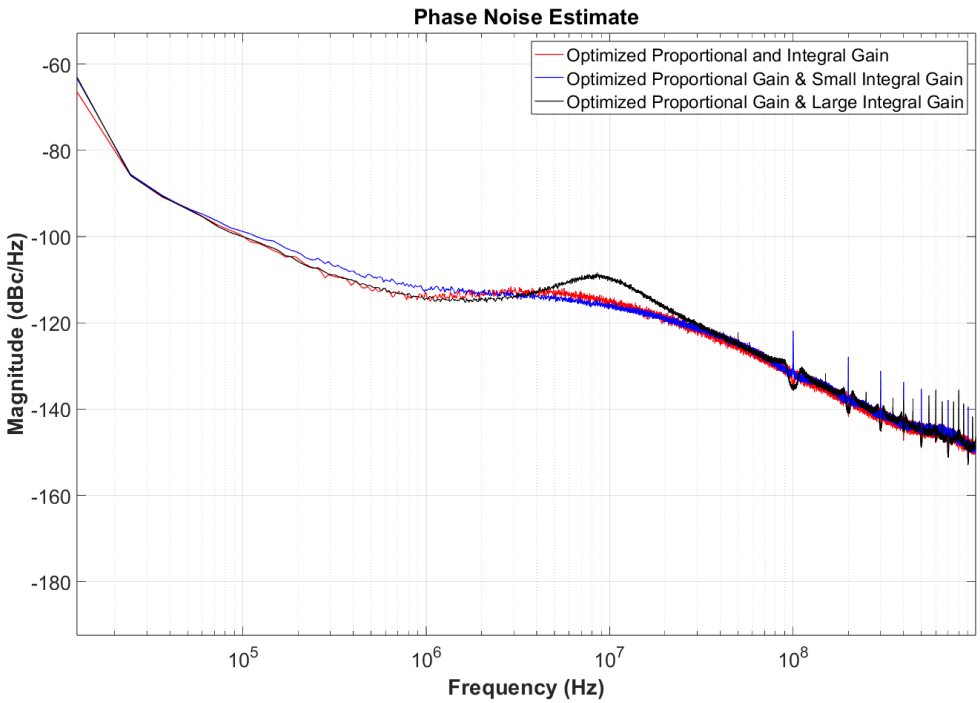



Figure 5.3 Phase noise plot of the output

Behavioral simulation was done to measure the phase noise performance of the PLL. Figure 5.3 shows the phase noise plot of the 3.2GHz output clock using 100MHz reference clock 1) when both proportional and integral gain are optimized, 2) when only the proportional gain is optimized and integral gain is set large, and 3) when only the proportional gain is optimized and integral gain is set small. When proportional and integral gain were optimized, 100kHz and 1MHz phase noise were -100.2dBc/Hz

and -113.4dBc/Hz respectively. When only the proportional gain was optimized and integral gain was set as small value, 100kHz and 1MHz phase noise were -98.6dBc/Hz and -112.3dBc/Hz respectively. When integral gain is set as small value, low-frequency noise of an oscillator is not filtered-out well leading to in-band noise degradation. When only the proportional gain was optimized and integral gain was set as large value, 100kHz and 1MHz phase noise were -100.1dBc/Hz and -114.5dBc/Hz respectively. In-band phase noise was similar to phase noise when both gains were optimized but out-band noise exhibited a peaking giving -110dBc/Hz at 10MHz. Integrated RMS jitter (integrated from 20kHz to 100MHz) for each case are 857fs, 890fs, and 1.05ps respectively.

# 5.3 Power Breakdown

Power breakdown was performed through post-layout simulation. The power consumption of each circuit block is as shown in Table 5.1. Figure 5.4 shows the portion that each block occupies in total power consumption.

Table 5.1 Power Breakdown

| **Block** | **Power consumption** |
|---|---|
| Divider | 178.6uW |
| BBPFD | 34.4uW |
| Oscillator | 6.6mW |
| Digital Loop Filter | 430uW |

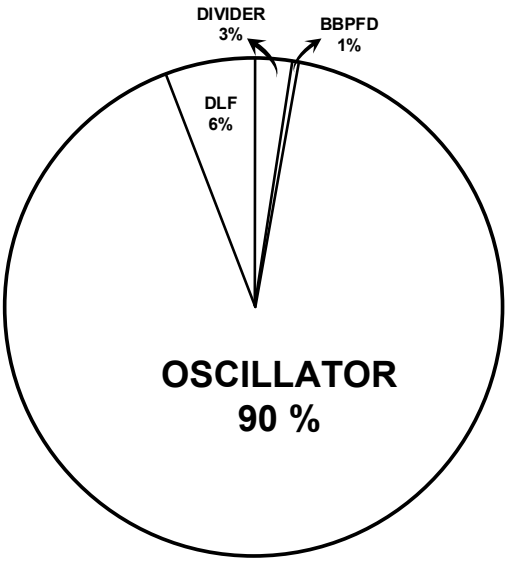



Figure 5.4 Power breakdown of the PLL

## 5.4 Die micrograph

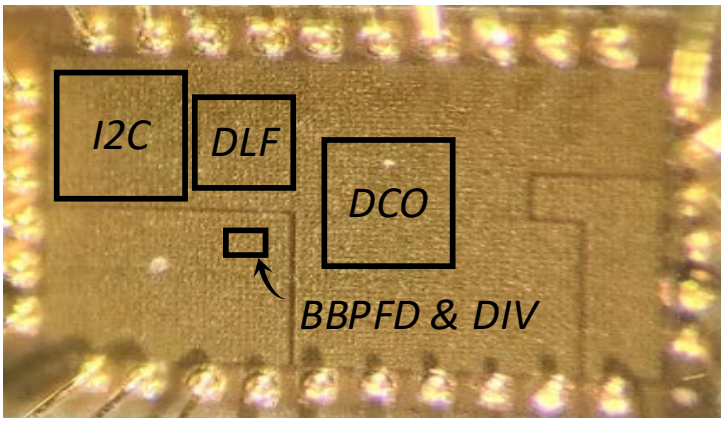



Figure 5.5 Die Micrograph

The chip was fabricated in 28-nm CMOS process and occupies $0.033\text{mm}^2$ of area. Digital blocks including digital loop filter and gain optimizer was coded by Verilog. After, the RTL code was synthesized and P&R process was done. On the other hand, the layouts of analog blocks were manually designed using virtuoso. Figure 5.5 shows the die micrograph of the fabricated chip.

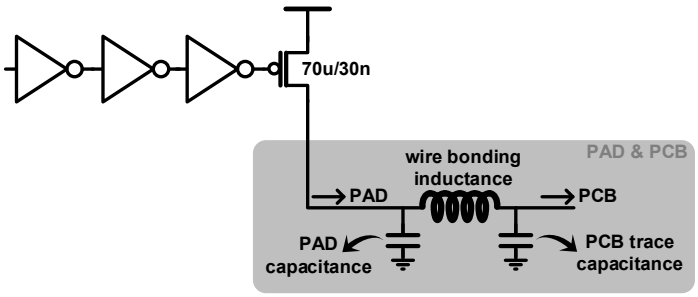



Figure 5.6 Diagram of open drain driver, PAD, and PCB

The output clock is buffered by clock buffers which are placed on the output path. The PLL output is transferred to PAD by open drain driver which is shown in Figure 5.6. Transistor size in open drain driver is designed big(70u/30n) to ensure output clock is driven well to the huge PAD and PCB capacitance.

I2C block tunes the parameters of the PLL through the registers. Registers are written by i2c communication. The register values are read by the PLL and utilized for tuning various parameters.

# 5.5 Performance Summary

The proposed AD-PLL occupies area of 0.033mm$^2$. 100kHz and 1MHz phase noise were -98.6dBc/Hz and -112.3dBc/Hz respectively and integrated RMS jitter was 857fs. The total power consumption was 7.6mW(Divider : 178.6uW, BBPFD : 34.4uW, Oscillator : 6.6mW, Digital Loop Filter : 430uW). Comparison table is shown in Table 5.2.

Table 5.2. Comparison Table

| | ASSCC'15 [19] | TCAS II'15 [10] | ISIC'14 [20] | JSSC'16 [18] | This Work |
|---|---|---|---|---|---|
| **Technology (nm)** | 40 | 65 | 65 | 65 | 28 |
| **Architecture** | DBPLL | DBPLL | DBPLL | CPPLL | DBPLL |
| **DCO Type** | Ring | Ring | Ring | Ring | Ring |
| **f_OUT / f_REF (GHz / MHz)** | 5 / 250 | 2.5 / 25 | 2.6 / N/A | 10 / 625 | 3.2 / 100 |
| **Mult. Factor(N)** | 25 | 100 | N/A | 16 | 32 |
| **Power (mW)** | 3.34 | 5 | 18 | 7.6 | 7.6 |
| **Area (mm$^2$)** | 0.005 | 0.038 | 0.098 | 0.009 | 0.033 |
| **RMS Jitter (fs)** | 1242 | 1720 | 2100 | 414 | 857 |
| **FoM$_{jit}$ (dB)** | -232.9 | -228 | -220.6 | -238.8 | -232.5 |

* $\mathrm{FOM_{jit}} = 10\log_{10}\left[\left(\frac{\text{RMS Jitter}}{1s}\right)^2 * \frac{\text{Power}}{1\text{mW}}\right]$.

*DBPLL stands for Digital Bang-Bang PLL.

# Chapter 6

# Conclusion

In this thesis, All-Digital PLL with proportional and integral gain co-optimization technique is presented. As it is nearly impossible to predict the noise environment priorly, proportional and integral gains must be tracked adaptively. The proposed PLL utilizes statistical information of BBPFD output to optimize proportional and integral gains adaptively. The gains were converged to the optimum value successfully. According to behavioral simulation the PLL exhibits RMS jitter of 857fs and achieves FoM of -232.5dB. The proposed PLL was fabricated in 28-nm CMOS process and occupies 0.033 $mm^2$ of area.

# Bibliography

[1] J. Kim, M. A. Horowitz, and G.-Y. Wei, “Design of CMOS adaptive-bandwidth PLL/DLLs: A general approach,” IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 50, no. 11, pp. 860–869, Nov. 2003. doi:10.1109/tcsii.2003.819120

[2] N. D. Dalt, “Markov chains-based derivation of the phase detector gain in Bang-Bang PLLs,” IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 53, no. 11, pp. 1195–1199, 2006. doi:10.1109/tcsii.2006.883197

[3] B. Razavi, “Jitter-power trade-offs in Plls,” IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 68, no. 4, pp. 1381–1387, 2021. doi:10.1109/tcsi.2021.3057580

[4] N. Da Dalt, “A design-oriented study of the nonlinear dynamics of Digital Bang-Bang PLLs,” IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 52, no. 1, pp. 21–31, 2005. doi:10.1109/tcsi.2004.840089

[5] T.-K. Kuan and S.-I. Liu, “A loop gain optimization technique for integer-N TDC-based phase-locked loops,” IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 62, no. 7, pp. 1873–1882, 2015. doi:10.1109/tcsi.2015.2423793

[6] B. Kim, T. C. Weigandt, and P. R. Gray, “PLL/DLL System Noise Analysis for Low Jitter Clock Synthesizer Design,” Proceedings of IEEE International

Symposium on Circuits and Systems - ISCAS '94. doi:10.1109/iscas.1994.409189

[7] G. Marucci, S. Levantino, P. Maffezzoni, and C. Samori, "Exploiting stochastic resonance to enhance the performance of Digital Bang-Bang PLLs," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 60, no. 10, pp. 632–636, 2013. doi:10.1109/tcsii.2013.2273732

[8] G. Wei, "Estimations of frequency and its drift rate," IEEE Transactions on Instrumentation and Measurement, vol. 46, no. 1, pp. 79–82, 1997. doi:10.1109/19.552161

[9] Y. Lee et al., "17.1 A −240db-fomjitter and −115dBc/Hz PN @ 100khz, 7.7ghz ring-DCO-based digital PLL using P/I-gain co-optimization and sequence-rearranged optimally spaced TDC for flicker-noise reduction," 2020 IEEE International Solid- State Circuits Conference - (ISSCC), 2020. doi:10.1109/isscc19947.2020.9062966

[10] S. Jang et al., "An optimum loop gain tracking all-digital PLL using autocorrelation of Bang–Bang Phase-frequency detection," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 62, no. 9, pp. 836–840, 2015. doi:10.1109/tcsii.2015.2435691

[11] T.-K. Kuan and S.-I. Liu, "A bang bang phase-locked loop using automatic loop gain control and loop latency reduction techniques," IEEE Journal of Solid-State Circuits, vol. 51, no. 4, pp. 821–831, 2016.

doi:10.1109/jssc.2016.2519391

[12] Y. Song, H.-G. Ko, C. Kim, and D.-K. Jeong, “A 1.05-to-3.2 ghz all-digital PLL for DDR5 registering clock driver with a self-biased supply-noise-compensating ring DCO,” IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 69, no. 3, pp. 759–763, 2022. doi:10.1109/tcsii.2021.3123610

[13] D.-H. Oh, D.-S. Kim, S. Kim, D.-K. Jeong, and W. Kim, “A 2.8GB/s all-digital CDR with a 10B Monotonic DCO,” 2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers, 2007. doi:10.1109/isscc.2007.373374

[14] B. Razavi, “The strongarm latch [a circuit for all seasons],” IEEE Solid-State Circuits Magazine, vol. 7, no. 2, pp. 12–17, 2015. doi:10.1109/mssc.2015.2418155

[15] S. Bashiri, S. Aouini, N. Ben-Hamida, and C. Plett, “Analysis and modeling of the phase detector hysteresis in Bang-Bang Plls,” IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 62, no. 2, pp. 347–355, 2015. doi:10.1109/tcsi.2014.2365871

[16] W. Bae, H. Ju, K. Park, S.-Y. Cho, and D.-K. Jeong, “A 7.6 MW, 214-FS RMS jitter 10-ghz phase-locked loop for 40-GB/S serial link transmitter based on two-stage ring oscillator in 65-NM CMOS,” 2015 IEEE Asian Solid-State Circuits Conference (A-SSCC), 2015. doi:10.1109/asscc.2015.7387448

[17] A. Hajimiri and T. H. Lee, “A general theory of phase noise in electrical oscillators,” IEEE Journal of Solid-State Circuits, vol. 33, no. 2, pp. 179–194, 1998. doi:10.1109/4.658619

[18] W. Bae, H. Ju, K. Park, S.-Y. Cho, and D.-K. Jeong, “A 7.6 MW, 414 fs RMS-jitter 10 ghz phase-locked loop for a 40 GB/S serial link transmitter based on a two-stage ring oscillator in 65 nm CMOS,” IEEE Journal of Solid-State Circuits, vol. 51, no. 10, pp. 2357–2367, 2016. doi:10.1109/jssc.2016.2579159

[19] C.-H. Chiang, C.-C. Huang, and S.-I. Liu, “A Digital Bang-bang phase-locked loop with bandwidth calibration,” 2015 IEEE Asian Solid-State Circuits Conference (A-SSCC), 2015. doi:10.1109/asscc.2015.7387450

[20] Y. Kim, M.-K. Jeon, and C. Yoo, “Digital Phase Locked Loop (DPLL) with offset dithered Bang-Bang Phase Detector (BBPD) for Bandwidth Control,” 2014 International Symposium on Integrated Circuits (ISIC), 2014. doi:10.1109/isicir.2014.7029467

# 초 록

위상 고정 루프의 지터 성능은 위상 고정 루프 설계 시 고려해야할 중요한 요소다. 클럭 신호 엣지의 편차는 시스템 성능을 악화시킨다. 높은 지터 성능을 위해 위상 고정 루프의 비례 이득과 적분 이득은 최적값을 가져야 한다. 하지만 노이즈 환경에 대한 제한된 정보로 인해 설계 단계에서 최적값을 결정하기 어렵다. 따라서, 비례 이득과 적분 이득의 최적값은 실시간으로 주변 환경을 모니터링함으로써 결정되어야 한다. 따라서 본 논문은 비례 이득과 적분 이득을 실시간으로 최적화하는 디지털 위상 고정 루프를 제안한다.

본 논문에서는 최적화 기술에 대한 자세한 분석이 진행된다. 분석을 바탕으로, 최적화 기술의 효용성이 검증된다. 또한, 위상 고정 루프의 회로가 동작 원리와 함께 주어진다.

제안된 위상 고정 루프는 100MHz 레퍼런스 클럭을 통해 3.2GHz 클럭을 출력한다. 전력 소비는 7.6mW 이며, 1V 의 공급 전압을 사용한다. 20kHz 에서 100MHz 까지의 주파수 대역을 고려하였을 때 857fs 의 RMS 지터를 갖는다. 제안된 위상 고정 루프는 28nm CMOS 공정에서 구현되었으며 0.033$mm^2$의 유효 면적을 갖는다.

**주요어** : 디지털 위상 고정 루프, 이득값 최적화, 확률적 공진, 플리커 노이즈, 뱅뱅 위상주파수 검출기, 디지털 루프 필터

**학 번** : 2021-27376