國立交通大學

電子工程學系　電子工程研究所

碩士論文

適用於展頻時脈產生器之全數位鎖相迴路

All-Digital Phase-Locked Loop for Spread-Spectrum Clock Generator

研 究 生：蘇明銓

指導教授：周世傑　教授

中華民國 九十八年 九月

適用於展頻時脈產生器之全數位鎖相迴路

All-Digital Phase-Locked Loop for Spread-Spectrum Clock Generator

研究生：蘇明銓　　Student：Ming-Chiuan Su

指導教授：周世傑　教授　　Advisor：Prof. Shyh-Jye Jou

國立交通大學

電子工程學系　電子工程研究所

碩士論文

A Thesis
Submitted to Department of Electronics Engineering & Institute of Electronics
College of Electrical and Computer Engineering
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of Master of Science
in
Department of Electronics Engineering
September 2009
Hsinchu, Taiwan, Republic of China

中華民國 九十八年 九月

適用於展頻時脈產生器之全數位鎖相迴路

研究生：蘇明銓　　　　　　　　　　　　指導教授：周世傑 教授

國立交通大學

電子工程研究所碩士班

摘要

系統晶片中隨著內部參考時脈的提升，為了對抗電磁波干擾的問題，展頻技術被應用在時脈的產生。傳統類比鎖相迴路較易受到製程/電壓/溫度變化的影響。因此在使用深次微米互補式金氧半製程時，鎖相迴路遂逐漸向全數位式設計。

全數位鎖相迴路由 bang-bang 相位頻率偵測器、使用累加器為主的數位迴路濾波器、差動式數位控制震盪器和除頻器等組成。Bang-bang 相位頻率偵測器產生相位比較訊號，控制使用累加器為主的數位迴路濾波器。然後，數位迴路濾波器輸出粗調以及微調的控制碼，用以改變差動式數位控制震盪器的頻率。使用和差調變器進一步控制數位控制震盪器可增加全數位鎖相迴路的頻率解析度。展頻時脈的產生需要應用一個低抖動的全數位鎖相迴路。

展頻技術是對時脈信號的中心頻率做微量的調變，使時脈信號的頻譜展開成較寬的頻帶範圍。因此可降低時脈信號在頻譜上的能量峰值，減少時脈信號所造成的高頻電磁雜訊干擾(Electro-Magnetic Interference, EMI)。未來提出的時脈產生器以全數位鎖相迴路為基本架構，可能使用和差調變器及調變多重相位的方法來實現展頻，並且以符合 Serial-ATA 6Gbps 的規格和 USB 3.0 5Gbps 的規格作為參考設計。

展頻時脈產生器可依照系統對低功率需求或是對低抖動需求，選擇以 10 個輸出相位或是 20 個輸出相位作展頻。本論文探討全數位鎖相迴路及展頻時脈產生器的設計，以及使用 TSMC 65nm 1P9M CMOS 製程的模擬。

All-Digital Phase-Locked Loop for Spread-Spectrum Clock Generator

Student：Ming-Chiuan Su　　　　Advisor：Prof. Shyh-Jye Jou

Department of Electronics Engineering
Institute of Electronics
National Chiao Tung University

Abstract

As SOC(System On Chip) works with increasing internal reference clocks, the spread-spectrum clocking technique is used to mitigate EMI(Electro-Magnetic Interference) effect. Conventional analog PLLs are likely to be affected by PVT (process/voltage/temperature) variations. Hence, when using deep-submicron CMOS process, PLLs are prone to all-digital design.

All-digital PLL consists of bang-bang PFD, accumulator-based digital loop filter, differential DCO and divider. Bang-bang PFD generates phase compare signals to control accumulator-based digital loop filter. Then, DLF outputs coarse-tune and fine-tune control code to change differential DCO's frequency. Using $\Sigma\Delta$ modulator to further control DCO enhances ADPLL's frequency resolution. A low-jitter ADPLL is desired under spread-spectrum clocking application.

Spread-spectrum technique is to modulate clock's center frequency, and the spectrum of the clock is spread over a broader range. Therefore, clock's peak energy is reduced and it also mitigates EMI effect. Based on an ADPLL, the proposed spread spectrum clock generator (SSCG) is fulfilled using $\Sigma\Delta$ modulator and multiple phases. This SSCG uses Serial-ATA 6Gbps and USB 3.0 5Gbps specifications as reference.

For different system requirements for low-power or low-jitter, the SSCG can do modulation on 10 or 20 phases. The thesis proposes novel ADPLL and SSCG architecture and the circuits are implemented with TSMC 65nm 1P9M CMOS process.

致　謝

首先感謝周世傑老師在我這兩年多的碩士生涯中不遺餘力細心指導，經由一次又一次的討論，使我慢慢累積一定程度的研究經驗以及學習如何研究的態度。老師總是認真、用心的指導我的論文研究中可能有盲點的地方，也讓我在研究上能更加進步。祝福老師身體健康，家庭幸福。在此也要感謝口試委員陳巍仁教授以及蔡嘉明教授撥空參加口試，並能在口試中給予我寶貴的意見而能讓我的論文更加完善。

再來要感謝我的父母，除了生活上讓我不虞匱乏，在學業上也是時時刻刻關心我。因為有他們的照顧以及關懷，我在求學的路上能走的更平穩順遂，真的很感謝我的父母。另外，也要感謝我的哥哥，在學習以及待人處事上總是給了我很多意見，也是我一個學習的好榜樣。

接著，要謝謝所有在Mixed-Signal Circuit and System Lab的學長姐們，因為有他們引導，讓我能很快融入這個大家庭。不管是在平常的哈拉當中或是在每次的實驗室出遊或是邀約外出吃飯，都讓我跟實驗室成員彼此更佳的熟悉；有了他們，研究的日子總是在歡樂當中度過。

最後，除了在碩士生涯的研究之外，我還想再更進一步提升自我研究能力。未來也將繼續在這裡攻讀博士。也希望藉著在這個我熟悉的環境，能更快步上研究軌道以及補強我不足的地方，並能再次學習更廣層面的課程！

人生的路上都會遇上一些過客，能感謝的人太多了，不管是所有的授課老師們、學長姐們、同學、學弟妹們，我也想同樣的感謝你們，有了你們的牽成，我的學習生涯會更好，謝謝你們。

蘇明銓

于 新竹 國立交通大學 2009.09

Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Background

PLL plays an important role in all kinds of integrated circuits, with the functions of frequency synthesis, duty-cycle correction, clock recovery and clock de-skewing. The realization of the PLL using a traditional analog architecture requires different demands on the process technology from those circuits using standard logic cells. Analog PLLs require elements that are not critical to standard logic circuits, such as resistors and low-leakage capacitors. In addition, analog PLL may use logic families like current mode logic other than static CMOS. With process evolves and grows in complexity, the challenge of maintaining performance for use in circuit increase dramatically. Furthermore, because the passive elements and logic families used by analog PLL are not likely to be reused by digital core, its yield and performance is worse as compared to those of the digital parts of the chip.

A number of all-digital PLLs have been proposed in the papers [10], [15], [18], [31], with target applications of fast lock-in time or low power. The major all-digital PLL design issues include phase error offset between reference and feedback divided clock, the algorithm of the fast lock-in process, the performance of digital loop filter, frequency resolution enhancement of digitally-controlled oscillator, and other critical concern like low-jitter and low-power considerations.

Modern high speed serial link is composed of a transmitter and a receiver and data are transmitted over different channels like PCB track, cable, fiber, etc, as shown in Fig 1.1. The transmitter generates a high bandwidth signal. A serious problem associated with high speed serial link is electro-magnetic interference (EMI). Electrical devices operating at high speed result in EMI, and such EMI may interfere with the operation of its source circuit and other equipments adjacent to the EMI source. Since heavy metal shielding is not a low-cost option in the lightweight portable device, spread spectrum technique [28], [29], [36] has been frequently used for EMI reduction.

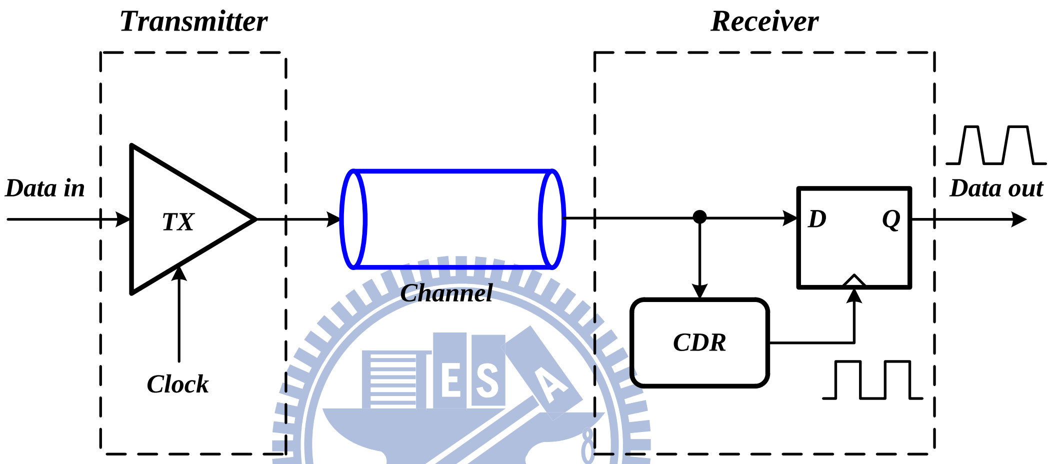


Fig 1.1 High speed serial link block diagram

1.2 Motivations and Goals

The demands on high data-rate and integration of modern high speed serial link activate the design of clock generator circuit being able to work at multi-Giga bits/s. The work of this thesis is motivated in two ways, one is to design a low-jitter all-digital PLL with working frequency of 1GHz or 1.2 GHz, and the other is to fulfill the spread-spectrum clocking technique referenced to Serial-ATA III(6Gbps data rate) and USB3.0(5Gbps data rate) specifications.

The proposed all-digital PLL is a PFD based all-digital PLL, with 10 or 20 phase outputs. A ΣΔ modulator is used to control DCO to enhance all-digital PLL frequency resolution. A low-jitter ADPLL is desired for spread-spectrum clocking application.

The goal of spread-spectrum clock generator (SSCG) is to design an all digital SSCG with small area and high EMI reduction.

1.3 Thesis Organization

Chapter 1 gives a brief introduction to the design demands of all-digital PLL and spread-spectrum clocking.

Chapter 2 describes spread-spectrum clocking theory. To reduce EMI effect, spread-spectrum clocking is used in wire-line communication. In this work, 1st order ΣΔ modulator is adopted for switching phase mechanism.

Chapter 3 reviews different kinds of all-digital PLL design. Digitally-controlled oscillator (DCO) is the most important block in all-digital PLL design. DCO is composed of delay cells. Chapter 3 will make comparisons of those delay cells claimed to have fine frequency resolution. Some all-digital PLL architecture are introduced like TDC-based all-digital PLL.

Chapter 4 brings up the proposed all-digital PLL. Initially, an overview of this all-digital PLL is discussed. Next, the components of the all-digital PLL are introduced. Phase frequency detector with threshold detector determines phase lead/lag condition and detects phase difference. Digital loop filter consists of accumulators and some logic cells. Differential digitally-controlled oscillator provides 10 or 20 output phases and has frequency range that covers 1GHz and 1.2GHz when process variation is considered. A divide-by-10 frequency divider is used. Finally, ΣΔ modulator is used to enhance the frequency resolution of DCO.

Chapter 5 shows the experimental results of the spread-spectrum clock generator and Chapter 6 draws a conclusion.

Chapter 2
Spread-Spectrum Clocking

2.1 Background: EMI Problem

All electronic devices operating at faster speed result in more Electromagnetic Interference (EMI). EMI emission can cause more electronic devices to interfere with each other and degrade their performance and operation. As the data rate of serial links increases, the clock source must increase its working frequency. Since frequency sources such as crystal oscillator, phase-locked loop frequency synthesizer or other clock generators are major sources of EMI in electric circuits, designers utilizing these above clock generation schemes must consider EMI effect.

The Federal Communications Commission (FCC) in the United States has regulation rules about the maximum power of EMI. The FCC's regulation has been divided into Class A and Class B for the electronic products. Industrial applications use the FCC's Class A regulation; while residential and consumer applications use the FCC's Class B regulation. This certification shows the electronic device conforms to standards which limit the amount of EMI that one can produce.

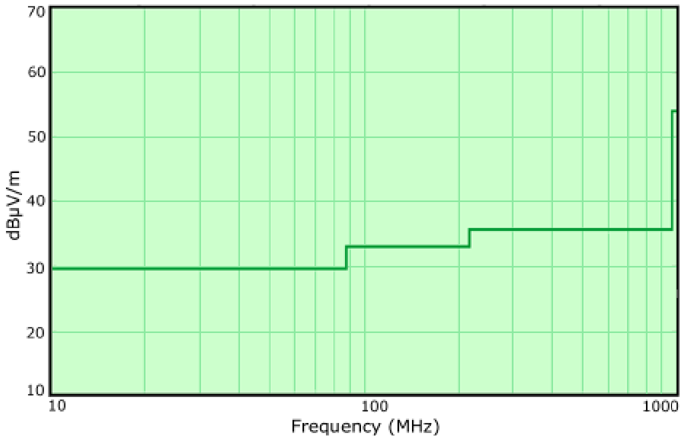


Fig 2.1 FFC class B EMI peak emission limit [4]

Fig 2.1 shows a FCC Class B plot of power (dBμV/m) versus frequency (MHz) for the peak emission requirements (at 10 meters).

Nowadays, FCC regulations only pay attention to peak emission power at given frequency but not the average emission power over the given frequency spectrum. Therefore, one circuit designer should focus on reducing peak EMI emission at any given frequency within the spectrum.

Today, several methods are developed to solve EMI problem, like metal shielding, multi-layer printed circuit boards, special casing, passive components, pulse shaping, slew-rate control, layout technique, and the spread spectrum clocking.

Spread spectrum clocking (SSC) scheme modulates the given frequency of the clock source slightly and so the energy of the given frequency on the spectrum will be dispersed to a controllable small range. Hence, the peak emission power within the spectrum degrades, and the EMI effect is weakened. This method is most popular and is our subject in this thesis.

2.2 Concept of Spread-Spectrum

Spread spectrum clocking (SSC) slightly modulates the frequency of the clock so as to spread its power over a range of frequencies on the spectrum such that the average power emitted at a specified frequency is reduced as shown in Figure 2.2.

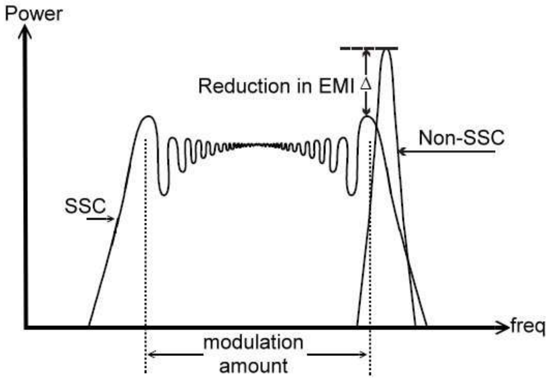


Fig 2.2 SSC frequency domain view

In general, spread spectrum clocking can be classified into three kinds of modulation modes: center-spread, up-spread and down-spread, which are based on the spreading frequency compared with nominal frequency. We adopt the down-spread mechanism according the specification of SATA-III and USB 3.0.

Fig 2.3 shows the triangular profile modulation in spread spectrum clocking. The triangular profile modulation will have the best EMI reduction because its frequency deviation is regular in a fixed time, so that the peak power of the spreading frequency can be reduced as a plane among the spectrum like that shown in Fig 2.2. Here we have some spread spectrum clocking parameters: f_{nom} is the nominal frequency of the clock, δ is the spread spectrum clocking frequency deviation and $\frac{1}{f_m}$ is the spread spectrum clocking modulation rate. The modulation frequency should be as low as possible due to the timing jitter issue. It is hard for clock data recovery (CDR) to recover data if the timing jitter is too large.

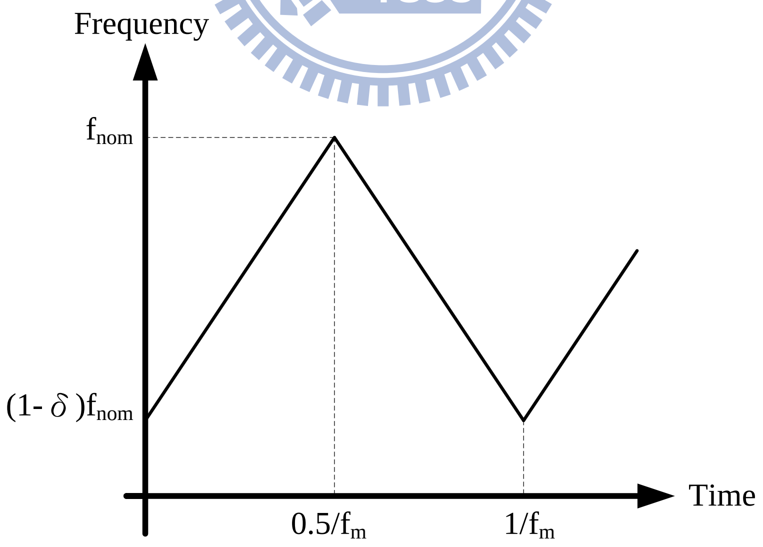


Fig 2.3 Triangular profile modulation

A widely adopted SSC profile proposed in an industry standard, Serial AT Attachment (SATA) [33], is shown in Fig 2.4. We choose the nominal clock frequency at 1.2GHz as the non-spread spectrum clocking frequency. The maximum spread spectrum clocking frequency deviation is 5000ppm and the modulation rate is 30~33 kHz as defined in SATA-III(6Gbps data rate) and USB 3.0(5Gbps data rate).

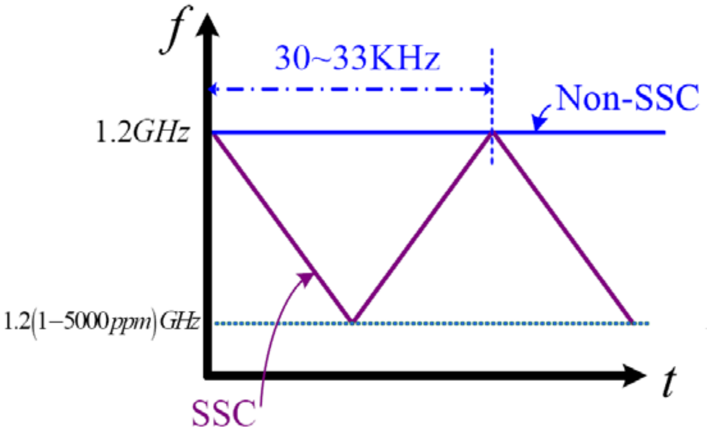


Fig 2.4 SATA-III triangular profile modulation [33]

As shown in Fig 2.3 the down spread technique is a way that the nominal clock frequency being moved below the nominal frequency between f_{nom} and (1-δ) f_{nom}, where δ is the maximum spread spectrum clocking frequency deviation with amount of 5000*ppm* down spread, and f_m is modulation frequency of 30~33 kHz respectively.

2.3 Phase Rotation Mechanism for Spread-Spectrum Clocking

In this section, we will overview the static analysis of using phase rotation to achieve spread-spectrum clocking. Fig 2.5 shows the timing diagram of 10(P) phases from DCO. The phase difference between any two adjacent phases is the same.

If the period of each phase is T_{DCO}, then the phase difference of two adjacent phases is $T_D = \frac{T_{DCO}}{10} = \frac{T_{DCO}}{P}$, where P is the number of phases provided by DCO. The output of DCO is fed into frequency divider which has a divide-ratio of N. The period of the frequency divider output without phase rotation is $N \times T_{DCO} = T_{REF}$. If the input of the frequency divider rotates one phase in one T_{REF}, the period of the frequency divider output will be $\left(N+\frac{1}{P}\right)\times T_{DCO}=\left(1+\frac{1}{N\times P}\right)\times N\times T_{DCO}=\left(1+\frac{1}{N\times P}\right)\times T_{REF}$.

The timing diagram is shown in Fig 2.6.

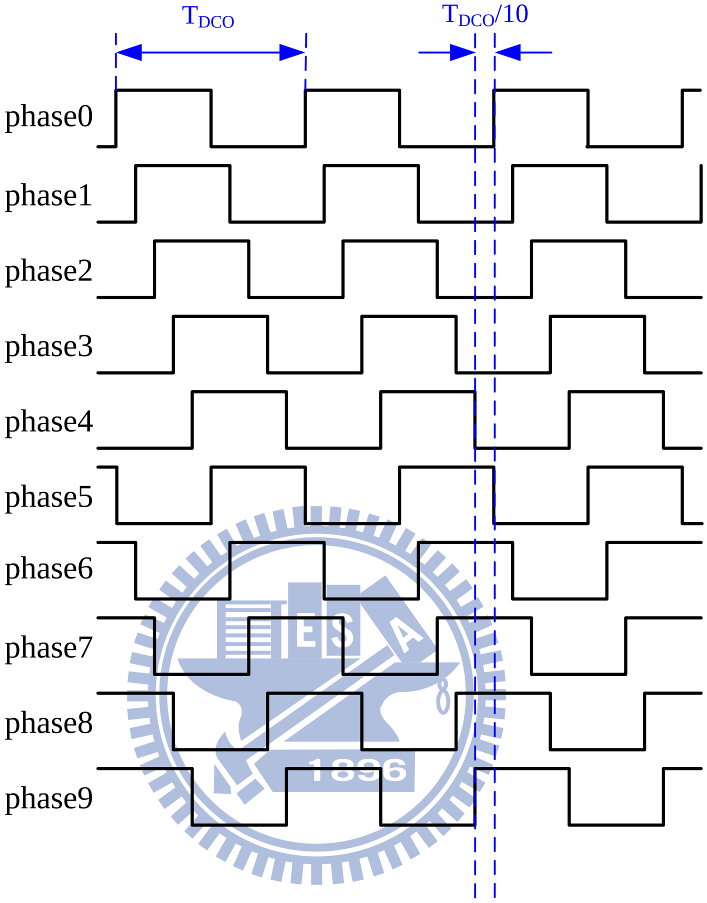


Fig 2.5 Timing diagram of 10 phases from DCO

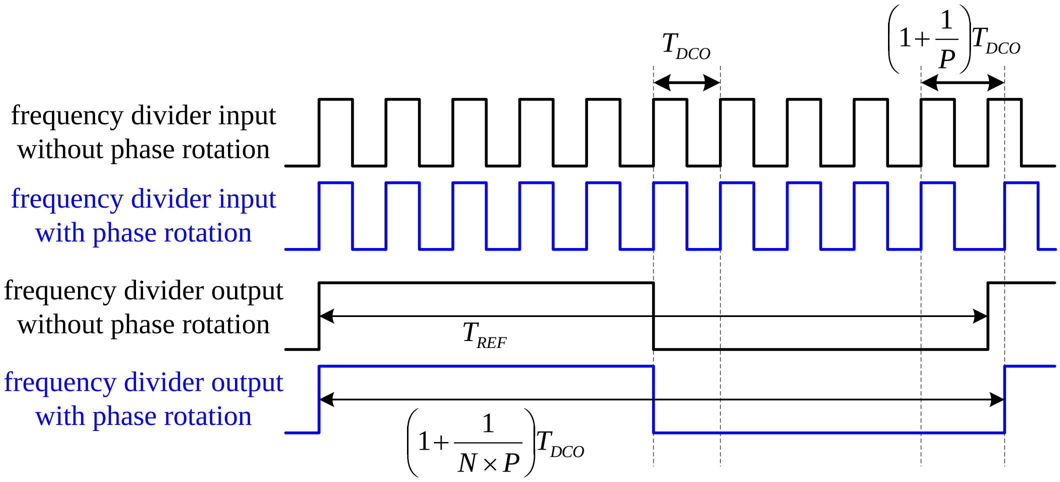

T_{DCO}
$\left(1+\frac{1}{P}\right)T_{DCO}$
frequency divider input without phase rotation
frequency divider input with phase rotation
frequency divider output without phase rotation
T_{REF}
frequency divider output with phase rotation
$\left(1+\frac{1}{N\times P}\right)T_{DCO}$

Fig 2.6 Timing diagram of phase rotation

As shown in Fig 2.6, the period of the output from the frequency divider with phase rotation is longer than that without phase rotation (T_{REF}). When the output from the frequency divider is sent back to the PFD, it will send out a “lag” signal so that the digital loop filter (DLF) will change the control code of the DCO which will speed up the DCO’s frequency, and the period of the frequency divider will decrease as well as the period of the DCO. The above mechanism is a transient behavior. When the ADPLL is stable, the period of the frequency divider’s output should be the same as T_{REF}. Thus, the new T_{DCO} must be less than the original T_{DCO} under the same T_{REF}. This method can be used to vary the oscillation frequency of the DCO.

Fig 2.7 shows the basic architecture of the phase rotation ADPLL for spread-spectrum clocking. When the ADPLL is locked, the frequency divider has the same frequency as F_{REF}. And $F_{DCO} = F_{REF} \times \left(N + \frac{\alpha}{P}\right) = F_{DCO\_original} \times \left(1 + \frac{\alpha}{N \times P}\right)$, where α is the phase rotation number. A multiplexer (MUX) is placed into the feedback path between the DCO and the frequency divider, and this MUX is controlled by phase rotation control logic circuit.

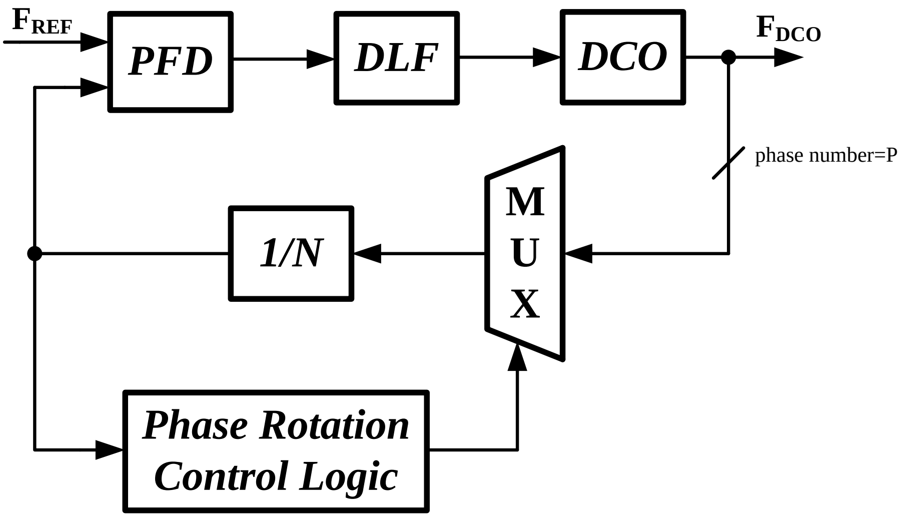


Fig 2.7 Phase rotation ADPLL for spread-spectrum clocking

For M sequence of F_{REF}, if the MUX rotates one phase for A sequence and do not rotate in the other (M-A) sequence, then the output frequency of the DCO can be derived as follows:

$$\left[A\times\left(1+\frac{1}{N\times P}\right)+(M-A)\times 1\right]\times\frac{N}{F_{DCO}}=\frac{M}{F_{REF}} \tag{2.1}$$

$$\therefore F_{DCO}=F_{REF}\times N\times\frac{\left[A\times\left(1+\frac{1}{N\times P}\right)+(M-A)\times 1\right]}{M}$$

$$=F_{REF}\times\left(N+\frac{\frac{A}{M}}{P}\right)=F_{REF}\left(N+\frac{\alpha}{P}\right) \quad , M>A \tag{2.2}$$

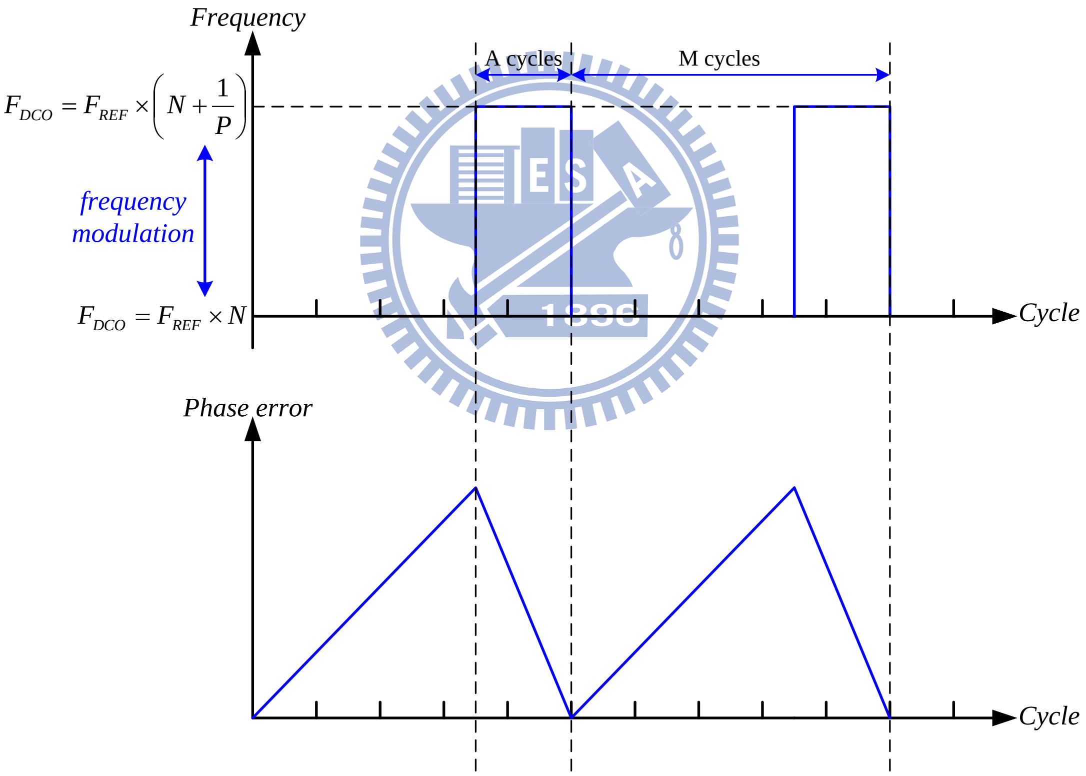

Frequency
A cycles
M cycles
$F_{DCO}=F_{REF}\times\left(N+\frac{1}{P}\right)$
frequency modulation
$F_{DCO}=F_{REF}\times N$
Cycle
Phase error
Cycle

Fig 2.8 Periodic phase error accumulation phenomenon

The equivalent rotation ratio is $\alpha=\frac{A}{M}$. This value can vary between 0 and 1 by proper choice of A and M. There is one problem when the phase rotation mechanism is used. If we use counters to control the MUX, there is a periodic phase error accumulation phenomenon, as shown in Fig 2.8.

For the first (M-A) output pulses from the frequency divider, the MUX rotates no phase and the phase error accumulates. After MUX rotates one phase in “A” cycles, the phase error would be gradually compensated.

Since the phase error would yield phase rotation spurs per $\frac{F_{REF}}{M}$ offset from the carrier frequency, and such phase rotation spurs will decrease the quality of spread spectrum clock generator, it should be reduced as much as possible. The most popular method for solving this problem is to use the ΣΔ modulator. It will be explained in the next section.

2.4 ΣΔ Modulator

ΣΔ modulators are well widely used in communication and especially for A/D and D/A conversion applications. The objective of ΣΔ modulators is to shape the quantization noise spectrum such that a small amount of noise power remains within the useful signal band while the rest of the quantization noise pushed to higher frequency band.

In the previous section, the phase rotation spurs originate from the regular sequence of the MUX should be eliminated as much as possible. Hence, ΣΔ modulator can be used to randomize the choice of phase rotation while F_{DCO} is still given by $F_{REF} \times \left(N + \frac{\alpha}{P} \right)$ as Eq. (2.2) shows. This means that individual multiplier factor occurs for only short period of time, the systematic fractional sideband would be converted to random noise. Further, one can shape the noise spectrum so that most of the noise’s energy could be moved to higher frequency offset. As a result, the noise in the vicinity of the F_{REF} tone is adequately small. Moreover, the noise that is put into high frequency offset can ideally be suppressed by the inherent low pass response of the ADPLL, as shown in Fig 2.9.

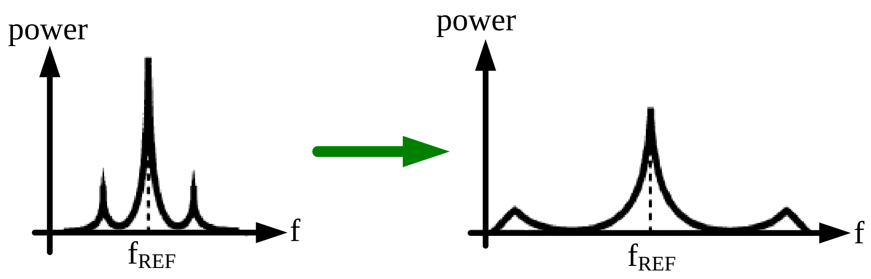


Fig 2.9 Noise shaping for eliminating spurs induced by randomization

With the use of ΣΔ modulator [5], the phase-selection signal used to control the MUX has pseudo-random sequence and the quantization noise is differentiated in the signal band. The quantization noise results from that the phase-selection signal can only change from i^{th} phase to $(i+1)^{th}$ phase but not fractional number, where i ranges from 0 to P-1 and P is the output phase number of DCO. Consequently, the ideal phase rotation of each selection is Y while it is quantized to either 0 or 1 (denoted as d(t)). Fig 2.10 shows one sequence example of d(t). Y is a time average value of d(t).

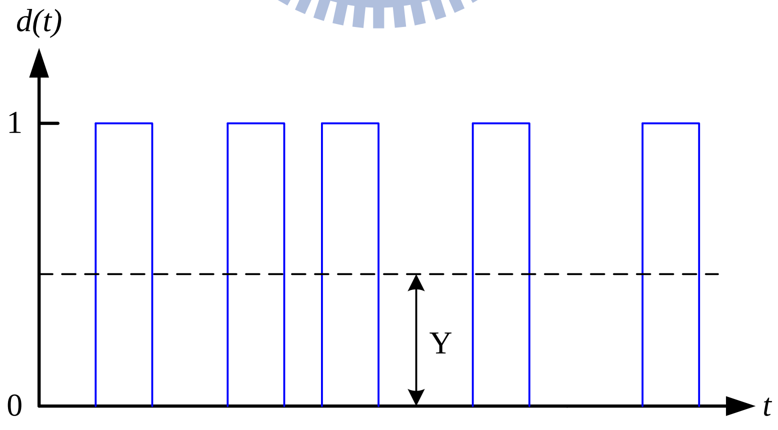


Fig 2.10 Waveform of d(t)

The implementation of ΣΔ modulator can be achieved by accumulator and D Flip-Flops (DFF). Fig 2.11 shows the architecture of the 1^{st} order ΣΔ modulator.

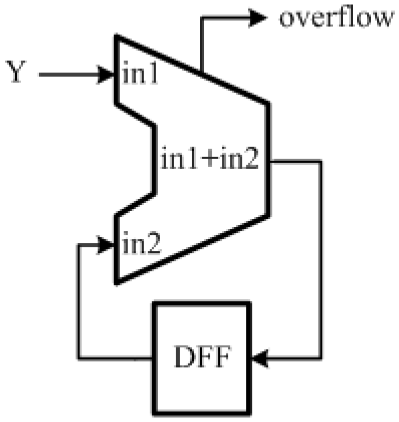


Fig 2.11 Architecture of 1st order ΣΔ modulator

The input normalized DC value of in1 is $Y = \frac{A}{2^X} = \frac{A}{M}$ and the overflow is either 0 or 1, where A is the input of the accumulator, X is the bit number of the accumulator and M is the maximum number of the accumulator. As shown in Fig 2.12, the ΣΔ modulator's overflow is sent to phase rotation control logic circuit to generate the phase-selection signal used to control the MUX for phase rotation mechanism.

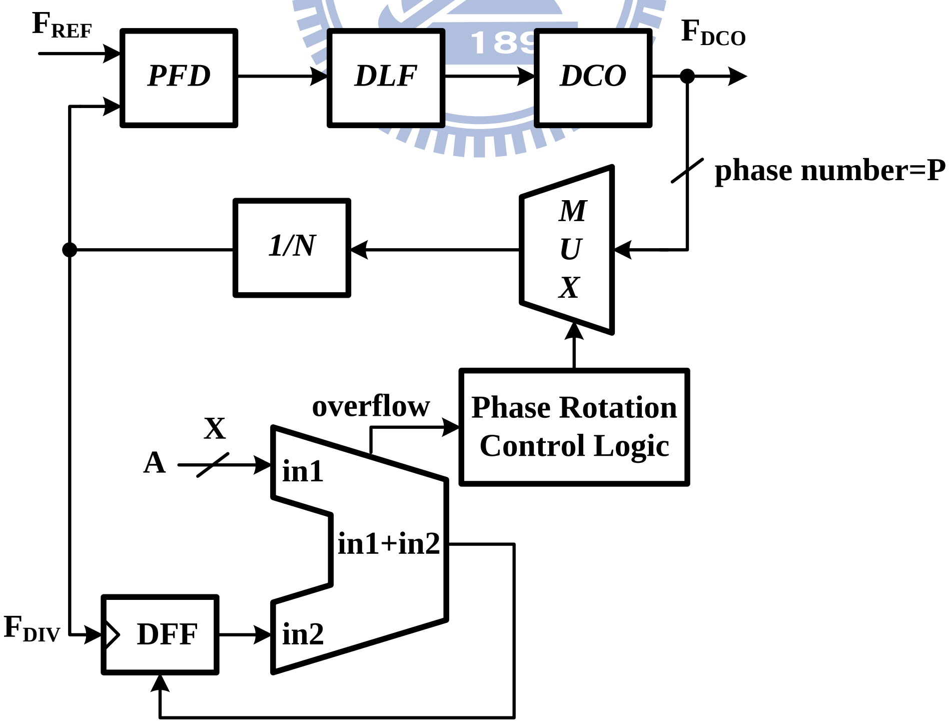


Fig 2.12 Realization of ΣΔ modulator for phase rotation mechanism

For an X-bit accumulator, the accumulator would generate an overflow of average value $\frac{A}{2^X}$ at every F_{DIV} clock. Therefore, the average number of phase rotation is $\frac{A\times\left(1+\frac{1}{P}\right)+\left(2^X-A\right)\times 1}{2^X}=1+\frac{A}{2^X\times P}$.

In order to obtain better noise shaping toward higher frequency offset, the higher order ΣΔ modulator is required. Fig 2.13 shows the implementation of 2nd order ΣΔ modulator.

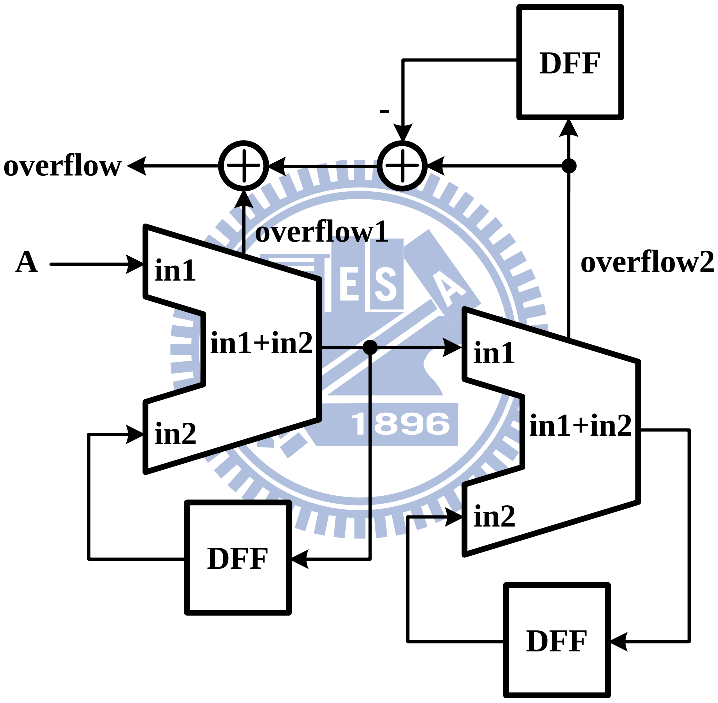


Fig 2.13 Implementation of 2nd order ΣΔ modulator

Here we use 1st order ΣΔ modulator to control the number of phase rotation. The reason of choosing 1st order ΣΔ modulator but not higher order ΣΔ modulator is that SATA-III specification must have down spread frequency modulation. The output of 2nd order ΣΔ modulator are -1, 0, 1, 2, and the "-1" term will have up spread frequency modulation even though the average is down spread. All of them could not plus 1; otherwise, much more jitter would be produced. For example, if the original output is 2, it would become 3 after it adds 1. It means that MUX will rotate 3 phases in one PFD comparison, so the spread spectrum modulation deviation would be $\frac{\alpha}{N\times P}=\frac{3}{10\times 20}=15000\,ppm$ and is beyond the scope.

2.5 Timing Jitter Issue

In the SSCG system, the output frequency is spread over time. If the transmitter uses the SSCG system to generate the clock, then the data is affected by the clock's performance. For example, as shown in Fig 2.14, the clock's waveform would vary over time.

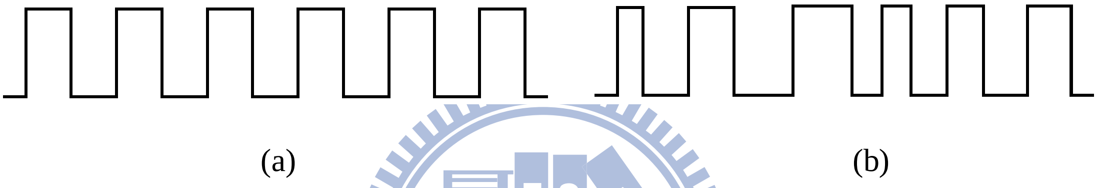


Fig. 2.14 Clock's Waveform in Time Domain (a) without frequency modulation

(b) with frequency modulation

A. Cycle-to-Cycle Jitter

The cycle-to cycle jitter can vary over time, because its measurement depends on the relationship between the present cycle time and the previous cycle time. Fig 2.15 shows the relationship between cycle time and jitter.

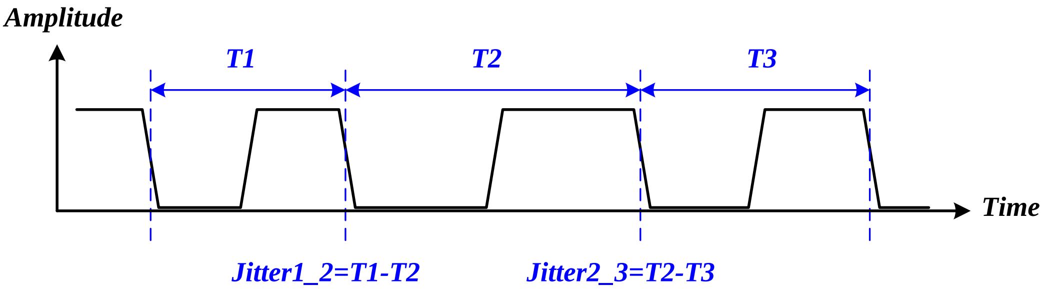


Fig 2.15 Cycle-to-cycle jitter diagram

Below we derive the formulas about cycle-to-cycle jitter. The period difference between the normal frequency and the maximum modulated frequency is:

$$\Delta T_{total}=\frac{1}{(1-\delta)f_{normal}}-\frac{1}{f_{normal}}=\frac{\delta}{(1-\delta)f_{normal}}\approx\frac{\delta}{f_{normal}} \tag{2.3}$$

δ is the frequency modulation deviation, and f_{normal} is the normal frequency.

The number of cycles (N) that exists in the time interval that the modulated clock moves from f_{normal} to (1-δ) f_{normal} is:

$$N=\frac{f_{avg}}{2f_m} \tag{2.4}$$

Where f_{avg} is the average frequency of the spread spectrum clock, and f_m is the modulated frequency. According to triangular modulation profile, we could derive the average frequency as

$$f_{avg}=(1-0.5\delta)f_{normal} \tag{2.5}$$

Therefore, the cycle-to-cycle jitter induced from spread spectrum clock can be expressed as

$$\Delta T_{cycle-to-cycle}=\frac{\Delta T_{total}}{N}=\frac{\delta}{f_{normal}}\cdot\frac{2f_m}{(1-0.5\delta)f_{normal}}=\frac{2f_m\delta}{(1-0.5\delta)f_{normal}{}^2} \tag{2.6}$$

For a 1.2GHz spread spectrum clock with 0.5% triangular modulation and 31.25 KHz modulation frequency, the cycle-to-cycle jitter is:

$$\Delta T_{cycle-to-cycle}=\frac{2\times31.25\times10^3\times0.5\%}{(1-0.5\times0.5\%)\times\left(1.2\times10^9\right)^2}=2.175578\cdots\times10^{-16}\approx0.218(fs) \tag{2.7}$$

B. Long-Term Jitter

Long-term jitter measures the maximum change in a clock's output transition from its ideal position. Fig 2.16 shows the diagram of peak-to-peak jitter. As a result, equation (2.3) can be viewed as the long-term jitter of a down-spreading clock.

Analogously, for a 1.2GHz spread spectrum clock with 0.5% triangular modulation and 31.25 KHz modulation frequency, the long-term jitter is:

$$\Delta T_{total} = \frac{\delta}{f_{normal}} = \frac{0.5\%}{1.2\times 10^{9}} = 4.166\cdots\times 10^{-12} \approx 4.2(ps) \tag{2.8}$$

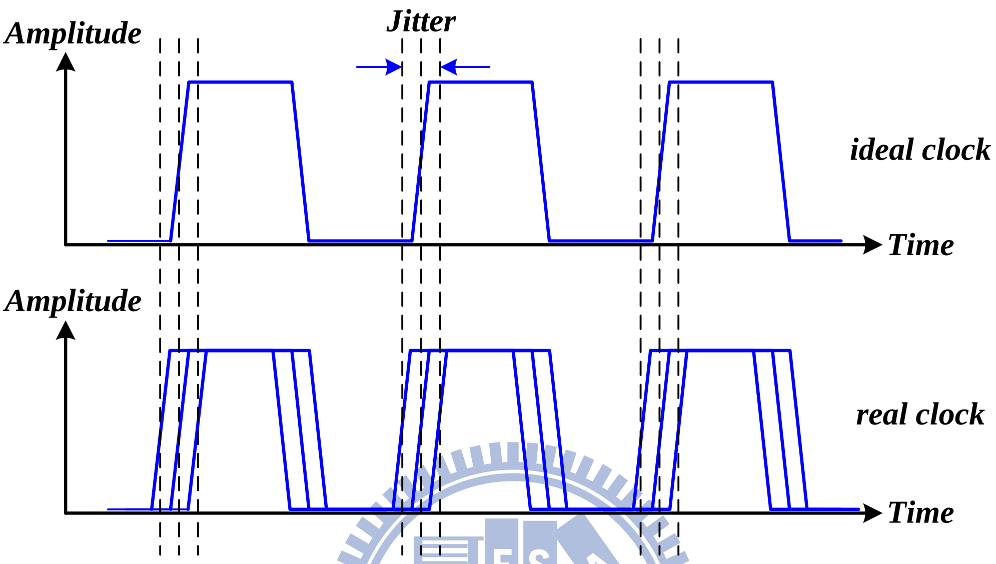


Fig 2.16 Diagram of peak-to-peak jitter

2.6 Conclusions

In this chapter, we introduce the EMI problem and spread-spectrum clocking is used to reduce the EMI. Phase rotation mechanism is applied to the proposed ADPLL to construct the spread-spectrum clock generator (SSCG). The SSCG can turn on or turn off its spread-spectrum clocking under the "switch" signal. Also, the SSCG can choose 10 phases or 20 phases spread-spectrum clocking under the "select" signal. Our goal is to design an all-digital SSCG with small area, less power consumption and good EMI reduction.

Chapter 3
All-Digital Phase-Locked Loop Basics

3.1 Introduction

Phase-locked loops (PLLs) have been used in different kinds of IC for many years. PLLs are widely used in frequency synthesis and timing recovery filed, especially for the serial link transceiver.

To date, most PLLs are analog feedback circuits that have a particular system to track with reference signal. In other words, PLLs generate an output clock that is synchronized with a reference clock in frequency as well as in phase. Once the PLLs are locked, the phase error between the frequency divider output clock and the reference clock is ideally zero.

The main applications of PLL are as follows:

1. Clock generation: Most electronic systems have processors operating at various clock frequencies. PLLs serve to provide them with clock frequency. Usually, a lower reference clock frequency (usually 50 or 100 MHz) is multiplied by N (multiplication factor), so as to generate a higher operating frequency the processor needs. The multiplication factor can be an integer or a fractional number.

2. Spread spectrum: Electronic devices working at high frequency will emit some unwanted electro-magnetic wave. Such emitted electro-magnetic wave generally appears as sharp spectral peaks (usually at the device’s working frequency, and its harmonics). For an economic way to reduce EMI, PLLs are modified for spread-spectrum clocking. By spreading the sharp spectral peak energy over a wider portion of the spectrum, the EMI problem is released. For example, by changing the operating frequency up and down by a small amount (about 1%), a device running at hundreds of megahertz can spread its interference evenly over a few megahertz of spectrum, which drastically reduces the amount of noise seen by FM receivers which have a bandwidth of tens of kilohertz.

3. Clock recovery: Some data streams, especially high-speed serial data streams, (such as the raw stream of data from the magnetic head of a disk drive) are sent without an accompanying clock. The receiver generates a clock from an approximate frequency reference, and then phase-aligns to the transitions in the data stream with a PLL. In order for this scheme to work, the data stream must have a transition frequently enough to correct any drift in the PLL's oscillator. Typically, some sort of redundant encoding is used; 8B10B is very common.

4. De-skewing: If a clock is sent in parallel with data, that clock can be used to sample the data. Because the clock must be received and amplified before it can drive the flip-flops which sample the data, there will be a finite, and process-, temperature-, and voltage-dependent delay between the detected clock edge and the received data window. This delay limits the frequency at which data can be sent. One way of eliminating this delay is to include a de-skew PLL on the receive side, so that the clock at each data flip-flop is phase-matched to the received clock.

3.2 All-Digital Phase-Locked Loop

Many kinds of all-digital PLLs are presented so far, and some are suitable for specific applications. When well-controlled bandwidth is required for wireless applications, a high resolution time-to-digital converter (TDC) [6] with multi-bits is used. For the requirement on short lock times, a delicate digital search algorithm and a control scheme may be needed [7]. Fast lock-in time can also be achieved by the use of dual-loop architecture [8] (one loop for frequency acquisition, and the other for phase alignment) with different loop filter characteristics to support frequency and phase acquisition, respectively. The above architectural choices significantly increase area, power consumption and circuit complexity without commensurate benefit for the clock generation application described here.

The proposed all-digital PLL is designed for spread-spectrum clock generation, and in this application, the critical specification is peak to peak period jitter. The realized ADPLL uses single loop architecture, based on a bang-bang phase/frequency detector (bang-bang PFD). Since no fast frequency hopping is required, this work does not acquire a high resolution multi-bit TDC. The proposed DCO has differential signaling, which has at most 20 phases for requirement on spread-spectrum clocking. With the use of ΣΔ modulator, the DCO could increase its frequency resolution as required. Fig 3.1 shows the architecture of the proposed ADPLL.

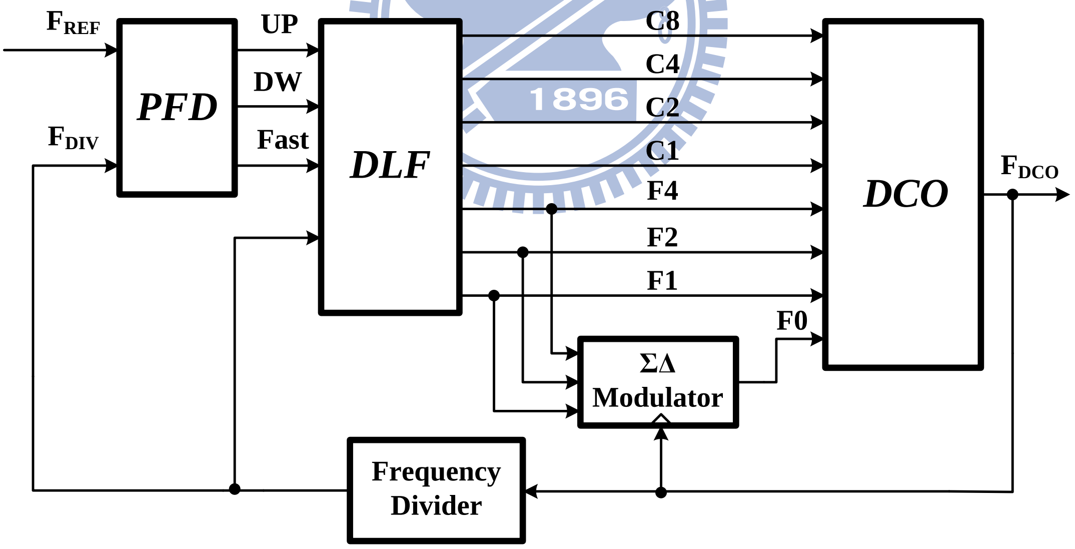


Fig 3.1 Architecture of proposed ADPLL

3.2.1 Digitally-Controlled Oscillator

The most critical component of the ADPLL is no doubt the digitally-controlled oscillator (DCO). It is impossible for the DCO to be realized in an ADPLL with extremely poor noise performance under free-running condition. Also, the tuning range and the number of frequency control bits decide the DCO's performance in a low-jitter ADPLL design.

Conventional DCO is composed of odd number of inverting cells, as shown in Fig 3.2. When changing its turn-on number of inverters, the DCO can change the path's driving ability [9], so as to change its delay time as well as frequency. It has the advantage of using all-digital standard cells. But, it has limited frequency resolution and it is not suitable for high-frequency circuit application.

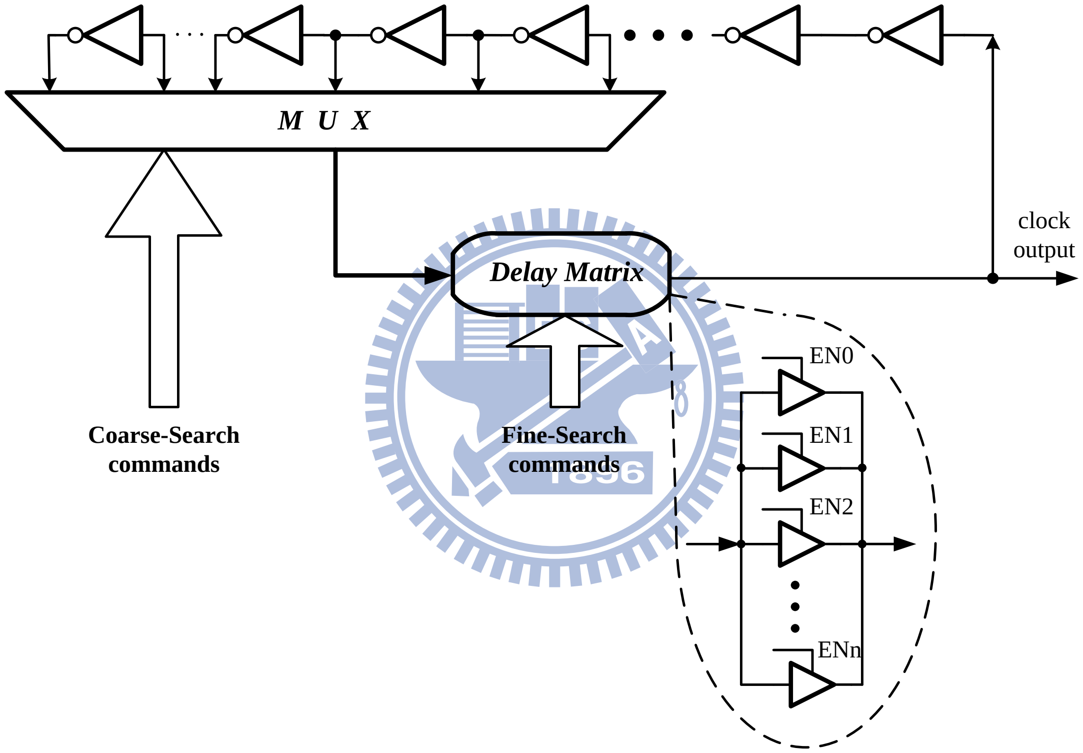


Fig 3.2 Conventional DCO structure [9]

Another DCO has two control parts, as shown in Fig 3.3(a) [10]. One is coarse tune part and is composed of a group of delay. The other is fine tune part, which is designed for enhancing the DCO's frequency resolution, as shown in Fig 3.3(b). By choosing different number of delay stages, DCO can change frequency. Using digital code to control the turn-on or turn-off conditions of the tri-state buffers, the delay time of the ring DCO can determine its oscillating frequency. However, once the inverter delay time in a fine tune stage is the finest resolution delay time that the DCO can have, the DCO could not have very precise frequency.

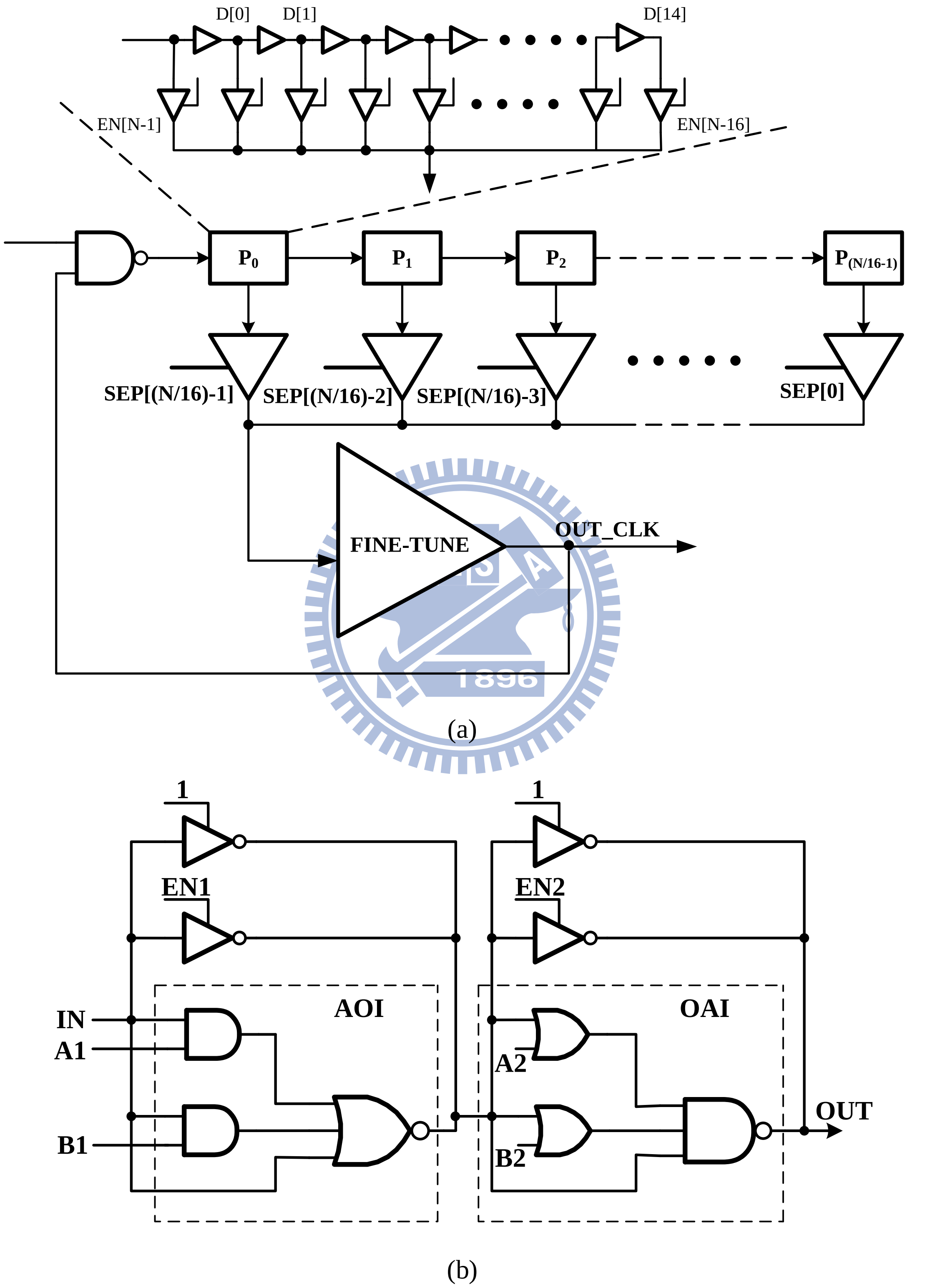
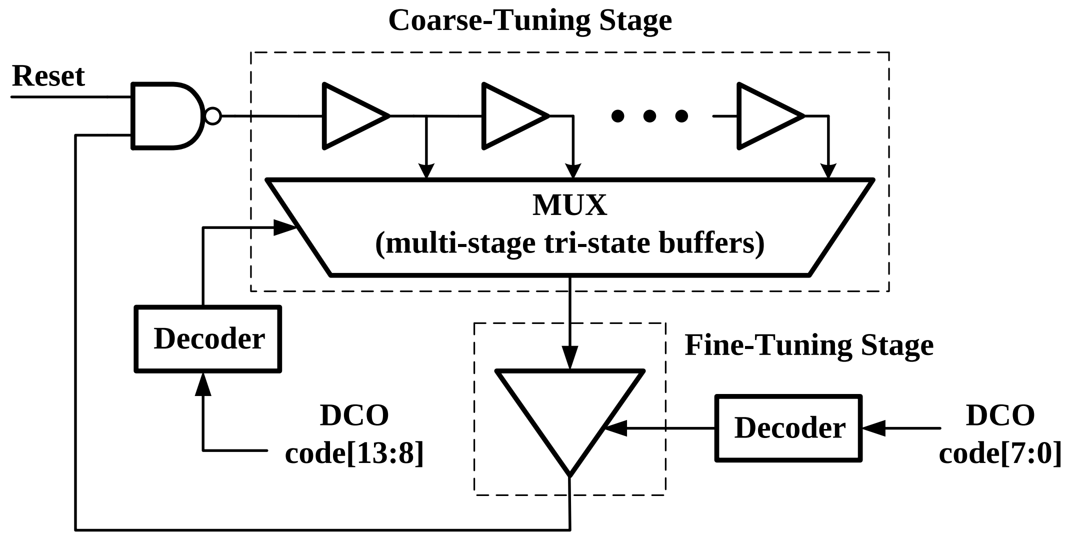
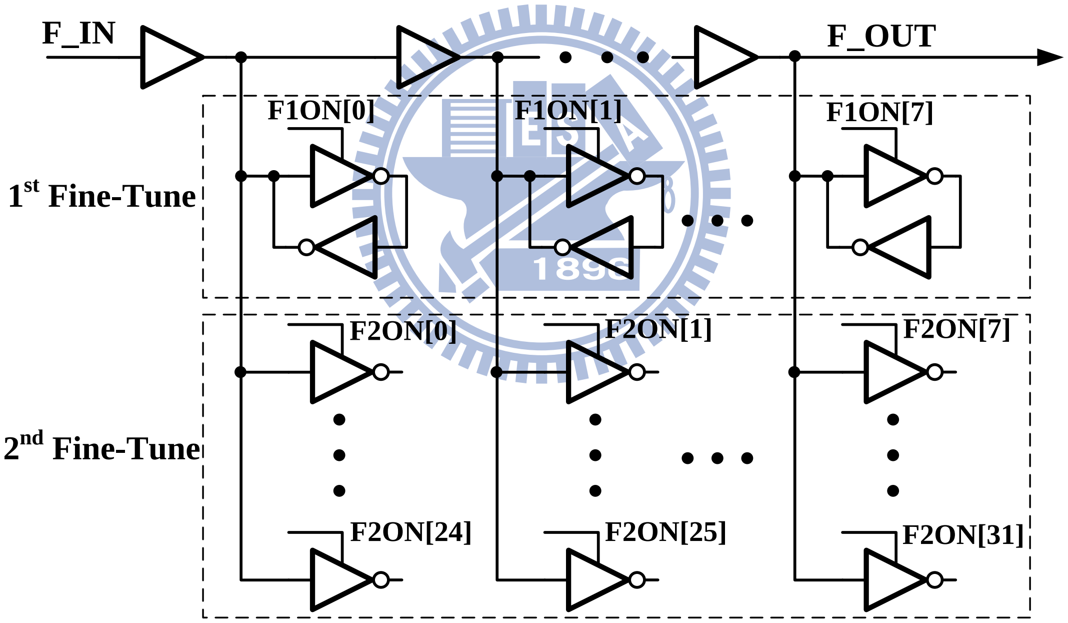


Fig 3.3(a) DCO architecture in [10] and (b) fine-tuning delay cell in [10]



(a)



(b)

Fig 3.4 (a) DCO architecture in [11] and (b) fine-tuning delay cell in [11]

The coarse tune part is used to increase adjustable frequency range, so the delay cells in the coarse tune stage may have larger delay time as compared to fine tune stage. The fine tune stage serves to enhance frequency resolution; therefore, the delay cells in fine tune stage need smaller delay time as possible.

Essentially, there are two main kinds for designing fine-tuning delay cell. One method is to change the path's driving strength dynamically with a fixed capacitance loading. The other method is to change the effective loading capacitance so as to have fine-tuning ability [11], [12], [13]. Fig 3.4(a) is the DCO architecture and Fig 3.4(b) is its fine-tuning delay cell.

3.2.2 High Resolution Delay Cell

To apply ADPLL into higher frequency systems, designing a high resolution delay cell is essentially required. In an analog delay cell, the delay time which is controlled by its operating current or voltage is a continuous-time parameter. Nevertheless, in a digital delay cell, the delay time is quantized. The ADPLL could have smaller timing jitter if the DCO could achieve as more precise frequency resolution as possible. Some examples of digital high resolution delay cell are introduced below.

One digital realization of high resolution delay cell, as shown in Fig 3.3(b), is combined by one And-Or-Inverter (AOI) cell and one Or-And-Inverter (OAI) cell [10]. The usage of two parallel tri-state inverters is to increase the delay cell's adjustable frequency range. Upon using the AOI and OAI cells can have the advantage of precisely changing delay cell's driving ability and thus can improve delay time control, but with a disadvantage of being more sensitive to power-supply variation.

Another digital realization of high resolution delay cell, as shown in Fig 3.5 [12], uses the concept of changing the number of turn-on or turn-off loading cells between inverting cells in a DCO so as to change its effective loading capacitance as well as the delay time. Due to tiny capacitance quantity change, such DCO could have resolution smaller than 1ps and more linear DCO characteristic (digitally controlled word vs. frequency range). For the sake of using of many loading transistors, the nodes within DCO may have large effective capacitance and result in large area and power consumption.

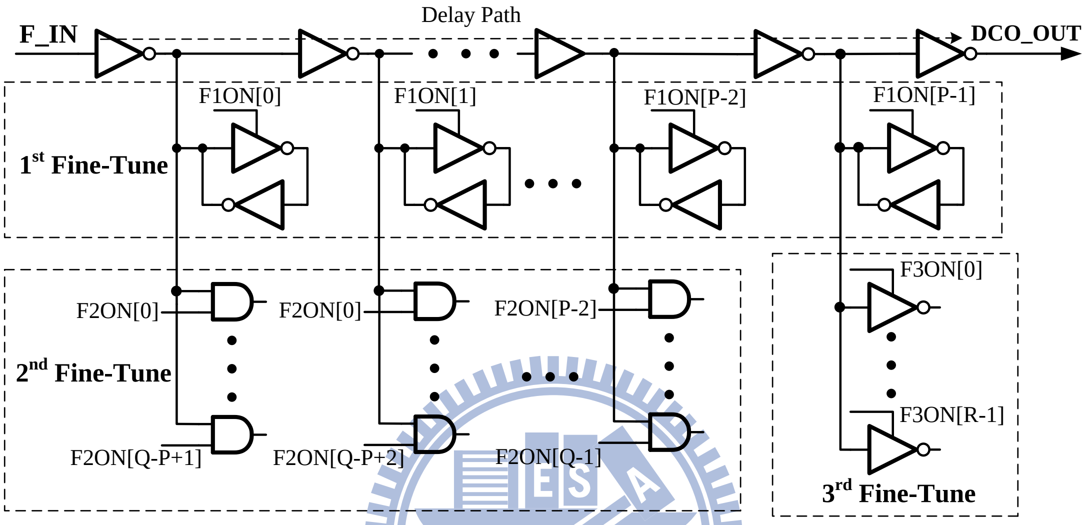


Fig 3.5 DCO fine tune stage in [12]

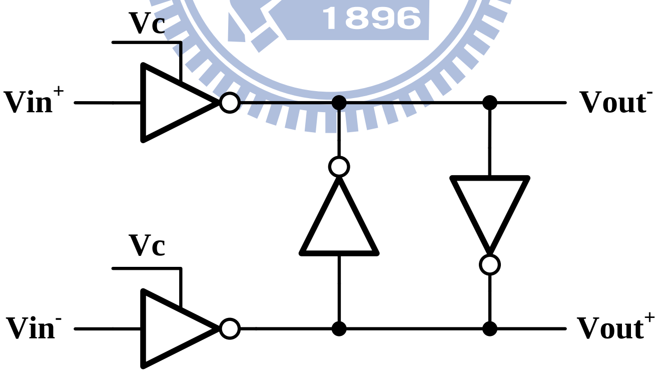


Fig 3.6 Differential delay cell [14]

In order to operate the DCO with higher frequency and have multiple phase outputs, we often use differential input/output delay cell to build a DCO [14]. As shown in Fig 3.6 is the differential delay cell. Differential input/output delay cell can construct a DCO without considering an odd or an even number of delay cells that should be used. Therefore, an even number of multiple phase outputs can be achieved.

3.3 All-Digital PLL with Time-to-Digital Converter

Fig 3.7 shows the TDC-based ADPLL block diagram from [2]. The ADPLL uses the time-to-digital converter (TDC) to detect not only the phase lead/lag information but also the phase difference between the reference clock and the feedback clock. Also, the phase difference is quantized into digital codes and these digital codes are sent to the cascaded digital loop filter, which is designed case by case. Then the digital loop filter manipulates one set of new DCO control words to control the DCO. This ADPLL features faster dynamics and is used where fast frequency and phase acquisition are required.

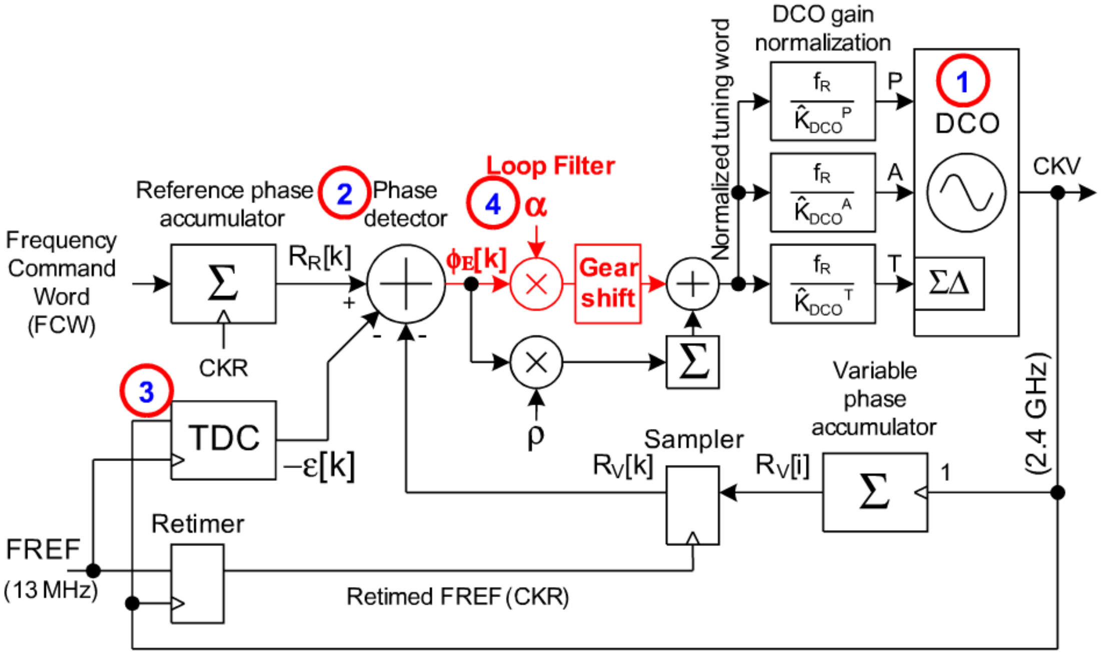


Fig 3.7 TDC-based ADPLL [2]

Fig 3.8 shows the time-to-digital converter from [2]. The TDC has quantized phase detector with resolution of 20ps. The DCO sends a clock and it passes through the inverter chain. Then, the delayed outputs are sampled by reference clock and generate a group of digital codes.

Because the TDC's finest resolution decides the minimum phase difference that the ADPLL can detect, the delay inverters and DFFs need to be designed carefully.

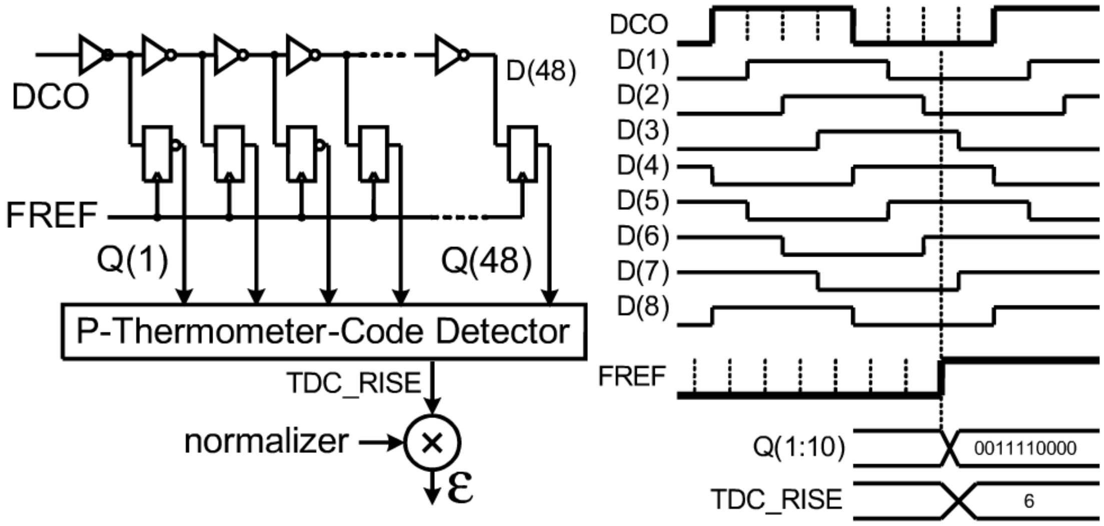


Fig 3.8 Time-to-digital converter (TDC) [2]

The TDC's design can be affected by the DCO frequency and the delay of the inverters in TDC. When the DCO frequency decreases (which means the period of DCO clock increases) and the delay of inverters in TDC is constant, then the number of inverters needed by TDC's delay chain should increase for the sake of covering one full DCO clock cycle. Increasing the number of delay chain inverters can increase the TDC's power consumption. Furthermore, when the inverter delay decreases, the TDC can have less quantization noise; however, the number of inverters needed by TDC's delay chain should increase for the sake of covering one full DCO clock cycle and this can also increase TDC's power consumption.

Chapter 4
Proposed All-Digital Phase-Locked Loop

4.1 Architecture Briefs

Fig 4.1 shows the architecture of the proposed low-complexity ADPLL. This work is simply composed of phase/frequency detector (PFD), digital loop filter (DLF), frequency divider and the digitally-controlled oscillator (DCO). A ΣΔ modulator is used so that the DCO can have more precise frequency resolution. The DCO can have at most 20 phase outputs as needed by spread-spectrum clocking.

Based on SATAIII (data rate: 6Gbps) or USB3.0 (data rate: 5Gbps) specifications, the ADPLL should be able to lock at 1.2GHz or 1GHz clock frequency, which shall have 5 phases to reach the so-called data-rate. The divider ratio is 10 and is determined according to the spread spectrum clocking modulation deviation.

The phase/frequency detector (PFD) can not only detect the phase difference information (phase lead or phase lag) between the reference clock (F_{REF}) and the feedback clock (F_{DIV}) but can also detect the phase difference extent and represent this condition using one control bit, “Fast.” This seems like a one-bit time to digital converter.

After each comparison in PFD, the DLF is sampled by “F_{DIV}” clock and changes its control bits (C8, C4, C2, C1, F4, F2, and F1) based on the latest information (UP, DW, and Fast). The three fine-tune bit, F4, F2, and F1 are sent into 1st order ΣΔ modulator, which is sampled by “F_{DCO}” clock, to generate one dithering bit, F0.

Totally, there are 8-bit control codes to the DCO and dominate its oscillation frequency.

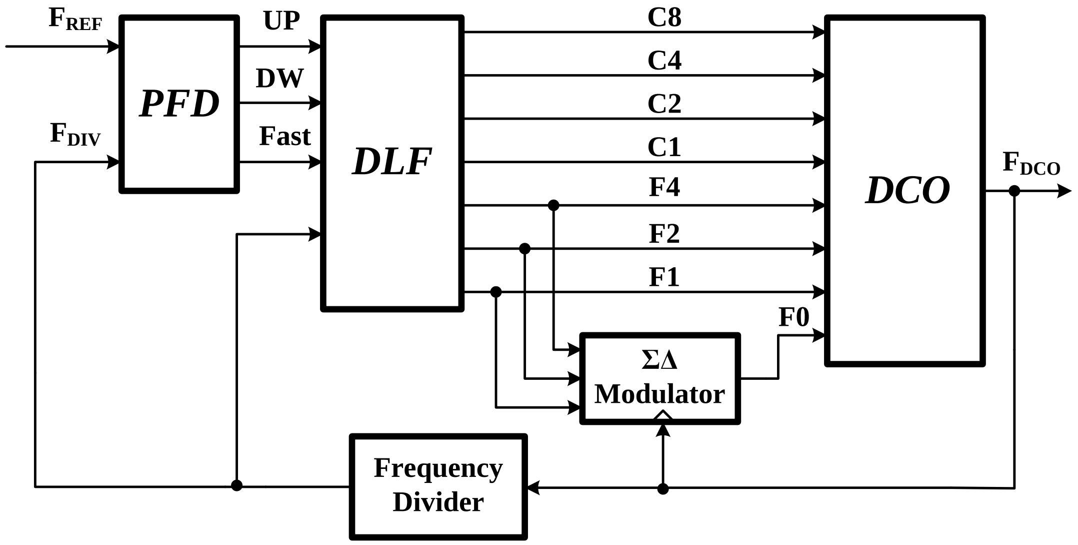


Fig 4.1 ADPLL block diagram

The DCO is a differential signaling architecture [14] [31], which has ten stages, and has at most 20 output phases. Owing to the DCO's differential signaling architecture, these 20 output phases can have more precise duty cycle and more uniform phase difference between two adjacent phases. The working frequency range of the DCO should cover 1.2GHz and 1GHz clock frequency for SATA-III and USB 3.0 specifications respectively. Taking into account the process variation, the DCO's frequency range should be large enough to overcome all process corners.

All the component blocks in the proposed ADPLL are built from digital logic cells, and through the customized design, this ADPLL can reach the low-complexity goal. Finally, the peak-to-peak period jitter and the power consumption are two main issues in the proposed ADPLL design.

4.2 Phase Frequency Detector with Phase Threshold Detector

Fig 4.2 shows the PFD architecture. The PFD is composed of a conventional tri-state PFD and a phase threshold detector [15]. This PFD acts as the traditional PFD would be.

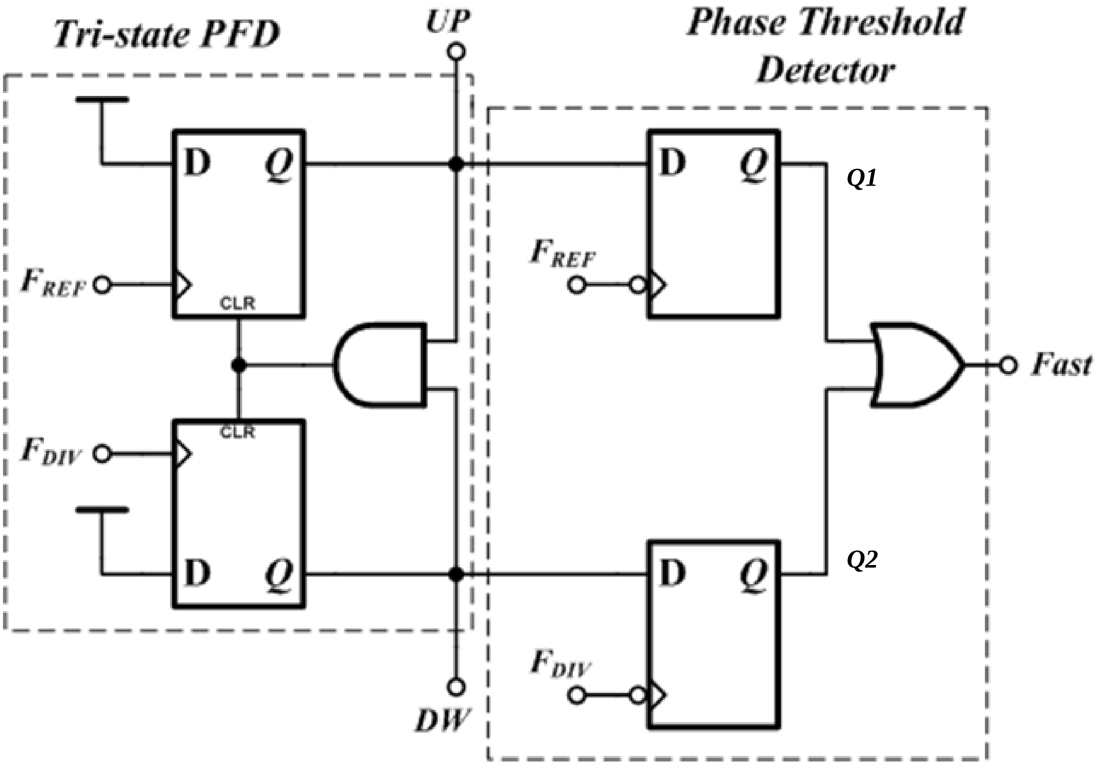


Fig 4.2 PFD architecture with tri-state PFD and phase threshold detector

Initially, the “UP” and “DW” signals are both “0”.

If the reference clock’s rising edge (F_{REF}) comes first, then the “UP” signal will become “1”. Then when the divided clock’s rising edge (F_{DIV}) comes, the “DW” signal will also become “1”. Upon this condition, the PFD’s DFFs will be reset and the “UP” and “DW” signals will return to “0”.

Oppositely, if the divided clock’s rising edge (F_{DIV}) comes first, then the “DW” signal will become “1”. Then when the reference clock’s rising edge (F_{REF}) comes, the

"UP" signal will also become "1". Similarly, the PFD's DFFs will also be reset and the "UP" and "DW" signals also return to "0" again.

The above operations of the PFD are inadequate in the lock-in process of the ADPLL. Because these operations only indicate the phase lead or lag information and are unable to show the extent of the phase difference. As a result, a phase threshold detector is added to the PFD to indicate the phase difference extent.

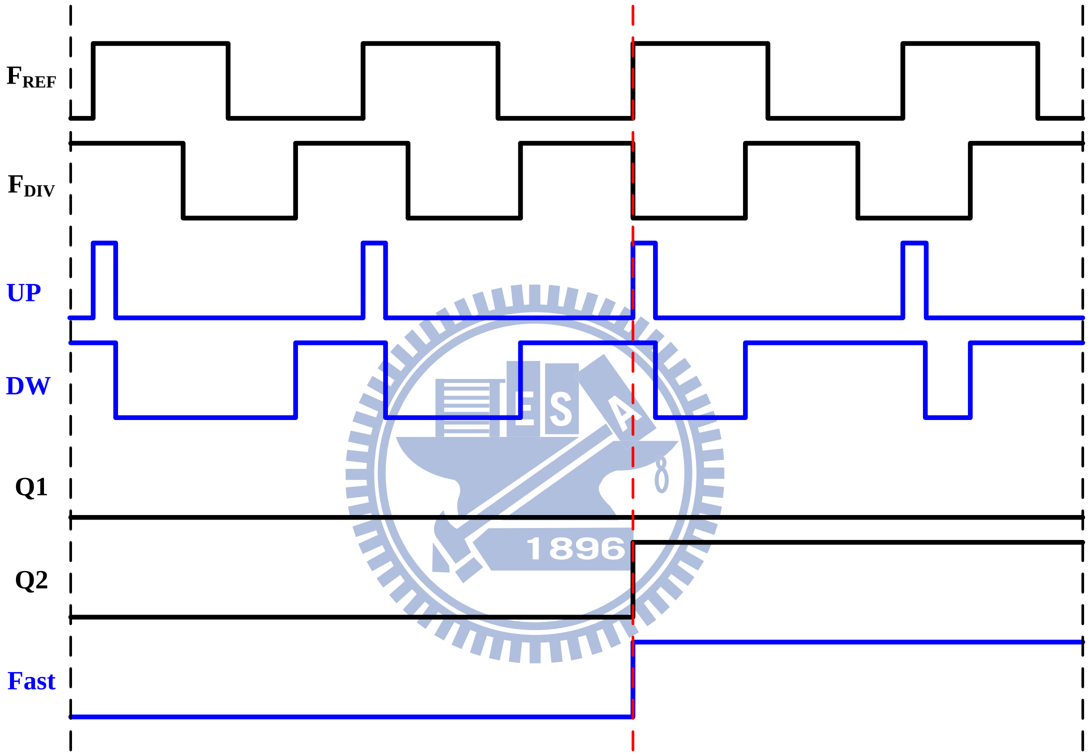


Fig 4.3 PFD's operation waveform to show the behavior of "Fast"

The proposed ADPLL has coarse tune codes and fine tune codes in DCO. In order to use coarse tune codes for the frequency acquisition and fine tune codes for the phase acquisition of the ADPLL, a phase threshold detector is applied in the PFD to generate a "Fast" signal to enable frequency acquisition to enhance the lock-in time. "Fast" signal becomes tunable when the phase difference between F_{REF} and F_{DIV} is beyond ±π, as shown in Fig 4.3. In the next section, the usage of the "Fast" signal will be explained.

In the phase threshold detector, the "UP" and "DW" signals are sampled at the falling edges of the reference clock (F_{REF}) and the divided clock (F_{DIV}) respectively. The sampled results are Q1 and Q2 respectively. These two signals are sent to the "OR" gate to generate the "Fast" signal.

4.3 Digital Loop Filter Design

Fig 4.4 shows digital loop filter architecture. The digital loop filter has three inputs: UP, DW and Fast. It generates 4-bit coarse tune codes: (C8, C4, C2, C1) and 3-bit fine tune codes: (F4, F2, F1). These control codes are sent into the DCO to generate the necessary oscillating frequency.

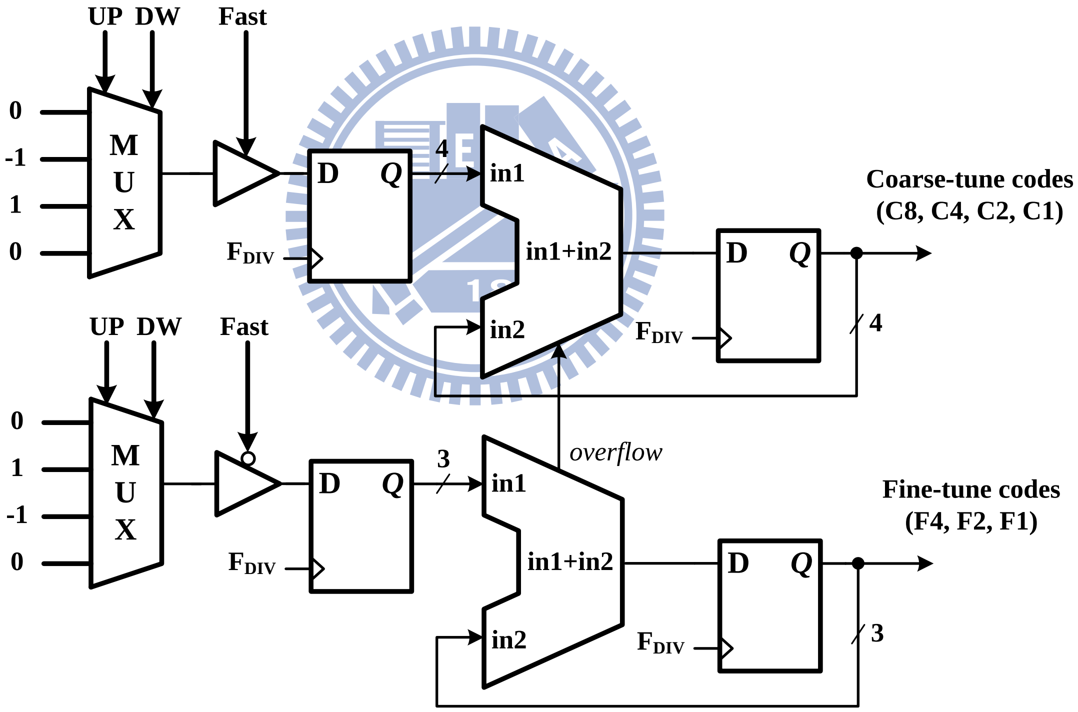


Fig 4.4 Digital loop filter architecture

Initially, the digital loop filter sets a condition that the DCO can work at one frequency that is near its mid-frequency. There are basically two procedures for the digital loop filter to generate the coarse tune codes and fine tune codes. These two procedures are controlled by "Fast" signal.

According to the DCO's characteristics in the next section, the larger coarse tune codes the DCO has, the higher oscillating frequency the DCO generates. On the other hand, the larger fine tune codes the DCO has, the lower oscillating frequency the DCO generates.

Under different conditions, the digital loop filter will operate differently as described below. One procedure is coarse tuning mechanism, or it may be called the "frequency acquisition."

When "Fast" signal is high, the digital loop filter changes coarse tune codes (C8, C4, C2, C1) and maintains fine tune codes (F4, F2, F1) under the following situations:

1. When UP=1 and DW=0→increase coarse tune code by 1

2. When UP=0 and DW=1→decrease coarse tune code by 1

3. Others → coarse tune code holds

The above operations occur when the divided frequency (F_{DIV}) from the frequency divider and the reference frequency (F_{REF}) sent into the PFD are quite different and have phase difference more than $\pm\pi$. Under this situation, the digital loop filter should change the coarse-tuning codes immediately so that the divided frequency (F_{DIV}) from the frequency divider can be almost the same as the reference frequency (F_{REF}).

At the end of coarse tuning procedure, we assume that the divided frequency (F_{DIV}) and the reference frequency (F_{REF}) sent into the PFD shall have only slightly frequency difference and this can be viewed to be the phase difference.

Next, the procedure is fine tuning mechanism, or it may be called the "phase acquisition."

When "Fast" signal is low, the digital loop filter maintains coarse codes (C8, C4, C2, C1) and changes fine codes (F4, F2, F1) under the following conditions:

1. When UP=0 and DW=1→increase fine tune code by 1

2. When UP=1 and DW=0→decrease coarse tune code by 1

3. Others → fine tune code holds

The above operations occur when the divided frequency (F_{DIV}) from the frequency divider and the reference frequency (F_{REF}) sent into the PFD have quite the same frequency value and have phase difference within $\pm\pi$. Under this situation, the digital loop filter should change the fine-tuning codes so that the phase difference between the divided frequency (F_{DIV}) and reference frequency (F_{REF}) can be eliminated gradually.

At the end of fine tuning procedure, we assume that the divided frequency (F_{DIV}) and the reference frequency (F_{REF}) sent into the PFD shall have only slightly phase difference. Ultimately, the ADPLL can be in lock-in state when frequency acquisition and phase acquisition are done step by step. But one inherent drawback of the ADPLL is that the DCO of the ADPLL can only have certain discrete frequency under these coarse tune codes and fine tune codes combinations.

Since the proposed ADPLL is used for the application of spread spectrum clocking, the issue of low jitter performance of the proposed ADPLL is quite important. Therefore, due to limited frequency resolution of the DCO, we adopt a 1st order $\Sigma\Delta$ modulator to enhance the DCO's frequency resolution. By way of using the 1st order $\Sigma\Delta$ modulator to enhance DCO's frequency resolution, the proposed ADPLL can have better jitter performance.

4.4 Differential Digitally-Controlled Oscillator

The proposed ADPLL is used in spread spectrum clock generator, thus the number of ADPLL's output phases is an important parameter for spread spectrum clocking. In the DCO design, we utilize the differential delay cell to construct a ten stage DCO with differential outputs. Furthermore, the design of the control codes of the DCO is another important issue in ADPLL design. Finally, we should take care of the DCO' output frequency range under the effects of process variation.

Fig 4.5 shows the DCO's differential delay cell [14] [31] and Fig 4.6 shows the ten-stage architecture of the DCO. The coarse tune signals (C8, C4, C2, C1) control a couple of parallel tri-state inverters, whose on-off state would determine the differential delay cell's driving strength. While the fine tune signals (F4, F2, F1) control a group of tri-state inverter based latches, whose on-off state would determine the differential delay cell's loading quantity. The dithering-bit (F0) generated by ΣΔ modulator is used to control the first stage of the DCO and the rest 9 stages of the DCO have their F0 connected to ground so as to enhance the DCO's frequency resolution.

The parallel driving cells with control-bit (C8, C4, C2, C1) are combined with the driving cell as shown in Fig 4.5; also, the loading cells with control-bit (F4, F2, F1) are connected with one loading cell. The advantage of using differential cell is that each stage of the DCO can have differential outputs and each stage's output can have good duty cycle performance. Differential architecture of the DCO can make its output's number to be double the DCO's stage number. Furthermore, the good duty cycle performance of each output from the DCO that results from differential architecture is important for switch phase mechanism used in spread spectrum clocking.

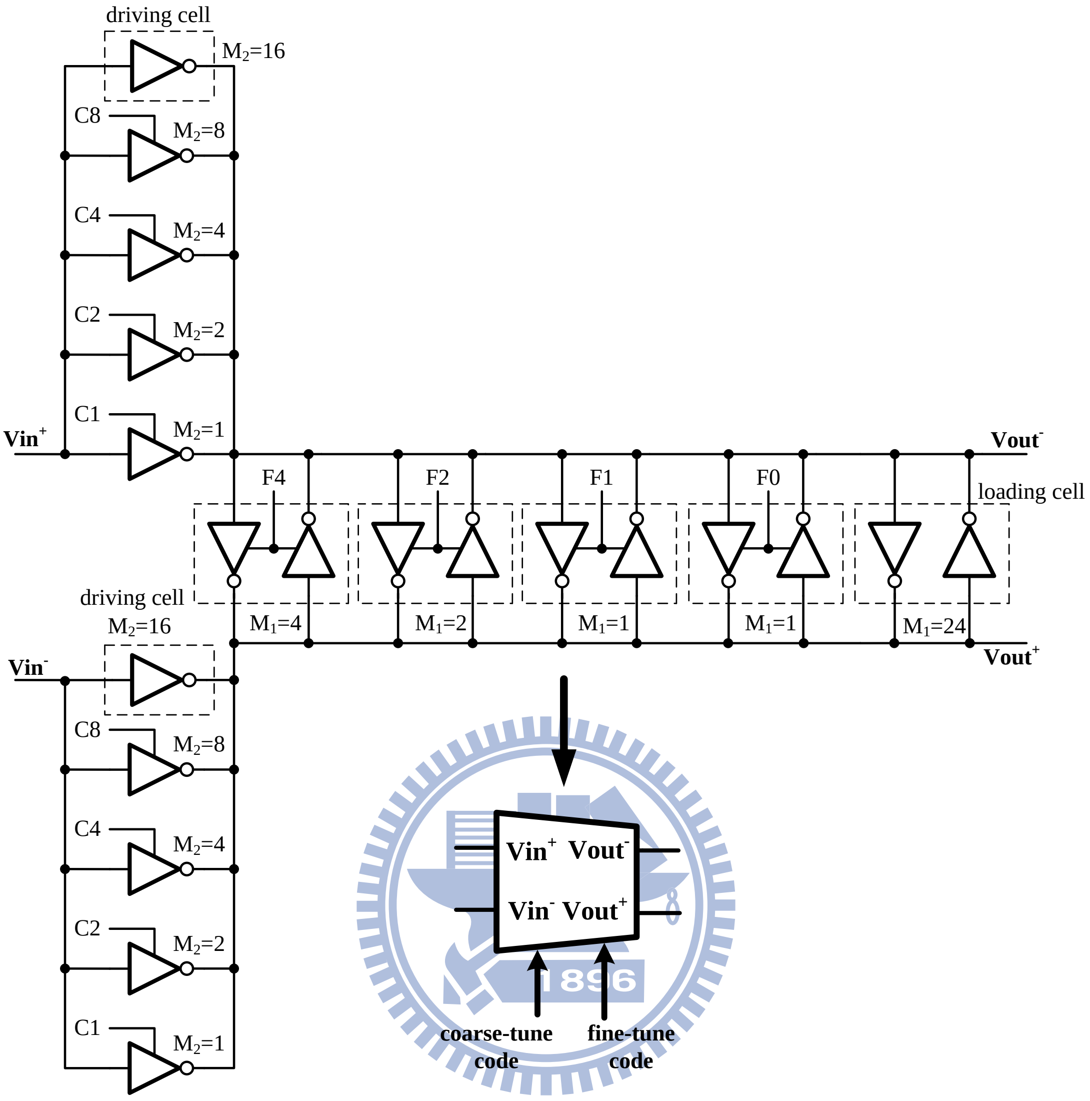


Fig 4.5 DCO differential delay cell [14] [31]

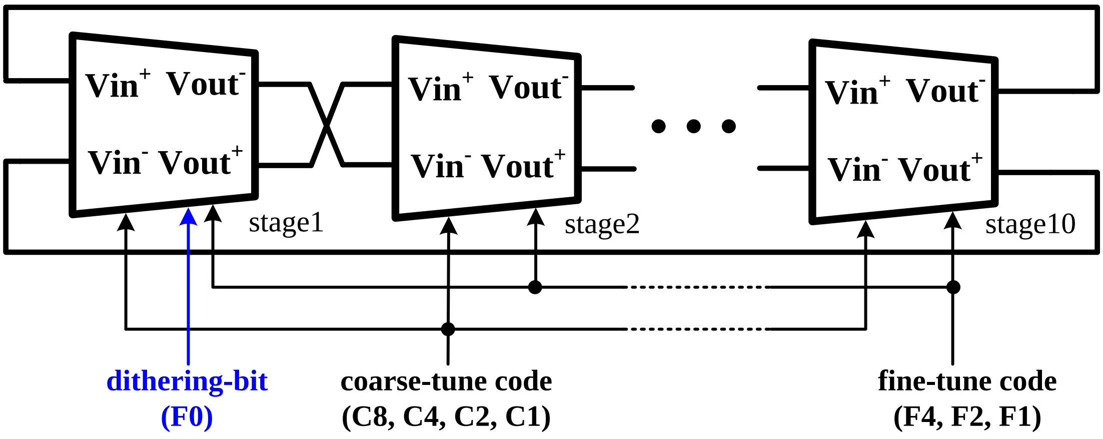


Fig 4.6 DCO architecture

The stronger the parallel driving cells' driving strength is, the smaller delay time the differential delay cell has. These parallel driving cells are served to be coarse tuning cells. Hence, the higher the coarse tune codes are, the stronger the driving strength is and as well as the higher oscillating frequency. On the opposite, the smaller the coarse tune codes are, the weaker the driving strength is and as well as the lower oscillating frequency.

The less the loading latches are turned on, the smaller delay time the differential delay cell has. These parallel loading cells are served to be fine tuning cells. Hence, the smaller the fine tune codes are, the smaller the effective loading is and as well as the higher oscillating frequency. On the opposite, the higher the fine tune codes are, the larger the effective loading is and as well as the lower oscillating frequency. According to above features, one can decide the DCO's frequency range.

To further increase the DCO's frequency resolution, we use one dithering bit (F0) controlled by 1^{st} order ΣΔ modulator. Also, only the first stage of the DCO has the dithering bit control and the rest stages of the DCO do not have the dithering bit control. The control mechanism of the dithering bit from 1^{st} order ΣΔ modulator is explained in next section.

Finally, the frequency range of the DCO should be carefully determined due to the effect of process variation. Since the frequency range of the DCO in the proposed ADPLL should cover 1.2GHz and 1GHz frequency, the maximum and minimum frequency are limited so that the frequency range still can cover 1.2GHz and 1GHz when the process is in different corners.

Except for the concern of process variation, the combination of coarse tune codes and fine tune codes should let the DCO to have an overlapped and continuous frequency range. In that way, the ADPLL may have correctly lock-in state. Fig 4.7 shows the DCO's coarse tune and fine tune mechanism chart.

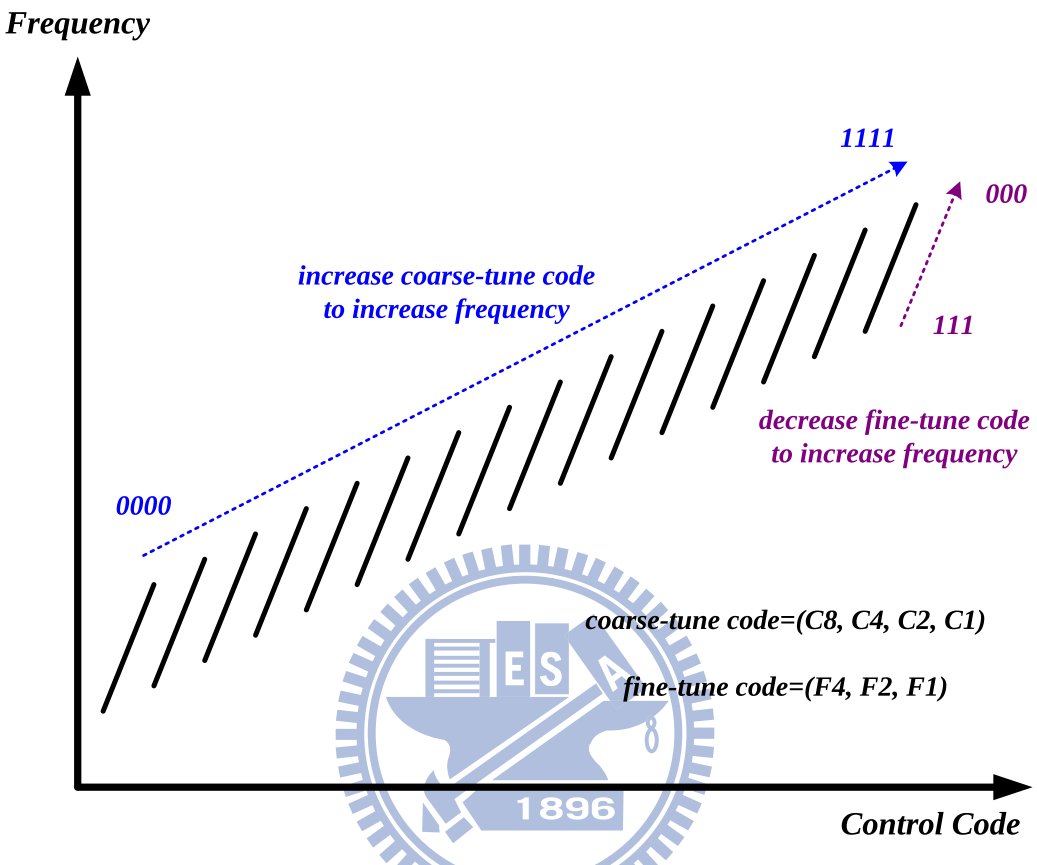
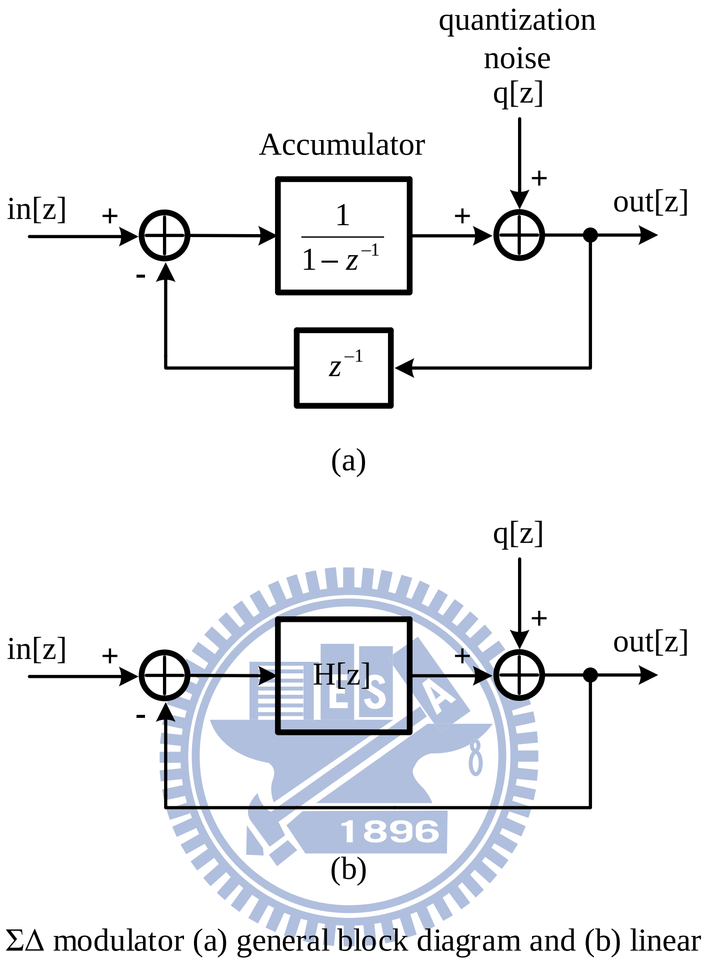


Fig 4.7 DCO's coarse tune and fine tune mechanism chart

4.5 ΣΔ Modulator

The oversampling noise shaping technique has been widely used in converting signals between the analog and digital domains. This method has led to the quick development of the sigma-delta (ΣΔ) modulator based converter. The ΣΔ modulator does the coarse signal quantization with negative feedback at one high sample rate that can shape the quantization noise away from the baseband frequency. In other words, the input signal is sampled at one rate which is higher than the Nyquist rate so as to spread its quantization noise over the bandwidth that is larger than signal bandwidth.

Fig 4.8(a) shows the accumulator based ΣΔ modulator block diagram with quantization noise, and Fig 4.8(b) shows its linear model.



(a)

(b)

Fig 4.8 ΣΔ modulator (a) general block diagram and (b) linear model

By doing z-domain analysis of Fig 4.8(a), we can get

$$out[z] = \left(in[z] - out[z]\cdot z^{-1}\right)\cdot\frac{1}{1-z^{-1}} + q[z] \tag{4.1}$$

$$out[z] = in[z] + q[z]\cdot\left(1-z^{-1}\right) \tag{4.2}$$

$$out[z] = in[z] + q[z]\cdot\frac{1}{H[z]+1} \tag{4.3}$$

, where in[z], out[z] and q[z] are the z-transforms of the input, the output and the quantization noise, respectively. The filter H[z] in Fig 4.8(b) is called the feedforward filter, which is a discrete-time integrator in a 1st order ΣΔ modulator with transfer function $H[z]=\frac{z^{-1}}{1-z^{-1}}$. As equation (4.2) shows, the quantized signal is an integrated version (sigma) of the difference (delta) between the input signal and the analog representation of the binary coded output.

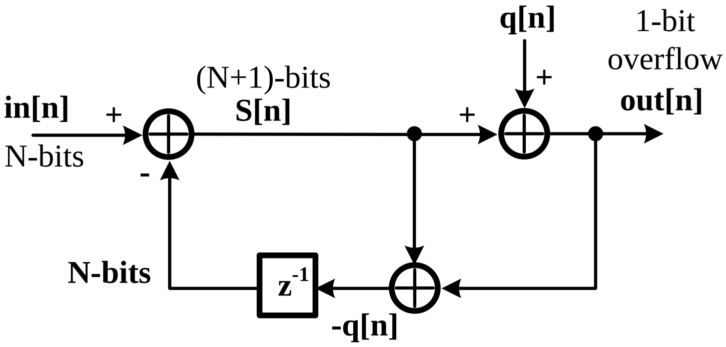


Fig 4.9 1st order ΣΔ modulator digital block diagram

Fig 4.9 is the 1st order ΣΔ modulator digital block diagram. With N bits input in[n] summed with N bits feedback output from the registers, the accumulator will generate N bits outputs S[n] and one overflow bit. q[n] is the quantization error, after the overflow, the rest value is saved in the registers for the next upcoming summation with another new input signal.

Since the DCO's frequency resolution is determined by the variation of the output cycle, which is dominated by the least-significant bit of the DCO's control codes, we need to generate one least significant bit to control the smallest variation of the DCO's output cycle.

Fig 4.10 shows the realization of 1st order ΣΔ modulator. It is built from a 3-bit accumulator and registers. The trigger clock (F_{DCO}) of the registers comes from the DCO. The inputs of the ΣΔ modulator are fine-tune codes (F4, F2, F1) and the overflow of the accumulator is the dithering bit (F0), which is used to control the DCO's finest frequency resolution.

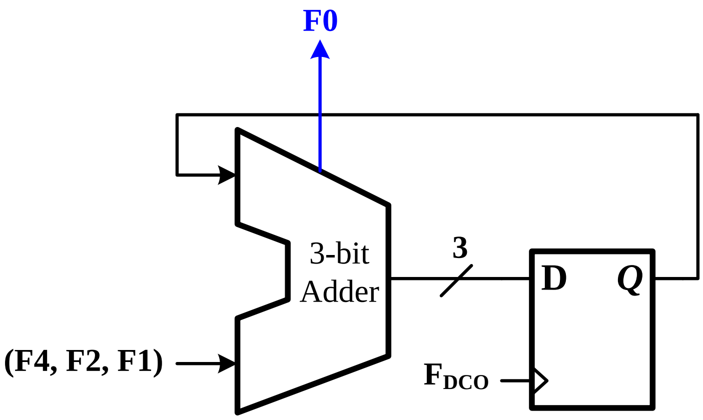


Fig 4.10 Realization of 1st order ΣΔ modulator

From section 4.3, we know that the DCO has 10 stages and each stage is controlled by the combination of coarse tune codes (C8, C4, C2, and C1) and fine tune codes (F4, F2, and F1). However, only one stage of the DCO has been controlled by dithering bit (F0). In that way, the DCO's finest cycle's variation is only one tenth of that the one-bit fine tune code can change.

Table 4.1 shows DCO's output cycle variation under control of 1st order ΣΔ modulator. We know that the divider ratio of the ADPLL is 10. Each time when one combination of (F4, F2 and F1) is brought out from the digital loop filter of the ADPLL, the 1st order ΣΔ modulator does summation ten times under that input combination. The DCO's output cycle changes only when overflow becomes "1", so the average cycle's variation shown in Table 4.1 is one tenth the difference of the DCO's output cycles. In this way, we can enhance the DCO's frequency resolution and minimize the jitter of the ADPLL due to inadequate frequency resolution as possible.

Table 4.1 DCO's output cycle variation under control of 1st order ΣΔ modulator

# of F_{DCO}	inputs (F4, F2, F1)	Accumulator's outputs	Overflow (F0)	DCO's output cycle(ps)
0	001	000	0	832.33
1	001	001	0	832.33
2	001	010	0	832.33
3	001	011	0	832.33
4	001	100	0	832.33
5	001	101	0	832.33
6	001	110	0	832.33
7	001	111	0	832.33
8	001	000	**1**	**842.33**
9	001	001	0	832.33
10	001	010	0	832.33
Average DCO's Output Cycle = 833.33 ps				

4.6 Frequency Divider

Fig 4.11 shows the frequency divider architecture. It is composed of one divided-by-two frequency divider cascaded with one divided-by-five frequency divider. Since the frequency divider usually has higher frequency input signal and the total power consumption is associated with operating frequency, the divided-by-two frequency divider is put in front of the divided-by-five frequency divider. The divider ratio can be integer or fractional number. In this ADPLL design, we choose the divider ratio as 10 to ease the calculation of spread spectrum's parameters.

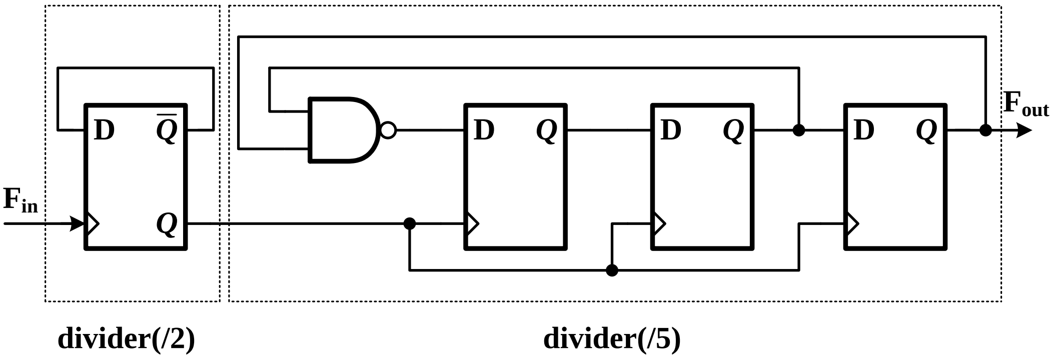


Fig 4.11 Frequency divider

In Chapter2, we know that the spread spectrum deviation is determined by $\dfrac{\frac{A}{M}}{N \times P}$, where A is the triangular wave profile generator's output, M is the maximum number in the accumulator of the 1st order ΣΔ modulator, P is the number of output phases of the DCO, and N is the frequency divider ratio. The detailed value will be shown in section 5.3.6.

Chapter 5

Spread-Spectrum Clock Generator

5.1 System Architecture

Generally, ADPLL should generate low-jitter output clocks in communication system. In the application of my proposed ADPLL, a spread spectrum clock generator (SSCG) for SATA-III specification with phase rotation mechanism is presented. The SSCG can turn-on or turn-off its spread-spectrum clocking function. Also, the SSCG can spread 10 phases or 20 phases of the proposed ADPLL.

The SSCG is down spread 5000 ppm with a triangular waveform of modulation frequency 30~33 KHz. The proposed SSCG is designed in TSMC 65nm CMOS process. The non-spread spectrum clock has a peak to peak jitter of 24.5ps and rms jitter of 3.96ps. The maximum EMI reduction is -18.6dB and -20.4dB in10 and 20 phase spread spectrum mode with power dissipation of 8.329mW and 8.618mW, respectively.

Fig 5.1 shows the architecture of spread spectrum clock generator with phase rotation mechanism based on ΣΔ modulator. With the “select” control ping as shown in Fig 5.1, we can choose 10 phases or 20 phases spread spectrum modes. Also, with the “switch” control ping as shown in Fig 5.1, we can turn-on of turn-off spread spectrum clocking function. In 10 phases spread spectrum mode, the SSCG has less power dissipation. While in 20 phases spread spectrum mode, the SSCG has better EMI reduction.

The SSCG is composed of the proposed ADPLL (with blocks of PFD, DLF, DCO, ΣΔ modulator, and frequency divider) and triangular modulation profile generator, 1st order ΣΔ modulator, MUX control circuit and MUX.

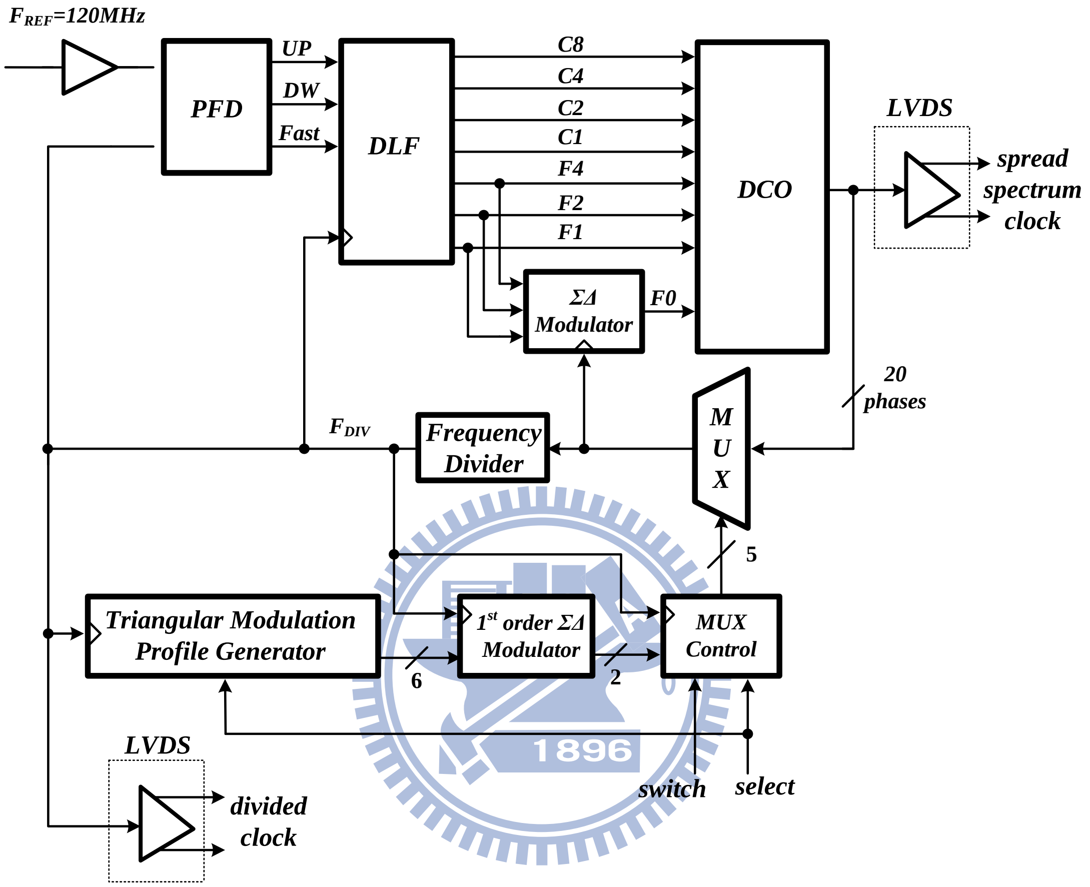


Fig 5.1 Architecture of spread spectrum clock generator

The triangular modulation profile generator sends a periodic profile (6-bits) into the 1st order ΣΔ modulator to produce the overflows (2-bits). And then these overflows are sent to the MUX control circuit to generate phase-selection signals (5-bits). Upon using 1st order ΣΔ modulator, the noise spectrum can be shaped to a higher frequency range and it can result in continuous frequency modulation from a discrete staircase input from the triangular modulation profile generator. The triangular modulation profile generator, the 1st order ΣΔ modulator, and the MUX control circuit are all synchronized with divided clock (F_{DIV}). The design of the triangular modulation profile generator can control modulation frequency and frequency modulation deviation. As mentioned in Chapter 2, the SSCG has some spread spectrum parameters to decide.

$$f_{spread} = f_{\mathrm{REF}} \cdot \left(N - \frac{\frac{A}{M}}{P} \right) = f_{non-spread} \left(1 - \frac{\frac{A}{M}}{N \times P} \right) \qquad (5.1)$$

The relationship between input and output of the 1st order ΣΔ modulator is $\frac{A}{M}$, where A is the output of the triangular modulation profile generator, and M is the maximum number of the accumulator in the 1st order ΣΔ modulator. According to Eq. (5.1), the output frequency of the SSCG will change smoothly according to the value of A, where f_{REF}=120MHz, $f_{non\text{-}spread}$=1.2GHz, N is the divider ratio, P is the number of DCO's output phases and A runs periodically through 0, 1, 2, …, A_{max}, A_{max}-1, …, 2, 1, 0, where A_{max} is the maximum value of the output of the triangular profile modulation generator.

If the output of the 1st order ΣΔ modulator is "0", the MUX control circuit selects the original phase as the MUX output, then $f_{spread} = N \times f_{\mathrm{REF}}$. When the output of the 1st order ΣΔ modulator is "1", the MUX selects an early phase (with down spread specification) as the MUX output, then $f_{spread} = f_{\mathrm{REF}} \cdot \left(N - \frac{1}{P} \right)$. In the same way, while the output of the 1st order ΣΔ modulator is "2", the MUX selects the phase which is two phases earlier compared to the original phase as the MUX output, then $f_{spread} = f_{\mathrm{REF}} \cdot \left(N - \frac{2}{P} \right)$.

For the sake of accomplishing SATA-III specification, the SSCG has a 5-bit accumulator [32] in the 1st order ΣΔ modulator ($M=2^5=32$). Based on the frequency deviation: $-\frac{\frac{A}{M}}{N \times P} = -5000\,ppm$, we set A to be 16 in 10 phases spread spectrum mode (P=10 and N=10), and set A to be 32 in 20 phases spread spectrum mode (P=20 and N=10). A is programmable in the triangular modulation profile generator under different spread spectrum mode.

Table 5.1 SSCG design parameters

Spread Spectrum **Parameters**	**10 Phases** **Spread Spectrum Mode**	**20 Phases** **Spread Spectrum Mode**
A	**16**	**32**
M	**2^5=32**	**2^5=32**
N	**10**	**10**
P	**10**	**20**
Modulation Deviation = -5000ppm ~ 0ppm **Modulation Frequency = 30KHz ~ 33KHz**		

5.2 ADPLL Behavioral Simulation

Because of a large amount of logic gate counts in the all digital phase-locked loop, the simulation should start from behavioral simulation and then do the Hspice simulation to save simulation time. In order to quickly and thoroughly understand the behavior of the ADPLL, we use the SIMULINK to test and verify the ADPLL's timing response to observe the ADPLL locking behavior. Fig 5.2 shows the ADPLL behavioral model built from SIMULINK.

The proposed ADPLL uses 1st order ΣΔ modulator to enhance the frequency resolution. Next, according to Section 5.3.3, we can estimate the DCO's coarse tune gain and fine tune gain as 330MHz/bit and 190MHz/bit, respectively.

Since the 1st order ΣΔ modulator's output (F0) only controls one of the ten stages of the DCO, its tuning gain can be set to 190MHz/bit×0.1=19MHz/bit. Fig 5.3 (a) shows the frequency lock-in state of the ADPLL with spread-spectrum clocking is turned off, and Fig 5.3 (b) is the enlarged figure of the locking behavior.

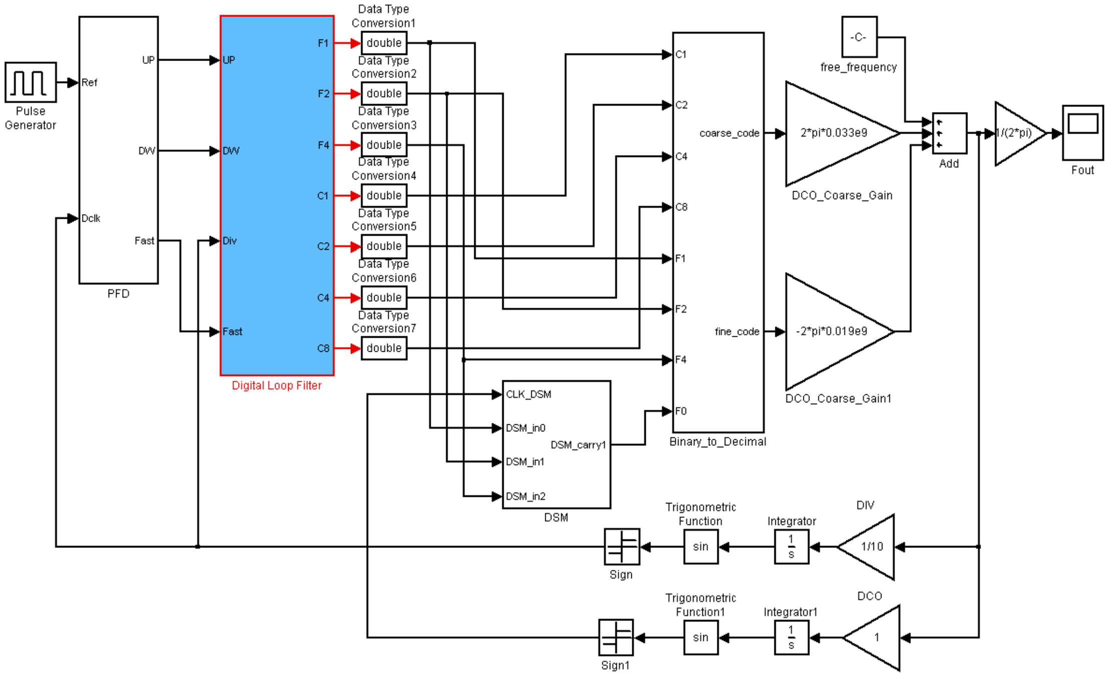
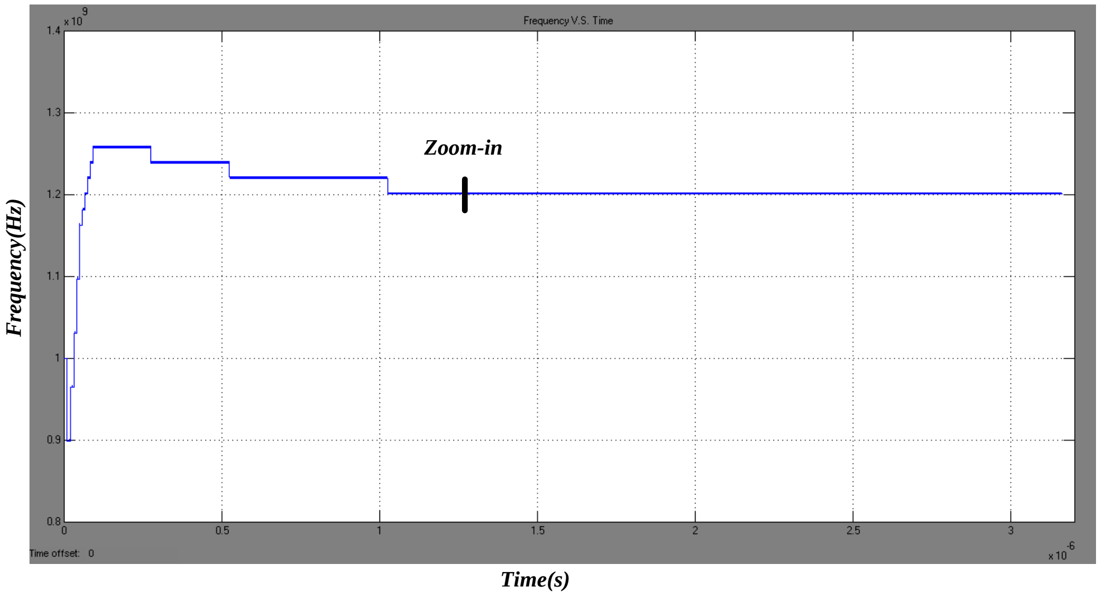


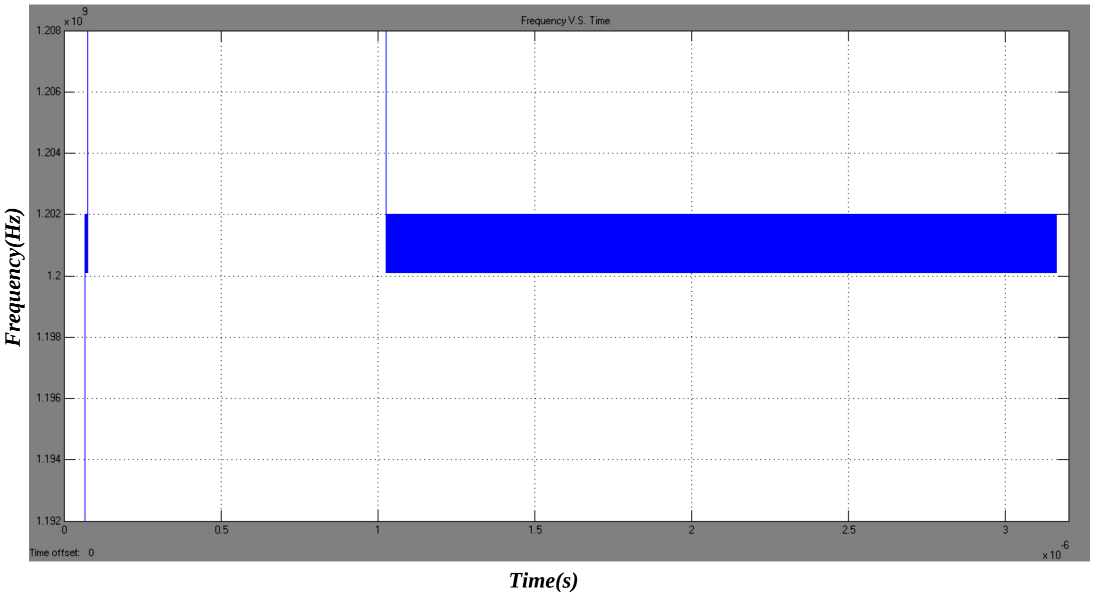
Fig 5.2 SIMULINK model of the proposed ADPLL

From Fig 5.3(a), we can find out that the ADPLL's frequency lock-in time is less than 2μs. As shown in Fig 5.3(b), when the ADPLL is in frequency lock-in state, its frequency jumps between two discrete frequencies. This phenomenon can be thought of as the source of the ADPLL's timing resolution. Although the frequency resolution of the ADPLL has enhanced, there is slightly frequency difference between the target frequency and the DCO frequency. That is the reason why the ADPLL designers have put all their attention on increasing the ADPLL's frequency resolution.

The target frequency of the ADPLL is 1.2GHz. Since the ADPLL has inherent timing resolution and SIMULINK has numerical resolution, the ADPLL is locked to an average frequency larger than 1.2GHz.



(a)



(b)

Fig 5.3 (a) The ADPLL's output frequency V.S. time plot and (b) zoom-in plot

5.3 Circuit Implementation

5.3.1 PFD with Phase Threshold Detector

In Section 4.2 we have introduced the phase/frequency detector with phase threshold detector, as shown in Fig 4.2. To do proper simulations, we set simulation parameters to test the phase/frequency detector with phase threshold detector.

In Fig 5.4, we have set the “F_{REF}” signal as 120MHz frequency and the “F_{DIV}” signal as 100MHz frequency. We can see that when the positive edge of “F_{REF}” leads that of “F_{DIV}”, the “UP” signal is set to high. And the frequency difference between “F_{REF}” and “F_{DIV}” gradually causes more and more phase difference between “F_{REF}” and “F_{DIV}”. As shown in Fig 5.4, once the phase difference is more than π (one clock cycle has 0~2π, and π means half clock cycle), then “Fast” is set to high as well as “Q1”. As a result, we have created a phase threshold “π” to enhance the speed of lock-in behavior.

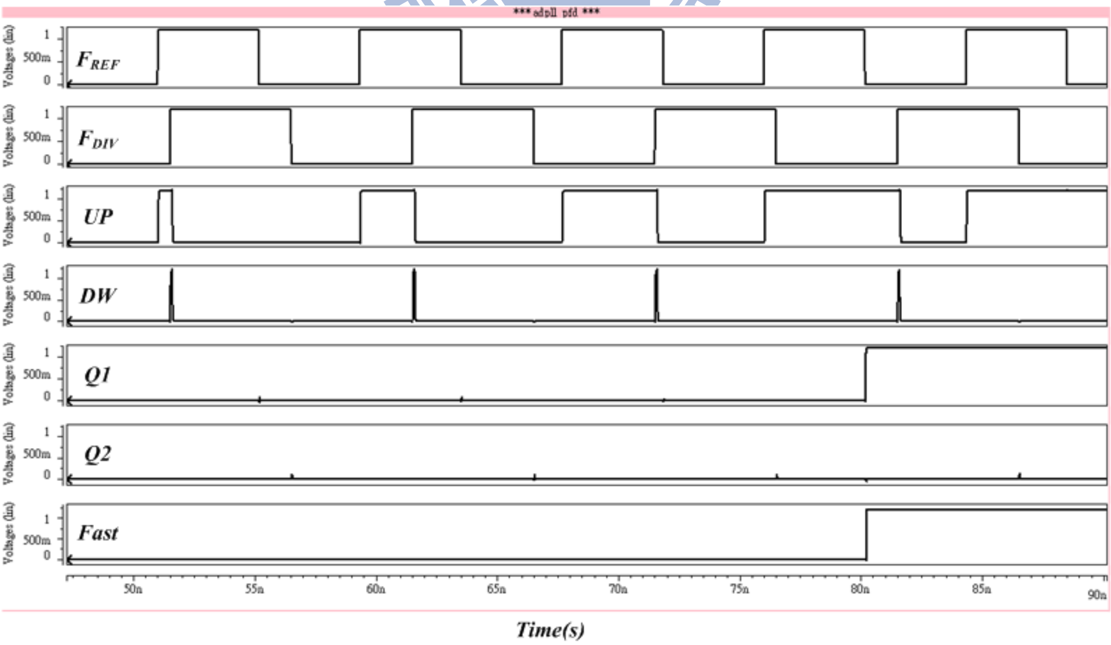


Fig 5.4 PFD simulation: “F_{REF}” has higher frequency than “F_{DIV}”

In the same way, we have set the “F_{REF}” signal as 100MHz frequency and the “F_{DIV}” signal as 120MHz frequency. We can see that when the positive edge of “F_{REF}” lags that of “F_{DIV}”, the “DW” signal is set to high. And as shown in Fig 5.5, once the phase difference is more than π (one clock cycle has 0~2π, and π means half clock cycle), then “Fast” is set to high as well as “Q2”. Also, we have verified a phase threshold “π” to enhance the speed of lock-in behavior.

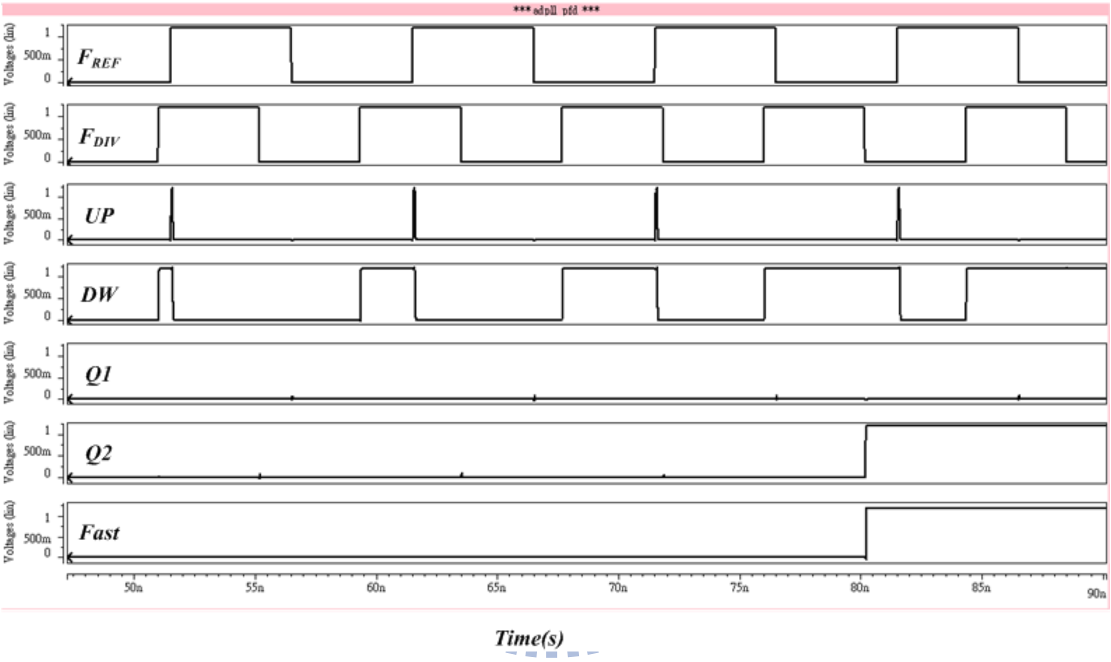


Fig 5.5 PFD simulation: “F_{DIV}” has higher frequency than “F_{REF}”

The tri-state PFD has dead zone (the minimum phase difference between “F_{REF}” and “F_{DIV}” that can make PFD to act) problem in charge-pump PLL. However, the PFD in the propose ADPLL only needs to know the phase lead or lag information and phase difference extent between “F_{REF}” and “F_{DIV}”, the dead-zone of the PFD does not affect the proposed ADPLL’s lock-in behavior.

5.3.2 The 1st Order ΣΔ Modulator Design

The proposed ADPLL uses coarse-tune codes (C8, C4, C2 and C1), fine-tune codes (F4, F2 and F1) and dithering bit (F0) to lock the target frequency in three procedures. We assume coarse-tune and fine-tune codes are fixed when the frequency of the proposed ADPLL is near target frequency and these fine-tune codes are chosen as the 1st order ΣΔ modulator's inputs.

In order to enhance the frequency resolution of the proposed ADPLL, the 1st order ΣΔ modulator uses fine tune codes (F4, F2 and F1) from digital loop filter to generate one dithering bit (F0) to control the DCO's finest frequency resolution. Since the "F0" dithering bit could affect the DCO's output frequency, its value must be generated correctly. Fig 5.6 shows the circuit of the 1st order ΣΔ modulator.

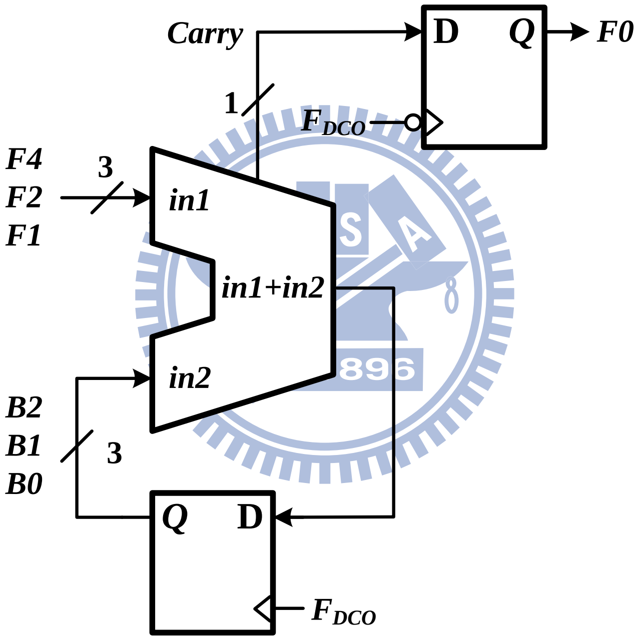


Fig 5.6 The 1st order ΣΔ modulator applied in ADPLL

The 1st order ΣΔ modulator has a 3-bit accumulator, and it sums the current input sets (F4, F2 and F1) with the previous sets once the positive edge clock comes. The "F_{DCO}" comes from the DCO. Since the frequency divider ratio is 10, the fine-tune codes (F4, F2 and F1) are changed every 10 "F_{DCO}" clock cycles. The 1st order ΣΔ modulator simulation result is shown in Fig 5.7. We set the fine-tune codes (F4, F2 and F1) as (0, 1, 1) and the "F_{DCO}" has 1.2GHz clock frequency.

In Fig 5.7, we can see in the circle area that when positive edge of “F_{DCO}” comes with fine-tune codes (F4, F2 and F1) as (0, 1, 1) and feedback codes (B2, B1 and B0) as (1, 0, 1), the accumulator sums these two codes and should generate the right “Carry” value as 1. However, the accumulator generates some unwanted “Carry” values during the calculation and if this “Carry” is used as dithering bit “F0”, the DCO’s operation may be affected.

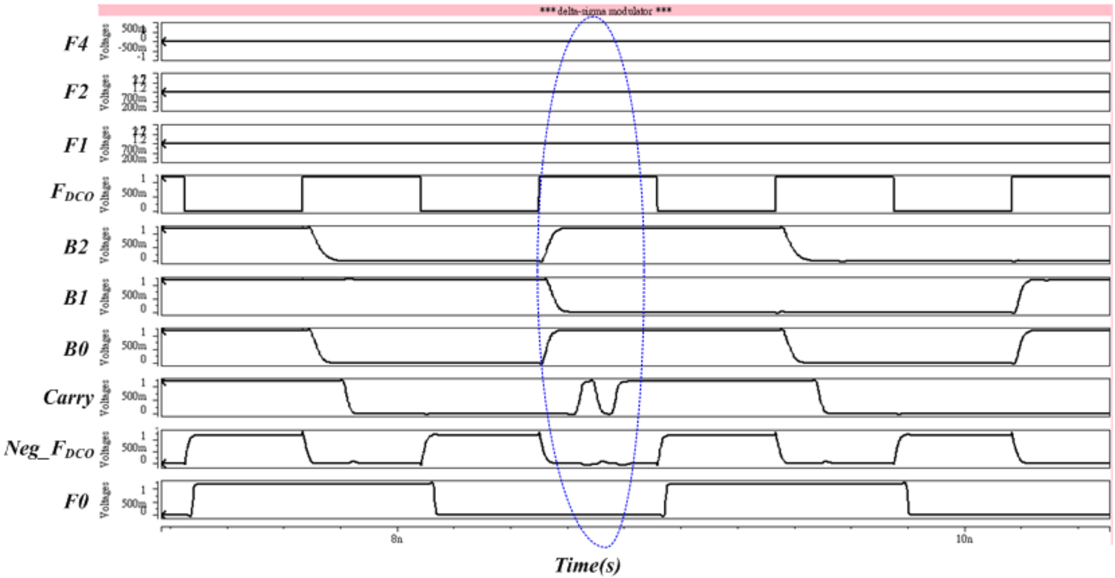


Fig 5.7 Simulation plot of 1st order ΣΔ modulator

Thus, we use the negative edge of the “F_{DCO}” (the positive edge of the “Neg_F_{DCO}”) to sample the “Carry” signal and generates the dithering bit “F0” to control the DCO’s finest frequency resolution. In that case, the 1st order ΣΔ modulator can generate the correct dithering bit “F0” without unwanted values.

5.3.3 Differential Digitally-Controlled Oscillator

In Section 4.4, we have shown the DCO differential delay cell and DCO architecture in Fig 4.5 and Fig 4.6, respectively. The DCO differential delay cell has two parts, one part is designed for the coarse tune mechanism and the other part is designed for the fine tune mechanism. For the coarse tune mechanism, the delay stage has 4 bits control code as (C8, C4, C2, and C1). For the fine tune mechanism, the delay stage has 3 bits control codes as (F4, F2 and F1) and 1 dithering bit control code as (F0). Basically, they are composed of tri-state inverters with size ratio in power of 2. All the 10 stages of the DCO are controlled by the coarse tune and fine tune control codes, but only the first stage of the DCO is controlled by the dithering bit (F0) for the sake of controlling minimum delay change as well as the finest frequency resolution.

In Fig 5.8, we simulate the DCO to vary its coarse tune codes (C8, C4, C2 and C1) with constant fine tune codes and dithering-bit (F4, F2, F1, F0) as (0, 1, 1, 0). We can see that when coarse tune codes increase, the DCO's frequency increases, too.

It is because that the coarse tune codes control the driving cells of the delay stage. When the coarse tune codes increase, more driving cells are turned on and the driving ability of the delay stage enhances. Hence, under the same effective loading condition (with constant fine tune codes), the effective delay of the delay stage is decreased and the DCO's frequency is increased.

Finally, the DCO's coarse-tune code delay is 22.68 (ps/code) in average.

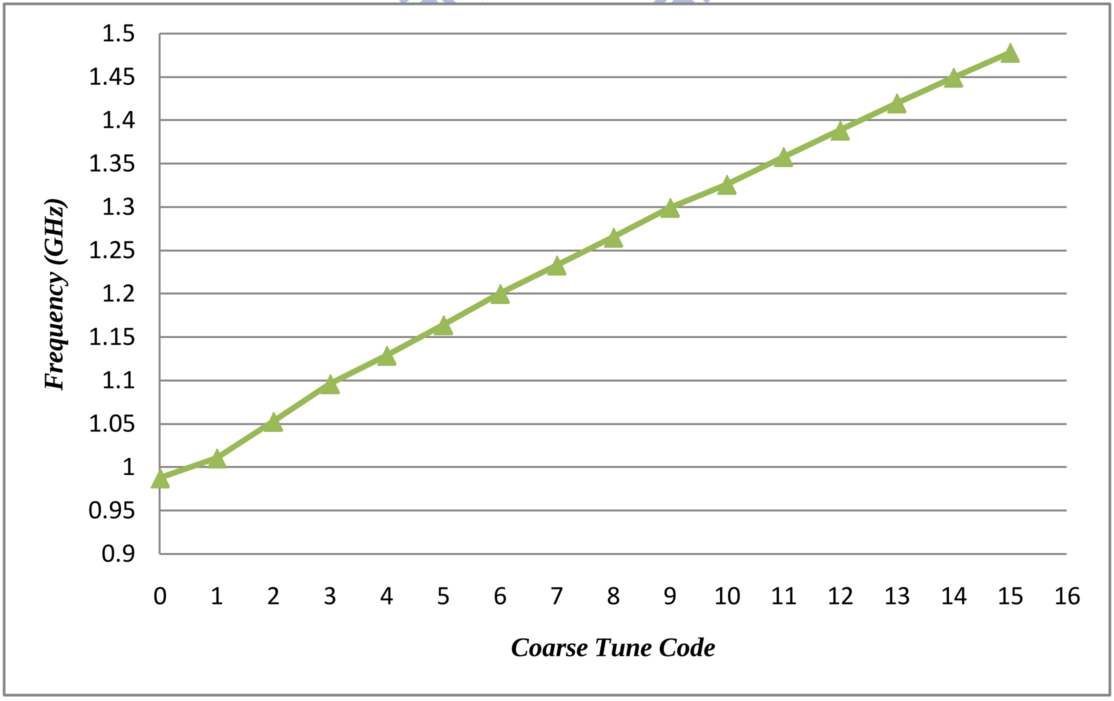


Fig 5.8 DCO's frequency V.S. coarse tune control code plot

Similarly in Fig 5.9, we simulate the DCO to vary its fine tune codes (F4, F2, F1) with constant coarse tune codes (C8, C4, C2, C1) as (0, 1, 1, 0) and with dithering-bit F0 set to 0. We can see that when fine tune codes increase, the DCO's frequency decreases. This is because that the fine tune codes control the loading cells of the delay stage. When the fine tune codes increase, more loading cells are turned on and the effective loading effect strengthens. Hence, under the same driving ability condition (with constant coarse tune codes), the effective delay of the delay stage is increased and the DCO's frequency is decreased.

Finally, the DCO's fine-tune code delay is 13.02 (ps/code) in average.

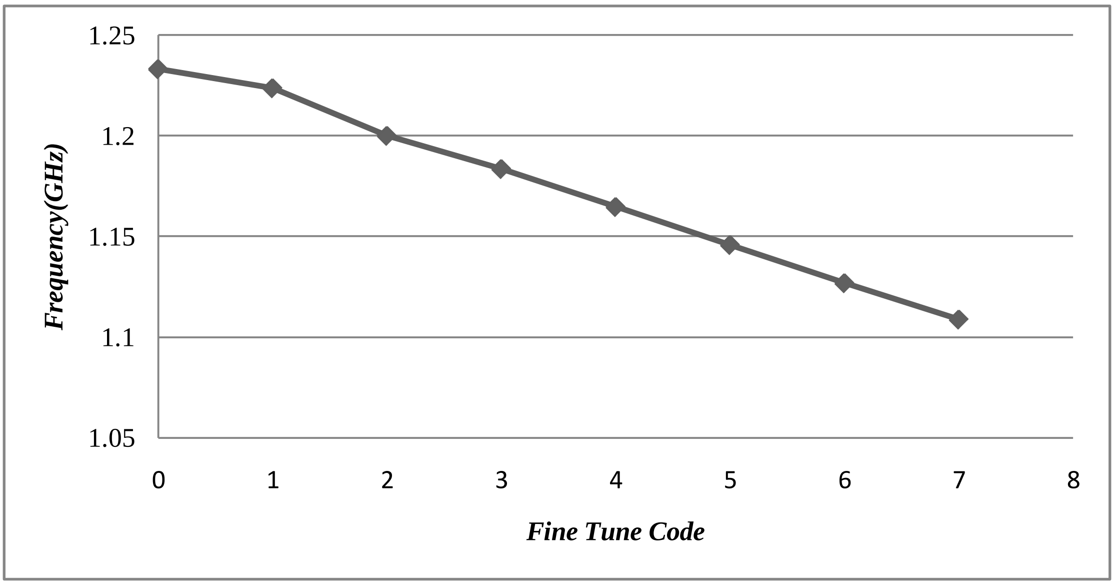


Fig 5.9 DCO‘s frequency V.S. fine tune control code plot

In Fig 5.10, we simulate the DCO to vary its coarse tune codes (C8, C4, C2, C1) with constant fine tune codes (F4, F2, F1) as (0, 1, 1). The dithering bit (F0) can vary the DCO's clock cycle delay time and its gain can also be roughly be estimated as 1.30 (ps/code).

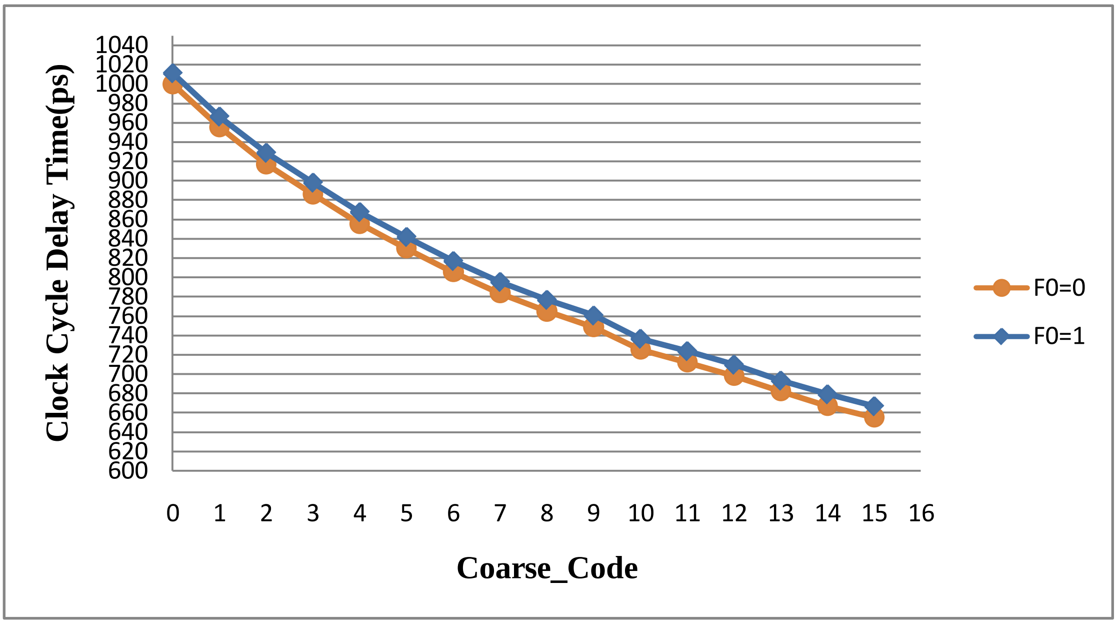


Fig 5.10 Dithering-bit characteristic of DCO

Finally, we combine the coarse tune and fine tune codes and vary them in sequence. The coarse tune codes (C8, C4, C4 and C1) are varied in increasing order from (0, 0, 0, 0) to (1, 1, 1, 1) and the fine tune codes (F4, F2 and F1) are varied in decreased order from (1, 1, 1) to (0, 0, 0). Fig 5.11 shows the DCO's overall frequency tuning range at "TT" corner.

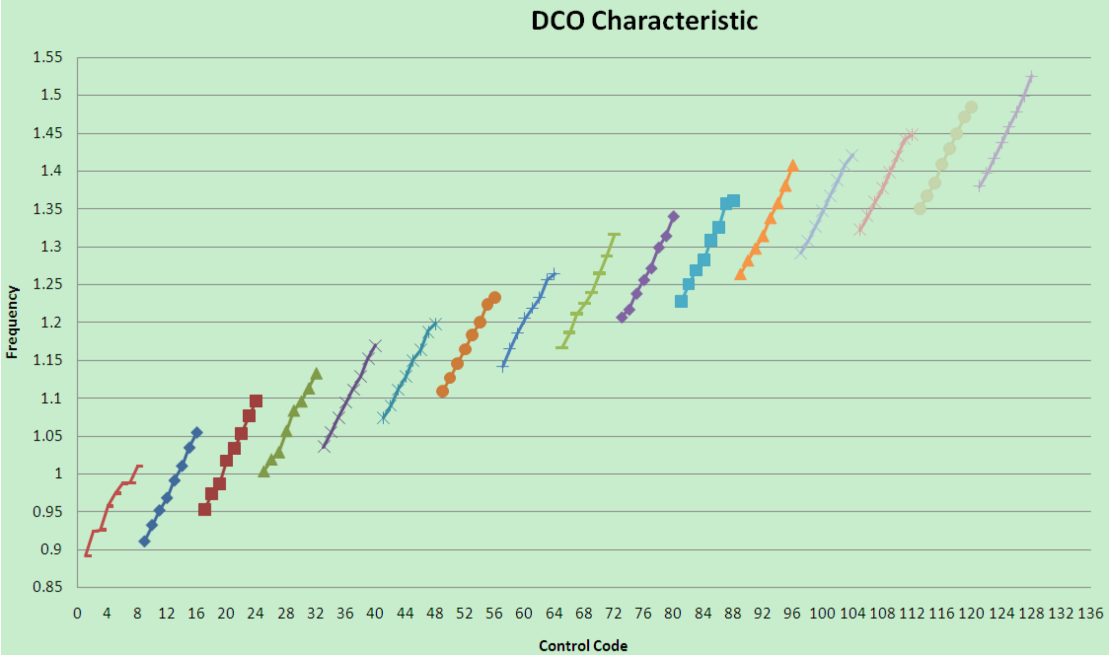


Fig 5.11 DCO characteristic: frequency V.S. control code plot

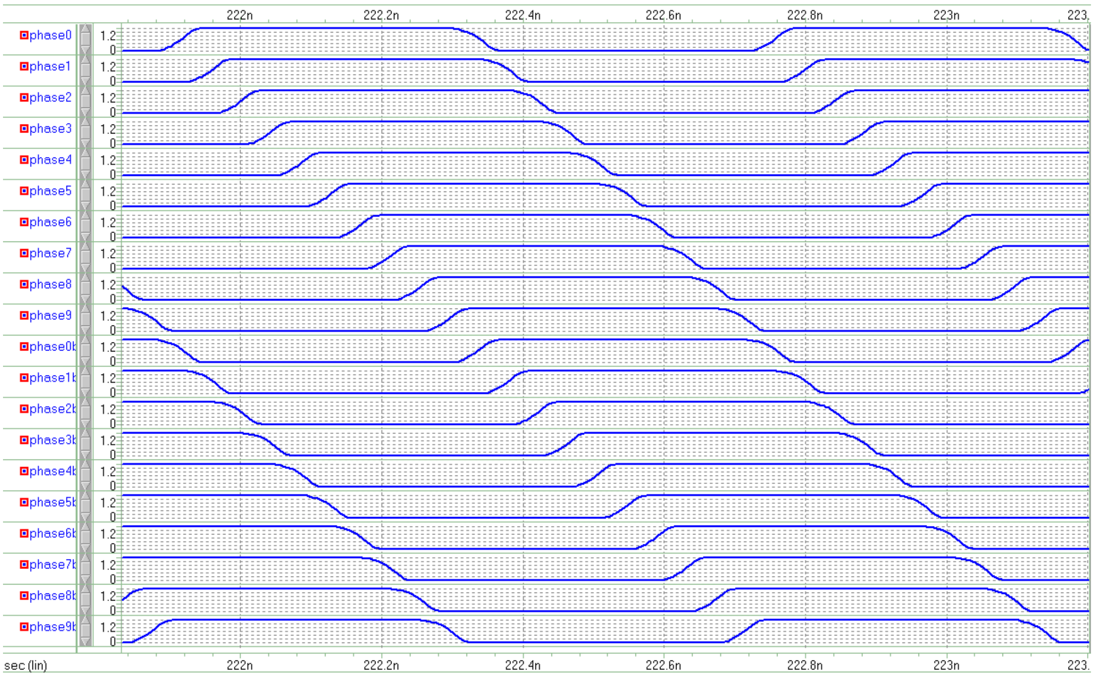


Fig 5.12 DCO's 20 output phases

Since the ADPLL is used for spread spectrum clocking application, the proposed ADPLL must generate multiple output phases. Thanks to the DCO's differential structure, these multiple output phases are separated uniformly, as shown in Fig 5.12.

The above simulation results are summarized in Table 5.2. Also, we simulate the DCO's frequency tuning range in five different process corners (TT, SS, FF, SF and FS). To make sure the frequency tuning range of the DCO can still cover the 1.2GHz and 1GHz frequencies with process variation, the frequency tuning range should be simulated in five different process corners. Table 5.3 shows the simulated results under different process corners.

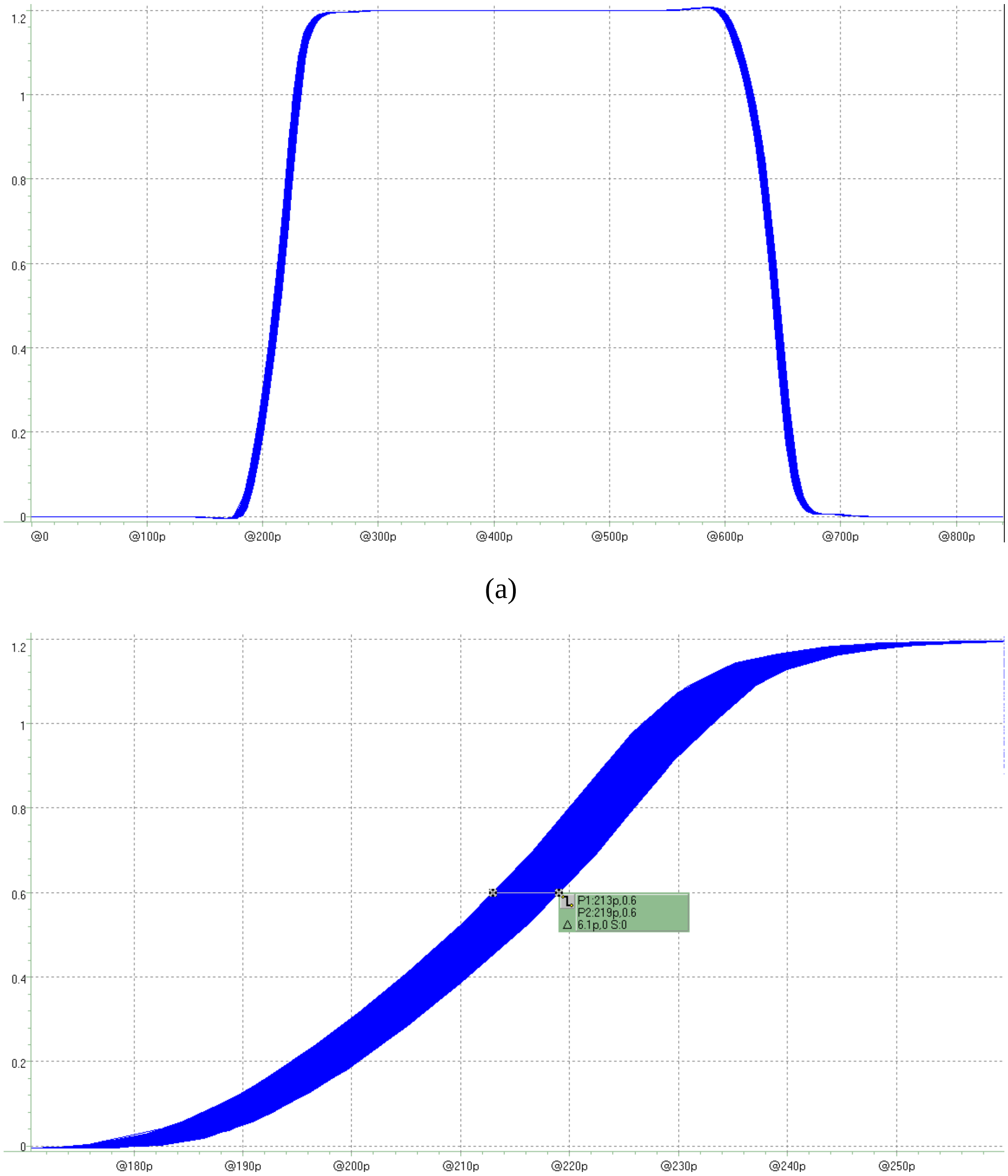
Table 5.2 DCO specification

Mechanism	***Coarse Tune***	***Fine Tune***	***Dithering***
Bit Number	***4***	***3***	***1***
Code Type	***Binary***	***Binary***	***Binary***
Average Gain	***22.68 ps/code***	***13.02 ps/code***	***1.30 ps/code***
Frequency Range	***0.888GHz ~ 1.526GHz @ TT corner***		
Output Phases	***20***		
Jitter$_{\text{peak-to-peak}}$	***6.1ps***		

Table 5.3 DCO frequency range with different corner

Corner		***TT***	***FF***	***SS***	***SF***	***FS***
Frequency Range(GHz)	Max.	***1.526***	***1.838***	***1.251***	***1.453***	***1.543***
	Min.	***0.888***	***0.978***	***0.751***	***0.857***	***0.841***

In addition, we have considered the jitter performance of the DCO. We set the coarse-tune codes (C8, C4, C2 and C1) to be (0, 1, 1, 0), fine-tune codes (F4, F2, and F1) to be (0, 1, 0) and dithering bit (F0) to be “0”. The DCO’s free-running frequency is 1.19GHz. Fig 5.13 (a) shows the DCO’s overall peak-to-peak jitter diagram and Fig 5.13 (b) gives the zoom-in plot. The measured peak-to-peak and RMS jitter is 6.1ps and 1.0ps, respectively.



(a)

(b)

Fig 5.13 (a) DCO’s peak-to-peak jitter and (b) zoom-in plot

5.3.4 Frequency Divider

The frequency divider ratio of the proposed ADPLL is set to be 10 for the convenience of designing the spread spectrum clock generator with integer parameters. In Section 4.6, we have shown the frequency divider in Fig 4.11 and it is composed of one divide-by-two frequency divider cascaded with one divide-by-five frequency divider. All DFFs in the frequency divider are TSPC based so that the divider can handle high frequency division. Fig 5.14 shows the frequency divider simulation.

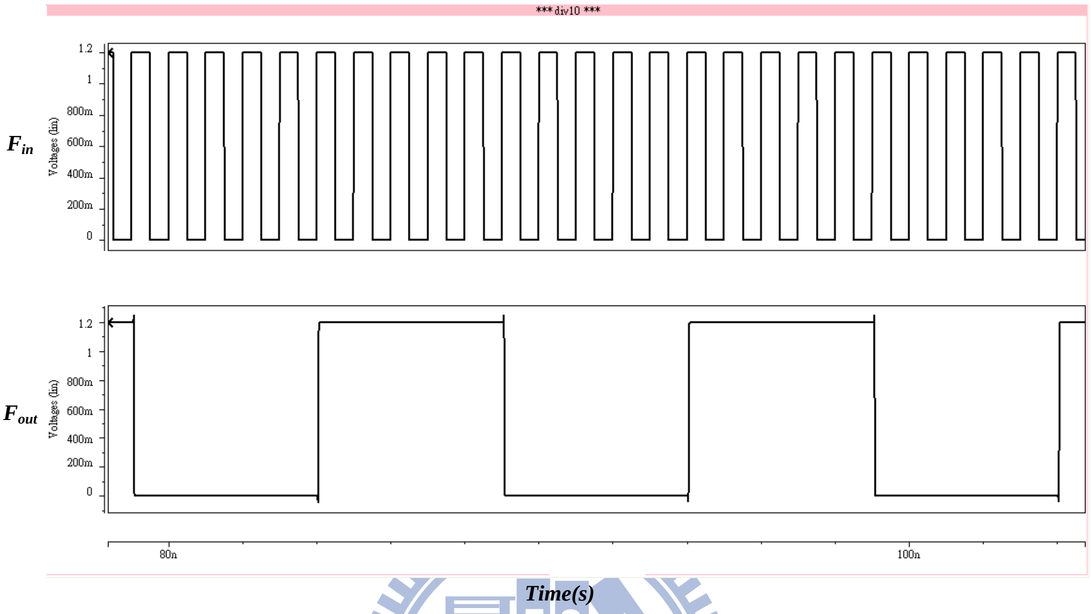
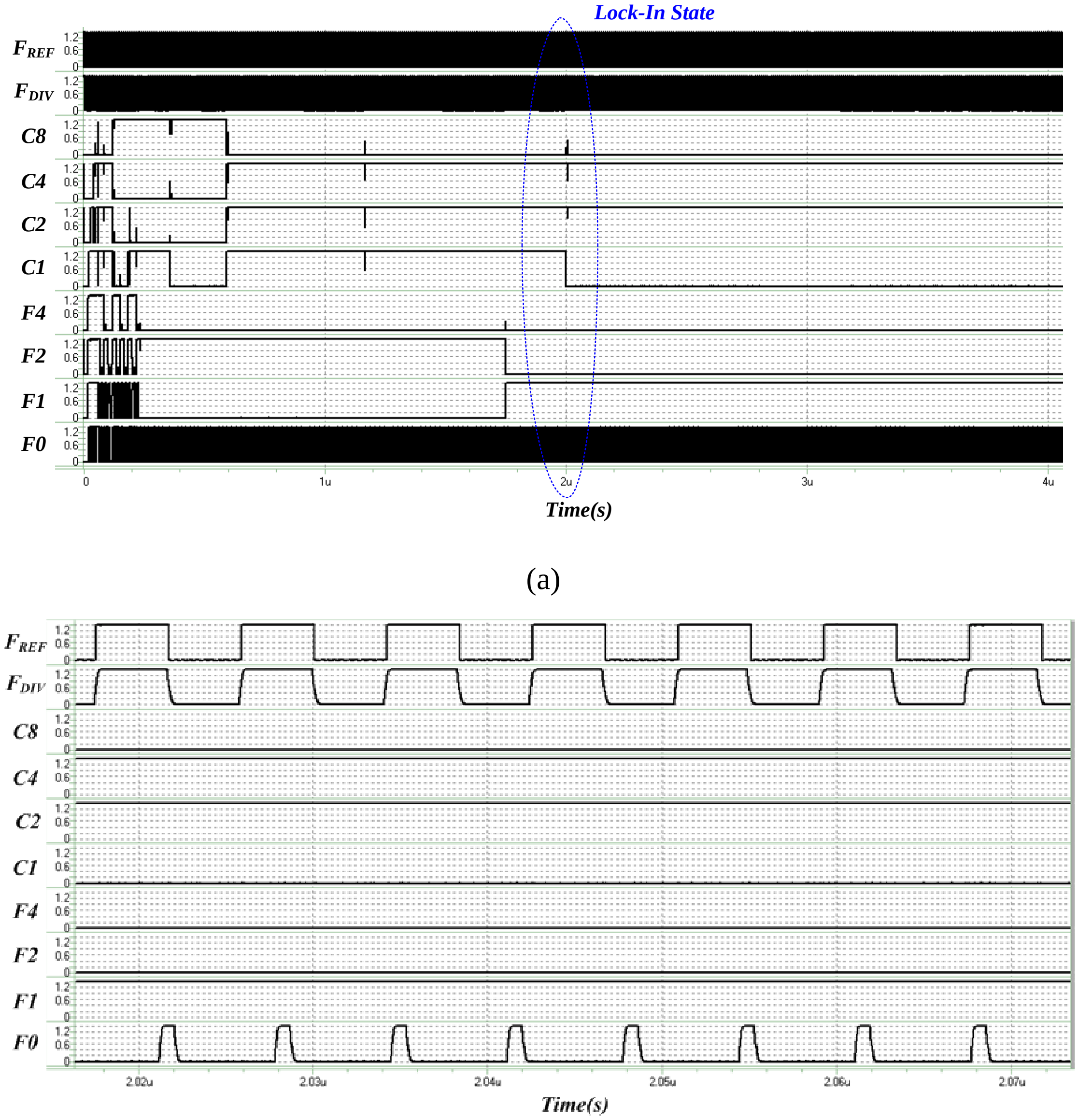


Fig 5.14 Frequency divider simulation

5.3.5 ADPLL Summary

Here we show some important simulation results about the proposed ADPLL. The ADPLL's DCO control words acquisition is shown in Fig 5.15 (a) and Fig 5.15 (b) shows the zoom-in plot. In Fig 5.15 (a), we can see that the DCO coarse tune codes (C8, C4, C2, C1) are locked to (0, 1, 1, 0) and fine tune codes (F4, F2, F1) are locked to (0, 0, 1) and dithering bit (F0) jumps between 0 and 1 in a constant manner when the ADPLL is at lock-in state. In Fig 5.15 (b), the zoom-in plot of the "Lock-In State" from 5.15(a) is shown more clearly.



(a)

(b)

Fig 5.15 (a) ADPLL's DCO control words acquisition and (b) zoom-in plot

The phase difference between "F_{REF}" (reference clock) and "F_{DIV}" (divider output) signals are almost eliminated and these two signals have almost the same frequency when at "Lock-In State". Since the reference clock frequency is 120MHz and the "Lock-In State" is at almost 2μs, the ADPLL finishes the lock-in process within almost 240 reference clock cycles.

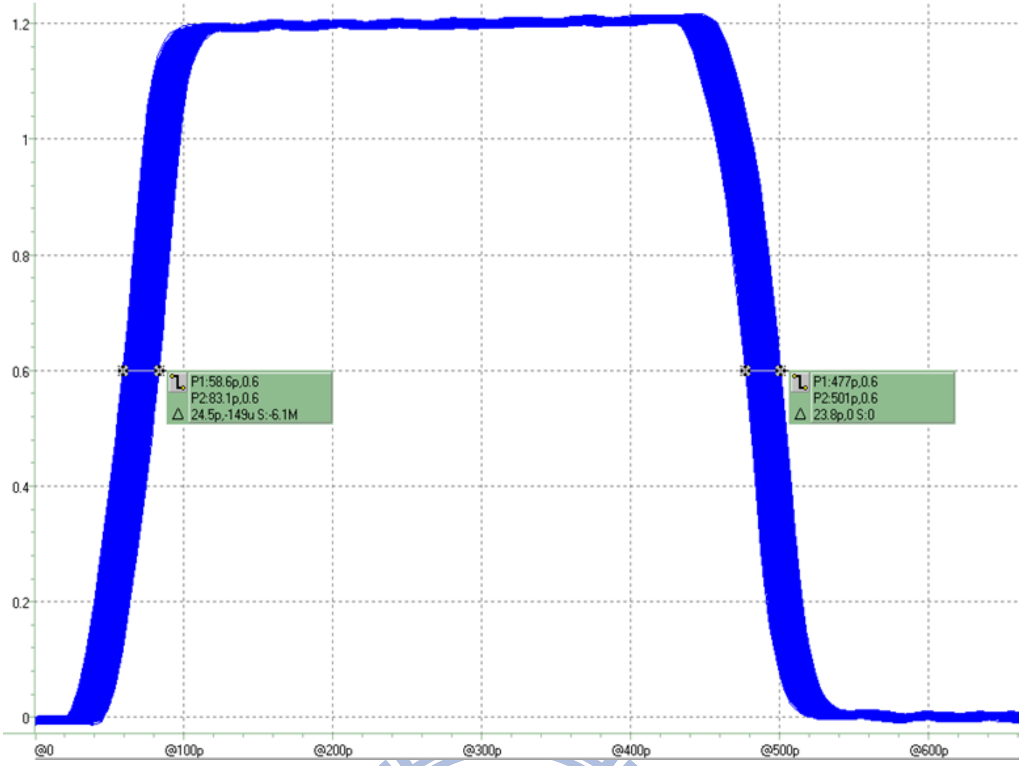


Fig. 5.16 ADPLL peak-to-peak jitter plot

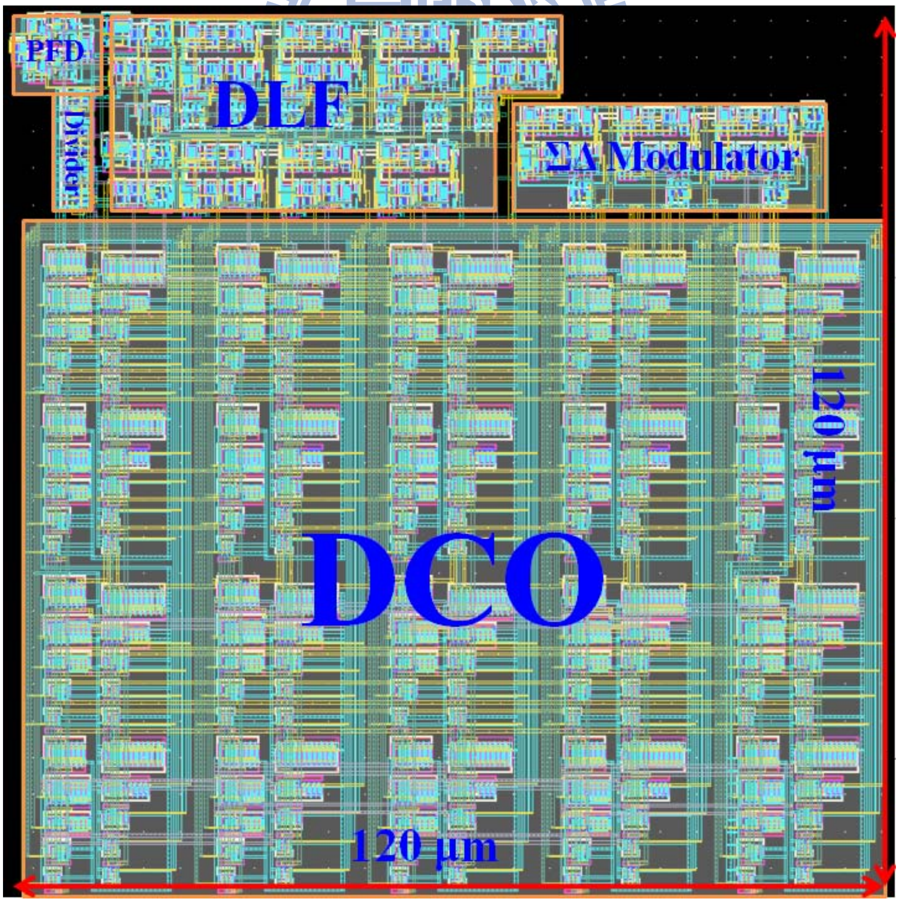


Fig 5.17 Layout of proposed ADPLL

In Fig 5.16, we measure the ADPLL's peak-to-peak jitter from one of the twenty output phases. The peak-to-peak jitter of the ADPLL is 24.5ps and 23.8ps at the rising edge and falling edge, respectively. And, the RMS jitter is 3.96ps and 3.85ps at the rising edge and falling edge, respectively.

Fig 5.17 shows the layout of the ADPLL, the core area is 120µm×120µm. In Table 5.4, the proposed ADPLL's performance is listed. And Table 5.5 shows the performance comparison result of the proposed ADPLL with other ADPLLs.

Table 5.4 ADPLL performance summary

Items	Performance
Technology	***TSMC 65nm 1P9M CMOS***
Power Supply	***1.2V***
Reference Frequency	***120MHz***
DCO target Frequency	***1.2GHz***
DCO Tuning Range	***0.888GHz ~ 1.526GHz***
Lock-In Time	***<240 reference cycles***
Peak to Peak Jitter	***24.5ps***
RMS Jitter	***3.96ps***
Core Area	***Total: 120um × 120um*** ***DCO: 120um × 90um***
DCO Control Words	***Coarse Tune Bits: C8, C4, C2, C1*** ***Fine Tune Bits: F4, F2, F1*** ***Dithering Bit: F0***
Power Consumption	***7.546mW***

Table 5.5 Comparison of ADPLL performance

ADPLL	**Proposed ADPLL**	JSSC 2008 [18]	JSSC 2009 [34]	ISSCC 2004 [23]
Technology	***TSMC 65nm 1P9M CMOS***	65nm digital CMOS SOI	65nm CMOS	90nm/ standard digital CMOS
Power Supply	***1.2V***	1.3V (programmable)	1.2V	0.7V ~ 2.4V (typical 1.2V)
Reference Frequency	***120MHz***	N/A	25MHz	0.03MHz ~ 65MHz
DCO target Frequency	***1.2GHz***	N/A	3GHz	N/A
DCO Tuning Range	***0.888GHz ~ 1.526GHz***	500MHz ~ 8GHz (programmable with Power Supply)	N/A	0.18MHz ~ 600MHz
Lock-In Time	***<240 cycles***	N/A	N/A	N/A
Jitter	$J_{peak\text{-}to\text{-}peak}$=24.5ps J_{RMS}=3.96ps	period jitter: 0.7ps rms long-term jitter: 6ps rms	N/A	66.7ps
Core Area	***Total: 120um×120um*** ***DCO: 120um×90um***	200um×150um	0.4mm^2	0.18mm^2
Power Consumption	***7.546mW @*** F_{out}=1.2GHz	8mW/GHz @ 1.2V and 1.6mW/GHz @ 0.5V	<10mw	1.7mW @ 1V, REF=13MHz, Fout=520MHz
Differential Output	***Yes***	No	No	No
Multiple Output Phase	***20***	1	1	1

5.3.6 Triangular Wave Profile Generator

The modulation profile will have dominant effect on the spreading clock's energy spectrum distribution. Because the triangular modulation profile has a regular frequency deviation in a fixed time, it is well known that the triangular modulation profile can get the most flattened spreading clock's energy spectrum. In actual, we adopt an up/down counter to achieve the triangular wave. Fig 5.18 shows the block diagram of the triangular modulation profile generator [32].

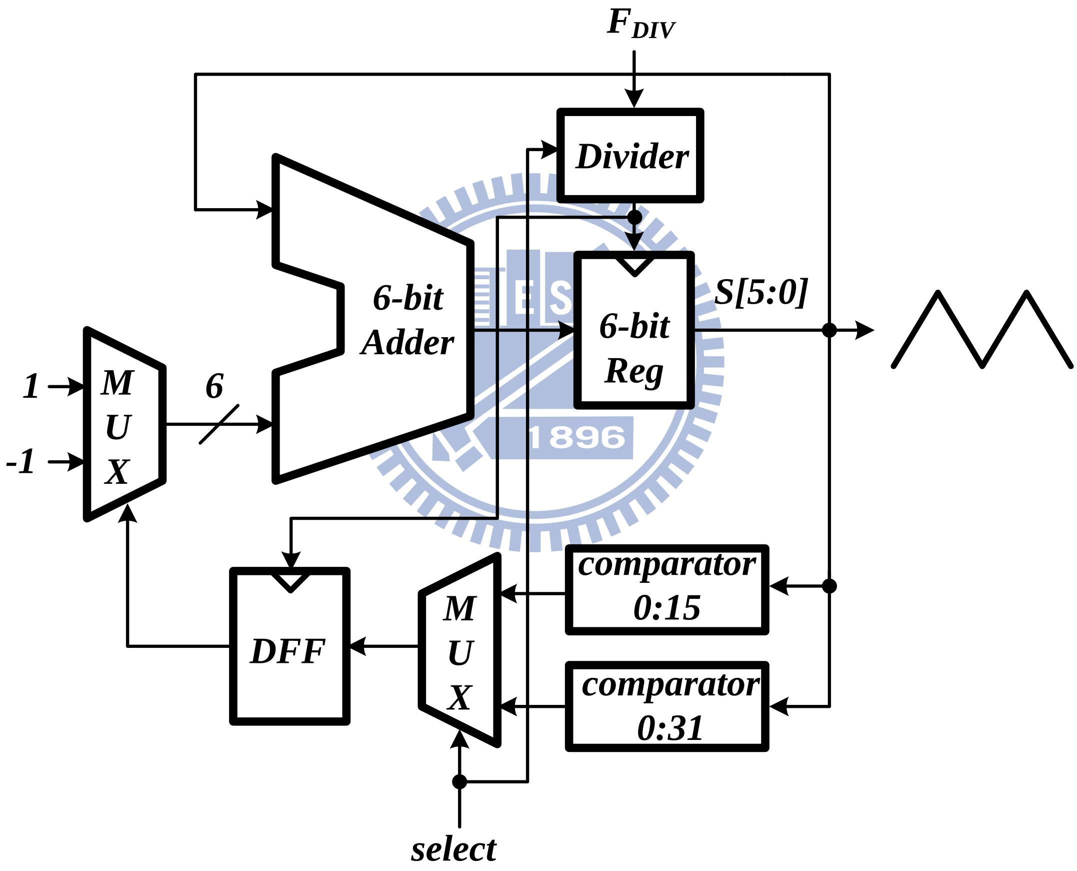


Fig 5.18 Triangular modulation profile generator

According to the SATA-III specification, the modulation frequency is 30~33 KHz, and the modulation deviation is no more than 5000ppm with down spread requirement. In the circuit, the divider determines the modulation frequency of the spread spectrum clock generator. The divider is programmable (10 phases: divider ratio=120; 20 phases: divider ratio=60) when different phases are used.

The modulation deviation is $-\frac{\frac{A}{M}}{N\times P}$, where A is the output number in modulation profile generator, M is the accumulator's maximum number in 1^{st} order ΣΔ modulator, N is the ADPLL's divider ratio, and P is the phase number provided by ADPLL's DCO. Thus, as shown in Table 5.6, in 10 phase SSCG, the maximum number of the counter is 16 and the total counting number is $16\times 2=32$, so the divider ratio in Fig 5.18 is 120 and the modulation frequency can be within 30~33 KHz (31.25 KHz). Similarly, with 20 phase SSCG, the maximum number of the counter is 32 and the counting number is $32\times 2=64$, so the divider ratio is 60 with the same modulation frequency 31.25 KHz. (clock=120MHz is the ADPLL's reference clock's frequency)

Table 5.6 Parameters in programmable modulation profile generator

	Maximum number (modulation deviation)	**Divider number (modulation frequency)**
10 phase	**A_{max}=N×P×M×5000ppm =10×10×2^5×5000ppm=16**	**120MHz÷(16×2) ÷120 =31.25KHz**
20 phase	**A_{max}=N×P×M×5000ppm =10×20×2^5×5000ppm=32**	**120MHz÷(32×2) ÷60 =31.25KHz**

The following equations show the control Boolean equations that are used in the comparators when A_{max} is achieved.

In 10 phase SSCG case, “select” is zero and

$$output\_of\_comparator[0:15] \\ = \left(\overline{S_0}\,\overline{S_1}\,\overline{S_2}\,\overline{S_3}S_4\overline{S_5}\right)\times\overline{select}+\left(S_0+S_1+S_2+S_3+S_4+S_5\right)\times select \tag{5.2}$$

In 20 phase SSCG case, “select” is one and

$$output\_of\_comparator[0:31] \\ = \left(\overline{S_0}\,\overline{S_1}\,\overline{S_2}\,\overline{S_3}\,\overline{S_4}S_5\right)\times\overline{select}+\left(S_0+S_1+S_2+S_3+S_4+S_5\right)\times select \tag{5.3}$$

5.3.7 ΣΔ Modulator

The reason we use 1st order ΣΔ modulator is that due to down spread requirement [32]. In tradition, the input number to the accumulator should be less than the maximum number the accumulator. According the previous section, the output of the profile generator is a 6-bit word. This profile word has maximum number of no more than 32. So, we use a 6-bit accumulator in the ΣΔ modulator. Next, we adopt the carry bit and the MSB (S_5) of the accumulator to be the outputs, as overflow0 and overflow1. Fig 5.19 shows the block diagram of the ΣΔ modulator.

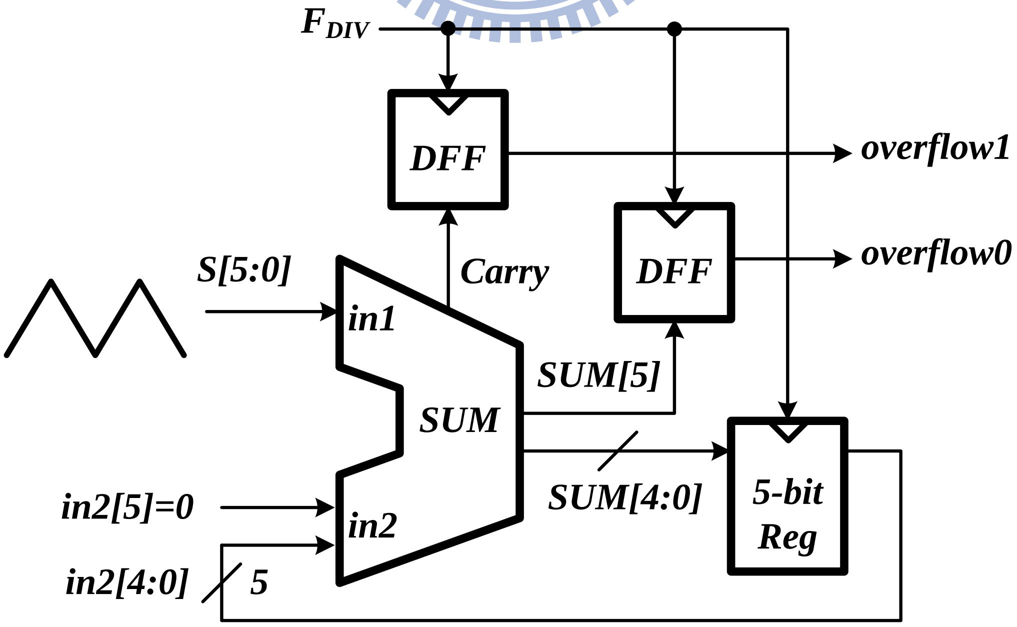


Fig 5.19 The 1st order ΣΔ modulator applied in SSCG

5.3.8 MUX Control Circuit

In the previous section, a 1st order ΣΔ modulator is used to generate two control codes for randomization. These two control codes (overflow0 and overflow1) are sent into the MUX control circuit, as shown in Fig. 5.20.

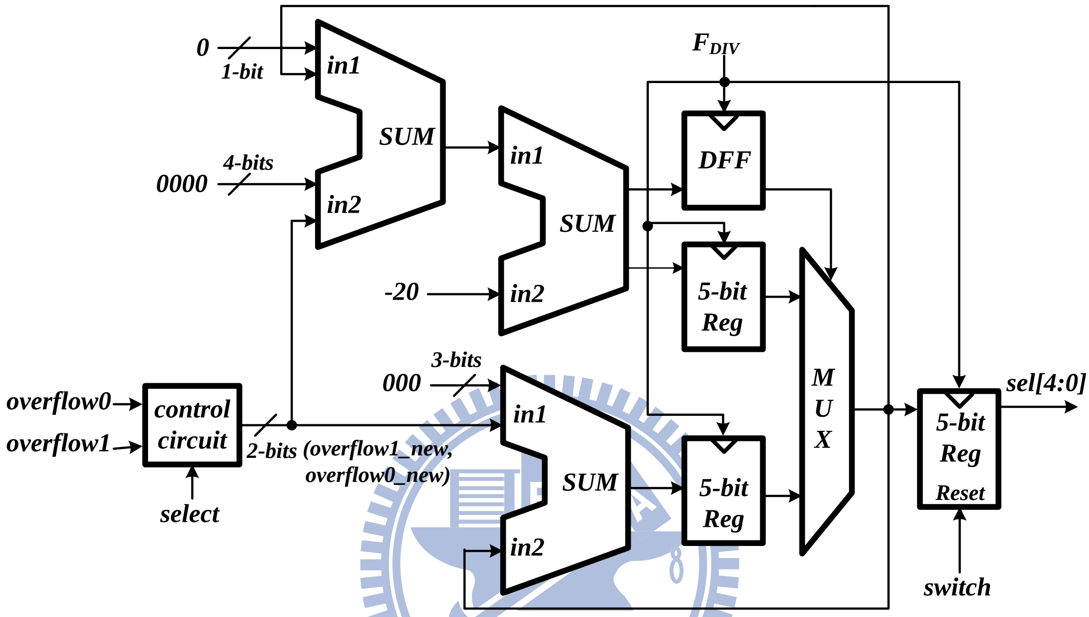


Fig 5.20 Programmable MUX control circuit

This MUX control circuit translates the input codes (overflow0, overflow1) from the ΣΔ modulator into the corresponding MUX selecting codes (sel0~sel4) in 10 or 20 phases SSCG. Normally, the design for controlling 10 phases SSCG is slightly different from that of 20 phases SSCG. For example, if the input code is “01”, which means that the MUX should rotate one phase, then we have to translate it into “10”, which means that the MUX would rotate two phases, in 10 phases SSCG.

The following equations show the control circuit’s transforming equations, the control circuit is shown in Fig 5.20. Table 5.7 is the truth table of the transforming equations of the control circuit.

In 10 phase SSCG case, "select" is zero and

$$overflow1\_new = overflow1 + overflow0 \times \overline{select} \tag{5.4}$$

In 20 phase SSCG case, "select" is zero and

$$overflow0\_new = overflow0 \times select \tag{5.5}$$

Table 5.7 Transforming equations of the control circuit

overflow1	overflow0	select	overflow1_new	overflow0_new
0	0	0	0	0
0	0	1	0	0
0	**1**	**0**	**1**	**0**
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	**1**	**0**	**1**	**0**
1	1	1	1	1

The reason for changing the input code is that the selecting MUX (introduced in next section) behind the MUX control circuit has different choices when selecting 10 and 20 phases SSCG.

5.3.9 Multiplexer

The use of the multiplexer is to select which phase from the DCO should be sent into the divider and latter be imported to the PFD in the ADPLL. That is why we call it a "phase rotation" mechanism in spread spectrum clocking. The multiplexer consists of logic circuits like that shown in Fig 5.21.

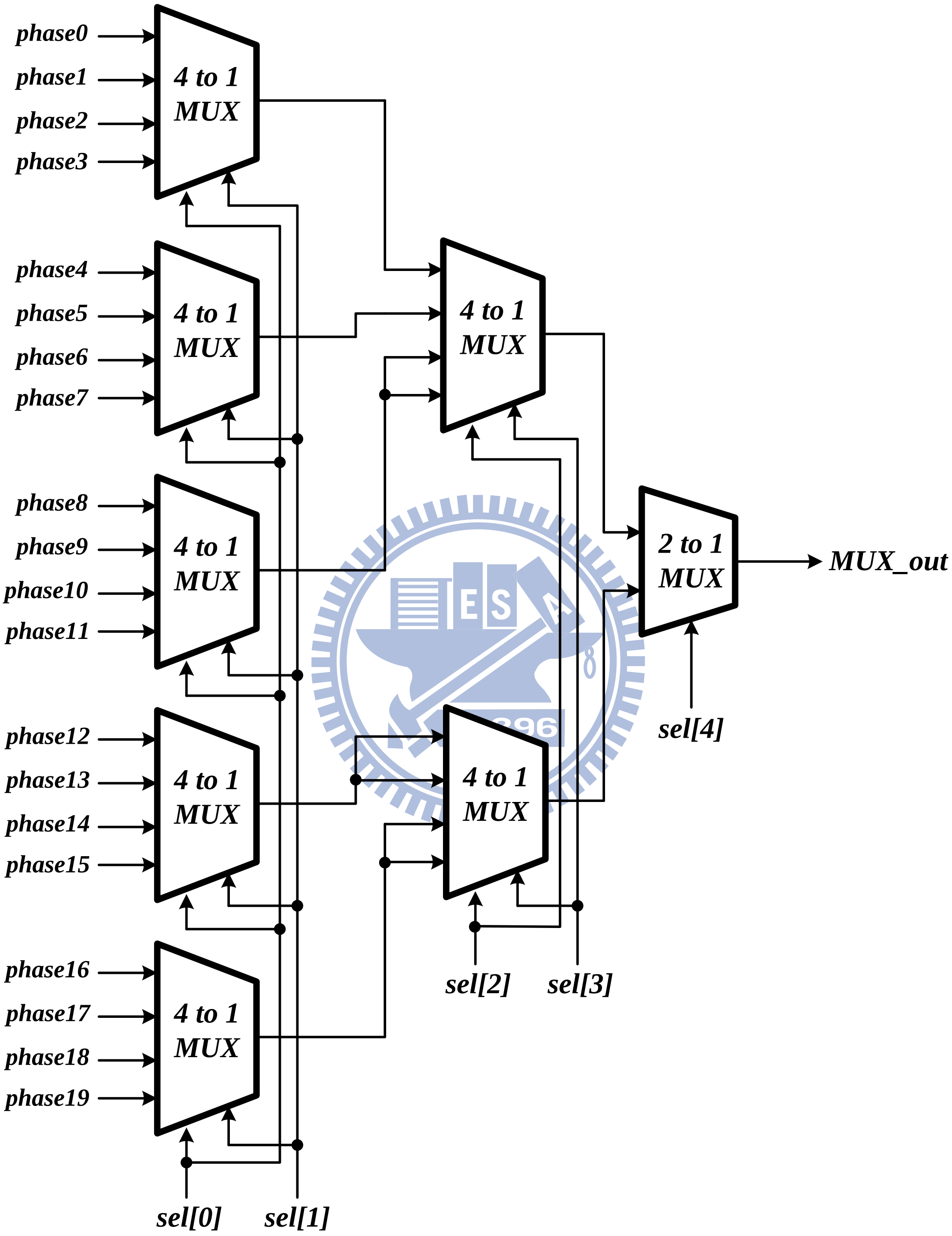


Fig 5.21 The 20 to 1 multiplexer

Once the new selected phase lags the original phase, the PFD would judge the frequency from the frequency divider lags reference frequency. Next, the PFD would send a “DW” signal and let the digital loop filter to change its control code sent to the DCO. Finally, the DCO’s oscillation frequency would increase. And it will become “up spread” specification.

Contrarily, if the new selected phase leads the original phase, the PFD would judge the frequency from the frequency divider leads reference frequency. Next, the PFD would send a “UP” signal and let the digital loop filter to change its control code sent to the DCO. Finally, the DCO’s oscillation frequency would decrease. And it will become “down spread” specification.

5.3.10 SSCG System

Here we show simulation results of the proposed SSCG. Fig 5.22 shows the carrier (F_{DCO}) spectrum without spread-spectrum clocking when the proposed ADPLL is locked. The PSD of “F_{DCO}” is 0dBv at 1.2GHz.

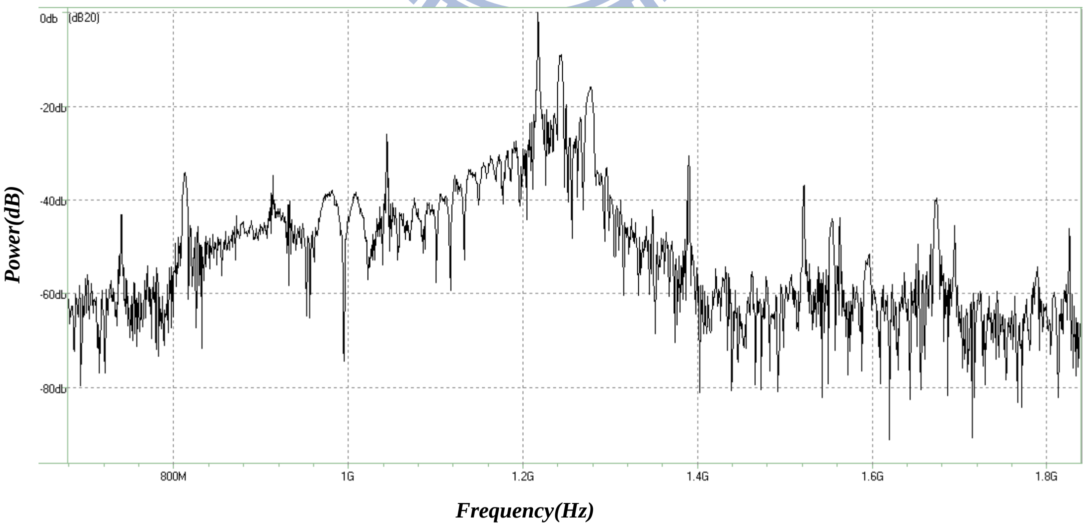
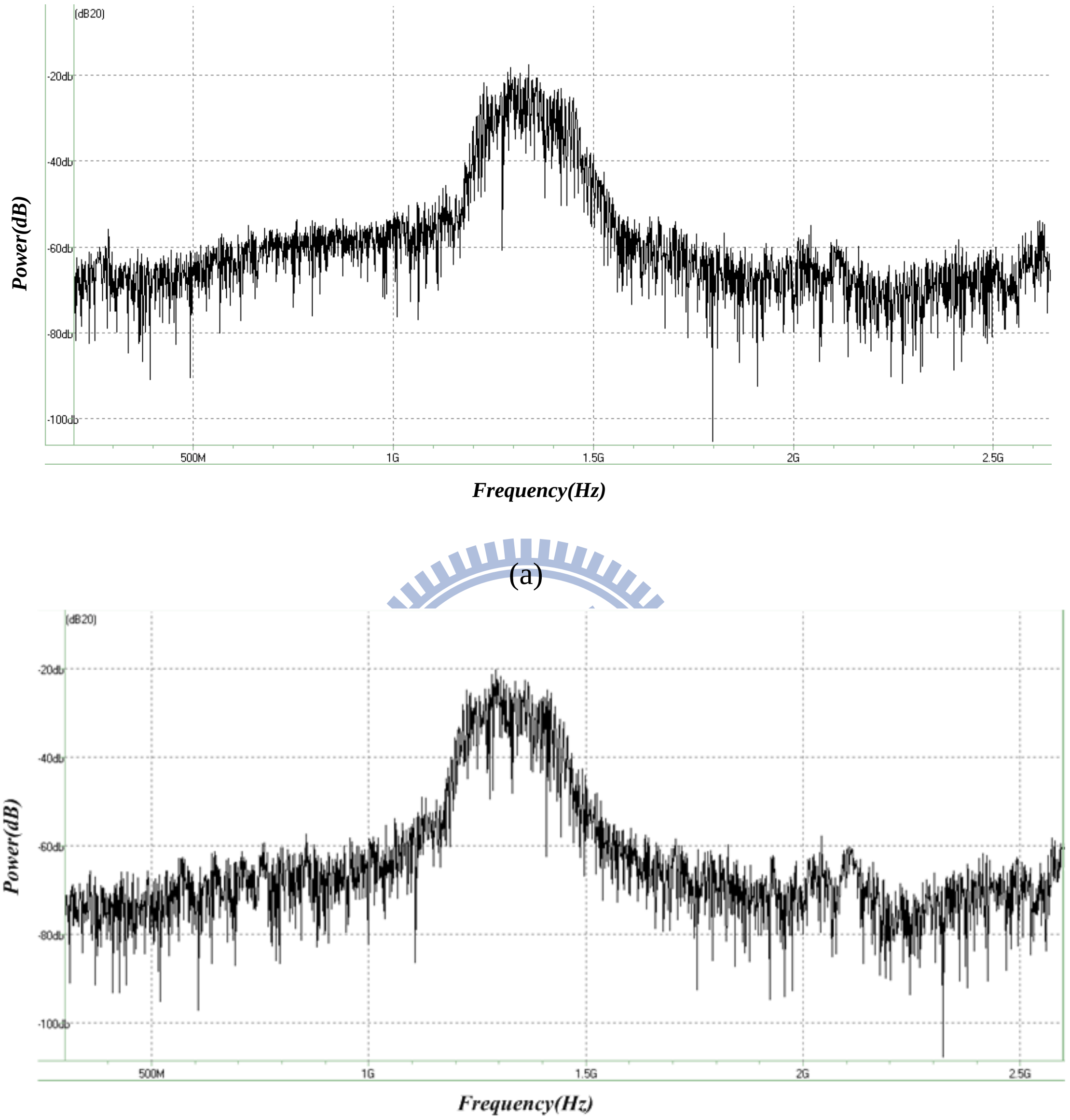


Fig 5.22 Carrier spectrum without spread spectrum (0dBV @ 1.2GHz)

Next, carrier spectrum with 10 phases and 20 phases spread-spectrum clocking are shown in Fig 5.23 (a) and Fig 5.23 (b), respectively. We can see that the peak magnitude of the carrier spectrum is reduced about -18.6dB and -20.4dB in Fig 5.23 (a) and Fig 5.23 (b), respectively. Fig 5.24 shows chip layout of the SSCG.



(a)

(b)

Fig 5.23 Carrier spectrum with spread spectrum in

(a) 10 phases SSCG and (b) 20 phases SSCG

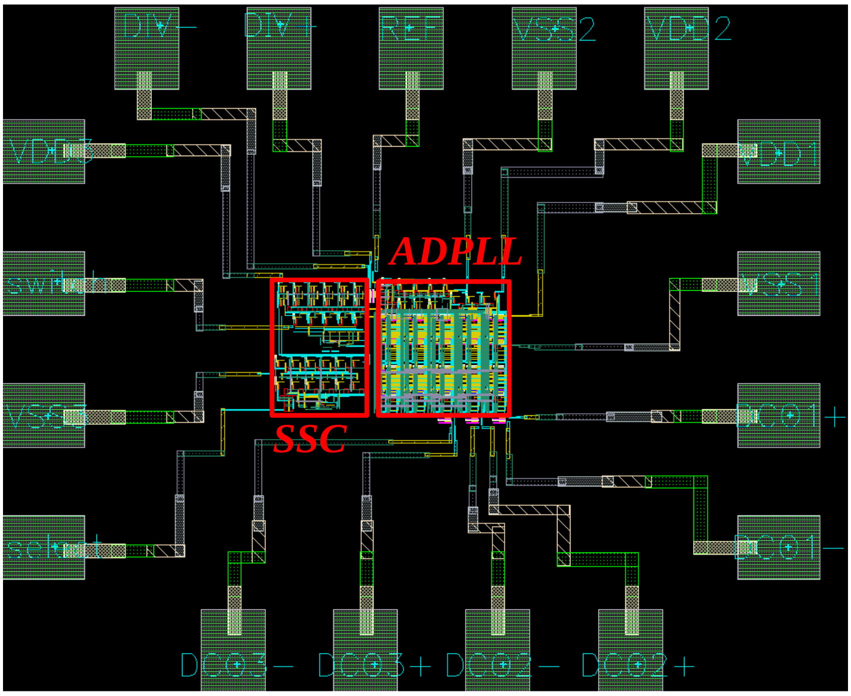


Fig 5.24 Chip layout of proposed SSCG

Table 5.8 shows the SSCG layout attributes, including PAD names and the number of PADs. Table 5.9 gives the SSCG performance in 10 phases and 20 phases spread-spectrum clocking. Table 5.10 shows the comparison of SSCG performance.

Table 5.8 SSCG layout attributes

Attribute	Pin Name	# of PAD
Power	***VDD1, GND1(for DCO)*** ***VDD2, GND2(for ADPLL control)*** ***VDD3, GND3(for SSC circuit)***	**6**
Input	F_{REF}	**1**
Output	F_{DCO1}, F_{DCO1b}, F_{DCO2},F_{DIV} ***(all with differential outputs)***	**8**
Control	***select, switch***	**2**

Table 5.9 Programmable SSCG performance comparison

	10 phase SSCG	20 phase SSCG
Total Power	***8.329mW***	***8.618mW***
EMI Reduction	***-18.6dB***	***-20.4dB***

Table 5.10 Comparison of SSCG performance

SSCG	***Our Work***	ISSCC 2005 [28]	ISSCC 2005 [29]	ISCAS 2005 [30]
Technology	***65nm***	0.15μm	0.18μm	0.18μm
Modulation Method	***Modulation on phase***	Modulation on divider	Modulation on phase	Modulation on divider
Divider Counter	***10***	37.5/75	60	73/75
Oscillator Output Phase	***10/20***	6	10	8
Oscillator Output Frequency	***1.2GHz***	1.5GHz	1.5GHz	1.5GHz
Modulation Deviation	***5000ppm (6MHz)***	5000ppm (7.5MHz)	5000ppm (7.5MHz)	5000ppm (7.5MHz)
EMI Reduction	***-18.6dB (10 phase)/ -20.4dB (20 phase)***	20.3dB	9.8dB	20.44dB
P-P jitter (non-SSC)	***24.5ps***	N/A	41ps	80ps
Power Consumption	***8.329mW (10 phase)/ 8.618mW (20 phase)***	54mW	N/A	55mW
Supply Voltage	***1.2V***	1.5V	1.8V	1.8V

5.4 Measurement Setup

Fig 5.25 shows the measurement setup of the SSCG. The input reference clock (F_{REF}) of the SSCG is generated by signal generator (Agilent 81130A). The power generator (Agilent 3610A) provides DC 1.2V supply and ground. The output spectrum (F_{DCO}) is observed by a Spectrum Analyzer (Agilent E4440A) and the time domain waveforms from ADPLL's divider (F_{DIV}) and DCO (F_{DCO}) is derived from the oscilloscope (Tektronix TDS9124C).

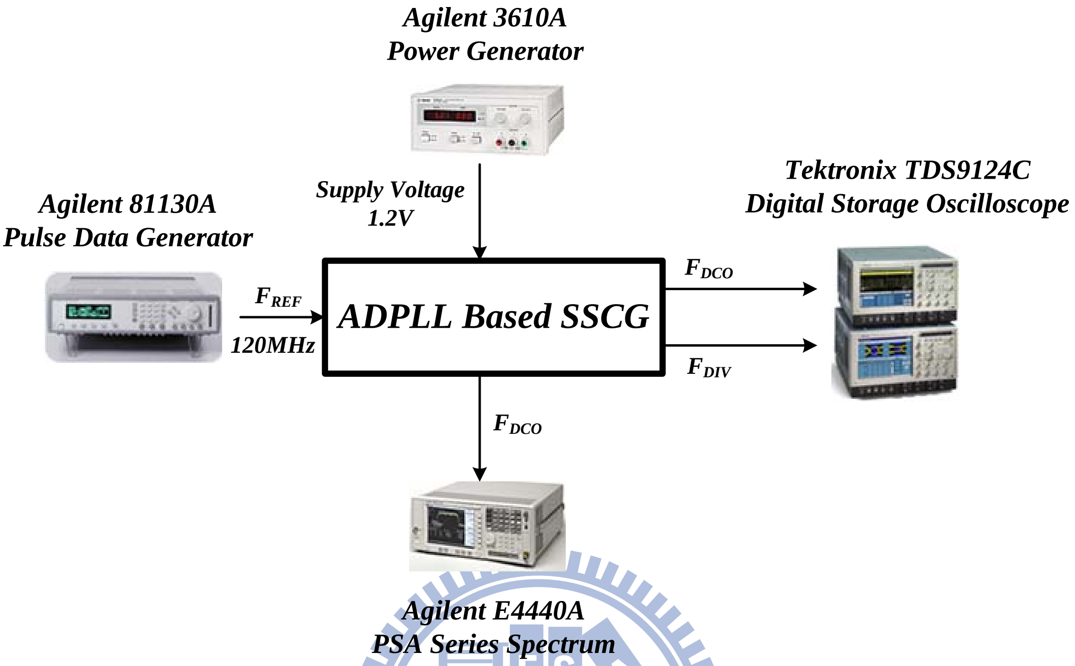


Fig 5.25 Measurement setup of the SSCG

Chapter 6
Conclusions

In this thesis, an all-digital phase-locked loop (ADPLL) and a programmable spread-spectrum clock generator (SSCG) for Serial-ATA with switching phase mechanism are presented. The proposed circuit is fabricated in a 65nm CMOS process.

An all-digital phase-locked loop is designed with multi-output phases by using differential DCO architecture. Also, the 1st order ΣΔ modulator is inserted between the digital loop filter and the DCO to control the DCO for enhancing its frequency resolution.

The frequency range of the ADPLL is from 0.888GHz to 1.526GHz under 1.2V supply voltage. The proposed ADPLL can lock the frequency within 240 reference clock cycles. The measured peak-to-peak and RMS jitter is 24.5ps and 3.96ps, respectively. The core area is 120μm×120μm (DCO area: 120μm×90μm) and power consumption is 7.546mw.

The SSCG can do spread-spectrum clocking with 10 phases or 20 phases. For Serial ATA specifications, our SSCG is designed with down spread 5000 ppm and a triangular waveform of modulation frequency 31.25kHz.

The SSCG is also operated under 1.2V supply voltage. The SSCG’s total power consumption is about 8.329mW and 8.618mW under the conditions of 10 phases and 20 phases, respectively. The entire core area of SSCG is 200μm×120μm, total pad number is 17. The EMI reduction is about -18.6dB and -20.4dB with 10 phases and 20 phases spread-spectrum clocking, respectively.

Bibliography

[1] Roland E. Best, “Phase Locked-Loops : Design, Simulation, and Applications,” McGraw-Hill, 6th edition

[2] Staszewski Balsara, “All-Digital Frequency Synthesizer in Deep-Submicron CMOS,” WILEY

[3] Weste/Harris, “CMOS VLSI Design : A Circuits and Systems Perspective,” Addison Wesley, 3rd edition

[4] Ecliptek Corporation. 2005. Programmable Spread Spectrum Quartz Crystal Oscillators Reduce EMI for High Speed Digital Systems. Retrieved Aug. 2007, from http://www.ecliptek.com/tech/ss_emi_digi.html

[5] T. A. Riley, M. A. Copeland, and T. A. Kwasniewski, “Delta-Sigma Modulation in Fractional-N Frequency Synthesis,” IEEE Journal of Solid-State Circuits, vol. 28, no. 5, pp. 553-559, May 1993.

[6] R. Staszewski, S. Vemulapalli, P. Vallur, J. Wallberg, and P. T. Balsara, “1.3 V 20 ps Time-to-Digital Converter for Frequency Synthesis in 90-nm CMOS,” IEEE Transactions on Circuits and Systems—II, vol. 53, no. 3, pp. 220-224, March 2006

[7] I. C. Hwang, S. H. Song, S. W. Kim, “A Digitally Controlled Phase-Locked Loop with a Digital Phase-Frequency Detector for Fast Acquisition,” IEEE Journal of Solid-State Circuits, vol. 36, no. 10, pp. 1574-1581, Oct. 2001

[8] J. Dunning, G. Garcia, J. Lundberg, Ed Nuckolls, “An all-digital PLL with 50 cycle lock time suitable for high performance microprocessors,” IEEE Journal of Solid-State Circuits, vol. 30, no. 4, pp. 412-422, April 1995

[9] T. Y. Hsu, C. C. Wang, and C. Y. Lee, “Design and Analysis of a Portable High-Speed Clock Generator,” IEEE Transactions on Circuits and Systems—II: Analog and Digital Signal Processing, vol. 48, no. 4, pp. 367-375, April 2001

[10] C. C. Chung and C. Y. Lee, “An all-digital phase-locked loop for high-speed clock generation,” IEEE Journal of Solid-State Circuits, vol. 38, no. 2, pp. 347-351, Feb. 2003

[11] D. Sheng, C. C. Chung, and C. Y. Lee, “A Fast-Lock-In ADPLL with High-Resolution and Low-Power DCO for SoC Applications,” IEEE APCCAS, pp. 105-108, Dec. 2006

[12] D. Sheng, C. C. Chung, and C. Y. Lee, “An Ultra-Low-Power and Portable Digitally Controlled Oscillator for SoC Applications,” IEEE Transactions on Circuits and Systems-II: Express Briefs, vol. 54, no. 11, pp. 954-958, Nov. 2007

[13] P. L. Chen, C. C. Chung, and C. Y. Lee, “A Portable Digitally Controlled Oscillator Using Novel Varactors,” IEEE Transactions on Circuits and Systems—II: Express Briefs, vol. 52, no. 5, pp. 233-237, May 2005

[14] L. Dai, “A Low Phase Noise CMOS Ring Oscillator with Differential Control and Quadrature Outputs,” IEEE ASIC/SOC Conference, pp. 134-138, 2001 Proceedings

[15] J. C. Zhuang, Q. J. Du, and T. Kwasniewski, “A 4GHz Low Complexity ADPLL-based Frequency Synthesizer in 90nm CMOS,” IEEE Custom Integrated Circuit Conference, pp. 543-546, Sep. 2007

[16] Stazewski, C. M. Huang, Barton, “A digitally controlled oscillator in a 90nm digital CMOS process for mobile phones,” IEEE Journal of Solid-State Circuits, vol. 40, no. 11, pp. 2203-2211, Nov. 2005

[17] T. Olsson, “A digitally controlled PLL for SOC applications,” IEEE Journal of Solid-State Circuits, vol. 39, no. 5, pp. 751-759, May 2004

[18] J. A. Tierno, A. V. Rylyakov, D. J. Friedman, “A wide power supply range, wide tuning range all static CMOS all digital PLL in 65nm SOI,” IEEE Journal of Solid-State Circuits, vol. 43, no. 1, pp. 42-51, Jan. 2008

[19] P. L. Chen, C. C. Chung, J. N. Yang, and C.Y. Lee, “A clock generator with cascaded dynamic frequency counting loops for wide multiplication range applications,” IEEE Journal of Solid-State Circuits, vol. 41, no. 6, pp.1275-1285, June 2006

[20] J. G. Maneatis, J. Kim, I. McClatchie, J. Maxey, and M. Shankarads, “Self-biased high-bandwidth low-jitter 1-to-4096 multiplier clock generator PLL,” IEEE Journal of Solid-State Circuits, vol. 38, no. 11, pp.1795–1803, Nov. 2003

[21] P. L. Chen, C. C. Chung, and C. Y. Lee, “A portable digitally-controlled oscillator using novel varactors,” IEEE Transactions on Circuits and Systems—II: Express Briefs, vol. 52, no. 5, pp. 233–237, May 2005

[22] T. Watanabe and S. Yamauchi, “An all-digital PLL for frequency multiplication by 4 to 1022 with seven-cycle lock time,” IEEE Journal of Solid-State Circuits, vol. 38, no. 2, pp. 198–204, Feb. 2003

[23] J. Lin, B. Haroun, T. Foo, J. S. Wang, B. Helmick, S. Randall, T. Mayhugh, C. Barr, and J. Kirkpartick, “A PVT tolerant 0.18 MHz to 600MHz self-calibrated digital PLL in 90 nm CMOS process,” IEEE International Solid-State Circuits Conference, pp. 488-489, Feb. 2004

[24] N. D. 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 , Jan. 2005

[25] T. Olsson, P. Nilsson, "A Digitally Controlled PLL for SoC applications," IEEE Journal of Solid-State Circuits, vol. 39, no. 5, pp. 751-760, May 2004

[26] T. Y. Hsu, C. C. Wang, and C. Y. Lee, "Design and Analysis of a Portable High-Speed Clock Generator," IEEE Transactions on Circuits and Systems—II: Analog and Digital Signal Processing, vol. 48, no. 4, pp. 367-375, April 2001

[27] V. Ramakrishnan, P. T. Balsara, "A Wide-Range, High-Resolution, Compact, CMOS Time to Digital Converter," Proceedings of the 19th International Conference on VLSI Design (VLSID'06)

[28] M. Kokubo, T. Kawamoto, T. Oshima, T. Noto, M. Suzuki, S. Suzuki, T. Hayasaka, T. Takahashi and J. Kasai, "Spread-Spectrum Clock Generator for Serial ATA using Fractional PLL Controlled by ΔΣ Modulator with Level Shifter," IEEE International Solid-State Circuits Conference, pp. 160-590, Feb. 2005

[29] H. R. Lee, O. Kim, G. Ahn, and D. K. Jeong, "A Low-Jitter 5000ppm Spread Spectrum Clock Generator for Multi-channel SATA Transceiver in 0.18um CMOS," IEEE International Solid-State Circuits Conference, pp. 162-590, Feb. 2005

[30] W. T. Chen, J. C. Hsu, H. W. Lune, and C. C. Su, "A Spread Spectrum Clock Generator for SATA-II," IEEE ISCAS, pp. 2643 – 2646, May 2005

[31] F. S. Kung, "A Programmable All-Digital Phase-Locked Loop," M. S. dissertation, Dep. of Electrical and Control Engineering, NCTU, Taiwan, Jan. 2008

[32] C. H. Chuang, "A Programmable Spread Spectrum Clock Generator for Serial ATA 6Gbps", M. S. dissertation, Institute of Electrical Engineering, NCTU. Taiwan, July 2006

[33] Serial ATA Workgroup, *SATA: High Speed Serialized AT Attachment*, Revision 2.6, March 2007

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

[35] S. Bolger and S. O. Darwish, “Use spread-spectrum techniques to reduce EMI,” EDN; Boston, vol.43, May 1998

[36] H. H. Chang, I. H. Hua, and S. I. Liu, “A Spread-Spectrum Clock Generator With Triangular Modulation,” IEEE Journal of Solid-State Circuits, vol. 38, no. 4, pp. 673-676, April 2003

[37] S. R. Li, “Digitally Phase Adjusted Clock Data Recovery for Receiver with Spread Spectrum Clocking”, M. S. dissertation, Institute of Electrical Engineering, NCTU. Taiwan, Oct. 2008