國 立 中 正 大 學

資訊工程研究所碩士論文

應用於寬頻操作之全數位時脈
責任週期校正與輸出相位對齊電路

A Wide-Range All-Digital Duty-Cycle Corrector with Output Clock Phase Alignment in 65nm CMOS Technology

研 究 生： 沈頌恩

指導教授： 鍾菁哲 博士

中華民國 一百 年 七 月

國立中正大學碩士班研究生

學位考試同意書

本人所指導 資訊工程學系

研究生 沈頌恩 所提之論文

應用於寬頻操作之全數位時脈責任週期校正與輸出相位對齊電路 (A wide-range all-digital duty-cycle corrector with output clock phase alignment in 65nm CMOS technology.)

同意其提付 碩 士學位論文考試

指導教授 蘇菁莎 簽章

100 年 6 月 2 日

國立中正大學碩士學位論文考試審定書

資訊工程學系

研究生沈頌恩 所提之論文

應用於寬頻操作之全數位時脈責任週期校正與輸出相位對齊電路 (A wide-range all-digital duty-cycle corrector with output clock phase alignment in 65nm CMOS technology.)

經本委員會審查，符合碩士學位論文標準。

學位考試委員會
召 集 人＿＿＿＿＿＿＿＿簽章

委 員

指 導 教 授＿＿＿＿＿＿＿＿簽章

中華民國 100 年 7 月 12 日

博碩士論文授權書

（本聯請裝訂於論文紙本書名頁前空白處，供學校圖書館做爲授權管理用）

ID:099CCU00392042

本授權書所授權之論文爲授權人在 國立中正 大學(學院) 資訊工程研究所 系所_____組 99 學年度第 二 學期取得 碩 士學位之論文。

論文題目：應用於寬頻操作之全數位時脈責任週期校正與輸出相位對齊電路

指導教授：鍾菁哲,Cheng, Ching-Che

茲同意將授權人擁有著作權之上列論文全文(含摘要)，提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印，此項授權係非專屬、無償授權國家圖書館及本人畢業學校之圖書館，不限地域、時間與次數，以微縮、光碟或數位化方式將上列論文進行重製，並同意公開傳輸數位檔案。

紙本論文：茲同意將授權人擁有著作權之上列論文全文(含摘要)，提供讀者基於個人非營利性質之閱覽或列印，此項授權系非專屬、無償授權國立中正大學圖書館做爲編目上架及公開陳列閱覽使用。

☐ 校內外立即開放

☐ 校內立即開放，校外於　年　月　日後開放

☑ 校內於 2016 年 07 月 27 日；校外於 2016 年 07 月 27 日後開放

☐ 其他 _____

授權人：沈頌恩

親筆簽名或蓋章：沈頌恩　　民國100年7月27日

應用於寬頻操作之全數位時脈
責任週期校正與輸出相位對齊電路

學生 ： 沈頌恩　　指導教授 ： 鍾菁哲

國立中正大學資訊工程學系研究所

摘要

在此論文中，我們針對應用於寬頻操作之全數位時脈責任週期校正(ADDCC)與輸出相位對齊電路的設計。在高速資料傳輸與資料擷取電路系統中，例如：雙倍數據率同步動態隨機存取(DDR SDRAM)與雙倍取樣類比至數位轉換器，會使用時脈訊號的正負緣來加速擷取資料，因而希望系統時脈的責任週期為百分之五十。但系統時脈訊號經過電晶體不平衡的充電與放電時間，和製程溫度電壓(PVT)漂移的改變，皆會使得時脈責任週期不為百分之五十，因而造成擷取資料發生錯誤。我們整理近年來相關論文研究架構，並討論先前研究的不同點與改善處。本論文提出使用全數位的控制方式，不但加快電路鎖定時間，並改善傳統由電壓控制充放電時所引起的漏電問題。此外提出創新的高解析度時脈責任週期校正方法，可提高校正後時脈訊號責任週期為百分之五十的精細度，並能解決先前架構中，使用量化時脈週期成數位訊息(TDC)，量化精細度不足的問題。以及在量化後，使用另一半時脈週期延遲電路(HCDL)，來所產生輸出訊號，因而產生校正誤差問題，尤其是在先進半導體製程當中，由於晶片中製成遷移的關係，這種電路不一致的問題會更加明顯。本論文所提出的 ADDCC 是使用 65nm 標準元件庫及標準 CMOS 製程來製作實現晶片，並驗證所提出的電路。

A Wide-Range All-Digital Duty-Cycle Corrector with Output Clock Phase Alignment in 65nm Technology

Student: Sung-En Shen Advisor: Dr. Ching-Che Chung
Department of Computer Science and Information Engineering,
National Chung Cheng University

Abstract

A Wide-Range All-Digital Duty-Cycle Corrector (ADDCC) with Output Clock Phase Alignment in 65nm Technology is presented in this dissertation. In high speed data transmitter application, such as double data rate (DDR) SDRAM and double sampling analog-to-digital converter (ADC), the positive edge and the negative edge of system clock are utilized for sampling the data. Thus, theses systems require an exact 50% duty-cycle of system clock. Nevertheless, system clock is affected by the unbalanced rise time and fall time of the clock buffers with process, voltage and temperature (PVT) variations, which cause error data latching when clock duty-cycle is not equal to 50%. We summarize some researches and architectures in prior years, moreover, discuss these differences and how to improve them. In this thesis, we use all-digital control method not only speed-up locking time than voltage control method, but also solve the leakage current problem of the voltage charge-pump control. Besides, we presented the novel high resolution ADDCC which can solve the restricted resolution of time-to-digital converter (TDC). The half-cycle delay line (HCDL) generates output clock signal by another mirror circuit will cause mismatch problem in nano-meter CMOS process when there has on-chip variations (OCVs). The proposed ADDCC is implemented on a standard performance (SP) 65nm CMOS process with standard cell library, and verify the performance of the proposed circuit.

Abbreviation

ADC	Analog-to-Digital Converter
ADDCC	All-Digital Duty-Cycle Corrector
DCCP	Digital-Controlled Charge Pump
DCD	Duty-Cycle Detector
DCDL	Digital-Controlled Delay Line
DDCC	Digital-controlled Duty-Cycle Correction
DDR	Double Data Rate
DFF	D flip-flop
DLL	Delay Locked Loop
HCDL	Half-Cycle Delay Line
OCVs	On-Chip Variations
PD	Phase Detector
PVT variations	Process, Voltage and Temperature variations
PWCL	Pulse-Width Control Loop
QDR	Quadrature Data Rate
SBDCD	Sampled-Based Duty-Cycle Detector
SBPD	Sampled-Based Phase Detector
SDRAM	Synchronous Dynamic Random Access Memory
SP	Standard Performance
TDC	Time-to-Digital Converter

Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Motivation

In high-speed devices, such as double data rate (DDR) SDRAM, double sampling analog-to-digital converter (ADC) and System-on-Chip (SoC) applications. The positive edge and the negative edge of clock are utilized for sampling the input data. Thus, these systems require an exact 50% duty-cycle of input clock which also ensure the system can operate on a right frequency. However, the clock signal is distributed over the chip using clock buffers, and thus, the duty-cycle of the clock is affected by the unbalanced rise time and fall time of the clock buffers with process, voltage and temperature (PVT) variations. In order to overcome such problem, many approaches have been proposed to adjust the clock duty-cycle to 50% to meet the system requirements, such as analog pulse-width control loop (PWCL) [1-5], all-digital PWCL [6-9], and all-digital duty-cycle corrector (ADDCC) [10-14]. In some applications, the DCC is combined with delay-locked loop (DLL) and located at the output side [15-20]. We will discuss these architectures in following sections.

Fig. 1.1 shows the block diagram of data latching. The CLK_IN's positive edge is utilized to trigger the DFF1, and latched the DATA_IN to Q1. After the Combination Circuit operation outputs the DATA_OUT. The CLK_IN passed through the Clock Tree and becomes the CLK_TREE. Because the CLK_IN is distributed over the chip by clock buffers, therefore, the duty-cycle of CLK_IN is affected by the unbalanced rise time and fall time of the clock buffers (Clock Tree or Wire Load) with PVT variations. Thus, the CLK_TREE's duty-cycle will be less or over 50%. When the CLK_TREE's negative edge is utilized to trigger the DFF2, it might happen to sample the incorrect DATA_OUT. Fig. 1.2 shows the DFF2 latched the incorrect DATA_OUT by CLK_TREE.

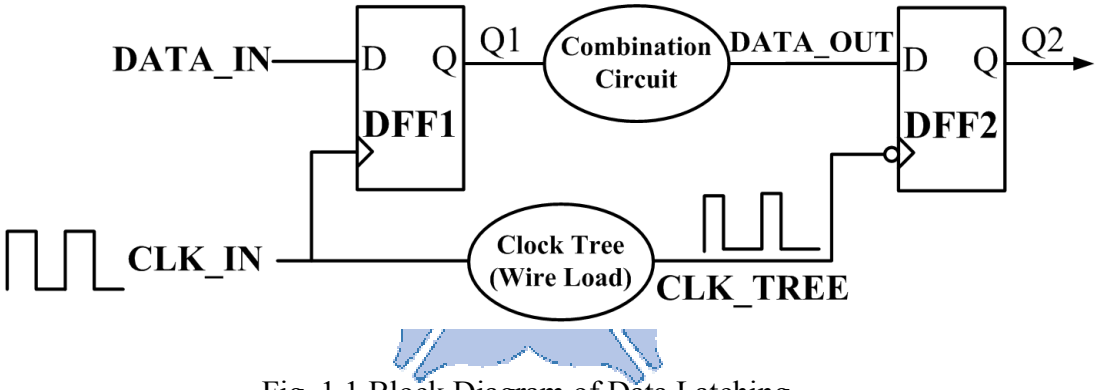


Fig. 1.1 Block Diagram of Data Latching

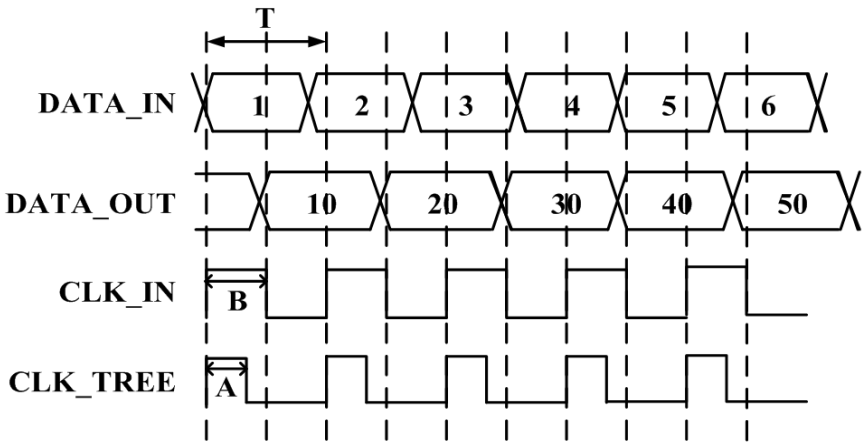


Fig. 1.2 Timing Diagram of Data Latching

1.2 Introduction to Duty Cycle Corrector

1.2.1 The Conventional Analog DCC

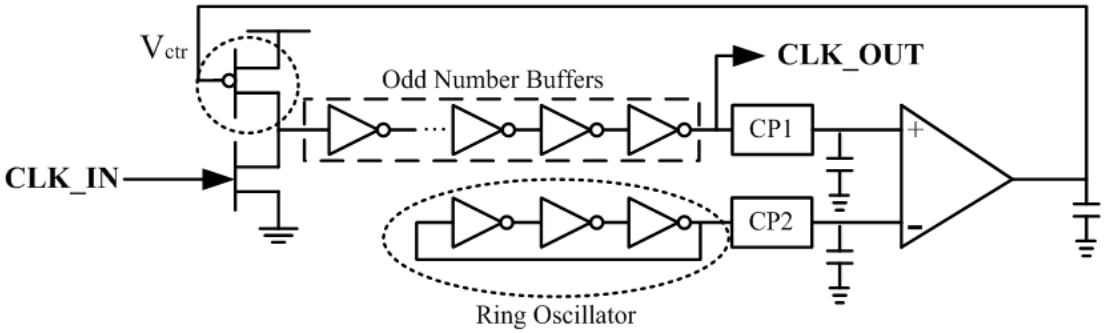


Fig. 1.3 Conventional Analog PWCL Architecture [1]

Fig. 1.3 shows the conventional analog PWCL architecture [1]. The conventional PWCL changes the feedback control voltage to adjust the duty-cycle of the input clock. Based on the architecture requirements, it requires a ring oscillator to produce 50% duty-cycle reference clock, and the operating range and the acceptable input duty-cycle error are very restricted in this architecture [1]. In recent year, the operating range of the PWCL can be improved by the linear control stage and the digitally controlled charge pump (DCCP) [4], the low-voltage designed PWCL can solve the power consumption [2] [5], and the fast-locking PWCL can speed-up the lock-in time [3]. However, these methods take long lock-in time, need for 50% duty-cycle reference clock and the leakage current problem of the charge-pump makes it not suitable for the nano-meter CMOS process.

1.2.2 The Half-Cycle Delay Line Based DCC

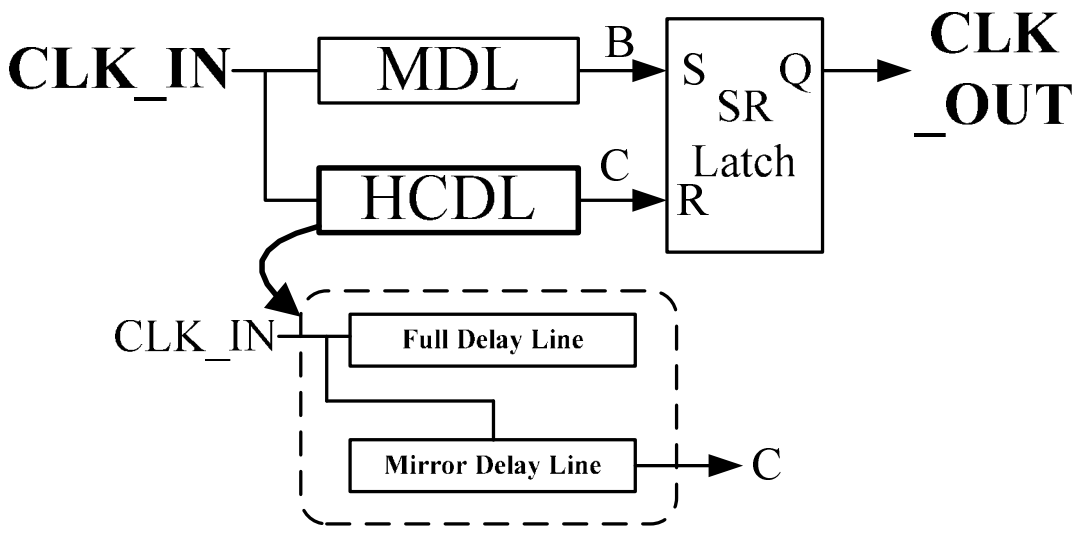


Fig. 1.4 Block Diagram of HCDL [12]

Fig. 1.4 shows the block diagram of the half-cycle delay line (HCDL) based DCC [12]. The HCDL-based DCC is consisting of HCDL and match delay line (MDL). In the HCDL which is consisted of a full delay line and a mirror delay line. The full delay line is used to detect the CLK_IN's period information, and the mirror delay line is used to generate half cycle delay time according to the period information. Therefore, the 50% duty-cycle clock is generated by SR Latch [6] [7] [10-13]. However, the two delay lines architecture has the mismatch problem especially in the nano-meter CMOS process with on-chip variations (OCVs).

1.2.3 The Time-to-Digital Converter Based DCC

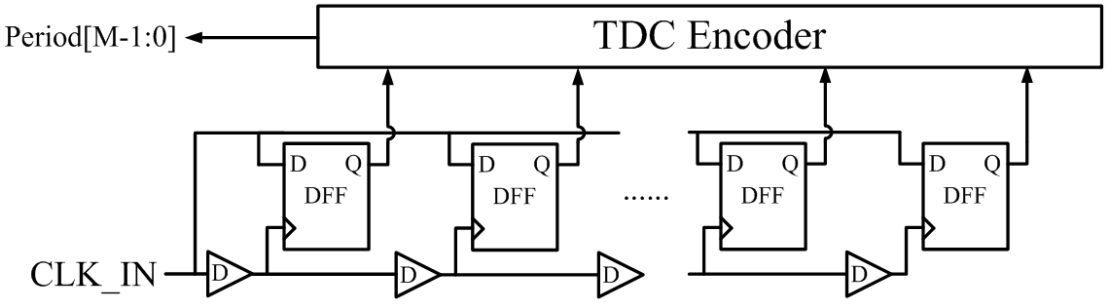


Fig. 1.5 Block Diagram of TDC [35]

Fig. 1.5 shows the block diagram of TDC [35]. The TDC-based DCC [10] [11] [13- 14] [16] [20] [21] quantizes the CLK_IN period information into M-bit digital code (Period[M-1:0]), and the TDC's resolution is restricted by D buffer delay time and the dead-zone of DFFs. Then the 50% duty-cycle clock is generated by the half quantified digital code Period[M-1:1] which right shifted one bit. Nevertheless, the TDC frequency range is limited by the delay line length, and the output 50% duty-cycle error is restricted by the resolution of TDC. In recent years, the TDC resolution is improved [22-24], but the high resolution TDC is not suitable for a wide frequency range. If the high resolution TDC is designed for wide-range, the delay line length, power consumption and chip area are also increasing.

1.3 DCC for DDR SDRAM Interface

The data transmit I/O speed goes higher in recent years, such as quadrature data rate (QDR) I/O [25] and DDR SDRAM [26-31]. The transmitter and receiver operation are restricted by clock phase-error and duty-cycle, therefore, the clock phase-error can be de-skewed by the DLL, and duty-cycle error can be corrected by the DCC. In this thesis, the proposed ADDCC is combined with the DCC and the DLL. The frequency range covers all of the DDR2/3 I/O bus clock rate specification. Table 1.1 shows the DDR2/3 I/O bus clock rate specification defined by the Joint Electron Devices Engineering Council (JEDEC) which is from 200MHz to 1066MHz.

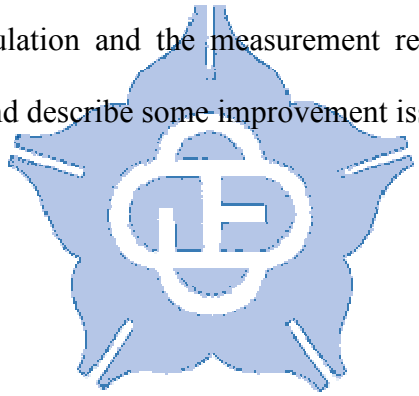
Table 1.1 The Standard of DDR2/3 I/O Bus

Standard name	Memory clock rate(MHz)	I/O Bus clock rate(MHz)
DDR2-400B/C	100	200
DDR2-533B/C	133	266
DDR2-667C/D	166	333
DDR2-800C/D/E	200	400
DDR2-1066E/F	266	533
DDR3-800D/E	100	400
DDR3-1066E/F/G	133	533
DDR3-1333F/G/H/J	166	667
DDR3-1600G/H/J/K	200	800
DDR3-1866J/K/L/M	233	933
DDR3-2133K/L/M/N	266	1066

1.4 Thesis Organization

In this thesis, we design a wide-range all-digital duty-cycle corrector Ver.1 and Ver.2 with output clock phase alignment in 65nm technology.

In chapter 2, the critical modules of the proposed ADDCC architecture are discussed. We will discuss the dead-zone of the phase detector (PD) and duty-cycle detector (DCD), and the resolution of the digital-controlled delay line (DCDL) and digital-controlled duty-cycle correction (DDCC). In chapter 3, the proposed ADDCC Ver.1 and Ver.2 are presented. The architecture and the controller design are discussed. In chapter 4, we will show the ADDCC Ver.1 and the ADDCC Ver.2 experimental results including the simulation and the measurement results. Finally, we make a conclusion of this thesis and describe some improvement issues in the further work.



Chapter 2 Critical Modules in ADDCC Architecture

2.1 Design of Digitally Controlled Delay Line

The Digital-Controlled Delay Line (DCDL) is the critical component in the proposed All-Digital Duty-Cycle Corrector (ADDCC). Because of the proposed ADDCC is combined with a Delay-Locked Loop (DLL). The DCDL is the most critical component in the DLL. For this reason, we should design a wide frequency range and high resolution DCDL in the proposed ADDCC. Thus the proposed ADDCC can improve the phase alignment accuracy.

2.1.1 MUX-Type DCDL Structure

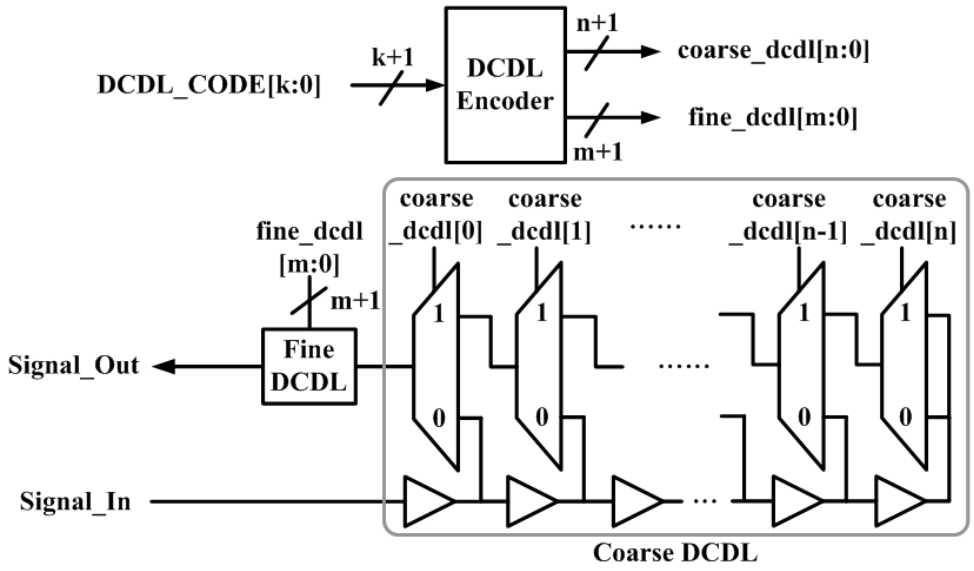


Fig. 2.1 MUX-Type Coarse-tuning Component of the proposed DCDL

Fig. 2.1 shows the MUX-Type DCDL architecture, which combined with the

coarse-tuning component Coarse DCDL and the fine-tuning component Fine DCDL. The circuit operating path is from Signal_In to Signal_Out which is selected by control code coarse_dcdl[X] (X is integer, from 0 to n). The Coarse DCDL has n+1 coarse-tuning delay cells, and for each one coarse-tuning delay cell is combined with a buffer and a multiplexer. Therefore, the Coarse DCDL can provide n+1 kinds of delay time and easily cover a wide frequency range. Nevertheless, the coarse-tuning component resolution is not good enough for our proposed ADDCC. For this reason, we combined the fine-tuning component to achieve a high resolution DCDL.

Furthermore, the DCDL is controlled by many digitally enable signals. In order to simplify DCDL's controller design, we combined a DCDL Encoder to encode DCDL_CODE[k:0] into coarse_dcdl[n+1] and fine_dcdl[m+1].

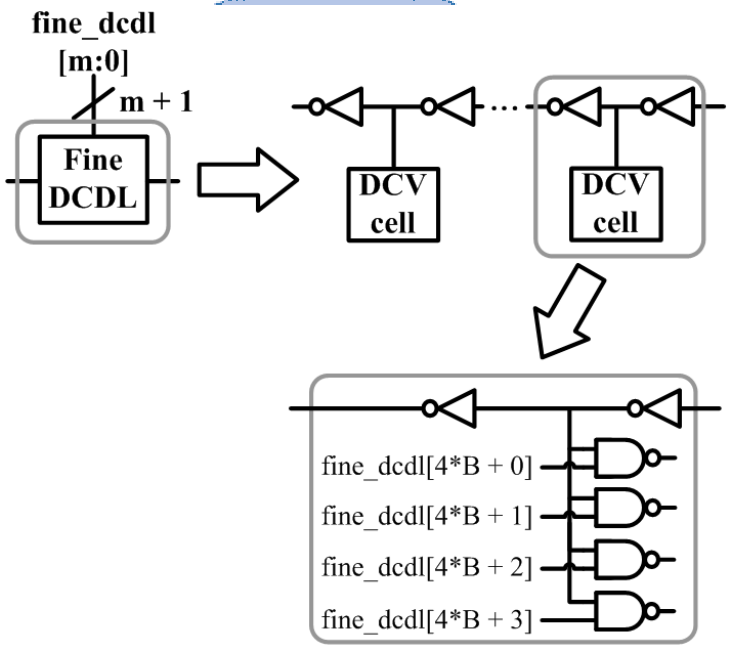


Fig. 2.2 Fine-tuning component of the proposed DCDL

Fig. 2.2 shows the fine-tuning component Fine DCDL which is based on digital-controlled varactors (DCVs) [32] architecture. For each one DCV cell is combined with four NAND gates which is controlled by control code fine_dcdl[Y] (Y

is integer, from 0 to m). Therefore, the Fine DCDL can provide m+1 kinds of delay time by each NAND delay cells. When control bit is enabled, the capacitance at inverter's output node is changed, and the delay time is increased. Therefore the DCV delay cell can increase the resolution of the DCDL.

2.1.2 Simulation Result

The proposed ADDCC operation frequency range is from 200MHz (period: 5000ps) to 1066MHz (period: 938ps) and the acceptable input duty-cycle range from 20% to 80%. Fig. 2.3 shows the scenario which is 20% duty-cycle input clock at 200MHz. Thus the proposed DCDL needs to provide the delay time from 4000ps (=5000ps*80%) to 750.4ps (=938ps*80%) at least.

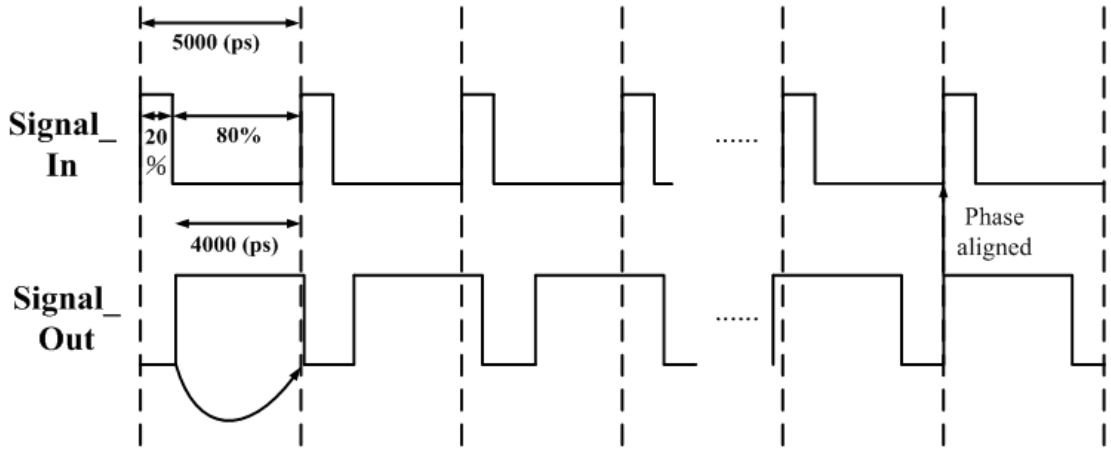


Fig. 2.3 DCDL Timing Diagram at Lowest Frequency Scenario

The MUX-Type DCDL is simulated by HSPICE. At each corner, the simulation parameters are process, voltage and temperature (PVT), respectively. The TT corner is 1.0V, 25°C. The FF corner is 1.1V, 0°C, and the SS corner is 0.9V, 125°C. Table 2.1 shows the Coarse DCDL delay range and the one coarse-tuning step delay time. Table 2.2 shows the Fine DCDL delay range and the one fine-tuning delay step.

Table 2.1 Properties of the DCDL Coarse-tuning Component

Coarse-Tuning Component Control Code : 0 to 116			
Mode	Max Delay (ps)	Min Delay (ps)	Step (ps)
TT Corner	6334	448	51
FF Corner	4564	322	37
SS Corner	9741	717	78

Table 2.2 Properties of the DCDL Fine-tuning Component

Fine-Tuning Component Control Code : 0 to 31		
Mode	Range (ps)	Step (ps)
TT Corner	88	2.75
FF Corner	42	1.31
SS Corner	95	2.96

The DCDL is designed in a cascaded architecture; therefore, the Fine DCDL delay range needs to overlap one Coarse DCDL band delay, and to make sure that there is no dead-zone delay in the DCDL. Nevertheless, overlapping method is causing the other non-monotonic problem. Hence we compensate a fixed code when the Coarse DCDL crosses to a different delay band, then the Fine DCDL's control code will add a fixed compensation code. The fixed compensation code can reduced the delay difference when the Coarse DCDL cross-band. The fixed compensation code is determined by simulation result with PVT variations.

We can get the compensation code in each PVT variation. In the TT corner, the compensation code is 14 ((88-51)/2.75≒14). In the FF corner, the compensation code is 4 ((42-37)/1.31≒4). In the SS corner, the compensation code is 6 ((95-78)/2.96≒6). Nevertheless, in the FF corner, if the compensation code 14 is determined by the TT corner that will cause dead-zone delay. For this reason, the compensation code should be determined by the minimum code from each PVT variations. As the result, the compensation code is 4 determined by FF corner. Fig. 2.4 shows the compensation result in FF corner.

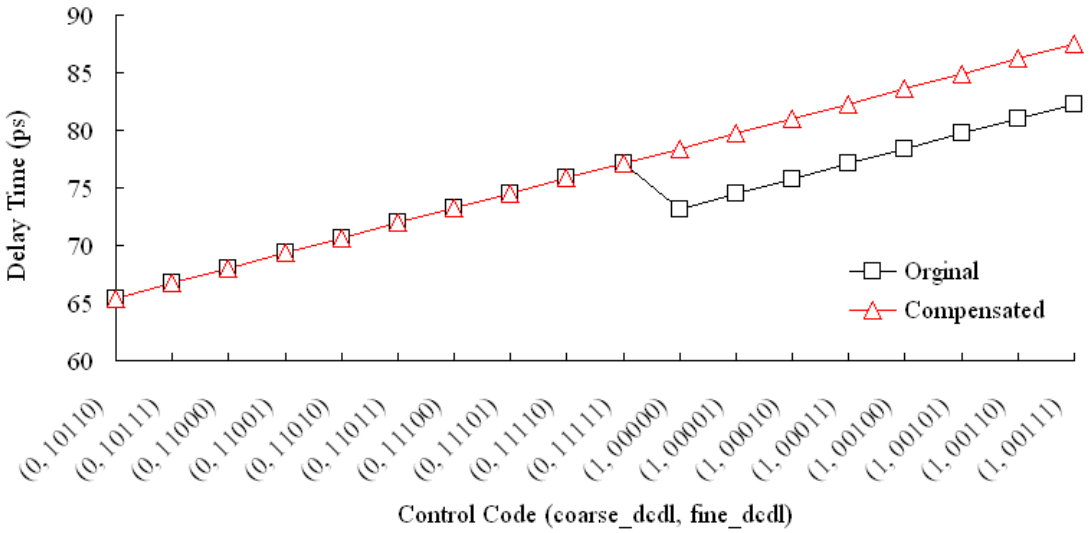


Fig. 2.4 Comparison between original and compensated in DCDL

2.2 Design of Digitally Controlled Duty-Cycle Correction Delay Line

The Digital-controlled Duty-Cycle Corrector (DDCC) Delay Line is another critical component in the proposed ADDCC. The DDCC is one of the DCC's components; hence, the DDCC is signal pulse-width adjusting which also named duty-cycle adjusting. Therefore, we should design a wide range and a high resolution DDCC delay line in the proposed ADDCC.

2.2.1 AND-OR-Type DDCC Structure

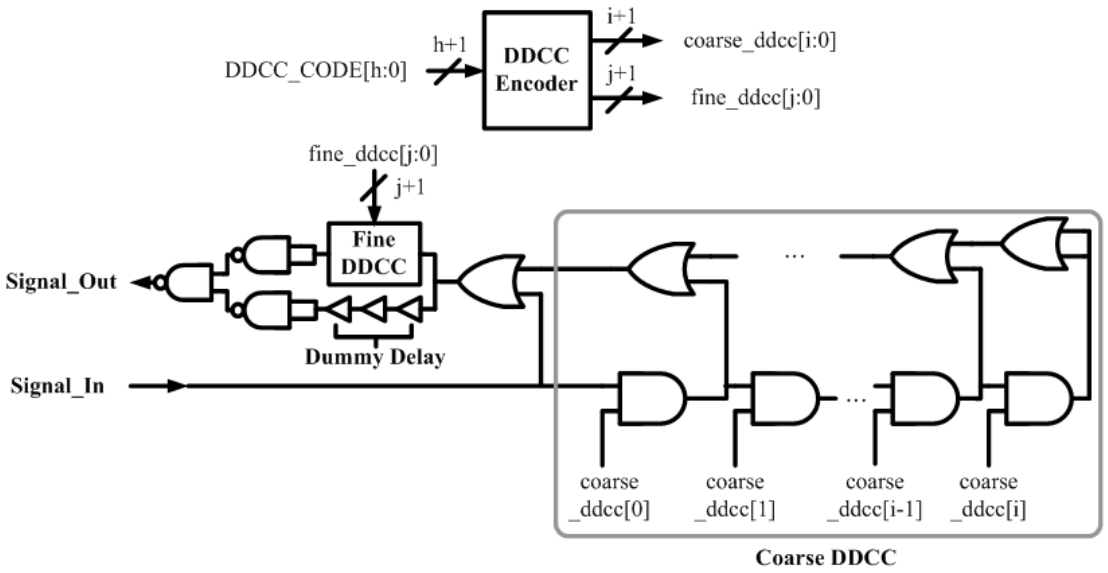


Fig. 2.5 AND-OR-Type Coarse-tuning component of the proposed DDCC

Fig. 2.5 shows the AND-OR-Type DDCC architecture, which is combined with the coarse-tuning component Coarse DDCC and the fine-tuning component Fine DDCC. The circuit operating path is from Signal_In to Signal_Out which is selected by enable code coarse_ddcc[A] (A is integer, from 0 to i). The Coarse DDCC has i+1 coarse-tuning delay cells, and for each one coarse-tuning delay cell is combined with an AND cell and an OR cell. The AND cell also can save power consumption of the unused delay cells. Therefore, the Coarse DDCC can provide i+1 kinds of pulse-width adjusting time and easily cover the wide pulse-width range. Nevertheless, the Coarse DDCC resolution is not good enough. For this reason, we combines the fine-tuning component to increase the DDCC resolution.

Furthermore, the DDCC is also controlled by many enable control signals, therefore, we also designed a DDCC Encoder to encode DDCC_CODE[h:0] into coarse_ddcc[i:0] and fine_ddcc[j:0].

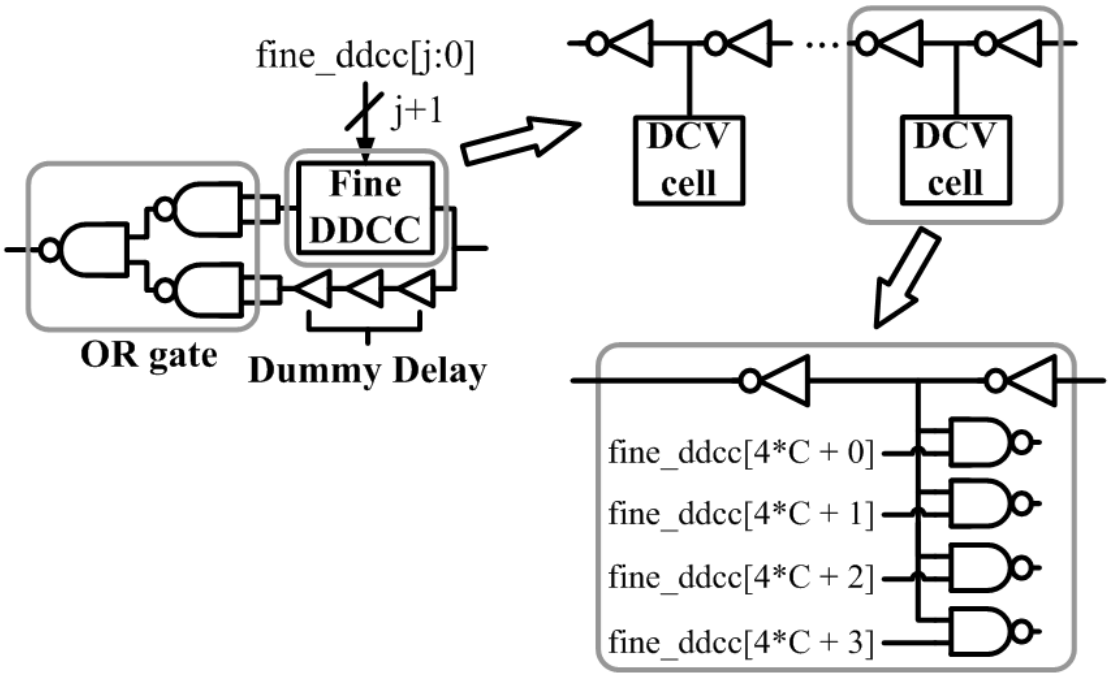


Fig. 2.6 Fine-tuning component of the proposed DDCC

Fig. 2.6 shows the Fine DDCC fine-tuning component which is based on digital-controlled varactors (DCVs) [32] architecture same as the Fine DCDL. Each one DCV cell is combined with four NAND gates which are controlled by enable code fine_ddcc[B] (B is integer, from 0 to j). The fine-tuning component has j+1 kind of delay cells. When the fine_ddcc enabled, it is changed the capacitance of inverter's output node, and also increased the delay time. Furthermore, an OR gate is connected after of the DCV output and Dummy Delay output. The Dummy Delay is used to reduce the DCV's intrinsic delay.

The proposed ADDCC is designed with standard performance 65nm cell library. In standard cell library, it is supports many kinds of standard cells for clock signal which has well balanced rising-time and falling-time. The well balanced rising-time and falling-time are good for the pulse-width adjusting, and therefore, we will choose these kinds of standard cells. Nevertheless, some common logic gates are not supported for clock signal. For this reason, we need to use the available standard cells for clock signal to create other logic cells. The OR gate in our design is an example,

we integrate three clock's NAND gates into an OR gate.

2.2.2 Simulation Results

The proposed ADDCC operation frequency range is from 200MHz (period is 5000ps) to 1066MHz (period is 938ps) and the acceptable input duty-cycle range from 20% to 80%. Fig. 2.7 shows the scenario which input duty-cycle 20% at lowest frequency 200MHz. Thus the DDCC at least to provide pulse-width adjusting time from 1500ps (=5000ps*30%) to 281.4ps (=938ps*30%).

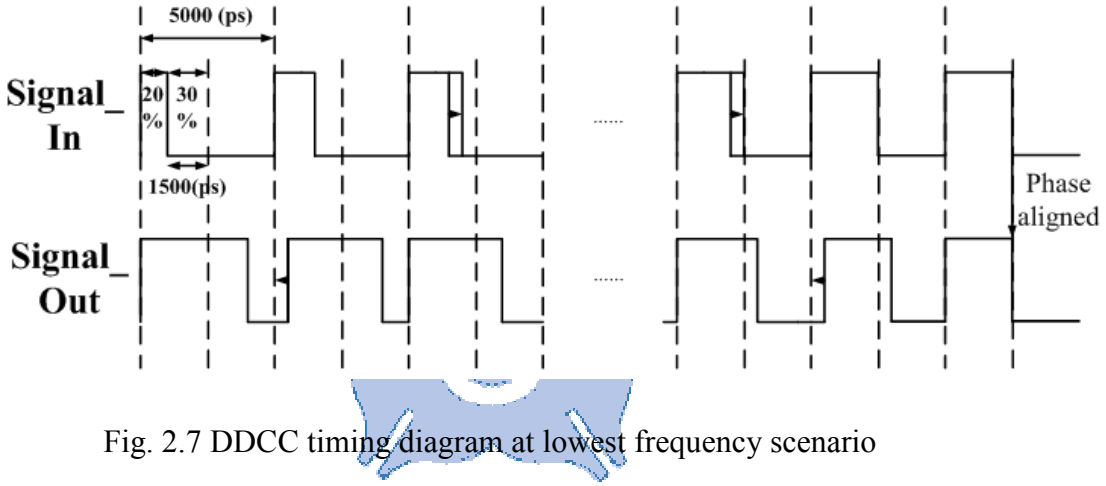


Fig. 2.7 DDCC timing diagram at lowest frequency scenario

The AND-OR-Type DDCC is simulated by HSPICE. At each corner, the simulation parameters are process, voltage and temperature (PVT), respectively. The TT corner is 1.0V, 25°C. The FF corner is 1.1V, 0°C, and the SS corner is 0.9V, 125°C. Table 2.3 shows the DDCC pulse-width adjusting range and the one coarse-tuning pulse-width adjusting time step in each corner. Table 2.4 shows the Fine DDCC cover range and the one fine-tuning pulse-width adjusting time step.

Table 2.3 Properties of the DDCC Coarse-tuning Component

Coarse-Tuning Component Control Code : 0 to 47			
Mode	Max Pulse-Width (ps)	Min Pulse-Width (ps)	Step (ps)
TT Corner	2272	16	47
FF Corner	1541	5	32
SS Corner	3624	24	75

Table 2.4 Properties of the DDCC Fine-tuning Component

Fine-Tuning Component Control Code : 0 to 31		
Mode	Pulse-Width Adjusting Range (ps)	Step (ps)
TT Corner	56	1.75
FF Corner	39	1.22
SS Corner	78	2.44

The DDCC is designed in a cascaded architecture; therefore, Fine DDCC delay range needs to overlap one Coarse DDCC band delay, and to make sure that there is no dead-zone delay in the DDCC delay line. Nevertheless, overlapping method is causing the other non-monotonic problem. Hence we compensated a fixed code when Coarse DDCC controlled code crosses to different band, then the Fine DDCC controlled code will add a fixed compensation code that can reduce the delay difference when Coarse DDCC cross-band. The fixed compensation code is determined by simulation result in each PVT variations.

We can get the compensation code in each PVT variations. In TT corner, the compensation code is 5 ((56-47)/1.75≒5). In FF corner, the compensation code is 6 ((39-32)/1.22≒6). In the SS corner, the compensation code is 1 ((78-75)/2.44≒1). Nevertheless, in SS corner, if the compensation code is 5 by TT corner that will cause dead-zone pulse-width adjusting. Therefore, the compensation code should be determined by the minimum code from each PVT variations. As the result, the compensation code is 1 determined by SS corner. Fig. 2.8 shows the compensation result in SS corner.

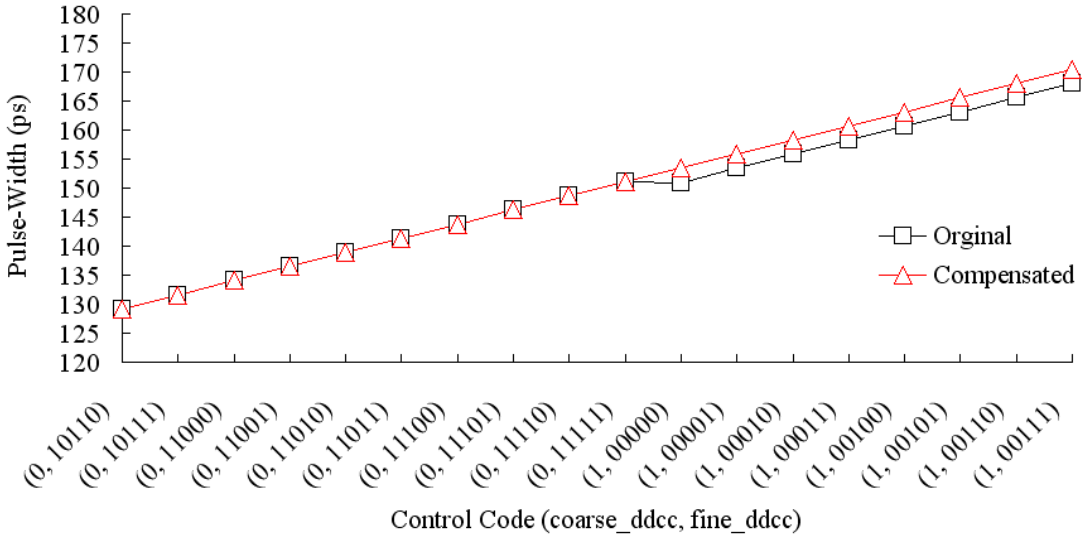


Fig. 2.8 Comparison between original and compensated in DDCC

2.3 Design of Phase Detector and Duty Cycle Detector

2.3.1 Designed Structure

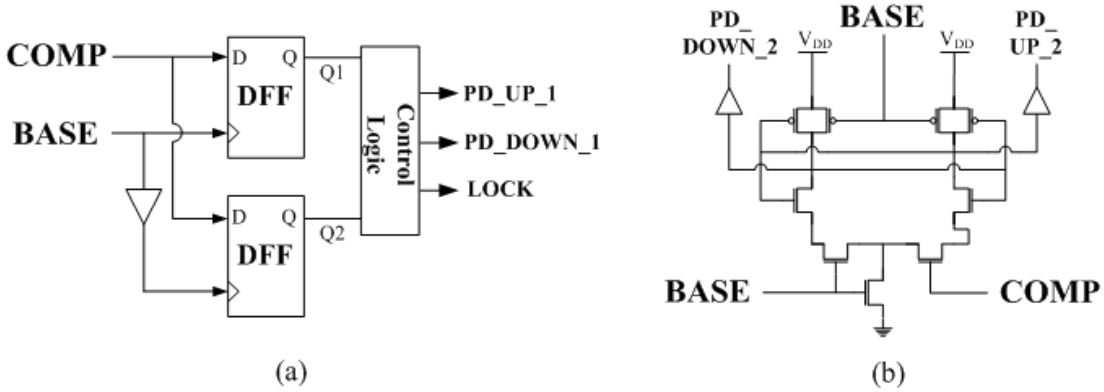


Fig. 2.9 The Proposed Phase Detector (PD)

(a) Sampled-based PD, (b) Sense-amplifier-based PD [33]

The proposed Phase Detector (PD) detects the positive phase error between COMP and BASE. The proposed PD is composed of a sampled-based PD (SBPD) and a sense-amplifier-based PD [33], as shown in Fig. 2.9. In order to improve the

detectable phase error, a sense-amplifier-based PD which can detect a phase error larger than 1ps in 65nm TT corner simulation is applied in the PD design.

The sense-amplifier-based PD seems to be good enough to detect phase error. Nevertheless, when the phase error is too large between COMP and BASE in FF corner simulation, the sense-amplifier-based PD has incorrect detecting result. For this reason, we used the SBPD to detect large phase error in the beginning. Although the SBPD does not have tiny dead-zone due to the dead-zone of the D-Flip/Flops, but the SBPD is easily designed and can be created with standard cells. It also can prevent the sense-amplifier-based PD incorrect detecting situation which mentioned before.

First of all, the PD controller receives the SBPD's outputs (PD_UP_1 and PD_DOWN_1). After the SBPD is locked, the PD controller will switch to receive the sense-amplifier-based PD's outputs (PD_UP_2 and PD_DOWN_2). As a result, the proposed PD can correctly detect a tiny phase error between COMP and BASE.

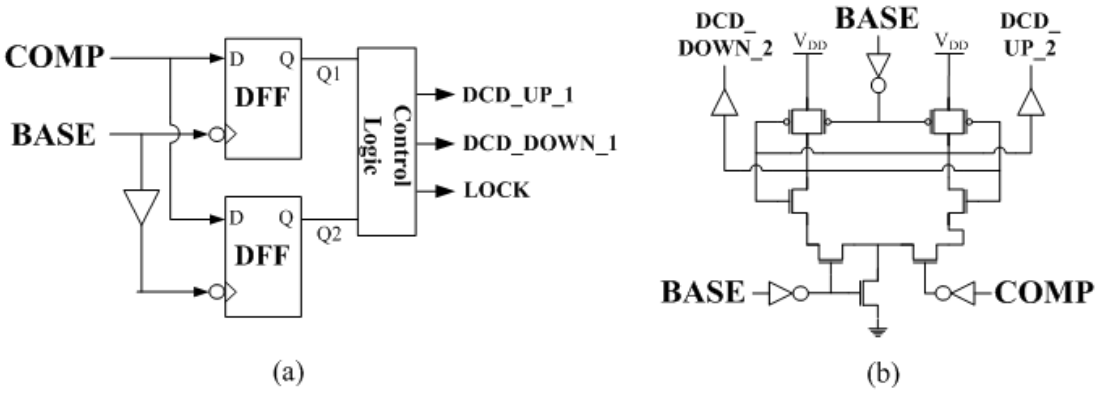


Fig. 2.10 The Proposed Duty-Cycle Detector (DCD)

(a) Sampled-based DCD, (b) Tiny dead-zone DCD

The proposed duty-cycle detector (DCD) architecture is similar to the proposed PD. It is also consisted of Sample-base DCD (SBDCD) and tiny dead-zone DCD, shows on Fig. 2.10. However, the proposed DCD detects the negative phase error between COMP and BASE. Thus in the proposed DCD, there are two inverters in front of PD's inputs (COMP and BASE). Then the PD can easily transform into the

proposed DCD and the operation behavior is same to the proposed PD.

In the beginning, the DCD controller is received the SBDCD's outputs (DCD_UP_1 and DCD_DOWN_1). After the SBDCD locked, the DCD controller switch to receive the tiny dead-zone DCD's outputs (DCD_UP_2 and DCD_DOWN_2). As a result, the proposed DCD can detect a tiny phase error between COMP and BASE.

2.3.2 Simulation Results

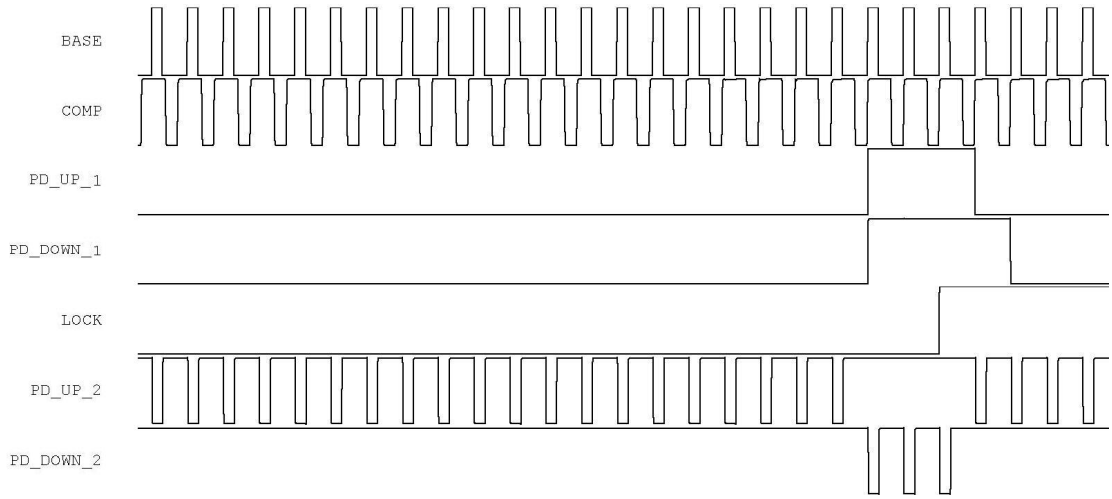


Fig. 2.11 Simulation result of the proposed PD

Fig. 2.11 shows the simulation result of the proposed PD with UltraSim SPICE mode. In the beginning, the positive phase error is detected by SBPD. Until the SBPD locked, then switch to the sense-amplifier-based PD's outputs (PD_UP_2 and PD_DOWN_2).

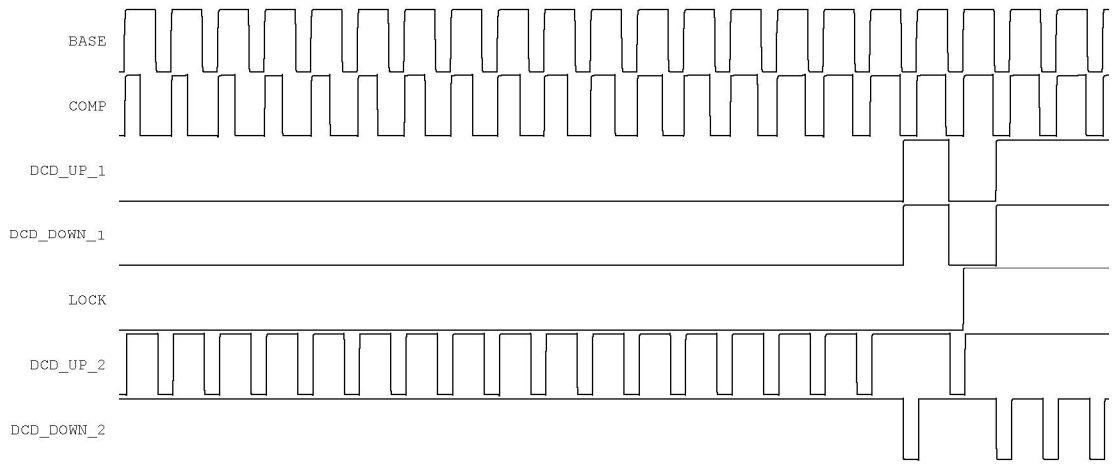


Fig. 2.12 Simulation result of the proposed DCD

Fig. 2.12 shows the simulation result of the proposed DCD with UltraSIM SPICE mode. In the beginning, the positive phase error is detected by SBDCD. Until the SBDCD locked, then switch to the tiny dead-zone DCD's outputs (DCD_UP_2 and DCD_DOWN_2).

2.4 Summary

The proposed ADDCC is consisted of the DLL and the DCC. Furthermore, the proposed PD and DCDL are critical components in the DLL. The proposed DCD and DDCC are also the critical components in the DCC. We verify each of components functional and simulate by HSPICE. Therefore, we ensure that these components can correctly operating with each PVT variation.

Chapter 3 All-Digital Duty-Cycle Corrector Design

3.1 The Proposed ADDCC Ver.1 with High Resolution Delay Line

3.1.1 Design of Controller

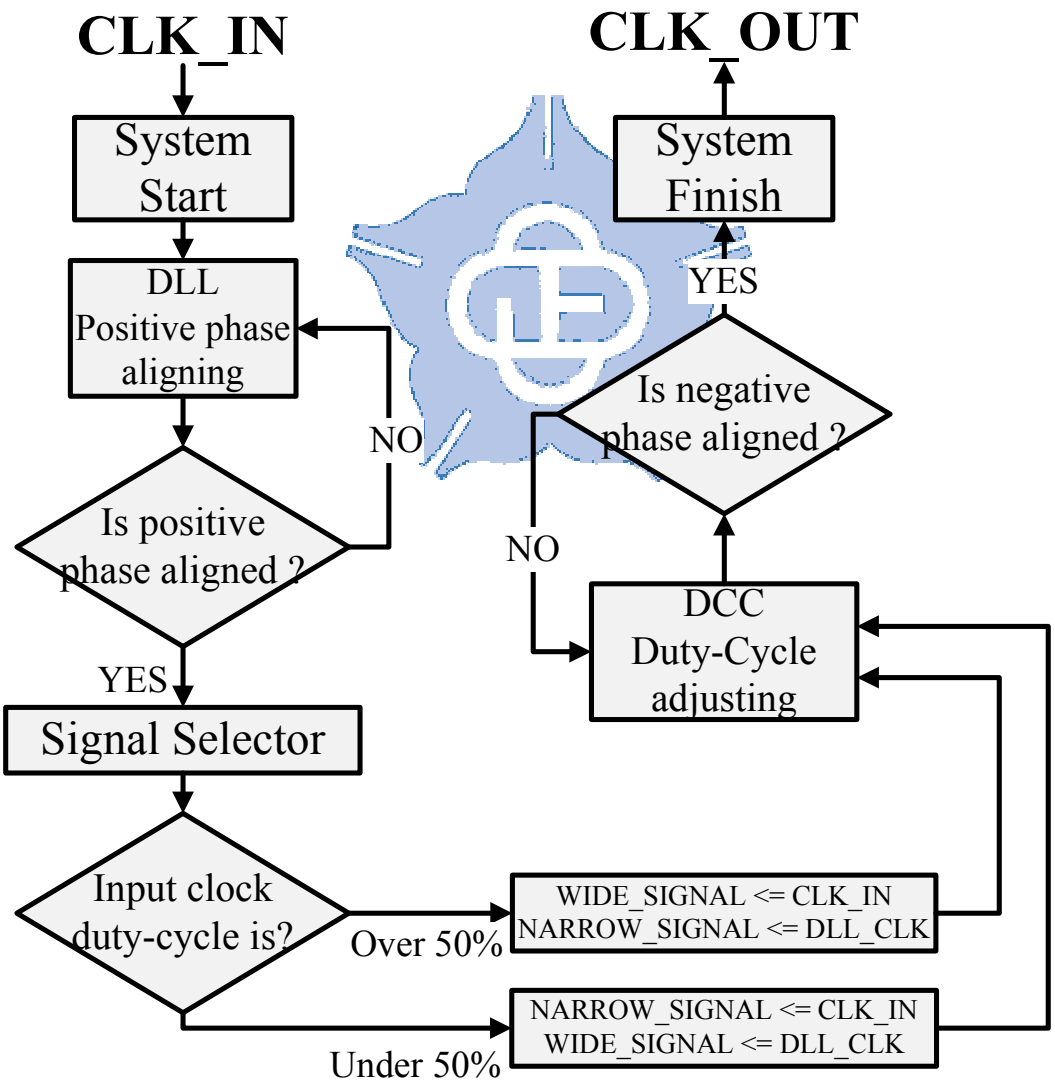


Fig. 3.1 Operation Flowchart of the ADDCC Ver.1

Fig. 3.1 shows the operation flowchart of the proposed ADDCC Ver.1. In the beginning when system starts, the DLL operates positive phase alignment. Until the positive phase aligned, two clocks (i.e. CLK_IN and DLL_CLK) with the complementary duty cycles are generated. At that time, the Signal Selector is determined that the CLK_IN's duty-cycle is under 50% or over 50%; moreover, it assigns the duty-cycle over 50% to WIDE_SIGNAL and the duty-cycle under 50% to NARROW_SIGNAL. These two signals are sent to DCC, NARROW_SIGNAL is compensated the duty-cycle until to the negative phase aligned to WIDE_SIGNAL's negative phase. Then the CLK_OUT is generated, and the system is locked.

3.1.2 Architecture

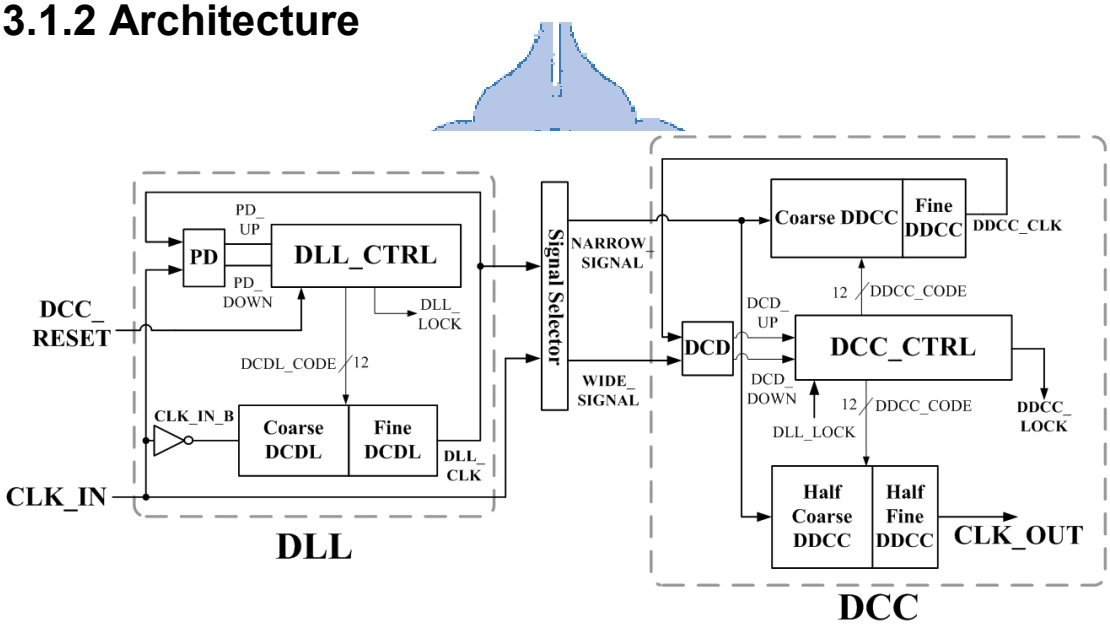


Fig. 3.2 The proposed ADDCC Ver.1 with high resolution delay line

Fig. 3.2 shows the block diagram of the proposed ADDCC Ver.1 with high resolution delay line. It is composed of an all-digital delay-locked loop (DLL), a signal selector and an all-digital duty-cycle corrector (DCC). The all-digital DLL consists of a phase detector (PD), a coarse-tuning digitally controlled delay line (Coarse DCDL), a fine-tuning digital controlled delay line (Fine DCDL) and a DLL controller (DLL_CTRL). The all-digital DCC consists of a duty-cycle detector (DCD),

a coarse-tuning Digitally-controlled Duty-Cycle Correction (Coarse DDCC), a fine-tuning DDCC delay line (Fine DDCC), a half coarse-tuning DDCC delay line (Half Coarse DDCC), a half fine-tuning DDCC delay line (Half Fine DDCC) and a DCC controller (DCC_CTRL).

The inverted input clock (CLK_IN_B) passes through the delay line of the DLL, then the DLL's output signal (DLL_CLK) is compared with the input clock (CLK_IN) to align the positive edge of DLL_CLK and CLK_IN. The PD of the DLL detects the positive phase error between CLK_IN and DLL_CLK which are the PD's BASE input and COMP input. Then outputs PD_UP/PD_DOWN control signals to the DLL_CTRL. Subsequently, the DLL_CTRL adjusts the delay line control code (DCDL_CODE) to compensate the phase error. When the positive phase error between CLK_IN and DLL_CLK is eliminated, the DLL is locked. Thus, two clocks (i.e. CLK_IN and DLL_CLK) with the complementary duty cycles are generated.

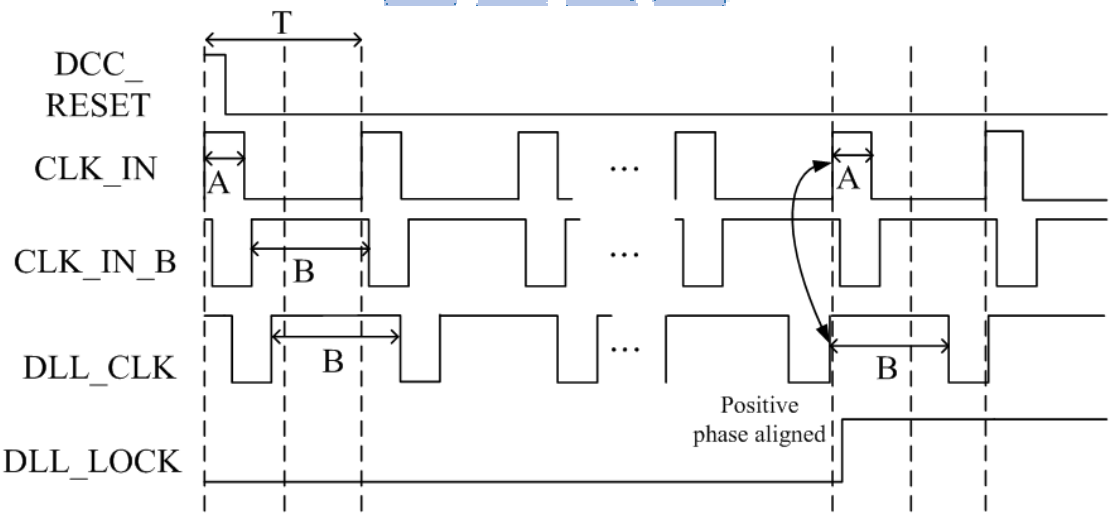


Fig. 3.3 Timing Diagram of DLL in ADDDC Ver.1

Fig. 3.3 shows the timing diagram of the DLL. The period of CLK_IN is T. If the duty-cycle of CLK_IN is A/T and the duty-cycle of DLL_CLK is B/T, the period T is equal to (A+B). After the DLL is locked, the positive edge of CLK_IN and DLL_CLK are phase aligned with complementary duty cycles. The signal selector

detects the pulse widths of these two clocks, and the clock with wider pulse width is outputted as WIDE_SIGNAL. Oppositely, the clock with shorter pulse width is outputted as NARROW_SIGNAL.

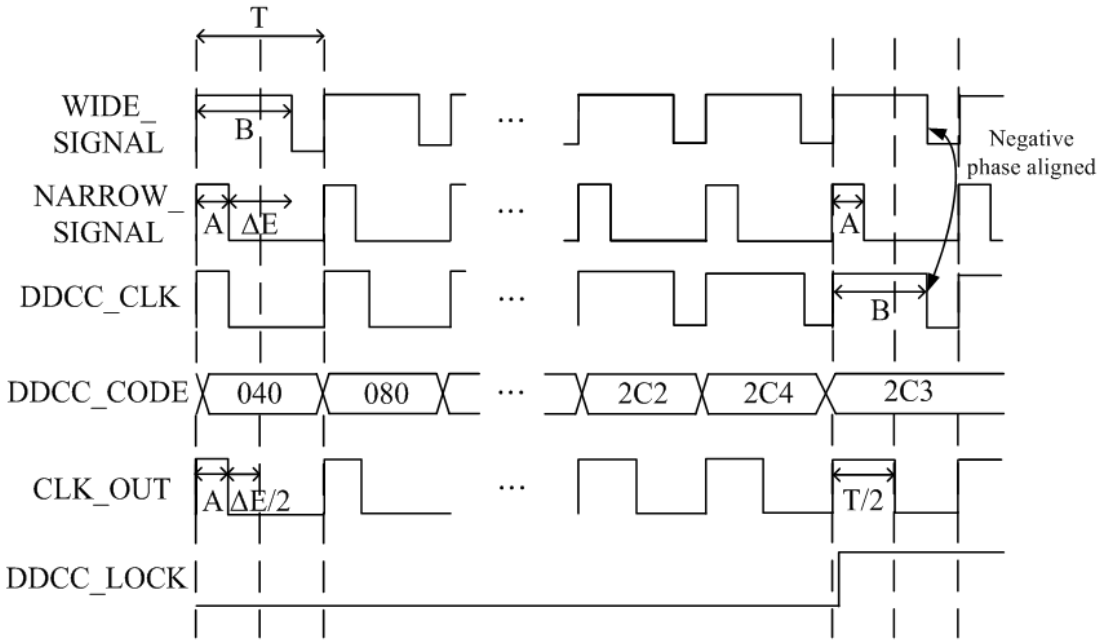


Fig. 3.4 Timing Diagram of DCC in ADDDC Ver.1

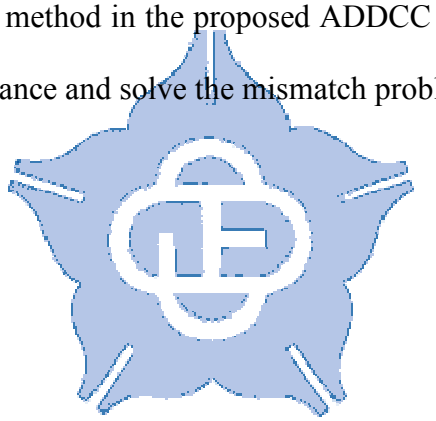
Fig. 3.4 shows the timing diagram of the DCC. After the DLL is locked, the proposed all-digital DCC starts to compensate the duty-cycle error of the output clock (CLK_OUT). The NARROW_SIGNAL passes through the Coarse DDCC and the Fine DDCC to increase the pulse-width then outputs as DDCC_CLK. The DCD detects the negative phase error between WIDE_SIGNAL and DDCC_CLK which are the DCD's BASE input and COMP input. Then outputs DCD_UP/DCD_DOWN control signals to DCC_CTRL. Subsequently, DCC_CTRL adjusts the duty-cycle correction delay line control code (DDCC_CODE) to increase the output pulse width of DDCC_CLK. When the negative phase error is eliminated between the WIDE_SIGNAL and DDCC_CLK, then DCC is locked.

The pulse-width of NARROW_SIGNAL is increased by ΔE, and ΔE is equal to (B-A). Since the period of CLK_IN is T, (A + ΔE/2) is equal to T/2 (= A+ (B-A)/2 =

(A+B)/2). Therefore, the proposed DCC utilizes Half Coarse DDCC and Half Fine DDCC to increase the pulse width of CLK_IN by ΔE/2. As a result, after the DCC is locked, the duty-cycle of CLK_OUT is 50%.

3.1.3 Summary of the Proposed ADDCC Ver.1

The proposed ADDCC Ver.1 uses the sequential search with the high resolution delay line, which can improve the accuracy of 50% duty-cycle. However, the ADDCC Ver.1 is used the HDCL, this architecture has the mismatch problem especially in the nano-meter CMOS process with on-chip variations (OCVs). Thus, we proposed the novel duty-cycle corrector method in the proposed ADDCC Ver.2 which can improve the ADDCC Ver.1 performance and solve the mismatch problem.



3.2 The Proposed ADDCC Ver.2 without Half Delay Line

3.2.1 Design of Controller

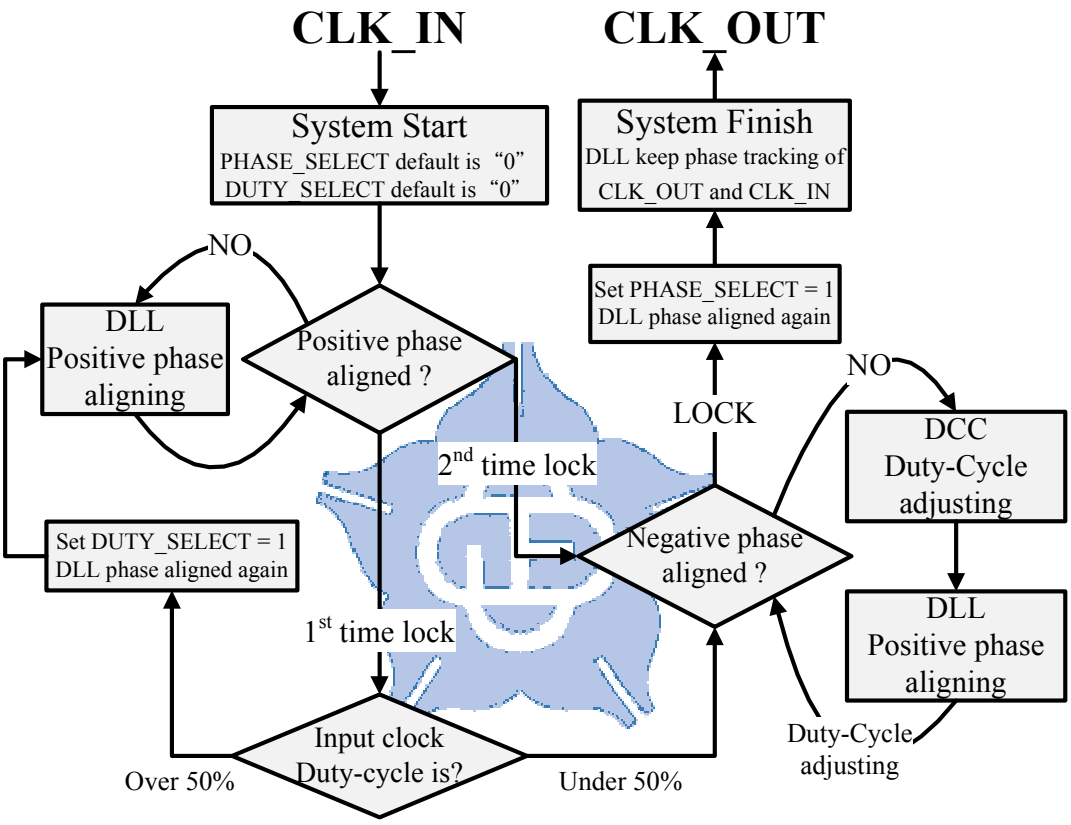


Fig. 3.5 Operation Flowchart of the ADDCC Ver.2

Fig. 3.5 shows the operation flowchart of the proposed ADDCC Ver.2. In the beginning, the DLL operates positive phase alignment. When the DLL is at first time lock, the DCC is determined whether the CLK_IN's duty-cycle is under 50% or over 50%. If the CLK_IN's duty-cycle is under 50%, then it goes to the DCC duty-cycle adjusting. Otherwise, the DCC set DUTY_SELECT to "1" and back to the DLL phase align until second time lock. When DLL phase aligned, start the DCC duty-cycle adjusting. In the first cycle, the DCC extends the pulse-width of the X signal. Then, in

the next cycle, the positive edge of the Y signal will lag behind the positive edge of the X signal due to the pulse extension in the previous cycle. Thus in the second cycle, the DLL aligns the positive edges of X and Y. The same process will be repeated until that both the positive edge and negative edge of X and Y are phase aligned, and then the DCC is locked. After the DCC lock, the DLL set PHASE_SELECT to “1” and keep tracking phase aligned between the output clock (CLK_OUT) and the input clock (CLK_IN).

3.2.2 Architecture

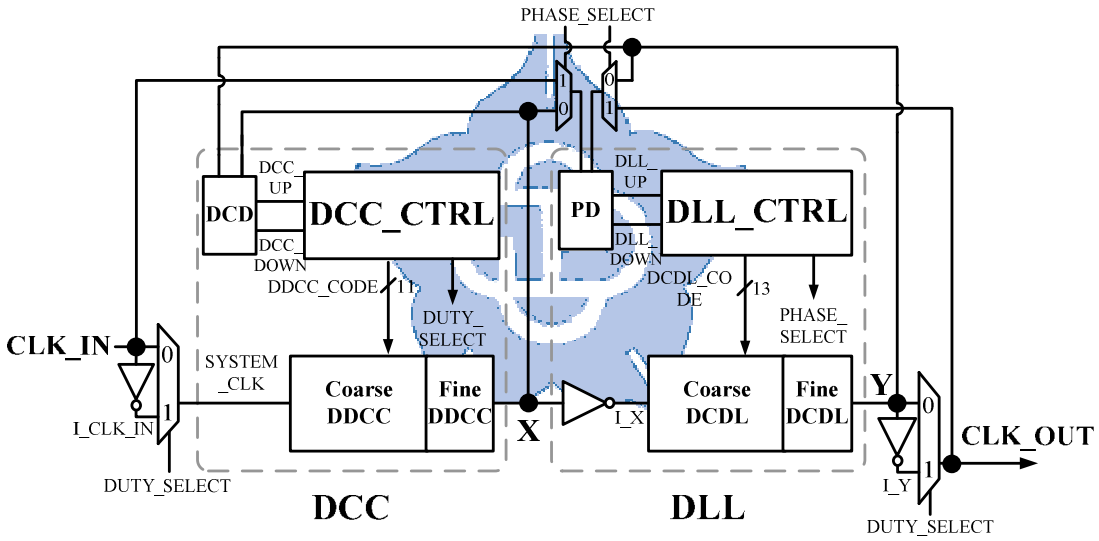


Fig. 3.6 The Proposed ADDCC Ver.2 without Half Delay Line

Fig. 3.6 shows the block diagram of the proposed ADDCC Ver.2 without half delay line. It is composed of an all-digital duty-cycle corrector (DCC) and an all-digital delay-locked loop (DLL). The all-digital DCC consists of a duty-cycle detector (DCD), a coarse-tuning digital controlled duty-cycle correction delay line (Coarse DDCC), a fine-tuning digital controlled duty-cycle correction delay line (Fine DDCC), and a DCC controller (DCC_CTRL). The all-digital DLL consists of a phase detector (PD), a coarse-tuning digitally controlled delay line (Coarse DCDL), a

fine-tuning digital controlled delay line (Fine DCDL), and a DLL controller (DLL_CTRL).

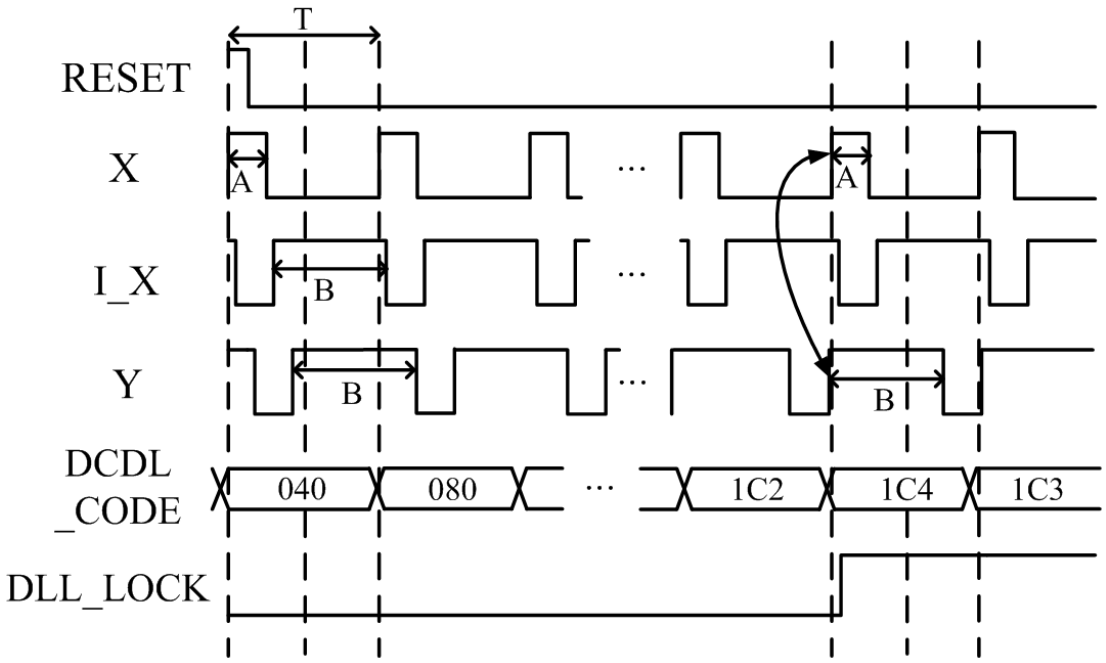


Fig. 3.7 Timing Diagram of DLL in ADDDC Ver.2

Fig. 3.7 shows the timing diagram for the DLL operation. After system is reset, the DUTY_SELECT signal is set to "0", and the PHASE_SELECT signal is also set to "0". The input clock (CLK_IN) is passed through the DCC's delay line and outputted as X signal. Subsequently, the inverted X signal is then passed through the DLL's delay line and outputted as Y signal. The phase detector (PD) of the DLL compares the phase error between the positive edges of X and Y, and then it outputs DLL_UP/DLL_DOWN control signals to the DLL_CTRL. The DLL_CTRL adjusts the delay line control code (DCDL_CODE) to compensate for the phase error. When the phase error between X and Y is eliminated, the DLL is locked. After that, two clocks (i.e. X and Y) with complementary duty cycles are generated. Thus, if the period of the input clock (CLK_IN) is T, and the duty-cycle of X and Y is A/T and B/T, respectively, the period T is equal to (A+B).

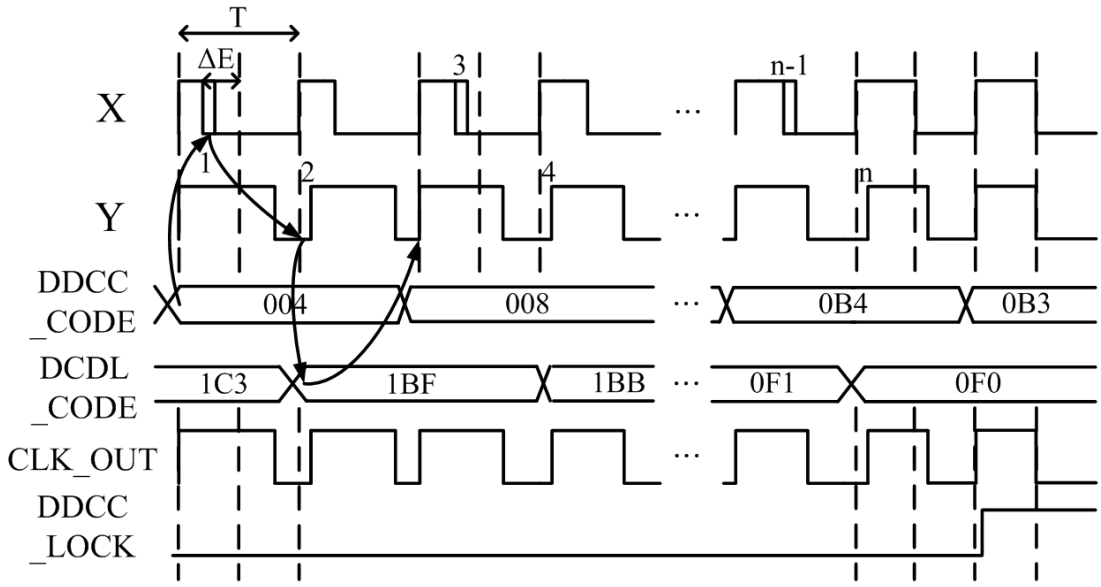


Fig. 3.8 Timing Diagram of DCC in ADDDC Ver.2

After the DLL is locked, the proposed all-digital DCC starts to compensate for the duty-cycle error of the output clock (CLK_OUT). The duty-cycle detector (DCD) detects the phase error between the negative edges of X and Y, and then it outputs DCC_UP/DCC_DOWN control signals to the DCC_CTRL. The DCC_CTRL adjusts the duty-cycle correction delay line control code (DDCC_CODE) to enlarge the pulse width of the X signal according to the outputs of the DCD. Fig. 3.8 shows the timing diagram for the DCC operation.

In the first cycle, the DCC extends the pulse width of the X signal. Then, in the next cycle, the positive edge of the Y signal will lag behind the positive edge of the X signal due to the pulse extension in the previous cycle. Thus in the second cycle, the DCDL_CODE is decreased to align the positive edges of X and Y. The same process will be repeated until that both the positive edge and negative edge of X and Y are phase aligned, and then the DCC is locked.

The pulse width of the X signal is increased by ΔE, and ΔE is equal to (B-A)/2. Since the period of input clock (CLK_IN) is T, (A+ΔE) is equal to T/2 (=A+(B-A)/2= (A+B)/2). As a result, after the DCC is locked, the duty-cycle of the CLK_OUT is

50%. Once the DCC is locked, PHASE_SELECT signal is set to “1”. The inputs of the DLL’s PD are switched to the CLK_IN and the CLK_OUT. Then, the DLL will adjust the DCDL_CODE to compensate for the phase error between the CLK_IN and the CLK_OUT. Therefore, the output clock (CLK_OUT) can be phase aligned with the input clock (CLK_IN).

Fig. 3.9 shows the DLL timing diagram when input clock duty-cycle over 50%. After the DLL is locked, if the negative edge of the X signal lags behind the negative edge of the Y signal, which means the duty-cycle of the input clock is larger than 50%. Then, the DUTY_SELECT signal is set to “1”, and therefore, the input clock is switched to the inverted CLK_IN (I_CLK_IN) to guarantee the duty-cycle of X signal is always smaller than 50%. In addition, the output clock is switched to the inverted Y (I_Y) signal, and the DLL will also eliminate the phase error between the CLK_IN and the CLK_OUT.

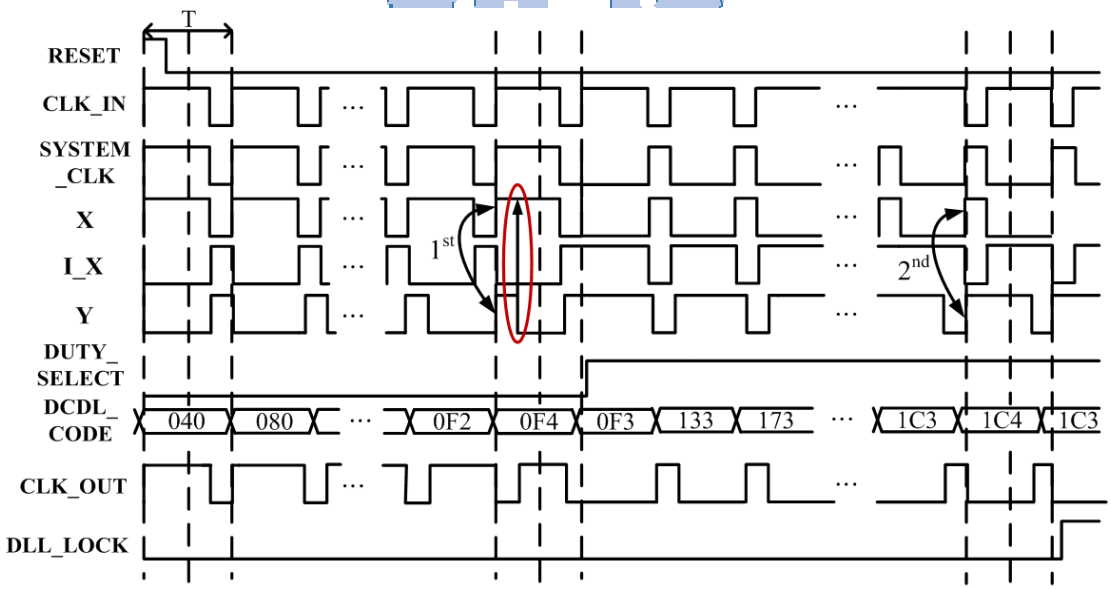


Fig. 3.9 DLL Timing Diagram when Input Clock Duty-Cycle Over 50%

Fig. 3.9 shows the DLL timing diagram when input clock duty-cycle over 50%. In this case of input clock (CLK_IN) duty-cycle is over 50%. When X and Y are positive phases aligned in first time (1st) by DLL. The DCC controller (DCC_CTRL)

detects the negative edge of Y, which is lead or lag to the negative edge of X. If the negative edge of Y is lead to X, then the DCC_CTRL will make sure that the duty-cycle of Y is less than X. Thus if the CLK_IN's duty-cycle is over 50%, the DUTY_SELECT control bit will switch to "1". Until X's and Y's positive phases are aligned in second time (2^{nd}), and DLL is locked.

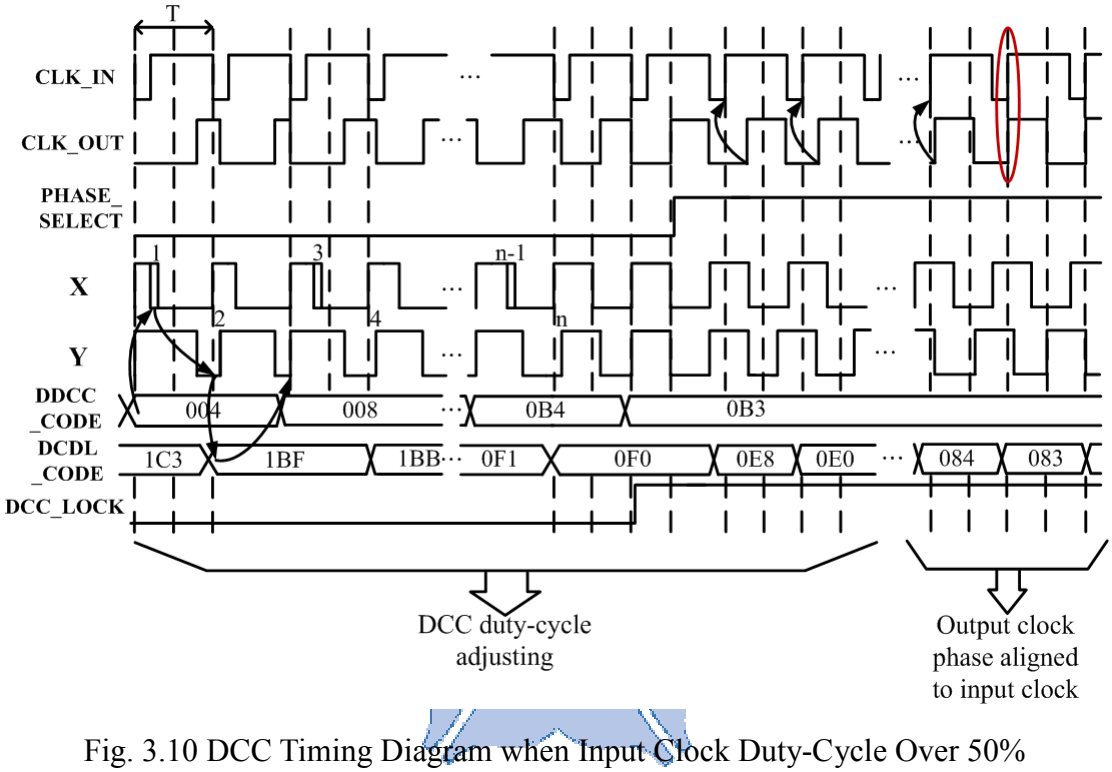


Fig. 3.10 DCC Timing Diagram when Input Clock Duty-Cycle Over 50%

Fig. 3.10 shows the DCC timing diagram when input clock duty-cycle over 50%. After the DLL locked, the DCC starts to correct duty-cycle. Before the DCC is locked, the correcting action is the same to the input clock duty-cycle under 50%. Nevertheless, the positive edge of CLK_OUT is not aligning to the positive edge of CLK_IN. Therefore, we selected the I_Y becomes CLK_OUT signal. Because the positive edge of inverted Y lags to CLK_OUT. Thus the DLL is just reduced DCDL_CODE which is also reduced the DCDL length. Until the positive edge between CLK_IN and CLK_OUT is aligned.

3.3 Comparison Summary of Ver.1 and Ver.2

In previous sections the proposed ADDCC Ver.1 and Ver.2 are introduced. In this section, we summarize the difference between them we also discuss the good and bad point between Ver.1 and Ver.2.

The proposed ADDCC Ver.1 is with the high resolution phase tracking method which can increase the accuracy of 50% duty-cycle correction. However, the duty-cycle detection is used by the full delay line and the output clock is compensated by the half delay line. The duty-cycle compensation code from the full delay line to the half delay line has shifted one bit. Thus the half delay line has reduced the correct precision and decreased the resolution two times. Moreover, if the detect circuit and the output circuit are not the same one that would cause delay mismatch problem, especially in the nano-meter CMOS process.

Therefore, the output clock duty-cycle error is comes from DCDL's resolution, PD's dead-zone, two times DDCC's resolution, DCD's dead-zone. The intrinsic delay from input clock to output clock which is comes from the Signal Selector and the Half DDCC.

In the proposed ADDCC Ver.1, the 50% duty-cycle error is comes from DLL's PD dead-zone, DCDL's resolution, DCC's DCD dead-zone, DDCC's resolution and the two times resolution of half DDCC.

The proposed ADDCC Ver.2 also uses the high resolution phase tracking method, but it improved the half delay line mismatch problem and two times DDCC's resolution problem. Therefore, the output clock duty-cycle error is comes from DCDL's resolution, PD's dead-zone, DDCC's resolution, and DCD's dead-zone. For this reason, the ADDCC Ver.2 can improve the accuracy of 50% duty-cycle correction

better than the ADDCC Ver.2. Besides, after the 50% duty-cycle correction, the proposed ADDCC Ver.2 can keep tracking the phase error between the external clock and the output clock. Thus the output clock can de-skew the phase error and reduce the circuit intrinsic delay.

In the proposed ADDCC Ver.2, the 50% duty-cycle error is comes from DLL's PD dead-zone, DCDL's resolution, DCC's DCD dead-zone and DDCC's resolution.

Table 3.1 Comparison between Ver.1 and Ver.2 by Theoretical Discussion

Item	ADDCC Ver.1	ADDCC Ver.2
Mismatch Problem	Yes	No
Output with Intrinsic Delay	No	Yes
Output Duty-Cycle Error	Ver.1 > Ver.2	
Power Consumption	Ver.1 > Ver.2	
Lock-in Time	Ver.1 < Ver.2	

Table 3.1 shows the comparison between Ver.1 and Ver.2 by theoretical discussion. In the proposed ADDCC Ver.1 and Ver.2 are used the binary search which the time complexity is $O(\log_2(n))$. Therefore, the ADDCC Ver.1 lock-in time is $K_1\log_2(n)$, and the ADDCC Ver.2 lock-in time $K_2\log_2(n)$, the K_1 and the K_2 are integer variables, n is the operation frequency and input duty-cycle error. Nevertheless, the Ver.2 duty-cycle correction algorithm has two steps positive and negative phase alignment, and the Ver.1 duty-cycle correction algorithm only has one step negative phase alignment. For this reason, the Ver.2's K_2 is larger than the Ver.1's K_1, thus the locking time of Ver.2 is longer than Ver.1.

Chapter 4 Circuit Implementation and Measurement Results

4.1 Test Circuit Implementation

The test chip is fabricated by UMC 65nm 1P10M standard performance (SP) CMOS process and the input frequency range from 250MHz to 1GHz. Because of the input and output frequency are restricted by the I/O pad max operation frequency which is approximate less than 300MHz. For this reason, the test circuit needs to provide the several kinds of high frequency clock and to generate the various duty-cycle clock inputs. Furthermore, if the output clock frequency is higher than 300MHz, it should be divided into low frequency to transmit the signal output.

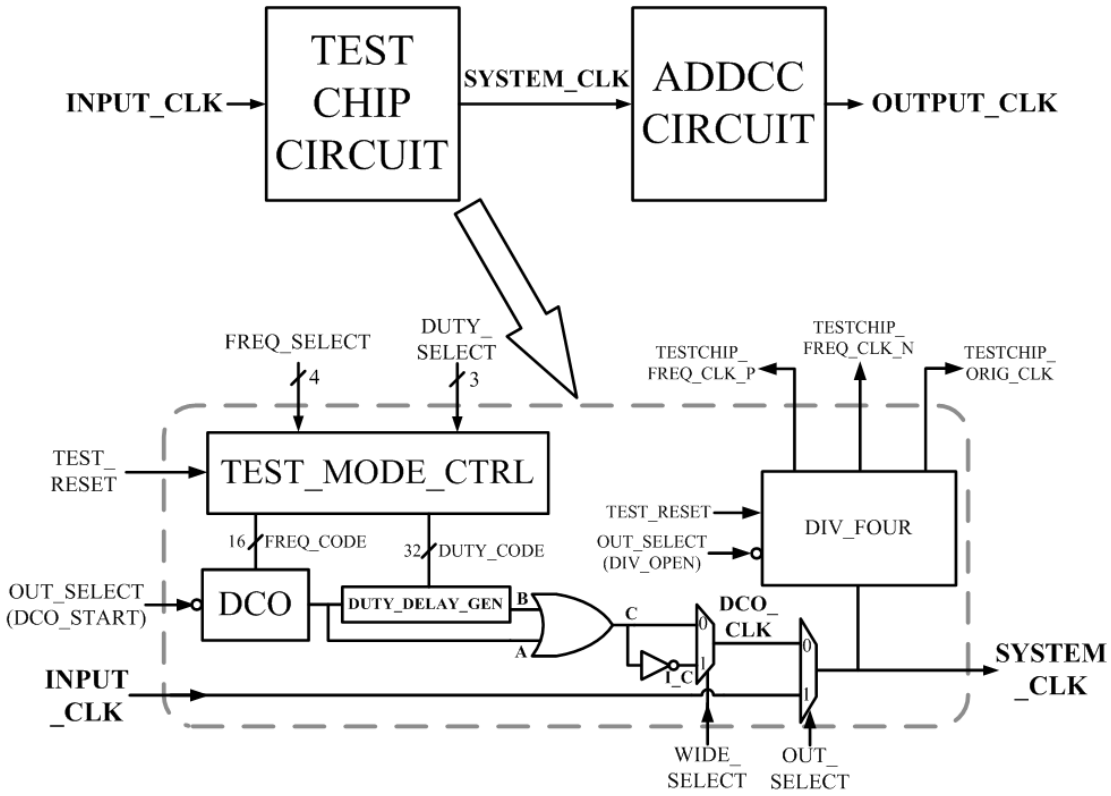


Fig. 4.1 Block Diagram of Test Chip

Fig. 4.1 shows block diagram of test chip. The test chip is combined with the test

mode control circuit (TEST_MODE_CTRL), digitally controlled oscillator (DCO), duty-cycle generator delay line (DUTY_DELAY_GEN) and a divider circuit (DIV_FOUR).

The input FREQ_SELECT is encoded into the FREQ_CODE which determined the DCO operation range. The DCO is designed in MUX-Type DCO and frequency range from 242MHz to 1094MHz by TT corner HSPICE simulation result. The input DUTY_SELECT is encoded into the DUTY_CODE which determined the delay time from A to B, then OR the two signals (A and B) into C. Thus the C's duty-cycle is over 50%, and inverted the C into the I_C which duty-cycle is under 50%. The duty-cycle range is from 15% to 85% by TT corner HSPICE simulation result. The ADDCC input clock (SYSTEM_CLK) is the external low frequency clock (INPUT_CLK) or the internal high frequency clock (DCO_CLK) which determined by the OUT_SELECT.

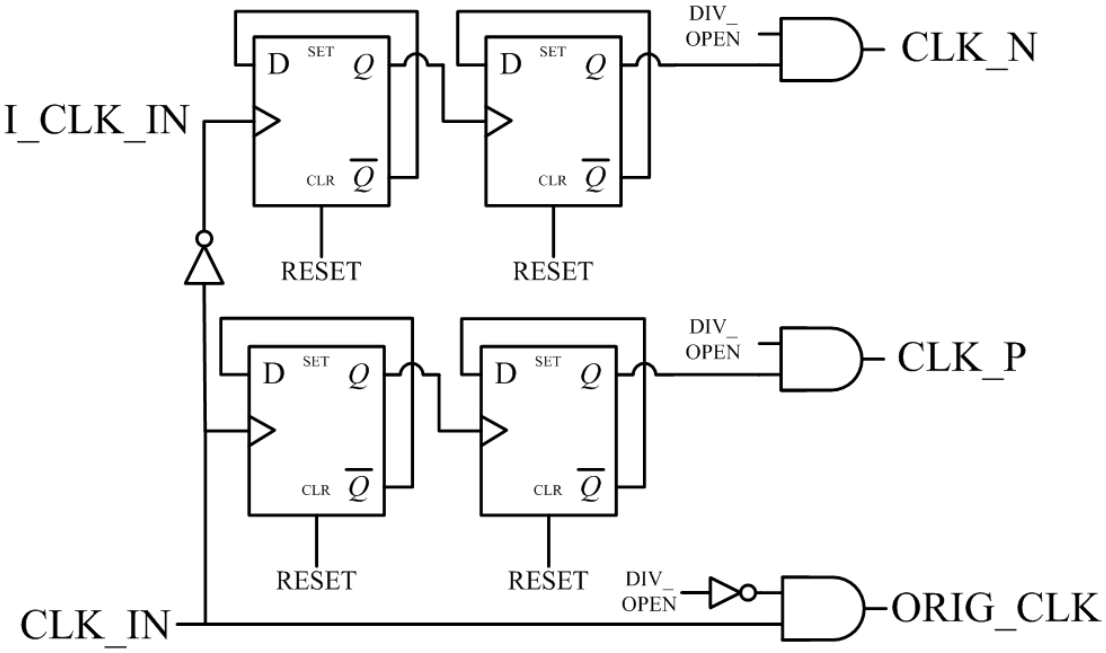


Fig. 4.2 Block Diagram of DIV_FOUR Circuit

The output clock frequency is restricted by the I/O pad speed so the internal DCO high frequency clock is divided by four into low frequency clock. Fig 4.2 shows

the block diagram of DIV_FOUR circuit which is designed with two divide-four circuit. The two dividers have two kinds of trigger signals CLK_IN (SYSTEM_CLK) and I_CLK_IN. CLK_P is triggered by CLK_IN's positive edge, CLK_N is triggered by CLK_IN's negative edge.

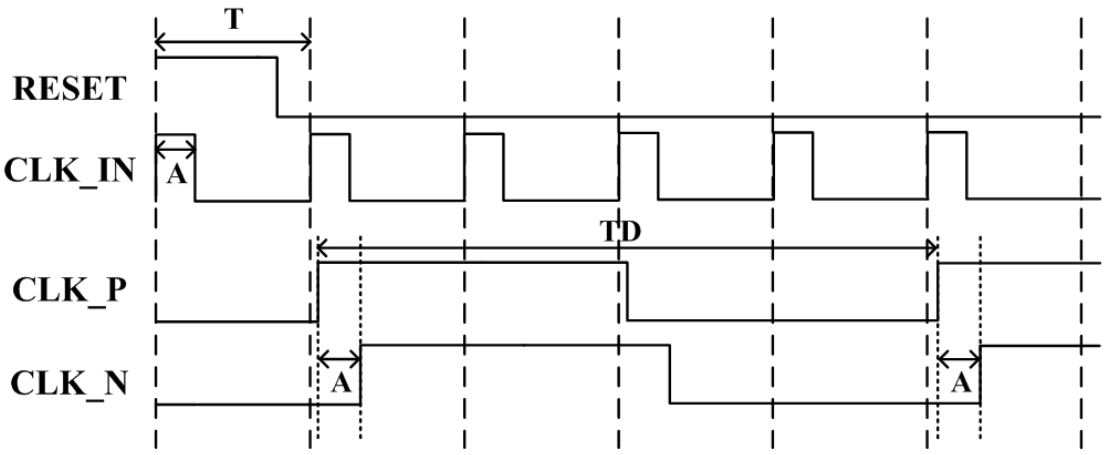


Fig. 4.3 Timing Diagram of DIV_FOUR Circuit

Fig. 4.3 shows the timing diagram of the DIV_FOUR circuit. The CLK_IN's period is T, the CLK_IN's duty-cycle is A/T and the CLK_P's (CLK_N's) period is TD (=4*T). The time difference between CLK_P's positive edge and CLK_N's positive edge is A which also is the CLK_IN's pulse-width. Then the CLK_IN's duty-cycle is A/T (=A/(TD/4)). Besides, the enable-bit DIV_OPEN (OUT_SELECT) which is controlled the output gating for saving power. If the DIV_OPEN set to "0", the CLK_P and the CLK_N are gated then the ORIG_CLK is enabled output. If the DIV_OPEN set to "1", the ORIG_CLK is gated then the CLK_P and CLK_N are enabled output.

4.2 The Proposed ADDCC Ver.1

4.2.1 Specifications

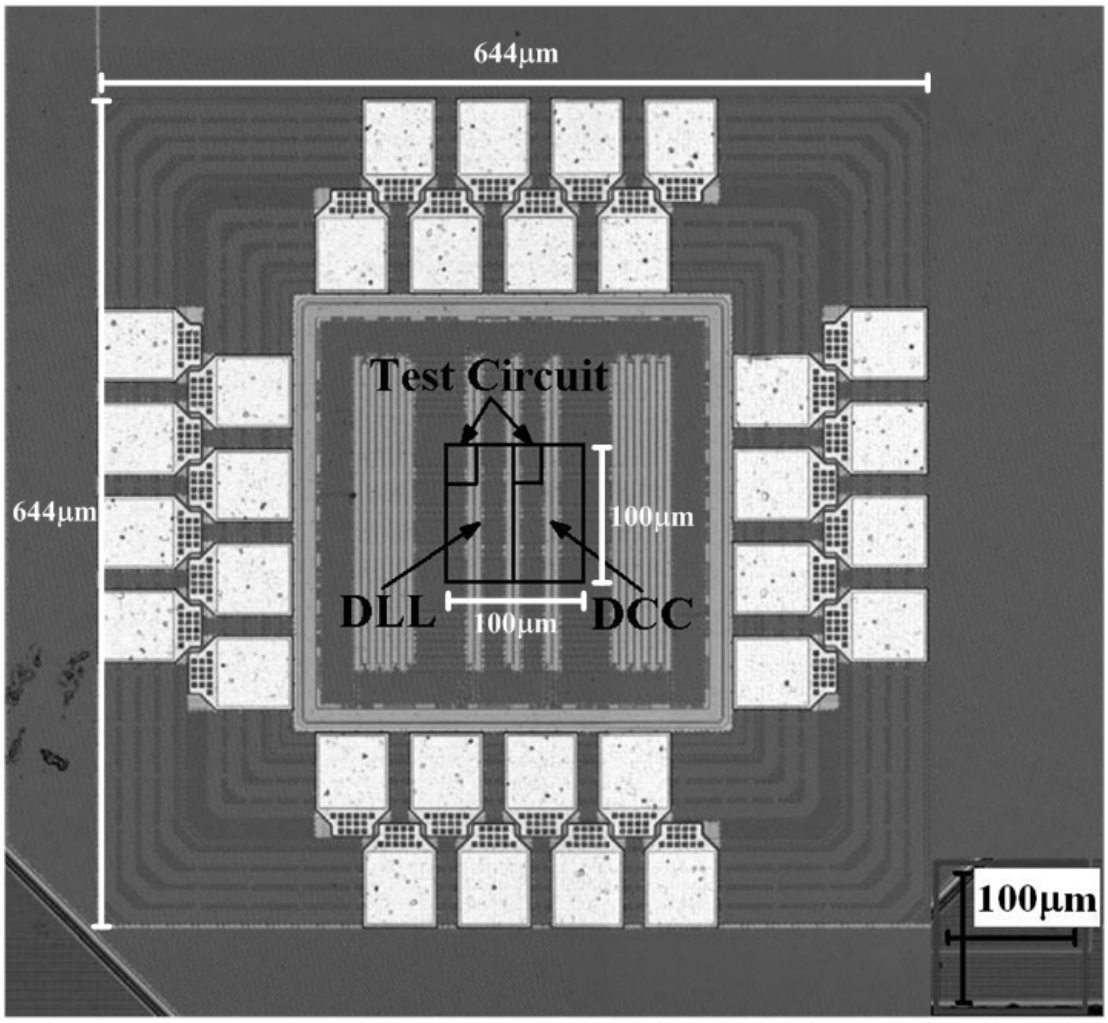


Fig. 4.4 Microphotograph of the ADDCC Ver.1

The proposed ADDCC Ver.1 is fabricated on UMC 65nm 1P10M standard performance (SP) CMOS process. Fig. 4.4 shows the microphotograph of the ADDCC Ver.1. The area of the core chip is 100 X 100 μm^2 and the area with I/O pads is 644 X 644 μm^2. The chip is consisted of a Test Circuit, a DLL and a DCC. The gate count is about 3748 (= 5398/1.44(1.44 is one NAND gate area)).

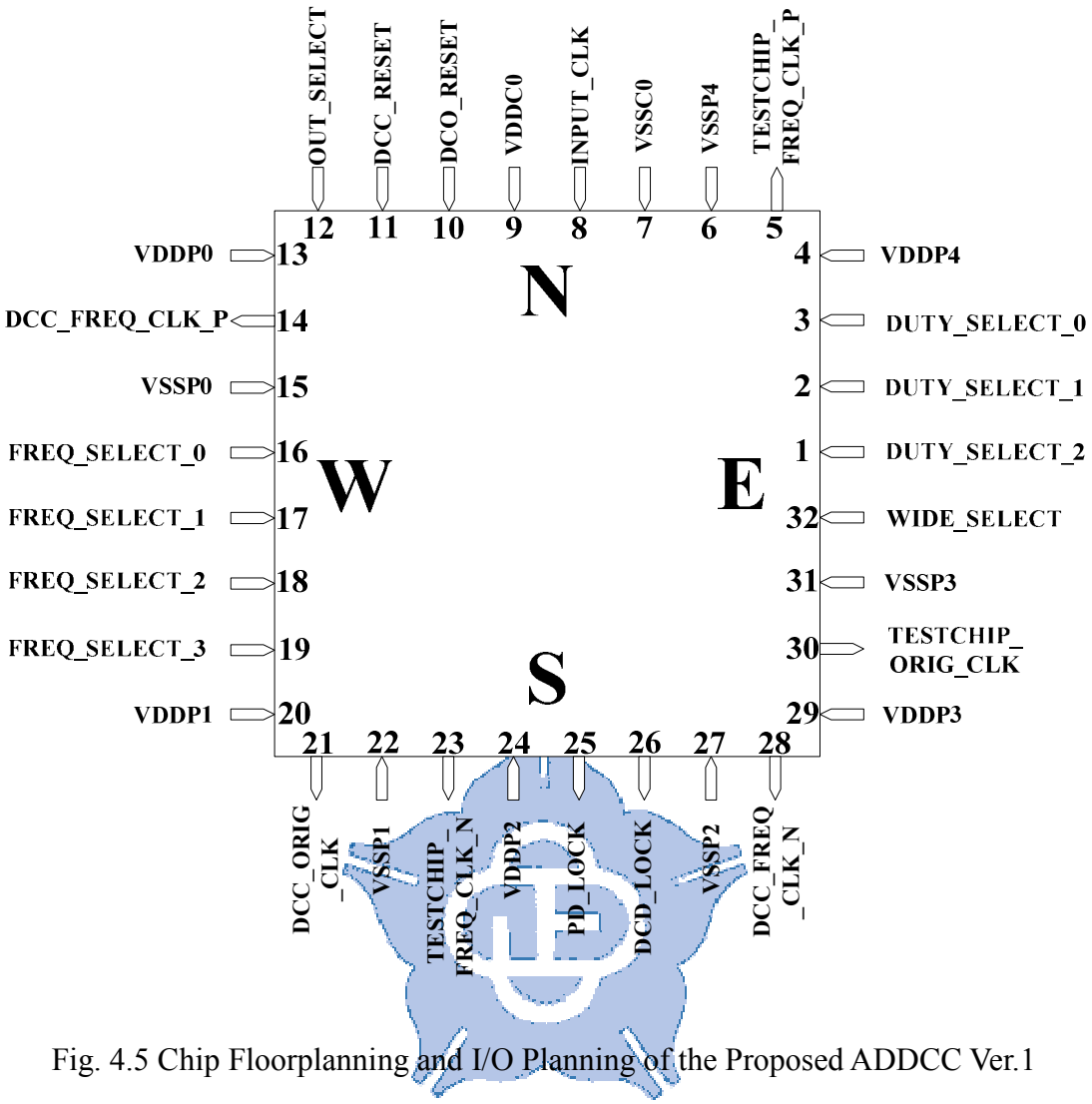


Fig. 4.5 Chip Floorplanning and I/O Planning of the Proposed ADDCC Ver.1

Fig. 4.5 shows the proposed ADDCC Ver.1 chip floorplanning and I/O planning which are 20 I/O Pads and 12 power Pads. Table 4.1 shows the I/O pads information of the proposed ADDCC Ver.2.

TESTCHIP_FREQ_CLK_P and TESTCHIP_FREQ_CLK_N which are the ADDCC Ver.1 input clock divide by DIV_FOUR circuit. DCC_FREQ_CLK_P and DCC_FREQ_CLK_N which are the ADDCC Ver.1 output clock divide by DIV_FOUR circuit.

Table 4.1 The I/O Pads Information of ADDCC Ver.1

Pin Number	Pin Name	Input/Output	Information
3~1	DUTY_SELECT[2:0]	Input	Duty Error Select (15%~85%, step 5%)
4	VDDP4	Input	Pad Power
5	TESTCHIP_FREQ_CLK_P	Output	Divided by FREQ_CLK Positive edge
6	VSSP4	Input	Pad Power
7	VSSC0	Input	Core Power
8	INPUT_CLK	Input	External CLK
9	VDDC0	Input	Core Power
10	DCO_RESET	Input	DCO Reset
11	DCC_RESET	Input	DCC Reset
12	OUT_SELECT	Input	Internal or External clock select
13	VDDP0	Input	Pad Power
14	DCC_FREQ_CLK_P	Output	Divided by CLK_OUT Positive edge
15	VSSP0	Input	Pad Power
19~16	FREQ_SELECT[3:0]	Input	Frequency select (250MHz~1GHz)
20	VDDP1	Input	Pad Power
21	DCC_ORIG_CLK	Output	Without Divided ADDCC clock out
22	VSSP1	Input	Pad Power
23	TESTCHIP_FREQ_CLK_N	Output	Divided by FREQ_CLK Negative edge
24	VDDP2	Input	Pad Power
25	PD_LOCK	Output	DLL Lock
26	DCD_LOCK	Output	DCC Lock
27	VSSP2	Input	Pad Power
28	DCC_FREQ_CLK_N	Output	Divided by CLK_OUT Negative edge
29	VDDP3	Input	Pad Power
30	TESTCHIP_ORIG_CLK	Output	Without Divider
31	VSSP3	Input	Pad Power
32	WIDE_SELECT	Input	Duty Wide Select (over 50% or under 50%)

4.2.2 Simulation Results

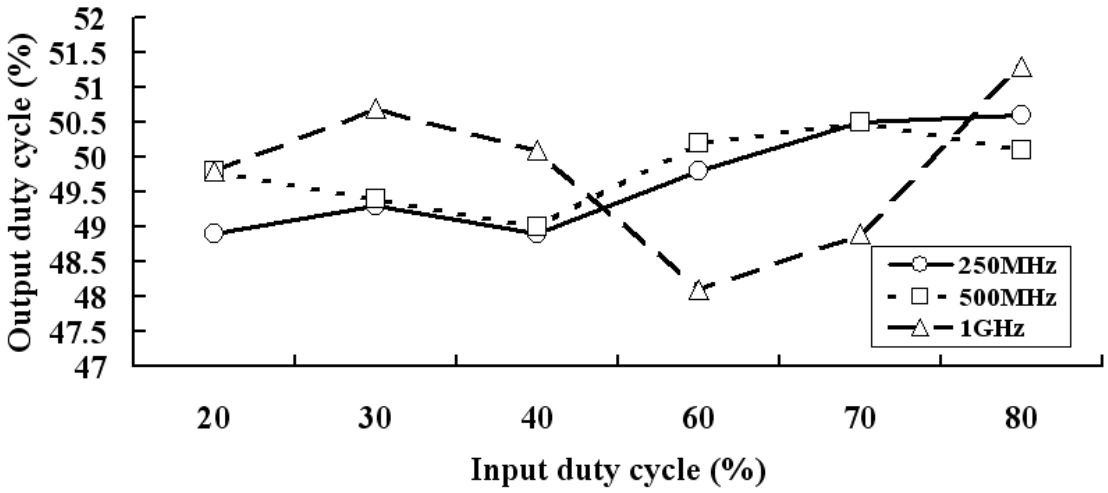
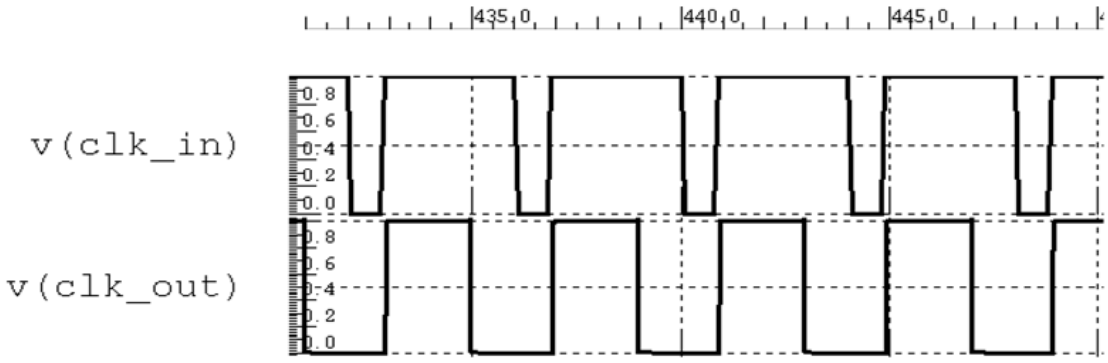
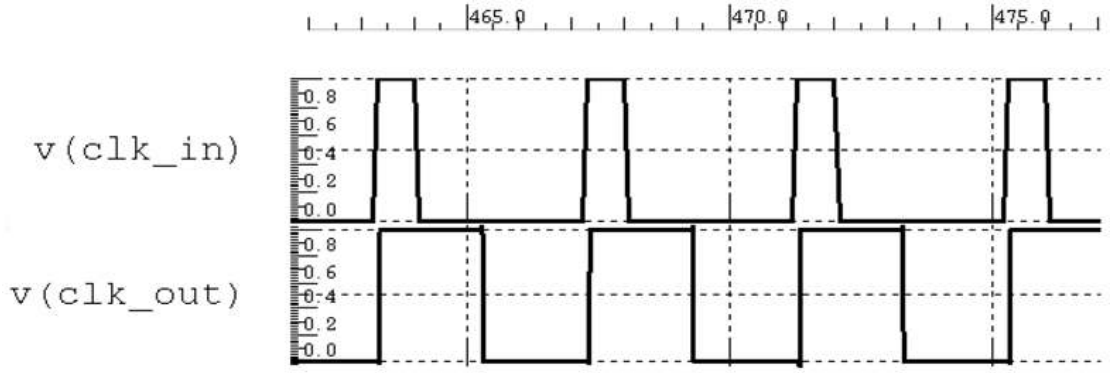


Fig. 4.6 Simulation Results of the Proposed ADDCC Ver.1

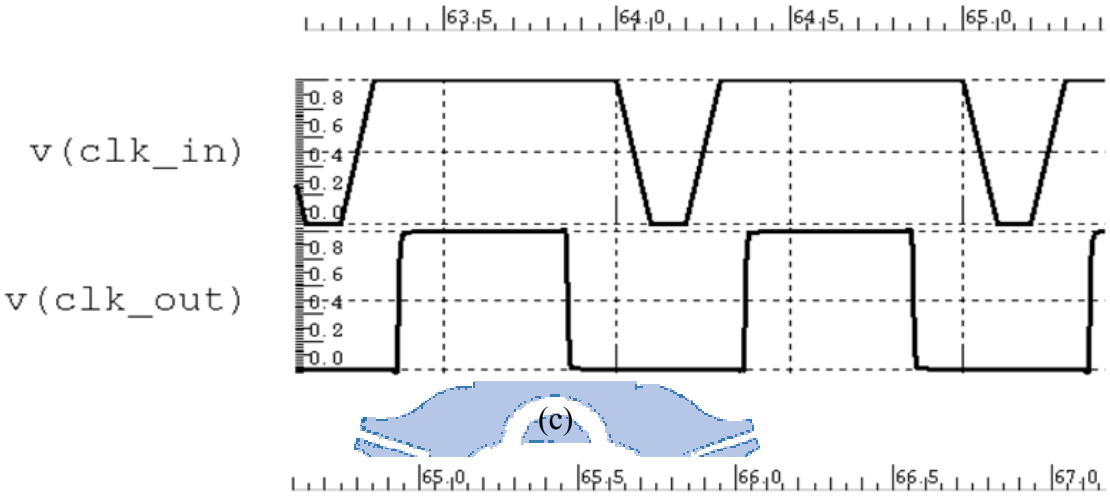
Fig. 4.6 summarized the simulation results of the proposed ADDCC Ver.1. It is simulated by UltraSIM SPICE mode in TT corner. The frequency range of the input clock is from 250MHz to 1GHz, and the duty-cycle range of the input clock is from 20% to 80%. Besides, the supply voltage is 1.0V; the power consumption of the proposed ADDCC Ver.1 is 3.15mW (@1GHz) and is 0.79mW (@250MHz).



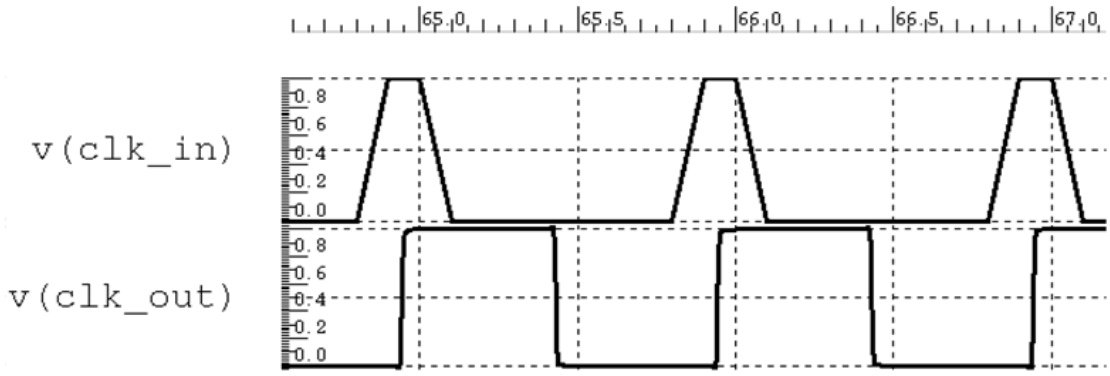
(a)



(b)



(c)



(d)

Fig. 4.7 Simulation Waveform of the Proposed ADDCC Ver.1

(a) 80% duty-cycle input clock at 250MHz, (b) 20% duty-cycle input clock at 250MHz, (c) 80% duty-cycle input clock at 1GHz, (d) 20% duty-cycle input clock at 1GHz

Fig. 4.7 shows the simulation waveform of the proposed ADDCC Ver.1 under different input frequencies and duty-cycle errors. Fig. 4.7(a) shows the input 80% duty-cycle clock at 250MHz, Fig. 4.7(b) shows the input 20% duty-cycle clock at

250MHz, Fig. 4.7(c) shows the input 80% duty-cycle clock at 1GHz, and Fig. 4.7(d) shows the input 20% duty-cycle clock at 1GHz.

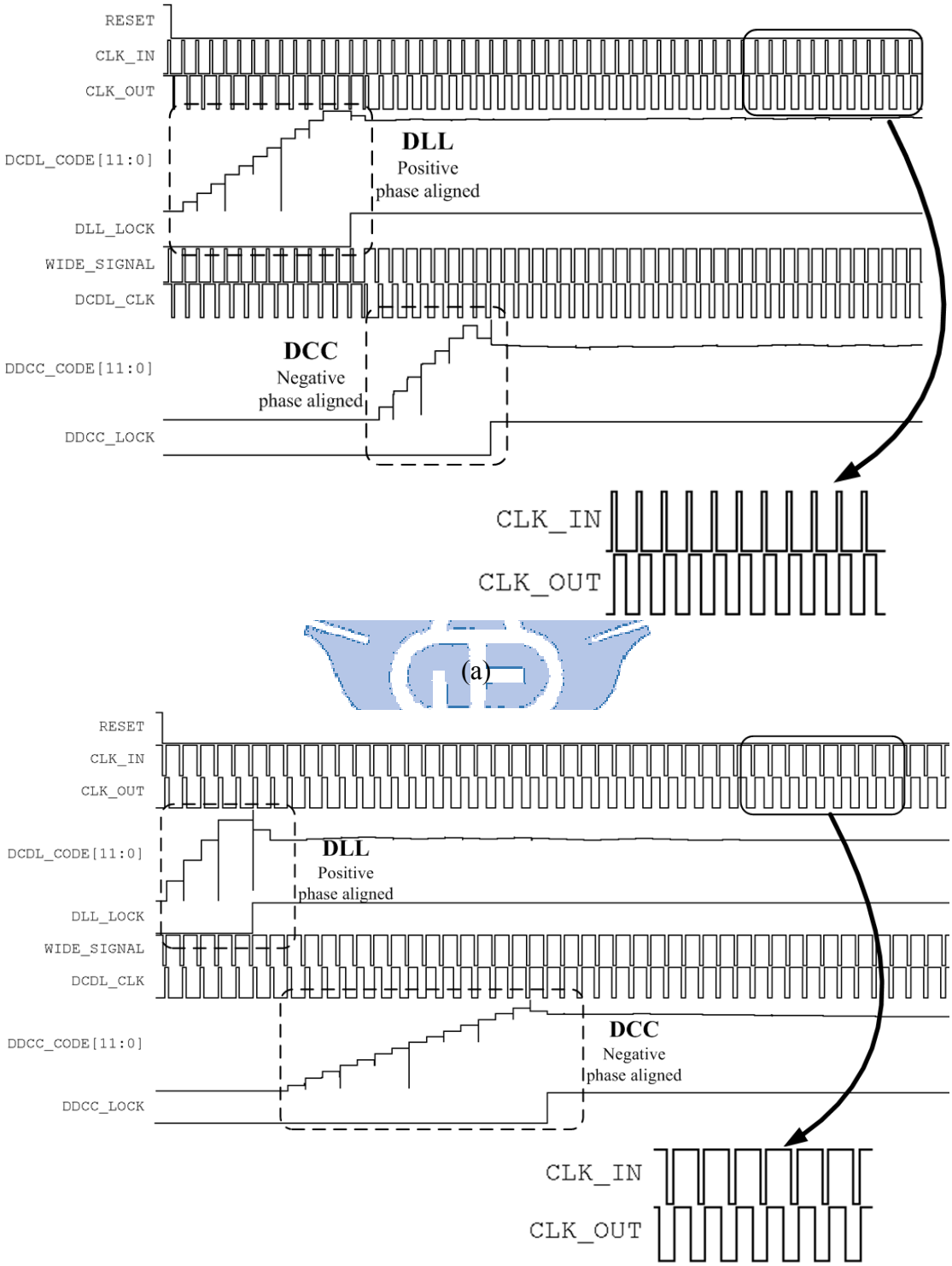


Fig. 4.8 Operation Waveform of the Proposed ADDCC Ver.1

(a) 20% duty-cycle input clock at 1GHz, (b) 80% duty-cycle input clock at 500MHz

Fig. 4.8 shows the operation waveform of the proposed ADDCC Ver.1 at the 1GHz input 20% duty-cycle clock at Fig. 4.8(a) and the 500MHz input 80% duty-cycle clock at Fig. 4.8(b). It is simulated by UltraSim SPICE mode TT corner is 1.0V, 25°C. First, the DLL is generated the two complementary signals by positive phase aligned. Second, the DCC is adjusted the DCDL_CLK's duty-cycle to align the WIDE_SIGNAL's negative phase. Finally, the CLK_OUT is 50% duty-cycle.

4.2.3 Measurement Results

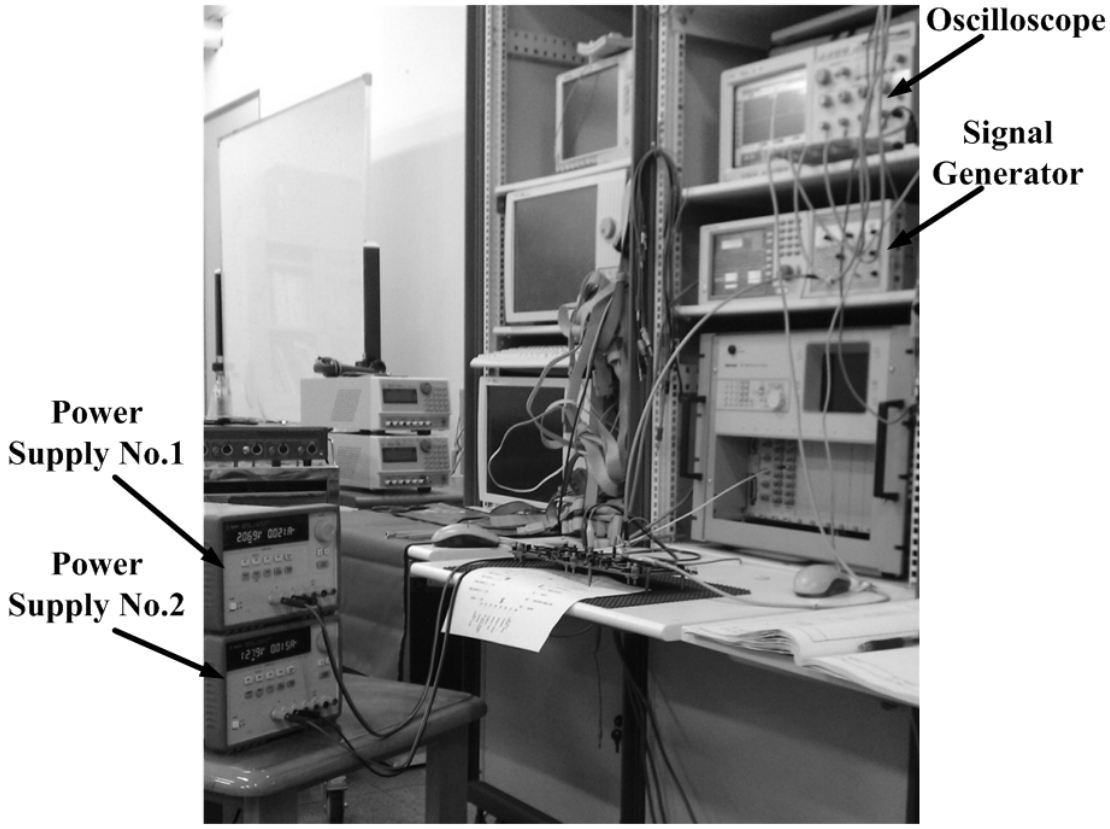


Fig. 4.9 Measurement Environment of the ADDCC Ver.1 Test Chip

Fig. 4.9 shows the measurement environment of the ADDCC Ver.1 test chip. There have two power supplies (Agilent E3631A), one signal generator (Agilent 81134A) and one oscilloscope (Agilent 54855A DSO). The one power supply is used for the Core power the other power supply is used for the Pad power. Furthermore, the Core power is 1.2V and the Pad power is 2.0V. The signal generator is supplied to the external input clock. The oscilloscope is used for sampling output clock waveform.

Table 4.2 Properties of the TEST CHIP Internal Clock Generator

FREQ_SELECT	(WIDE_SELECT, DUTY_SELECT)	Output Frequency	Output Duty-Cycle
0000	(1, 011)	1020MHz	14%
	(1, 010)		26%
	(1, 001)		43%
	(0, 001)		57%
	(0, 010)		74%
	(0, 011)		86%
0101	(1, 100)	442MHz	20%
	(1, 011)		30%
	(1, 001)		42%
	(0, 001)		58%
	(0, 011)		70%
	(0, 101)		82%
1011	(1, 111)	262MHz	15%
	(1, 101)		26%
	(1, 011)		36%
	(1, 001)		46%
	(0, 001)		54%
	(0, 011)		63%
	(0, 101)		74%
	(0, 111)		85%

Table 4.2 shows the properties of the TEST CHIP internal clock frequency and duty-cycle. The DCO can output frequency ranges from 262MHz to 1020MHz. The duty-cycle generator can output duty-cycle ranges from 15% to 85%.

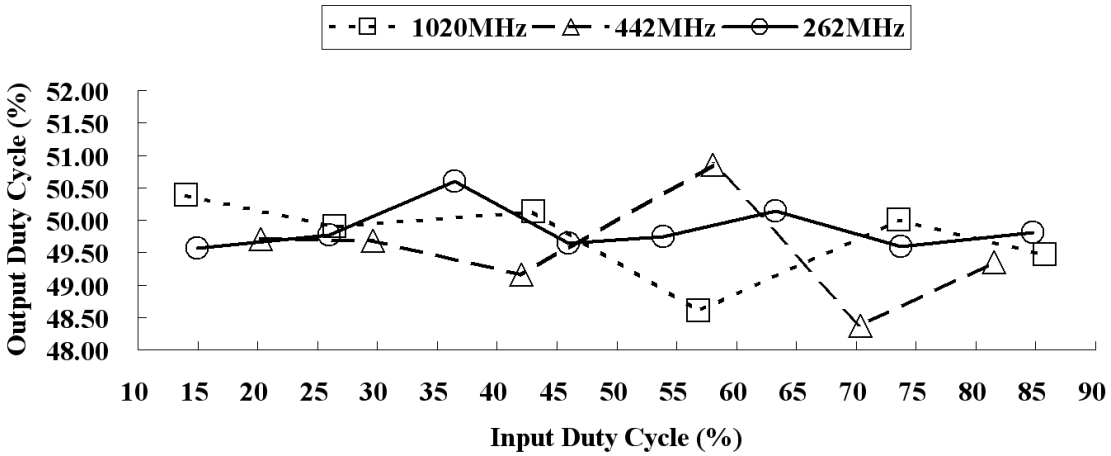


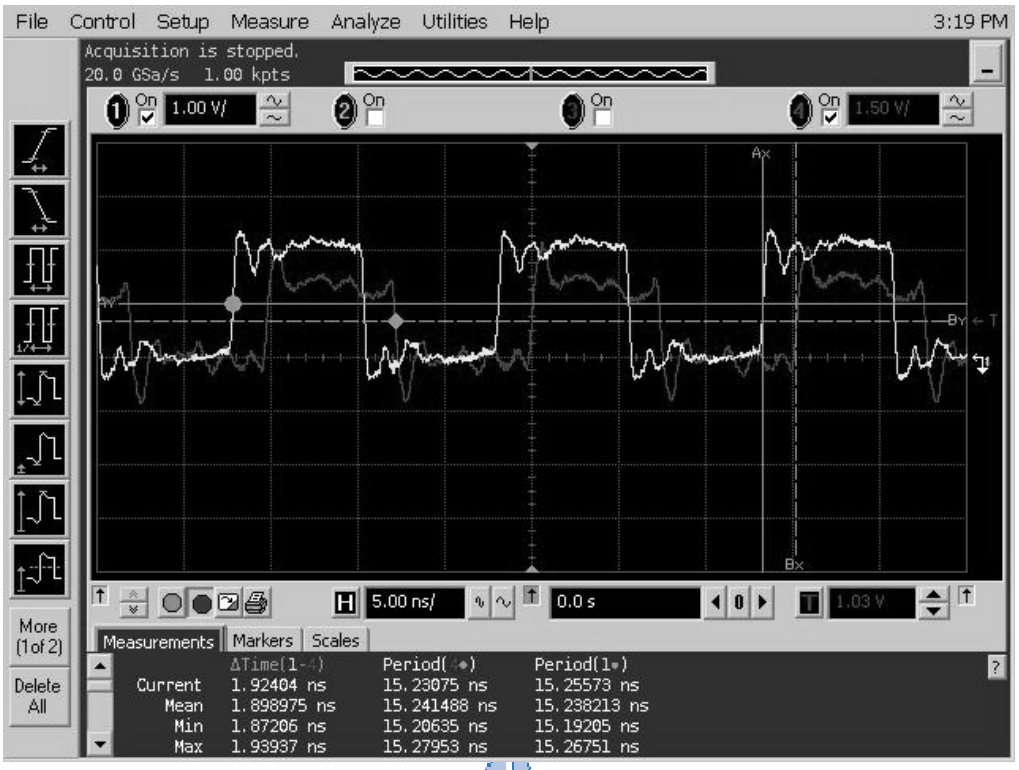
Fig. 4.10 Measurement Results of the Proposed ADDCC Ver.1

Fig. 4.10 summarized the measurement results of the proposed ADDCC Ver.1. The frequency range of the input clock is from 262MHz to 1020MHz, and the duty-cycle range of the input clock is from 15% to 85%. Besides, the core power supply voltage is 1.2V and the pad power supply voltage is 2.0V. The power consumption of the proposed ADDCC Ver.1 is 10mW (@978MHz), 5.6mW (@428MHz) and 3.4mW (@259MHz).

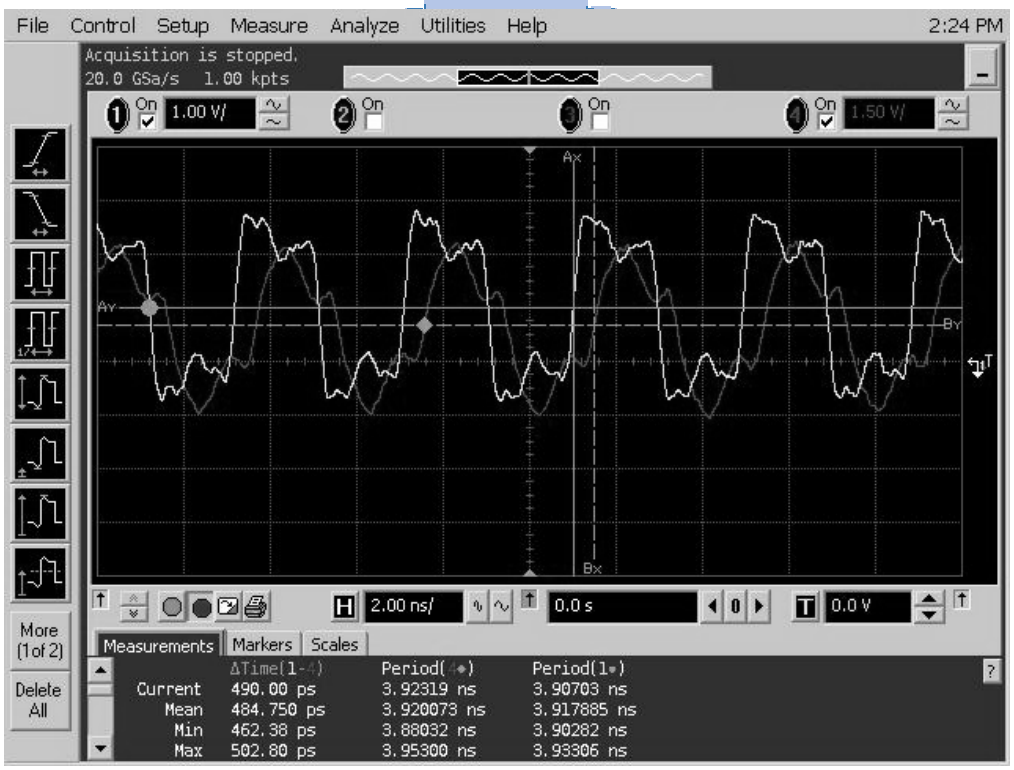
Table 4.3 Jitter Measurement Results of DCC Input Clock and Output Clock

Signal Type	Frequency	Positive edge		Negative edge	
		rms Jitter (ps)	p-p Jitter (ps)	rms Jitter (ps)	p-p Jitter (ps)
DCC Input Clock	262MHz	9.18	100	12.21	78.18
	442MHz	5.6	65.45	5.09	49.09
	1020MHz	3.66	25.45	4.31	29.09
DCC Output Clock	262MHz	8.85	85.46	8.16	81.82
	442MHz	5.58	45.45	4.82	41.82
	1020MHz	3.22	23.64	3.01	23.64

Table 4.3 shows the jitter measurement results of the DCC input clock (the test chip DCO clock generator) and DCC output clock. The frequency range from 262MHz to 1020MHz and the measurement objects are rms (root-mean-square) jitter and p-p (peak-to-peak) jitter. In order to measure the clock duty-cycle that is the delta time between positive edge and negative edge. For this reason, the duty-cycle error is concerned about positive edge jitter and negative edge jitter.



(a)

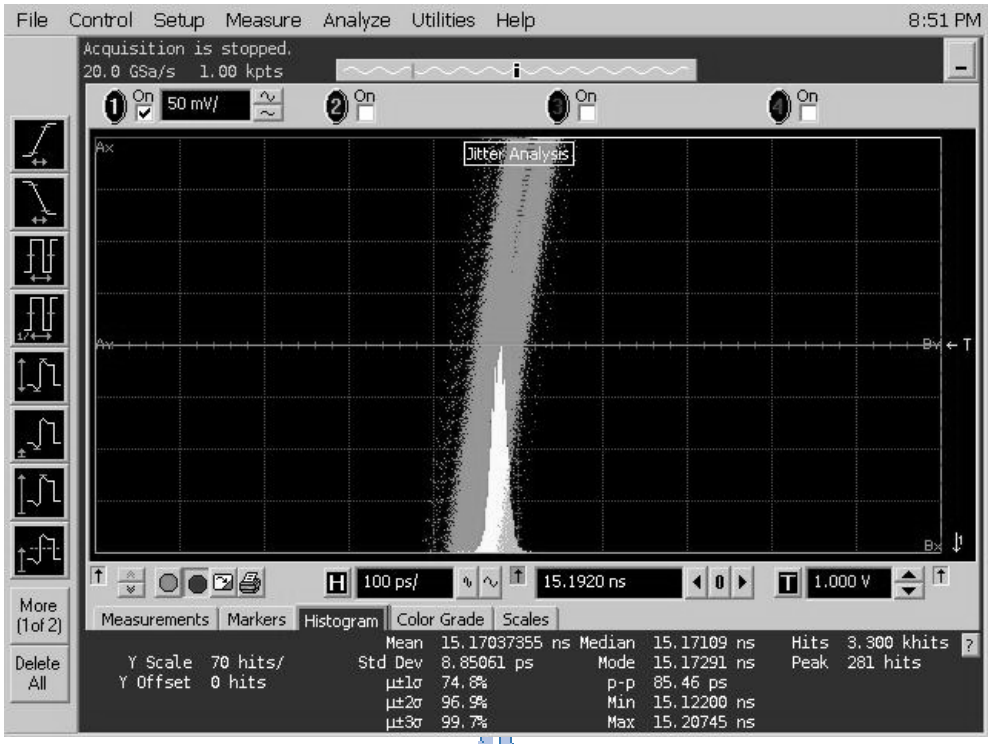


(b)

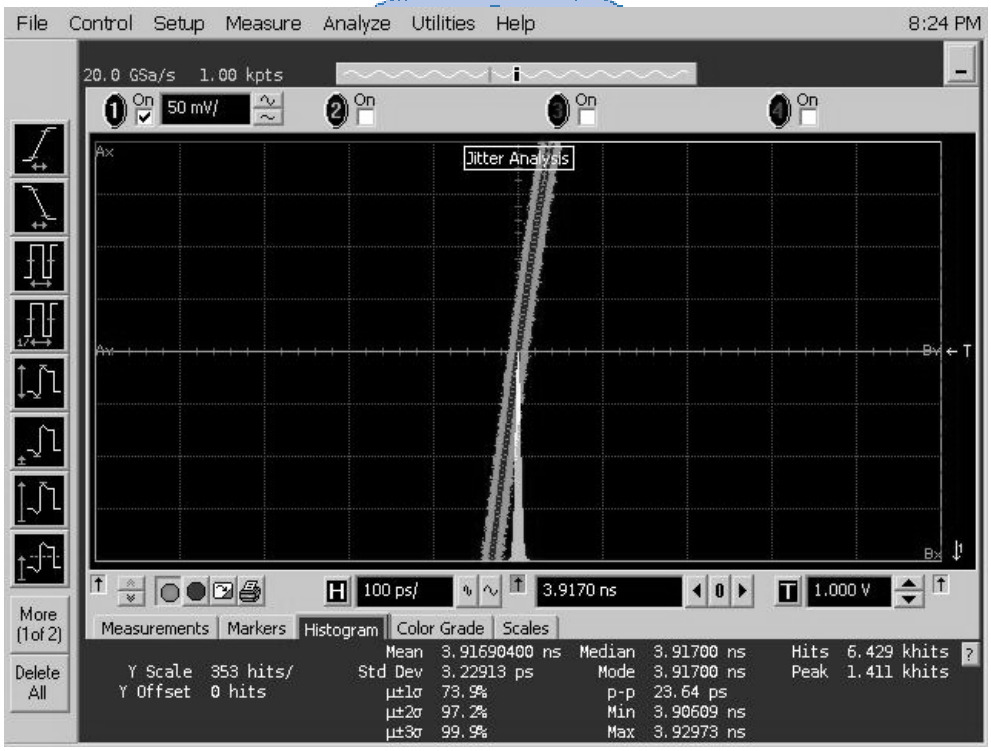
Fig. 4.11 Duty-Cycle Measurement Results of the Proposed ADDCC Ver.1

(a) The Duty-Cycle Measurement Results at 262MHz

(b) The Duty-Cycle Measurement Results at 1020MHz



(a)



(b)

Fig. 4.12 Jitter Histogram of the Proposed ADDCC Ver.1

(a) The Jitter Histogram of the proposed ADDCC Ver.1 at 262MHz

(b) The Jitter Histogram of the proposed ADDCC Ver.1 at 1021MHz

Fig. 4.11 shows the duty-cycle measurement results of the proposed ADDCC Ver.1 at 1020MHz and 262MHz. In the Fig 4.11, the signal No.1 is divided signal which triggered by positive edge and the signal No.4 is divided signal which triggered by negative edge. The measurement method is shown in Fig. 4.3. Fig. 4.11 (a) shows the delta time between signal No.1 positive phase and signal No.4 positive edge is 1.898 (ns). Hence the duty-cycle is 49.8% (=(1.898/(15.241/4))*100%). Fig 4.11 (b) shows the delta time between signal No.1 positive phase and signal No.4 positive edge is 0.4847 (ns). Hence the duty-cycle is 49.4% (=(0.4847/(3.92/4))*100%).

Fig. 4.12 shows the jitter histogram of the proposed ADDCC Ver.1 at 1021MHz and 262MHz. In Fig. 4.12(a), the signal mean period is 3.916 (ns), due to the output signal frequency is divided by four. So the actual on chip frequency is 1021MHz (=1/(3.916/4)). In Fig. 4.12(b), the signal mean period is 15.17 (ns), so the actual on chip frequency is 263MHz (=1/(15.17/4)).

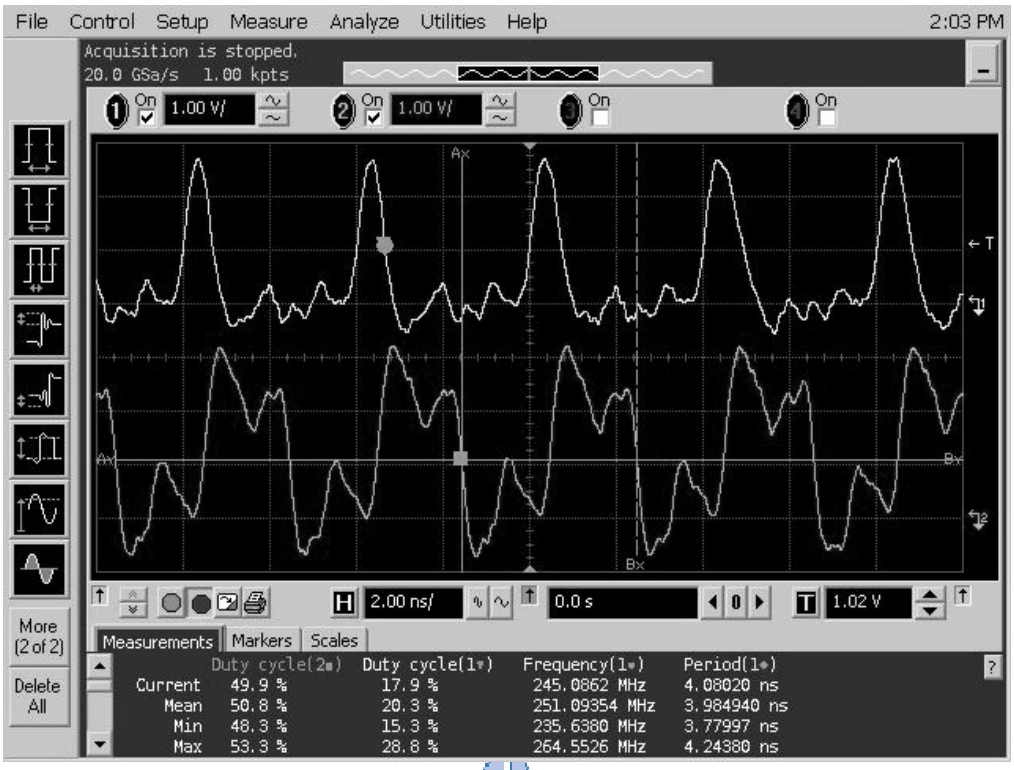


Fig. 4.13 Phase Error between Input Clock and Output Clock

Fig. 4.13 shows the phase error between input clock and output clock. The signal No.1 is the external 20% duty-cycle input clock and the signal No.2 is the ADDCC output. The phase error between signal No.1 and signal No.2 is about 400 (ps) which caused by the ADDCC intrinsic delay (Signal Selector and Half DDCC). Fig. 4.14 shows the block diagram of intrinsic delay from CLK_IN to CLK_OUT.

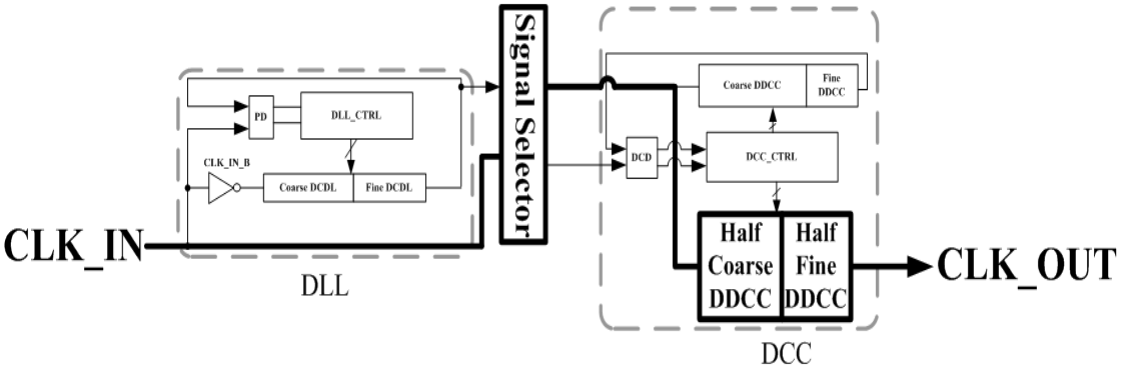


Fig. 4.14 Block Diagram of Intrinsic Delay from CLK_IN to CLK_OUT

4.3 The Proposed ADDCC Ver.2

4.3.1 Specifications

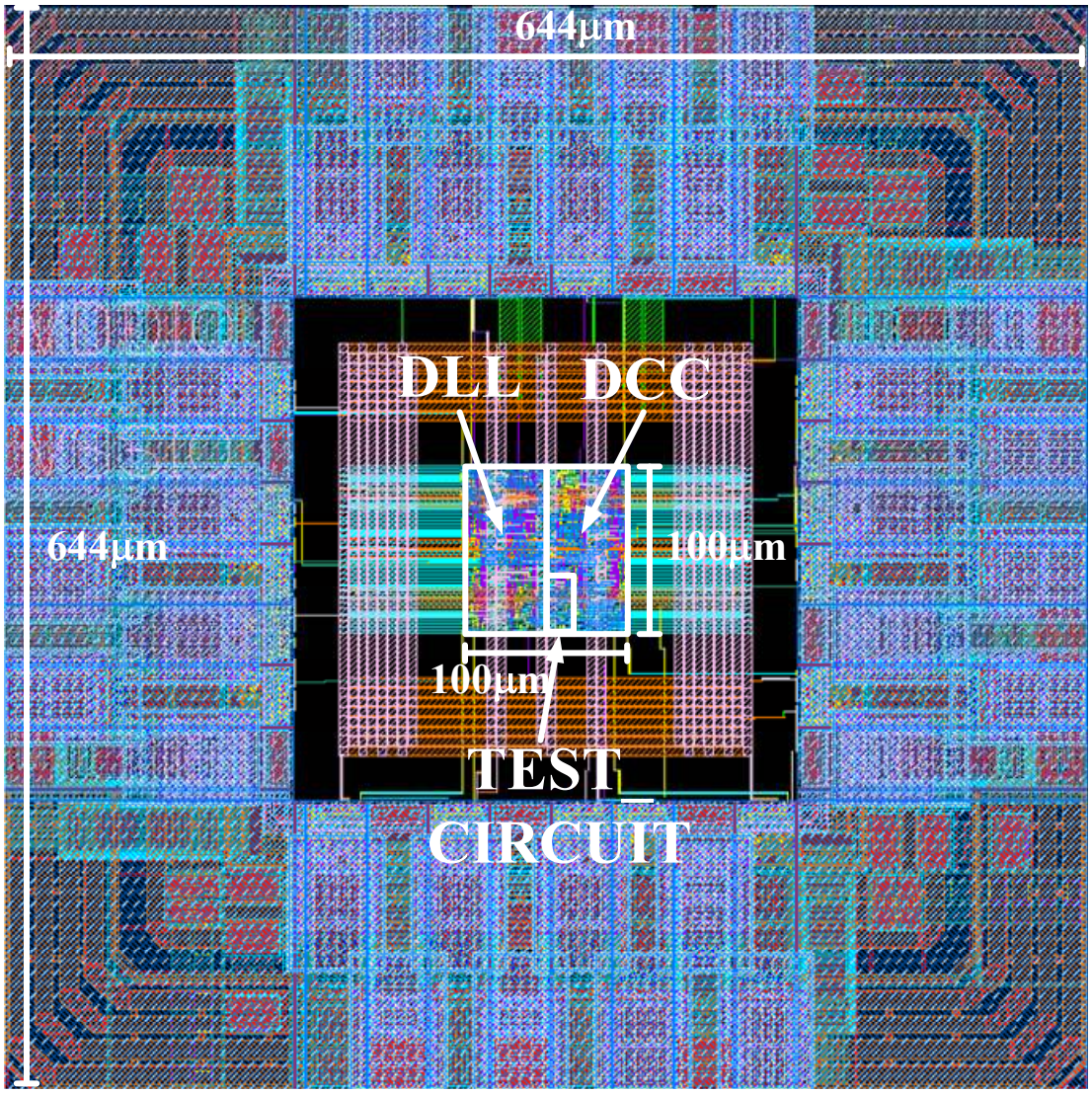


Fig. 4.15 Layout Diagram of the ADDCC Ver.2

The proposed ADDCC Ver.2 is implemented on UMC 65nm 1P10M standard performance (SP) CMOS process. Fig. 4.15 shows the layout diagram of the ADDCC Ver.2. The area of the core chip is 100 X 100 μm^2 and the area with I/O pads is 644 X 644 μm^2. The chip is consisted of a Test Circuit, a DLL and a DCC. The chip gate count is about 4149 (= 5975/1.44(1.44 is one NAND gate area)).

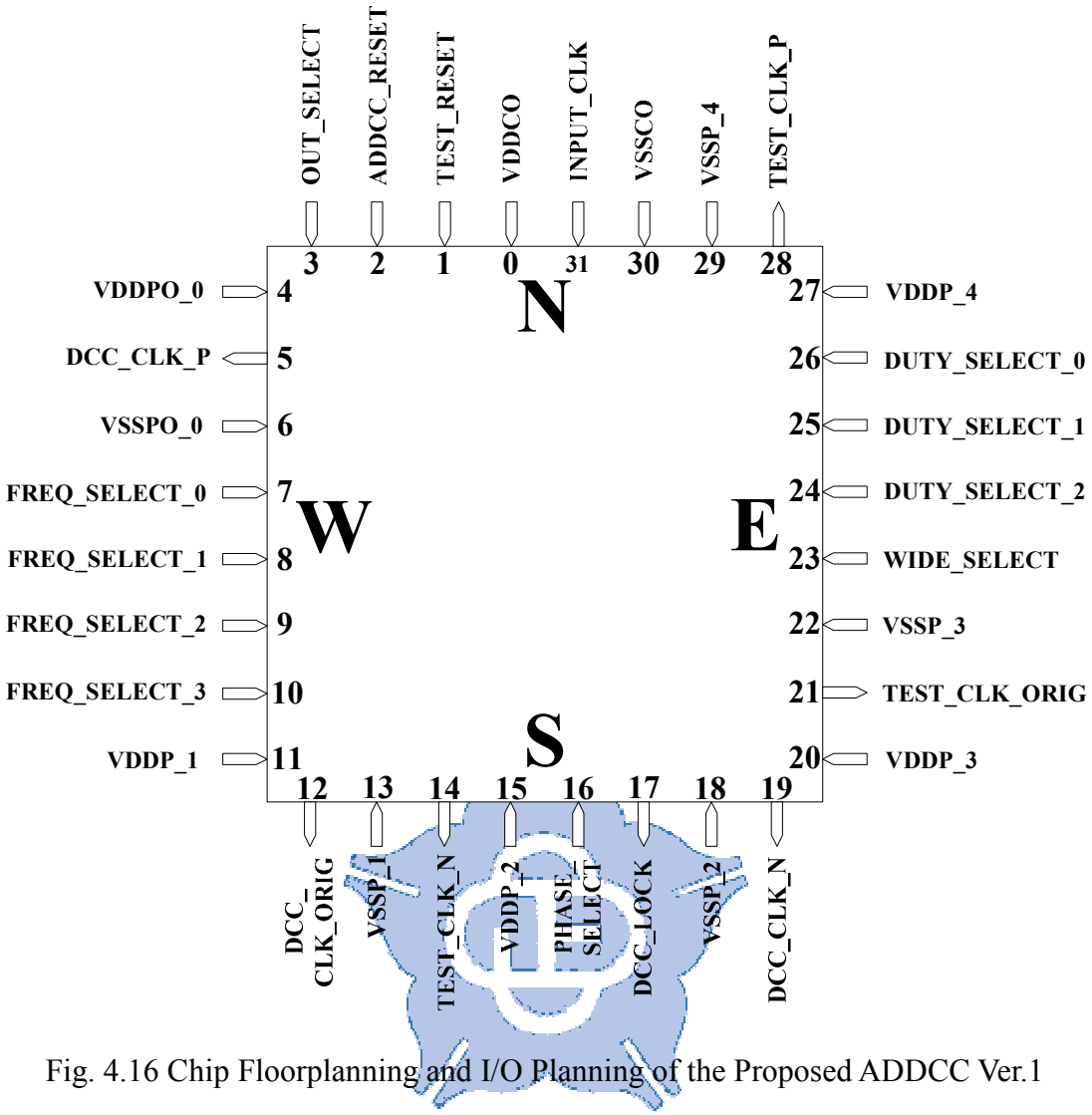


Fig. 4.16 Chip Floorplanning and I/O Planning of the Proposed ADDCC Ver.1

Fig. 4.16 shows the proposed ADDCC Ver.2 chip floorplanning and I/O planning which are 20 I/O Pads and 12 power Pads. Table 4.4 shows the I/O pads information of the proposed ADDCC Ver.2.

TEST_CLK_P and TEST_CLK_N which are the ADDCC Ver.2 input clock divide by DIV_FOUR circuit. DCC_CLK_P and DCC_CLK_N which are the ADDCC Ver.2 output clock divide by DIV_FOUR circuit.

Table 4.4 The I/O Pads Information of ADDCC Ver.2

Pin Number	Pin Name	Input/Output	Information
1	VDDC_0	Input	Core Power
2	TEST_RESET	Input	TEST CHIP Reset
3	ADDCC_RESET	Input	ADDCC Reset
4	OUT_SELECT	Input	Internal or External clock select
5	VDDP_0	Input	Pad Power
6	DCC_CLK_P	Output	Divided by CLK_OUT Positive edge
7	VSSP_0	Input	Pad Power
8~11	FREQ_SELECT	Input	Frequency select (200MHz~1066MHz)
12	VDDP_1	Input	Pad Power
13	DCC_CLK_ORIG	Output	Original System CLK_OUT
14	VSSP_1	Input	Pad Power
15	TEST_CLK_N	Output	Divided by TEST_CLK Negative edge
16	VDDP_2	Input	Pad Power
17	PHASE_SELECT	Input	DLL Phase Select
18	DCC_LOCK	Output	System DCC Lock
19	VSSP_2	Input	Pad Power
20	DCC_CLK_N	Output	Divided by CLK_OUT Negative edge
21	VDDP_3	Input	Pad Power
22	TEST_CLK_ORIG	Output	Without divided TEST_CLK
23	VSSP_3	Input	Pad Power
24	WIDE_SELECT	Input	Duty Wide Select (over 50% or under 50%)
25~27	DUTY_SELECT	Input	Duty Error Select (15%~85%, step 5%)
28	VDDP_4	Input	VDD Pad Power
29	TEST_CLK_P	Output	Divided by TEST_CLK Positive edge
30	VSSP_4	Input	Pad Power
31	VSSC_0	Input	Core Power
32	INPUT_CLK	Input	External CLK

4.3.2 Post-Layout Simulation Results

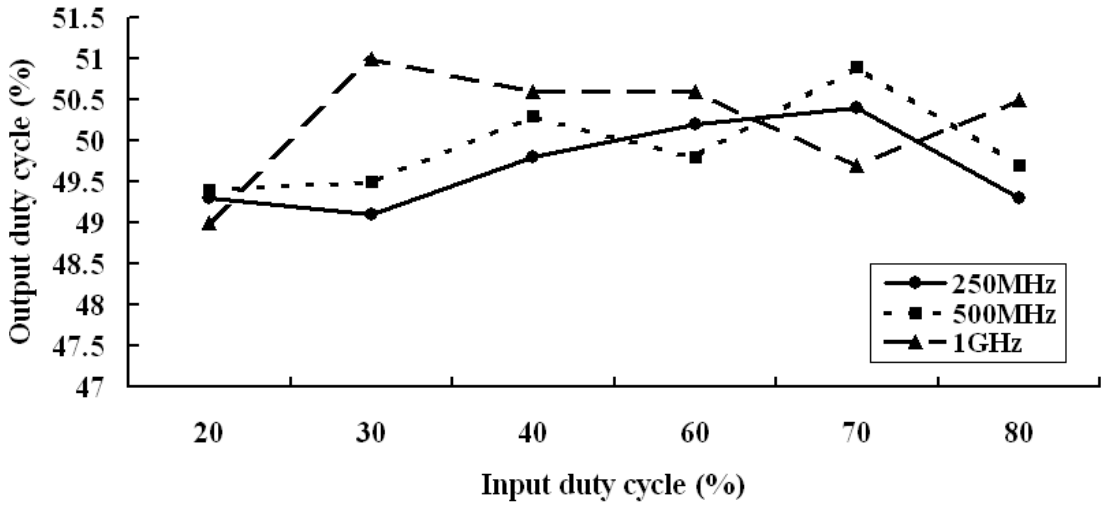
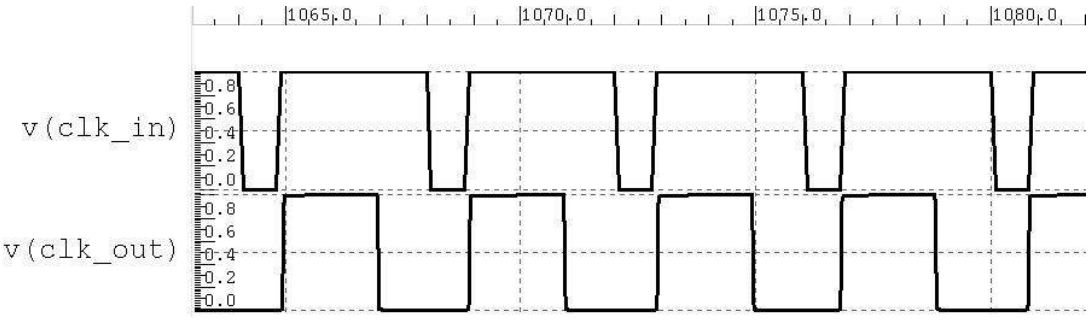
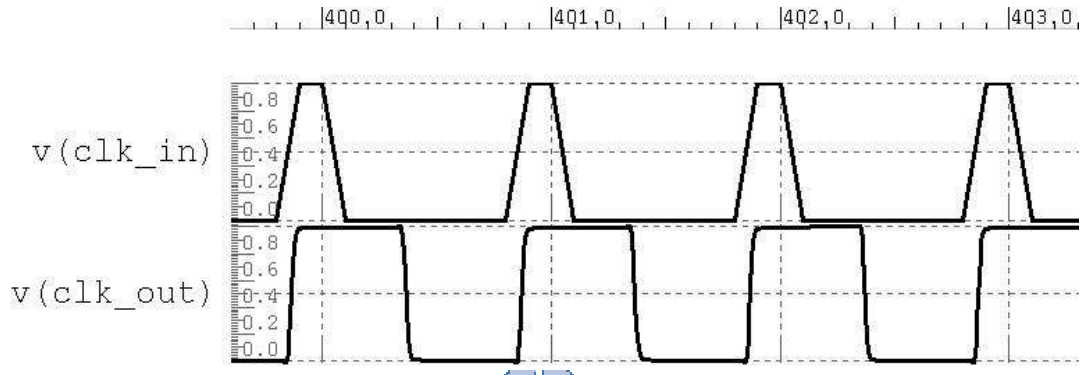


Fig. 4.17 Simulation Results of the Proposed ADDCC Ver.2

Fig. 4.17 summarized the simulation results of the proposed ADDCC Ver.2. It is simulated by UltraSim SPICE mode in TT corner. The frequency range of the input clock is from 250MHz to 1GHz, and the duty-cycle range of the input clock is from 20% to 80%. Besides, the supply voltage is 1.0V; the power consumption of the proposed ADDCC Ver.2 is 5.83mW (@1GHz) and is 1.52mW (@250MHz).



(a)



(b)

Fig. 4.18 Simulation Waveform of the Proposed ADDCC Ver.2

(a) 80% duty-cycle input clock at 250MHz

(b) 20% duty-cycle input clock at 1GHz

Fig. 4.18 shows the simulation waveform of the proposed ADDCC Ver.2 under different input frequencies and duty-cycle errors. Fig. 4.18(a) shows the input 80% duty-cycle clock at 250MHz and Fig. 4.18(b) shows the input 20% duty-cycle clock at 1GHz.

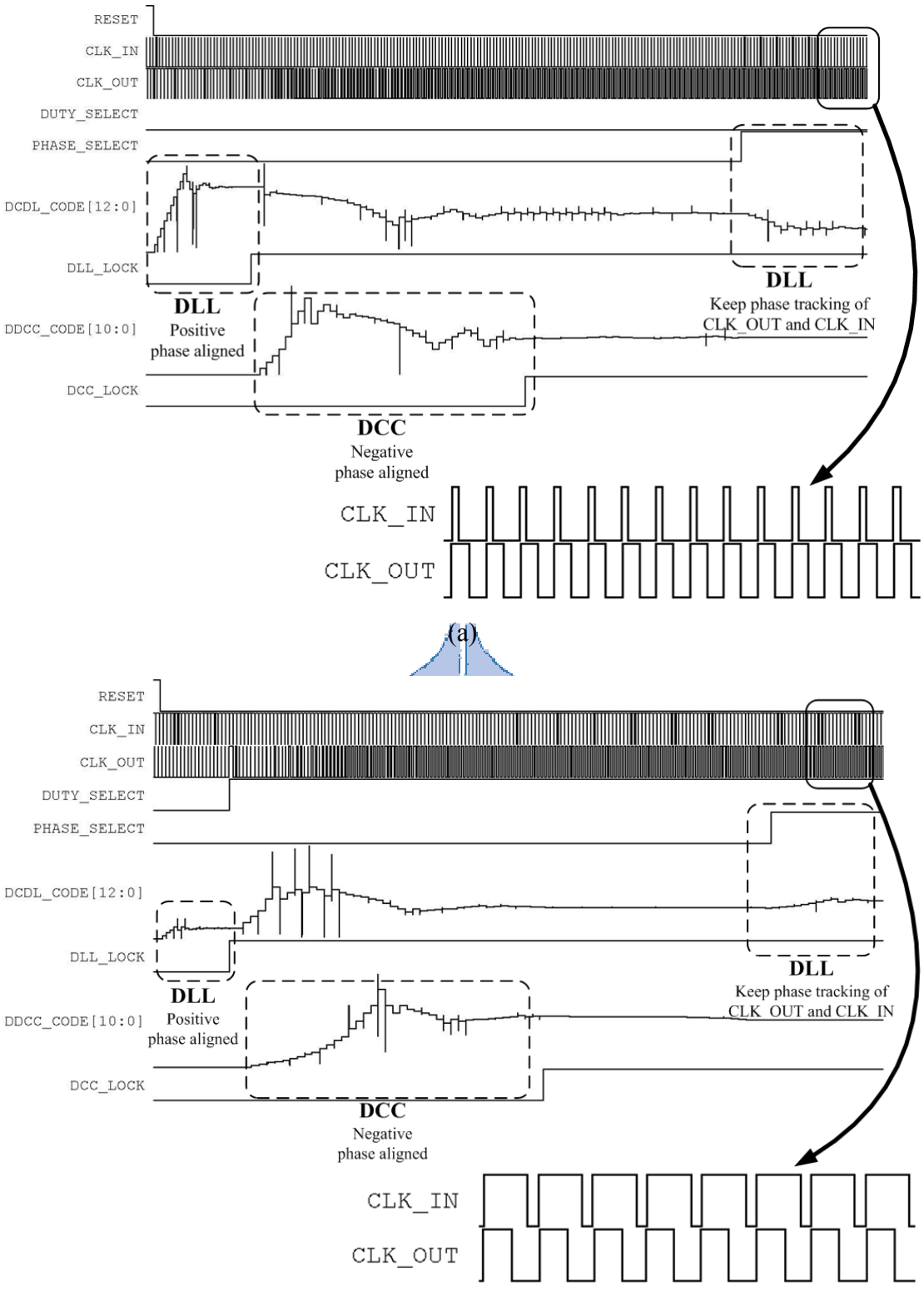


Fig. 4.19 Operation Waveform of the Proposed ADDCC Ver.2

(a) 20% duty-cycle input clock at 1GHz, (b) 80% duty-cycle input clock at 500MHz

Fig. 4.19 shows the operation waveform of the proposed ADDCC Ver.2 at the

1GHz input 20% duty-cycle clock at Fig. 4.19(a) and the 500MHz input 80% duty-cycle clock at Fig. 4.19(b). It is simulated by UltraSim SPICE mode TT corner is 1.0V, 25°C. In the beginning, the DLL is generated the two complementary signals by positive phase aligned. Second, the DCC is adjusted the CLK_OUT duty-cycle to 50% by negative phase aligned. Finally, the DLL keeps phase tracking of CLK_IN and CLK_OUT to de-skew the instinct delay.

4.4 Comparisons of Recent Research

Table 4.5 Performance Comparisons

Parameter	Ver.1	Ver.2	JSSC'06 [7]	TCAS-II'07 [14]	JSSC'09 [18]	VLSI'11 [10]	EL'08 [25]	JSSC'05 [8]
Type	Sequential Search/HCDL	Sequential Search	TDC	TDC	TDC/HCDL	TDC/HCDL	Digital PWCL	Analog PWCL
Process	65nm	65nm	0.35μm	0.18μm	0.18μm	0.18μm	0.18μm	0.35μm
Supply voltage	1.0V	1.0V	3.3V	1.8V	1.8V	1.8V	1.8V	3.3V
Max. Frequency (MHz)	1020	1066	600	1200	1500	2000	1250	1270
Min. Frequency (MHz)	262	200	400	800	440	400	800	1000
Input Duty Cycle Range	14%~86%	20%~80%	30%~70%	40%~60%	40%~60% @1.5GHz	10%~90% @400MHz 20%~80% @2GHz	40%~60%	N/A
Output 50% Duty Cycle Error	-0.4%~0.6% @262MHz -1.6%~0.8% @442MHz -1.4%~0.4% @1.02GHz	-0.9%~0.4% @250MHz -0.6%~0.9% @500MHz -1.0%~1.0% @1GHz	±0.6%	-1.5%~1.4% @1GHz	±1.8%	±1% @400MHz ±3.5 @1GHz	±0.6%	2% @1.25GHz
Duty Cycle Corrector Resolution	3.5ps	1.75ps	120ps	78.1ps*	17.75ps**	78.1***	N/A	N/A
Align with input clock	Yes	Yes	Yes	Yes	Yes	No	Yes	No
Power consumption	1.96mW @262MHz 2.68mW @442MHz 6.5mW @1.02GHz	1.52mW @250MHz 3.03mW @500MHz 5.83mW @1GHz	20mW @500MHz	15mW @1GHz	43mW @1.5GHz	1.76mW @400MHz 3.6mW @2GHz	16mW @1GHz	150mW @1.25GHz
p-p Jitter	23.64ps @1.02GHz	N/A	16.7ps @500MHz	12.9ps @1GHz	7ps @1.5GHz	28.45 @1GHz	N/A	19.6ps @1.25GHz
Area(mm^2)	0.01	0.01	0.68	0.23	0.053	0.025	0.09	0.141
Experimental Results Type	Measurement	Simulation	Measurement	Measurement	Measurement	Measurement	Simulation	Measurement

* : 1250ps / 16 = 78.1ps (@800MHz)

** : τ = 2272.7ps / 16 = 142ps (@440MHz)

resolution is τ * 0.125 = 142 * 0.125 = 17.75ps

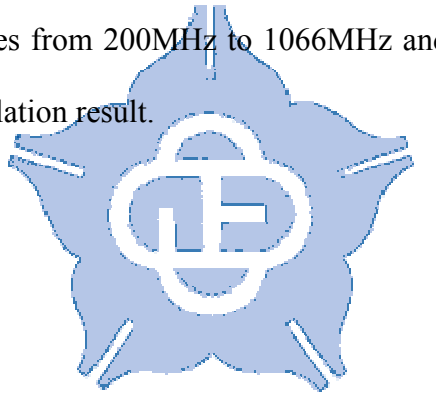
*** : 2500ps / 32 = 78.1 (@400MHz)

Table 4.5 shows the performance comparison with the prior studies. In [7, 10, 14, 18], the TDC-based all-digital DCC architecture must have a high resolution TDC to minimize the duty-cycle error. However, it is not easy to design a wide-range high resolution TDC. Therefore, they are not suitable for wide frequency range operation.

In [8, 25], the analog and digital PWCL has precisely 50% duty-cycle output, but the power consumption is large and frequency range is restricted. Compared to prior studies, the proposed ADDCC not only has a wider frequency range, but also has a wider input duty-cycle range.

4.5 Summary

In this chapter, we have shown the proposed ADDCC simulation results and measurement results. The ADDCC Ver.1 can achieve wide-range operation with input frequency ranges from 262MHz to 1020MHz and input duty-cycle ranges from 14% to 86% by measurement result. The ADDCC Ver.2 can achieve wide-range operation with input frequency ranges from 200MHz to 1066MHz and input duty-cycle ranges from 20% to 80% by simulation result.



Chapter 5 Conclusion and Future Works

In this thesis, we propose two kinds of all-digital duty-cycle corrector (ADDCC). One is the ADDCC Ver.1 with high resolution delay line; the other is the ADDCC Ver.2 without half delay line. The ADDCC Ver.1 solves the duty-cycle corrected to 50% precise problem, and the lock time is faster than the analog method. The ADDCC Ver.2 uses a novel correction method without half delay which can solves the half delay resolution problem and the delay mismatch problem in nano-meter CMOS process. The proposed ADDCC architectures can achieve wide frequency range and wide input duty-cycle error range. As a result, it is very suitable for duty-cycle correction applications in DDR2/3 I/O bus applications.

In some applications, the re-correction duty-cycle mechanism is requested. The ADDCC Ver.1 tracking method is faster than the analog method, but is slower than the TDC method. For this reason, we can use the TDC method to substitute for DLL phase tracking step.

The ADDCC Ver.1 and Ver.2 use the cascaded delay line structure. This kind of delay line has non-monotonic problem when crosses sub-band. In chapter 2, we have compensated a fixed code by simulation result. However, the delay line performance and accuracy are affected by the process, voltage and temperature (PVT) variations. The compensation code is determined by TT corner simulation result, but if the chip works on SS corner or FF corner. The compensation code will cause the delay dead-zone or the duty-cycle error. From the above problem, we can use the interpolation-type delay line to solve the non-monotonic problem.

Reference

[1] Fenghao Mu and Christer Svensson, “Pulsewidth control loop in high-speed CMOS clock buffers, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 35, no. 2, pp. 134-141, Feb. 2000.

[2] Po-Hui Yang and Jinn-Shyan Wang, “Low-voltage pulsewidth control loops for SOC applications, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 37, no. 10, pp. 1348-1351, Oct. 2002.

[3] Sung-Rung Han and Shen-Iuan Liu, “A 500-MHz-1.25-GHz fast-locking pulsewidth control loop with presettable duty cycle, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 39, no. 3, pp. 463-468, Oct. 2004.

[4] Kuo-Hsing Cheng, Chia-Wei Su and Kai-Fei Chang, “A high linearity, fast-locking pulsewidth control loop with digitally programmable duty cycle correction for wide range operation, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 43, no. 2, pp. 399-413, Feb. 2008.

[5] Poki Chen, Shi-Wei Chen and Juan-Shan Lai, “A low power wide tange duty cycle corrector based on pulse shrinking/stretching mechanism, “ *in Proceedings of IEEE Asian Solid-State Circuits Conference (ASSCC)*, Jan. 2008,pp. 460-463.

[6] Yi-Ming Wang, Chang-Fen Hu, Yi-Jen Chen and Jinn-Shyan Wang, “An all-digital pulsewidth control loop, “ *in Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, Jul. 2005,pp. 1258-1261.

[7] You-Jen Wang, Shao-Ku Kao and Shen-Iuan Liu, “All-digital delay-locked loop/pulsewidth-control loop with adjustable duty cycles, “ *IEEE Journal Solid-State Circuits (JSSC)*, vol. 41, no. 6, pp. 1262-1274, Jun. 2006.

[8] Sung-Rung Han and Shen-Iuan Liu, “A single-path pulsewidth control loop with a built-in delay-locked loop, “ *IEEE Journal Solid-State Circuits (JSSC)*, vol. 40,

no. 5, pp. 1130-1135, Jun. 2005.

[9] Kuo-Hsing Cheng, Cia-Wei Su , Kai-Fei Chang, Cheng-Liang Hung and Wei-Bin Yang, “A High Linearity and Fast-Locked PulseWidth Control Loop with Digitally Programmable Output Duty Cycle for Wide Range Operation, “ *in Proceedings of the 32th European Solid-State Circuits Conference (ESSCIR)*, Sept. 2006,pp. 178-181.

[10] Junhui Gu, Jianhui Wu, Danhong Gu, Meng Zhang, and Longxing Shi, “All-Digital Wide Range Precharge Logic 50% Duty Cycle Corrector, “ *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Mar. 2011,pp. 1-5.

[11] Shao-Ku Kao and Shen-Iuan Liu, “A Wide-Range All-Digital Duty Cycle Corrector with a Period Monitor, “ *in Proceedings of IEEE Conference on Electron Devices and Solid-State Circuits (EDSSC)*, Dec. 2007,pp. 349-352.

[12] Yi-Ming Wang and Jinn-Shyan Wang, “An all-digital 50% duty-cycle corrector, “ in Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS), May 2004,pp. 925-928.

[13] Bo-Jiun Chen, Shao-Ku Kao and Shen-Iuan Liu, “An All-Digital Duty Cycle Corrector, “ *in Proceedings of International Symposium on VLSI Design, Automation, and Test (VLSI-DAT)*, Apr. 2006,pp. 1-4.

[14] Shao-Ku Kao and Shen-Iuan Liu, “All-Digital Fast-Locked Synchronous Duty-Cycle Corrector, “ *IEEE Transactions on Circuits and Systems II: Express Briefs (TCAS-II)*, vol. 53, no. 12,pp. 1363-1367, Dec. 2006.

[15] Jun-Hyun Bae, Jin-Ho Seo, Hwan-Seok Yeo, Jae-Whui Kim, Jae-Yoon Sim and Hong-June Park, “An All-Digital 90-Degree Phase-Shift DLL with Loop-Embedded DCC for 1.6Gbps DDR Interface, “ *in Proceedings of IEEE Custom Integrated Circuits Conference (CICC)*, Sept. 2007,pp. 373-376.

[16] Jinn-Shyan Wang, Chun-Yuan Cheng, Je-Ching Liu, Yu-Chia Liu and Yi-Ming

Wang, “A Duty-Cycle-Distortion-Tolerant Half-Delay-Line Low-Power Fast-Lock-in All-Digital Delay-Locked Loop, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 45, no. 5, pp. 1036-1047, May 2010.

[17] Byung-Guk Kim, Kwang-Il Oh, Lee-Sup Kim and Dae-Woo Lee, “A 500MHz DLL with second order duty cycle corrector for low jitter, “ *in Proceedings of IEEE Custom Integrated Circuits Conference (CICC)*, Jan. 2006,pp. 325-328.

[18] Dongsuk Shin, Chulwoo Kim, Janghoon Song and Hyunsoo Chae, “A 7 ps Jitter 0.053 mm^2 Fast Lock All-Digital DLL With a Wide Range and High Resolution DCC, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 44, no. 9,pp. 2437-2451, Aug. 2009.

[19] Dongsuk Shin, Janghoon Song, Hyunsoo Chae, Kwan-Weon Kim, Young Jung Choi and Chulwoo Kim, “A 7ps-Jitter 0.053mm^2 Fast-Lock ADDLL with Wide-Range and High-Resolution All-Digital DCC, “ *in Digest of Technical Papers, IEEE Solid-State Circuits Conference (ISSCC)*, Feb. 2007, pp. 184-185.

[20] Ji-Wei Ke, Shi-Yu Huang and Ding-Ming Kwai, “A high-resolution all-digital duty-cycle corrector with a new pulse-width detector, ” *in Proceedings of IEEE Conference on Electron Devices and Solid-State Circuits (EDSSC)*, Dec. 2010,pp. 1-4.

[21] Dongsuk Shin, Won-Joo Yun, Hyun-Woo Lee, Young-Jung Choi, Suki Kim and Chulwoo Kim, “A 0.17–1.4GHz low-jitter all digital DLL with TDC-based DCC using pulse width detection scheme, “ *in Proceedings of the 34th European Solid-State Circuits Conference (ESSCIRC)*, Sept. 2008,pp. 82-85.

[22] Stephan Henzler, Siegmar Koeppe, Dominik Lorenz, Winfried Kamp, Ronald Kuenemund, and Doris Schmitt-Landsiedel, “A Local Passive Time Interpolation Concept for Variation-Tolerant High-Resolution Time-to-Digital Conversion, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 43, no. 7, pp. 1666-1676, Jul.

2008.

[23] Luca Vercesi, Antonio Liscidini and Rinaldo Castello, “Two-Dimensions Vernier Time-to-Digital Converter, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 45, no. 8, pp. 1504-1512, Aug. 2010.

[24] Jianjun Yu, Fa Foster Dai and Richard C. Jaeger, “A 12-bit vernier ring time-to-digital converter in 0.13μm CMOS technology, “ *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 45, no. 4, pp. 830-842, Apr. 2010.

[25] J.C. Ha, J.H. Lim, Y.J. Kim, W.Y. Jung and J.K. Wee, “Unified all-digital duty-cycle and phase correction circuit for QDR I/O interface, “ Electronics Letters, vol. 44, no. 22, pp. 1300-1301, Oct. 2008.

[26] Shin Dongsuk, Na, Kwang-Jin Kwon, Daehan Kang, Jong-Ho, Song Taeksang, Jung Ho-Don, Lee Woo-Young, Park Ki-Chon, Park Jung-Hoon, Joo Yong-Suk, Cha Jae-Hoon, Jung Youngho, Kim Youngran, Han Donghoon, Choi Byoung-Jin, Lee Geun-Il, Cho Joo-Hwan and Choi Young-Jung, “Wide-range fast-lock duty-cycle corrector with offset-tolerant duty-cycle detection scheme for 54nm 7Gb/s GDDR5 DRAM interface, “ *in Digest of Technical Papers, Symposium on VLSI Circuits (VLSIC)*, Aug. 2009,pp. 138-139.

[27] Chunseok Jeong, Changsik Yoo, Jae-Jin Lee and Joongsik Kih, “Digital delay locked loop with open-loop digital duty cycle corrector for 1.2Gb/s/pin double data rate SDRAM, “ *in Proceedings of the 30th European Solid-State Circuits Conference (ESSCIR)*, Nov. 2004,pp. 379-382.

[28] Kyu-Hyoun Kim, Geun-Hee Cho, Jung-Bae Lee and Soo-In Cho, “Built-in duty cycle corrector using coded phase blending scheme for DDR/DDR2 synchronous DRAM Application, “ *in Digest of Technical Papers, Symposium on VLSI Circuits (VLSIC)*, Aug. 2003,pp. 287-289.

[29] Seung-Jun Bae, Kwang-Il Park, Jeong-Don Ihm, Ho-Young Song, Woo-Jin Lee,

Hyun-Jin Kim, Kyoung-Ho Kim, Yoon-Sik Park, Min-Sang Park, Hong-Kyong Lee, Sam-Young Bang, Gil-Shin Moon, Seok-Won Hwang, Young-Chul Cho, Sang-Jun Hwang, Dae-Hyun Kim, Ji-Hoon Lim, Jae-Sung Kim, Sung-Hoon Kim, Seong-Jin Jang, Joo Sun Choi, Young-Hyun Jun, Kinam Kim and Soo-In Cho, "An 80 nm 4 Gb/s/pin 32 bit 512 Mb GDDR4 Graphics DRAM With Low Power and Low Noise Data Bus Inversion, " *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 43, no. 1,pp. 121-131, Jan. 2008.

[30] Won-Joo Yun, Hyun-Woo Lee, Dongsuk Shin, and Suki Kim, "A 3.57 Gb/s/pin Low Jitter All-Digital DLL With Dual DCC Circuit for GDDR3 DRAM in 54-nm CMOS Technology, " *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Mar. 2011,pp. 1-5.

[31] Won-Joo Yun, Hyun Woo Lee, Dongsuk Shin, Shin Deok Kang, Ji Yeon Yang, Hyeng Ouk Lee, Dong-Uk Lee, Sujeong Sim, Young Ju Kim, Won Jun Choi, Keun Soo Song, Sang Hoon Shin, Hyang Hwa Choi, Hyung Wook Moon, Seung Wook Kwack, Jung Woo Lee, Young Kyoung Choi, Nak Kyu Park, Kwan Weon Kim, Young Jung Choi, Jin-Hong Ahn and Ye Seok Yang, "A 0.1-to-1.5GHz 4.2mW All-Digital DLL with Dual Duty-Cycle Correction Circuit and Update Gear Circuit for DRAM in 66nm CMOS Technology, " *in Digest of Technical Papers, IEEE Solid-State Circuits Conference (ISSCC)*, Feb. 2009, pp. 282-283.

[32] Duo Sheng, Ching-Che Chung and Chen-Yi Lee, "An ultra-low-power and portable digitally controlled oscillator for SoC applications, " *IEEE Transactions on Circuits and Systems II: Express Briefs (TCAS-II)*, vol. 54, no. 11, pp. 954-958, Nov. 2007.

[33] Hsuan-Jung Hsu, Chun-Chieh Tu and Shi-Yu Huang, "A high-resolution all-digital phase-lock loop with its application to built-in speed grading for memory, " *in Proceedings IEEE Symposium on VLSI Design Automation and*

Test (VLSI-DAT), pp. 267-270, Apr. 2008.

[34] Yoo, C., Jeong, C. and Kih, J., "Open-loop full-digital duty cycle correction circuit," Electronics Letters, vol. 41, no. 11, pp. 635-636, May 2005.

[35] Ching-Che Chung and Chen-Yi Lee, "A new DLL-based approach for all-digital multi-phase clock generation, " *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 39, no. 3, pp. 469-475, Mar. 2004.