

**A 1.25-Gb/s Digitally-Controlled Dual-Loop  
Clock and Data Recovery Circuit  
with Enhanced Phase Resolution**

Chang-Kyung Seong

The Graduate School

Yonsei University

Department of Electrical and Electronic Engineering

**A 1.25-Gb/s Digitally-Controlled Dual-Loop  
Clock and Data Recovery Circuit  
with Enhanced Phase Resolution**

A Masters Thesis

Submitted to the Department Electrical and Electronic Engineering

and the Graduate School of Yonsei University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

**Chang-Kyung Seong**

January 2006

This certifies that the masters thesis of Chang-Kyung Seong is approved.

---

**Thesis Supervisor : Woo-Young Choi**

---

**Gun-Hee Han**

---

**Seung-Woo Lee**

**The Graduate School**

**Yonsei University**

**January 2006**

# Contents

<b>List of Figures</b>	<b>iv</b>
<b>List of Tables</b>	<b>viii</b>
<b>Abstract</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Review of Clock and Data Recovery Circuits for Multi-Channel Applications</b>	<b>5</b>
2.1 Dual-loop Clock and Data Recovery Circuits . . . . .	5
2.2 Digitally-Controlled Dual-Loop Clock and Data Recovery Circuits . . . .	8
2.2.1 Basic Structure . . . . .	8
2.2.2 Effect of Phase Resolution on Digitally-Controlled Loop . . . . .	9
2.2.3 Structure and Phase Resolution Limit of Phase Interpolator . . . .	14
2.3 System-level Design and Behavioral Simulations of CDR using CPPSIM	19
2.3.1 Effect of phase resolution . . . . .	22
2.3.2 Effect of counting number of up/down filter . . . . .	24

2.3.3	Effect of latency . . . . .	25
<b>3</b>	<b>Proposed Digitally-Controlled Dual-Loop Clock and Data Recovery Circuit</b>	<b>27</b>
3.1	Phase Resolution Enhancement . . . . .	29
3.2	Phase Interpolator Linearization . . . . .	33
<b>4</b>	<b>Circuit Design and Simulation Results</b>	<b>39</b>
4.1	Circuit-level Design of Clock and Data Recovery circuit . . . . .	39
4.1.1	Phase Interpolator . . . . .	39
4.1.2	Phase Controller . . . . .	42
4.1.3	Up/Down Filter . . . . .	50
4.1.4	Digitally-Controlled Delay Buffer . . . . .	54
4.2	Circuit-level Design and Simulations of Reference Phase-Locked Loop	55
4.2.1	Voltage-Controlled Oscillator and Voltage-to-Current Converter .	55
4.2.2	Charge Pump . . . . .	56
4.2.3	Loop Dynamics and Loop Filter . . . . .	61
4.2.4	Simulation of Reference Phase-Locked Loop . . . . .	62
4.3	Circuit-level Simulation of Clock and Data Recovery circuit . . . . .	64
<b>5</b>	<b>Experimental Results</b>	<b>72</b>
5.1	Measurement Setup . . . . .	74
5.2	Measurement Results . . . . .	76
<b>6</b>	<b>Conclusion</b>	<b>81</b>



# List of Figures

1.1	Example of switch system . . . . .	2
1.2	Level and timing recovery procedure in general serial link . . . . .	4
2.1	Basic configuration of multi-channel dual-loop clock and data recovery circuits . . . . .	6
2.2	Basic structure of conventional digitally-controlled dual-loop clock and data recovery circuit . . . . .	9
2.3	Phase dithering in digitally-controlled CDR due to clock latency on the loop . . . . .	11
2.4	Slew rate of reference clock versus number of reference clocks . . . . .	13
2.5	Reducing slew rate only at the last node in clock distribution for noise insensitivity . . . . .	13
2.6	Output phase vs. coefficient $\beta$ . . . . .	17
2.7	Schematic of general phase interpolator circuit . . . . .	17
2.8	Schematic of current digital-to-analog converter using in phase interpolator	18
2.9	Dynamic phase jump of binary-weighted phase interpolator . . . . .	18
2.10	CPPSIM schematic for behavioral simulation using CPPSIM . . . . .	21

2.11	Acquisition time vs. phase resolution . . . . .	23
2.12	RMS jitter generation vs. phase resolution . . . . .	23
2.13	Movement of control code during tracking frequency offset in locked state	24
2.14	RMS jitter generation of CDRs for various combinations of phase resolution and filtering order . . . . .	25
2.15	RMS jitter generation of CDR for various latencies and effects of filtering on it . . . . .	26
3.1	Block diagram of the proposed CDR . . . . .	28
3.2	Concept of phase resolution enhancement using digitally-controlled delay buffer . . . . .	31
3.3	Concept of DCDB delay error . . . . .	31
3.4	RMS Jitter generation vs. DCDB delay error . . . . .	32
3.5	Non-linear transfer curve of conventional phase interpolator . . . . .	35
3.6	Linear DAC . . . . .	36
3.7	Non-linear DAC . . . . .	36
3.8	Comparison of transfer curve : Conventional vs. linearized phase interpolator . . . . .	37
3.9	Comparison of nonlinearity : Conventional vs. linearized phase interpolator . . . . .	38
4.1	Schematic of linearized phase interpolator . . . . .	41
4.2	Phase transfer curve of PI with and without without phase offset . . . . .	42
4.3	Configuration of three controllers . . . . .	43

4.4	State-diagram of DCDB controller; 2-bit binary Up-Down counter . . . . .	45
4.5	Schematic of DCDB controller . . . . .	45
4.6	State-diagram of PI controller; 15-bit bidirectional shift register . . . . .	46
4.7	Flow-table of PI controller; 15-bit bidirectional shift register . . . . .	47
4.8	Schematic of PI controller . . . . .	47
4.9	State-diagram and Quadrant mapping of MUX controller . . . . .	48
4.10	Schematic of MUX controller and 4-input <i>Sum of Product</i> gate . . . . .	48
4.11	Operation of controller in $2\pi$ range . . . . .	49
4.12	State-diagram of Up/Down filter . . . . .	51
4.13	Schematic of Up/Down filter . . . . .	52
4.14	Schematic of 6-input <i>sum of product</i> gate . . . . .	52
4.15	Simulation results of Up/Down filter in locked state using HSPICE . . . . .	53
4.16	Schematic of digitally-controlled delay buffer . . . . .	54
4.17	Block diagram of reference phase-locked loop . . . . .	55
4.18	Block diagram of 4-stage ring oscillator . . . . .	57
4.19	Schematic of unit delay cell in voltage-controlled oscillator . . . . .	57
4.20	Transfer curve of the designed voltage-controlled oscillator: oscillation frequency vs. control voltage . . . . .	58
4.21	Schematic of voltage-to-current converter . . . . .	58
4.22	Schematic of charge pump . . . . .	59
4.23	Simulated current mismatch for whole range of control voltage . . . . .	60
4.24	Acquisition process of reference PLL for three process corners . . . . .	63
4.25	Accumulated waveform of the synthesized quadrature clocks . . . . .	63

4.26	Jitter generation of the CDR using 6-bit PI with 2-bit filter . . . . .	65
4.27	Jitter generation of the CDR using 6-bit PI with 3-bit filter . . . . .	65
4.28	Jitter generation of the CDR using both 6-bit PI and 2-bit DCDB with 2-bit filter . . . . .	66
4.29	Jitter generation of the CDR using both 6-bit PI and 2-bit DCDB with 3-bit filter . . . . .	66
4.30	Jitter rejection of the CDR using 6-bit PI with 2-bit filter . . . . .	67
4.31	Jitter rejection of the CDR using 6-bit PI with 3-bit filter . . . . .	67
4.32	Jitter rejection of the CDR using both 6-bit PI and 2-bit DCDB with 2-bit filter . . . . .	68
4.33	Jitter rejection of the CDR using both 6-bit PI and 2-bit DCDB with 3-bit filter . . . . .	68
4.34	Comparison of four CDRs in jitter performance . . . . .	69
5.1	Chip microphoto . . . . .	72
5.2	Layout of CDR core . . . . .	73
5.3	Layout of reference PLL . . . . .	73
5.4	Measurement setup . . . . .	74
5.5	Measured jitter of the reference PLL . . . . .	77
5.6	Measured waveform of output clock . . . . .	78
5.7	Output jitter vs. delay error of DCDB . . . . .	79
5.8	Measured eye-pattern of transmitted data and retimed data . . . . .	79

# List of Tables

4.1	Parameters of reference PLL . . . . .	61
4.2	Comparison of four CDRs in jitter performance . . . . .	71
4.3	Comparison of four CDRs in power consumption . . . . .	71
5.1	Chip summary . . . . .	80

# ABSTRACT

## A 1.25-Gb/s Digitally-Controlled Dual-Loop Clock and Data Recovery Circuit with Enhanced Phase Resolution

Chang-kyung Seong  
Dept. of Electrical and Electronic Eng.  
The Graduate School  
Yonsei University

In this thesis, a 1.25-Gb/s digitally-controlled dual-loop clock and data recovery circuit is proposed. In the proposed structure, a digitally-controlled delay buffer (DCDB) having 4-level variable delay tunes output phase for higher phase resolution. As a result, phase resolution is enhanced from 64-level to 256-level with little additional power consumption and die area using only the minimum number of reference clocks.

Furthermore, a new compensating technique is presented to linearize phase characteristic of phase interpolator. This improves Integran Nonlinearity(INL) and Differential Nonlinearity(DNL) of digitally-controlled phase interpolator by 71.2% and 55.2%, re-

spectively.

A prototype chip was fabricated with 1-poly 6-metal 0.18  $\mu m$  CMOS technology. In chip measurement, the CDR successfully operated for 1.25-Gb/s  $2^7 - 1$  PRBS with frequency offset tolerance of  $\pm 400$  ppm. The power consumption of the CDR core is 17.8mW with 1.8V supply and core occupies  $255 \times 165 \mu m^2$ .

---

**Key words : Clock and data recovery, Phase interpolation, Phase interpolator linearization**

# Chapter 1

## Introduction

Rapid growth of computation capability of microprocessors has driven the need for wide-band networks and high-speed communication systems. Especially, as the data rate increases, the switch is relied on to route large volumes of data traffic. Overall switch system is generally formed with plugged boards into the backplane as shown in Fig. 1.1. Switch chips soldered on *switch card* connect many pieces of communication equipments each other through *node cards*.

Ethernet standards are formed by two layers: Medium Access Control(MAC) and Physical(PHY) layer. The switching of data traffic is performed in MAC layer while physical transmission and reception are performed in PHY layer. Since most Ethernet controllers separate MAC and PHY chips, a standard is required to define the rule of communication between MAC and PHY chips such as : Medium Independent Interface(MII) standard [2].

Among MII standards used in 1-Gb/s Ethernet, Serial Gigabit MII(SGMII) simplifies interface of MAC-PHY by using only four data I/O pins [1]. Therefore, it is very suitable for switch fabric system suffering from insufficient I/O pins and complex trace routing

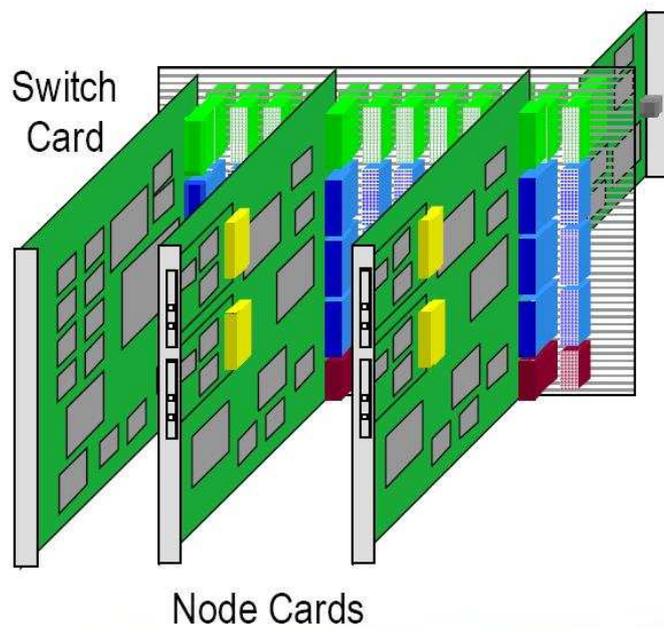


Figure 1.1: Example of switch system : a *switch card* connects many *node cards* each other

in PCB.

In a general high-speed serial link, including SGMII, a data stream is transmitted without clock signal so that transmission is free from phase skew. Consequently, timing information must be extracted from the data to allow synchronous operations in receiver stages. Furthermore, the data must be *retimed* to ensure a timing margin in the followed digital system by removing the jitter accumulated during transmission as illustrated in Fig. 1.2 [6]. In multi-channel applications, low power consumption and small chip area are very crucial issues for CDR since dozens or hundreds of transceivers should be integrated on a single die.

This paper presents a proposed configuration of digitally-controlled dual-loop CDR applicable to multi-channel applications. In Chapter 2, an overview of a conventional digitally-controlled dual-loop CDR are presented along with its problems. Basic ideas and behavioral simulation results of a new CDR are given in Chapter 3. Each block designed in transistor-level is described in Chapter 4. Chapter 5 shows experimental results of the prototype chip. Finally, conclusions are given in Chapter 6.

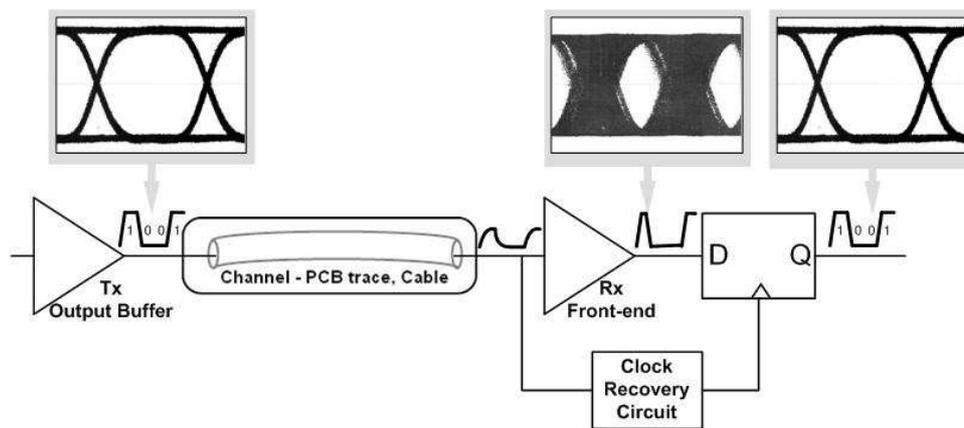


Figure 1.2: Level and timing recovery procedure in general serial link: During transmission, the NRZ data is distorted by band-limited channel. The Rx front-end recovers signal level by amplifying the transmitted waveform and clock recovery circuit extracts timing information from it by generating edge-aligned and jitter-suppressed clock signal. Finally, retimer makes a decision by sampling the amplified data with the recovered clock.

## **Chapter 2**

# **Review of Clock and Data Recovery Circuits for Multi-Channel Applications**

### **2.1 Dual-loop Clock and Data Recovery Circuits**

In the multi-channel serial interface application, numerous CDRs are integrated on single chip. Therefore, each CDR has to be robust to the noise coupled from adjacent blocks and has a low power consumption and small die area.

A dual-loop CDR is one of the most widely used structure in this application. As shown in the Fig. 2.1, the dual-loop CDRs share a common Phase-Locked Loop (PLL), called a reference PLL, and take reference clock signals from it. By using Delay-Locked Loop (DLL), each CDR recovers clock signal which is phase-aligned to the input data. In the clock recovery, phase difference between input data and reference clock is slipped continuously due to a frequency offset. Therefore, a DLL using a phase interpolator(PI) instead of Voltage-Controlled Delay line (VCDL) has to be used in the CDR, since PI

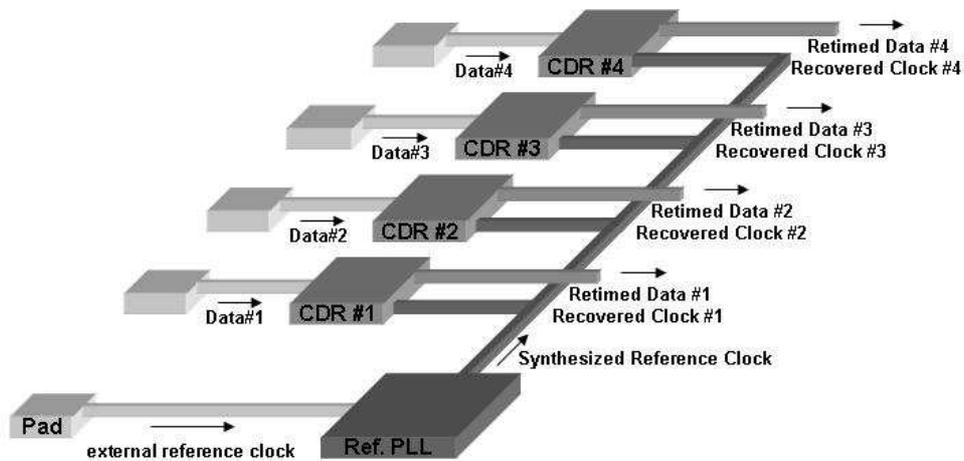


Figure 2.1: Basic configuration of multi-channel dual-loop clock and data recovery circuits

has continuous phase generation capability for the whole range unlike VCDL.

The dual-loop CDR has several advantages over the conventional PLL-based CDR.

First, the dual-loop CDR occupies a smaller die area. In the multi-channel application, it is too bulky to have dozens of PLL-based CDR's are in every channel since the PLL contains a large capacitor in its loop filter. On the other hand, the dual-loop CDR contains no capacitors or only a small one.

Second, the dual-loop CDR has more robustness in jitter performance than the PLL-based CDR. Generally, the CDR circuit and core digital logic in each channel operates at slightly different frequency due to the frequency offset. This means that the circuits generate switching noise of different frequency each other. In this case, jitter performance of the CDR and PLL is severely degraded due to mutual noise coupling [7]. The fact that the PI in the dual-loop CDR does not accumulate jitter as in the case of VCO in the PLL-based CDR is a great advantage in the noisy environments. For these rea-

sons, the dual-loop CDR is more suitable than the PLL-based CDR in the multi-channel application.

The dual-loop CDR is classified by the control method into analog and digitally-controlled type. The digitally-controlled CDR suffers from the quantization error and poor jitter performance since it generates discontinuous phases. However, it is simple, stable and robust to the noise as the circuit is controlled digitally [8][11]. It can be implemented in smaller size than analog-controlled one without a large loop filter. Moreover, it has a infinite open loop gain, so there is no static phase error. On the other hand, the analog-controlled type has better intrinsic jitter performance due to continuous phase generation and faster operation than digitally-controlled type [3][4][5]. Although analog-controlled dual-loop CDR has been investigated and implemented for higher data rates, it is neither simple nor robust to the noise relative to digitally-controlled one. It also generate a static phase error if the open-loop gain of the CDR is not large enough. This is an important drawback since larger open-loop gain means wider loop bandwidth thus less jitter rejection.

Without serious limit in the operating speed, digitally-controlled type is more suitable for data rate of 1.25-Gb/s. Therefore, only the digitally-controlled CDR is considered in this thesis.

## **2.2 Digitally-Controlled Dual-Loop Clock and Data Recovery Circuits**

### **2.2.1 Basic Structure**

A block diagram of the conventional digitally-controlled dual-loop CDR is shown in Fig. 2.2. It consists of a Bang-Bang Phase Detector (BBPD), controller, phase selection circuit and PI. The CDR receives even numbers of equally spaced and uniformly distributed reference phases from a reference PLL. It is assumed that the reference PLL is frequency- and phase-locked to an external reference clock having frequency very close to a fraction of the specified data rate. The phase selection circuit takes two adjacent phases that contain the desired output phase. The PI makes the target phase by interpolating selected two phases. The BBPD compares the phases of the interpolated clock and data so that the controller can produce the control word for the next output phase. It generates "UP" or "DOWN" pulse when the recovered clock has the later or earlier phase than data, respectively. Then, the controller decides the next phase that the phase selection circuit and PI should make together. As a whole, the CDR forms a negative feedback loop and aligns the clock to input data.

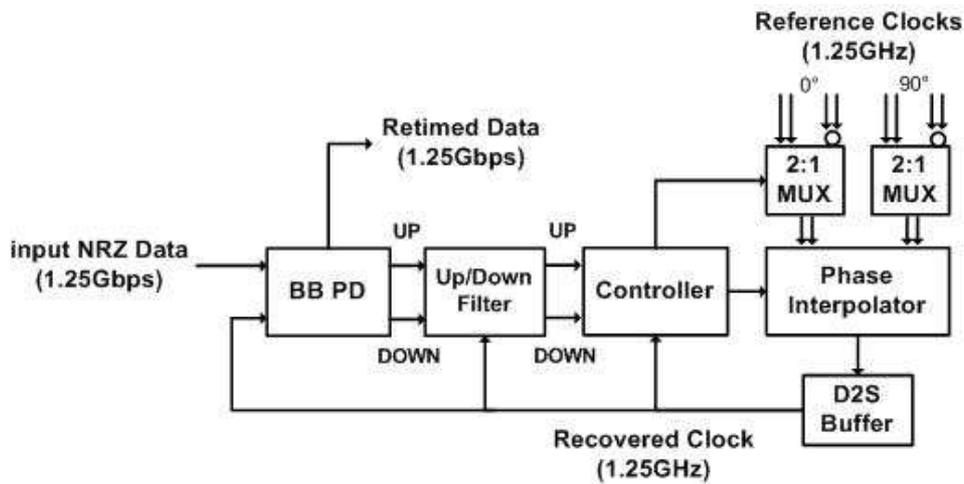


Figure 2.2: Basic structure of conventional digitally-controlled dual-loop clock and data recovery circuit

## 2.2.2 Effect of Phase Resolution on Digitally-Controlled Loop

The phase resolution is the most important factor that determines the dynamics and performance of the CDR. It is related to three issues: jitter generation, jitter suppression and frequency offset tracking ability.

Unlike analog-controlled CDR, the digitally-controlled CDR has non-zero jitter generation even for an ideally clean input data. Since it generates quantized phase, the edge of recovered clock dithers around the edge of input data with the quantization error even in locked state. This quantization error is inversely proportional to the phase resolution. Moreover, some clock latencies in the loop degrade the jitter generation performance [8]. As one more clock latency increases, generally, additional two steps of peak-to-peak phase dithering occurs and the CDR loop becomes more unstable, as described in Fig. 2.3.

In the aspect of the jitter suppression, the phase resolution is directly related to the open loop gain or loop bandwidth. The phase step that the CDR can jump in one clock cycle is determined by the phase resolution. For a higher resolution CDR, the phase movement of the recovered clock is insignificant even if there is jitter on the input data. This means that the recovered clock does not track the input jitter well. Therefore, the higher phase resolution leads a narrower jitter bandwidth and better jitter rejection ability.

Inversely, the CDR with the narrow jitter bandwidth can not track a large frequency offset. Since the transmitter and receiver are synchronized to different external oscillators, they operate at different frequencies. The frequency offset from positive to negative 100ppm is specified in most Ethernet standards. So, the CDR may track the maximum frequency offset of 200ppm in the worst case.

Consequently, the phase resolution of CDR is lower-bounded by the frequency offset tracking ability and upper-bounded by both the jitter generation and suppression performance. Phase resolution should be increased to reduce the jitter generation and reject the input jitter enough, but limited not to lose locking by frequency offset.

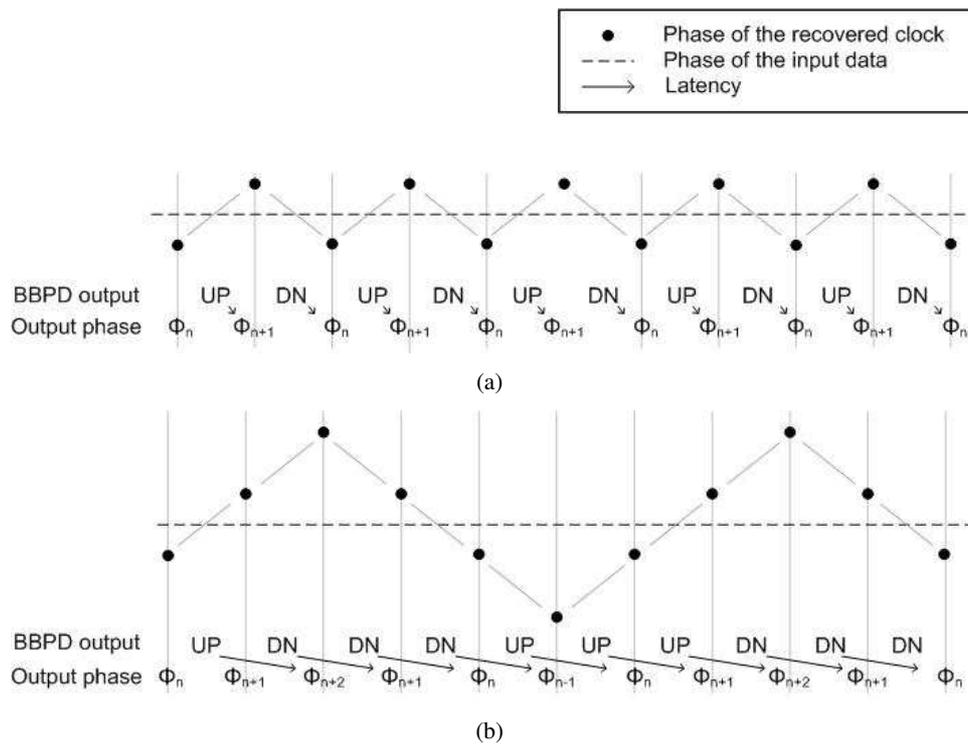


Figure 2.3: Phase dithering in digitally-controlled CDR due to clock latency on the loop (a) no clock latency, dithering of two phase-steps (b) one clock latency, dithering of four phase-steps

The phase resolution of CDR is determined by the number of the reference clock and the phase resolution of PI. Total number of phases is determined as follows.

$$N_{totalphase} = N_{reference\ phase} \times N_{PI\ resolution} \quad (2.1)$$

where  $N_{totalphase}$  is the number of total phase,  $N_{reference\ phase}$  is the number of reference phase from PLL, and  $N_{PI\ resolution}$  is the resolution of PI.

The number of the reference phase is the first factor in designing the system. It is related to the complexity and power consumption of clock distribution from PLL and the allowable slew rate of the reference clock. In the system using a fewer number of the reference clocks, more simple clock distribution and less power consumption are possible, but slower edge of the reference clock is unavoidable. As the phase difference between the adjacent phases clock increases, the slew rate of the reference clock signal should be limited as shown in the Fig. 2.4 since input signals to the phase interpolator has to be overlapped enough. The slower edge makes rectangular signals sensitive to jitters with a large converting-ratio from noise to jitter [9][10]. Thus, 4-phase reference clocks is the best solution in the aspect of complexity, while the worst in robustness to the jitter. Although the 4-phase reference clock is used in this work, the slew rate of the reference clock is reduced only at the last node, as illustrated in Fig. 2.5.

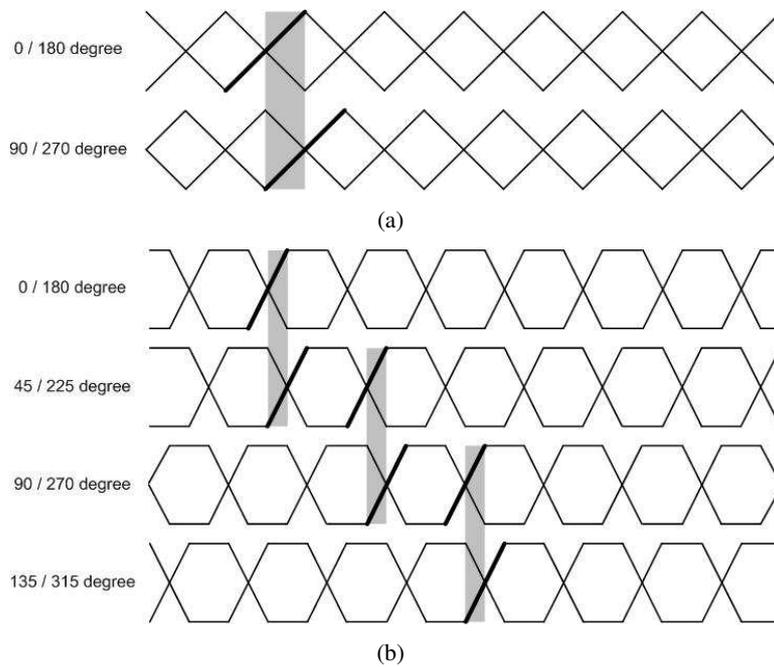


Figure 2.4: Slew rate of reference clock versus number of reference clocks; shadowed regions mean the minimum overlap for smooth interpolation (a) 4-phase reference clock (b) 8-phase reference clock

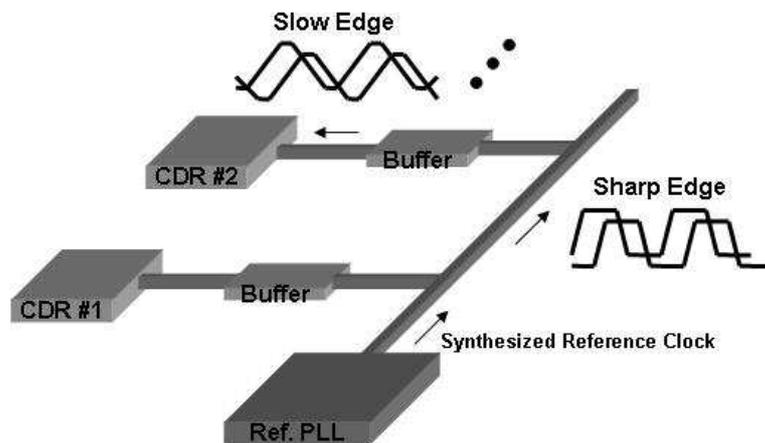


Figure 2.5: Reducing slew rate only at the last node in clock distribution for noise insensitivity

### 2.2.3 Structure and Phase Resolution Limit of Phase Interpolator

The phase interpolation is the weighted sum of two sinusoidal signals, as written in Eq. 2.2.

$$f(t) = \alpha \cos\omega t + \beta \sin\omega t \quad (2.2)$$

where  $f(t)$  is the output signal,  $\omega$  is the clock frequency, and  $\alpha$  and  $\beta$  are weights for two input signals.

Eq. 2.2 forms a single sinusoidal waveform as follows.

$$f(t) = \sqrt{\alpha^2 + \beta^2} \cos(\omega t + \theta) \quad (2.3)$$

where  $\theta$  is the phase of the output signal.

Then,  $\theta$  is determined by the inverse-tangent function of the ratio of  $\alpha$  and  $\beta$ .

$$\theta = \tan^{-1} \frac{\beta}{\alpha} \quad (2.4)$$

or

$$\frac{\beta}{\alpha} = \tan\theta \quad (2.5)$$

In the PI circuit, it is desired that the output phase is linearly controlled by  $\alpha$  and  $\beta$ . Assuming the linear relationship between the output phase and  $\beta$  in each single quadrant.

$$\theta = \frac{\pi}{2}\beta \quad (2.6)$$

where  $0 \leq \beta \leq 1$  and  $0 \leq \theta \leq \frac{\pi}{2}$ .

By substituting Eq. 2.6 into Eq. 2.5,

$$\frac{\beta}{\alpha} = \tan\left(\frac{\pi}{2}\beta\right) \quad (2.7)$$

The right-side of the Eq. 2.7 can be approximated to a fractional expression, as follows.

$$\frac{\beta}{\alpha} \simeq \frac{\beta}{1 - \beta} \quad (2.8)$$

And

$$\alpha \simeq 1 - \beta \quad (2.9)$$

Evidently, the target phase can be obtained by controlling two coefficients complementarily. It is noted that the relationship between the output phase and coefficient is not perfectly linear, however, due to the approximation of tangent to fractional expression, as shown in Fig. 2.6. The phase transfer of real interpolator, i.e. linearly-controlled PI, forms arc-tangent function, obtained by substituting Eq. 2.9 to the Eq. 2.4 inversely as follows.

$$\theta = \tan^{-1} \frac{\beta}{1 - \beta} \quad (2.10)$$

The schematic of conventional PI is shown in the Fig. 2.7. The PI comprises two variable current bias circuits (current DACs), two differential pairs, and two loads for each output node (omitted in the schematic). Current signals driven by input voltage

signals in differential pairs have amplitudes proportional to the bias currents. Since they are partitioned with a certain ratio by the control codes  $\alpha$  and  $\beta$ , weighted-summation or phase mixing is performed as presented in process from Eq. 2.2 to Eq. 2.10.

The current DAC can be constructed in two types : binary-weighted DAC and thermometer DAC. The binary-weighted DAC is more simple and efficient than the thermometer DAC. Using  $N$ -bit control word, binary-weighted DAC can represent  $2^N$  levels of phase while thermometer DAC can represent only  $N + 1$  levels. However, the binary-weighted DAC has a critical drawback in dynamical current-switching. When the Most Significant Bit (MSB) is turned on or off in the binary-weighted DAC, a large current source is activated instantly. It causes a current overshoot and a dynamic phase jump in PI. In the SPICE simulation, the PI with the binary-weighted DAC has a phase discontinuity as shown in the Fig. 2.9. On the other hand, there is no current overshoot in thermometer DAC since it has no large current source. But, it is very bulky especially for higher resolution DAC.

Therefore, it is very difficult to realize a PI which has phase resolution higher than 4-bit, i.e. 16-level, as well as both small-area and good dynamic performance in both types. By using 4-phase reference clocks, total phase resolution of CDR is increased to four times of PI resolution by the Eq. 2.1. So, the 6-bit CDR using 4-bit PI has the minimum phase step of  $5.63^\circ$ . Considering clock latencies more than two cycles by the BBPD and controller, it is not small enough since peak-to-peak self-dithering becomes  $\pm 3$  phase steps, or  $33.75^\circ$ , at least.

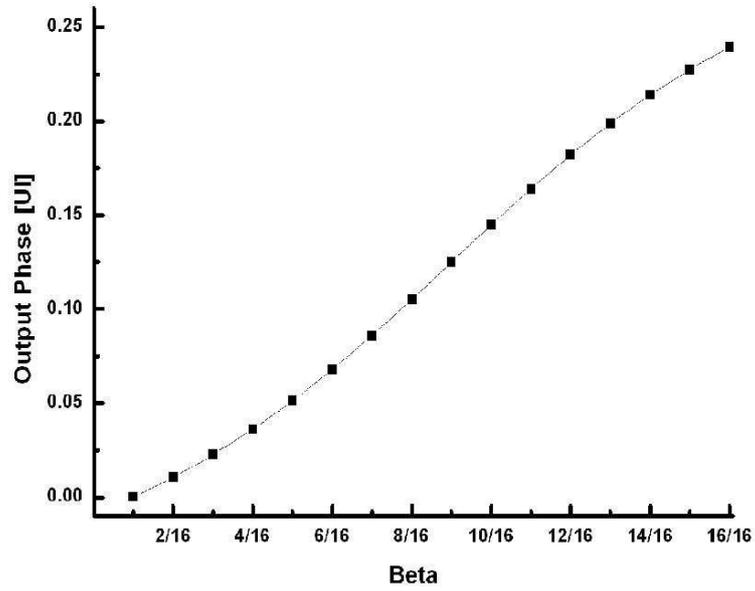


Figure 2.6: Output phase vs. coefficient  $\beta$

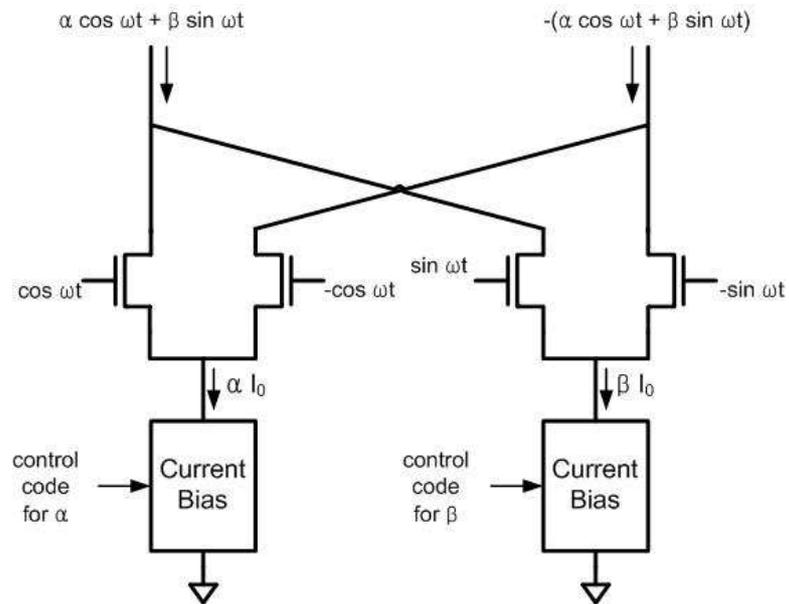


Figure 2.7: Schematic of general phase interpolator circuit

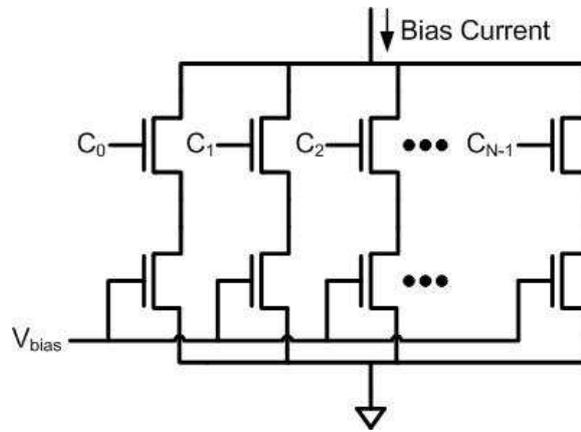


Figure 2.8: Schematic of current digital-to-analog converter using in phase interpolator

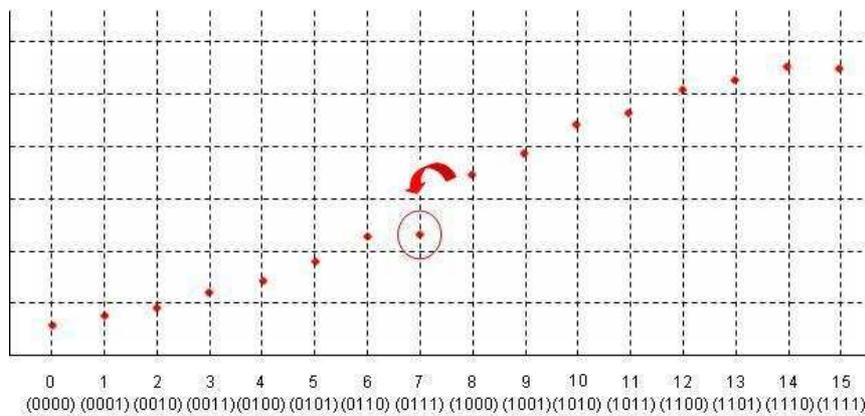


Figure 2.9: Dynamic phase jump of binary-weighted phase interpolator : As the MSB is turned off at the transition from 1000 to 0111, instantaneous phase is broken away from a trace

## 2.3 System-level Design and Behavioral Simulations of CDR using CPPSIM

Designing the digitally-controlled dual-loop CDR contains a lot of system parameters such as phase resolution, filtering length, clock latency in digital block and delay error of DCDB, as well as operating conditions such as frequency offset, input jitter and order of Pseudo-Random Bit Sequence (PRBS), and so on. Since it is difficult that all factors are considered and simulated in circuit-level simulation, the system-level simulation using mathematical models of circuits is very efficient for saving of time and effort.

CPPSIM is a simple, fast and accurate time-step simulator, based on C++ language. By using an interpolated time-step instead of constant one, it performs fine and fast transient simulation with relatively coarse simulation steps [12].

Fig. 2.10 is the schematic of conventional CDR for system-level, or behavioral, simulations. It consists of various models such as a signal source, BBPD, up/down filter, controller, reference PLL, 2:1 multiplexers and PI. Descriptions of each block are as follows.

- Signal source - Provides a periodic data or PRBS.
- BBPD - Constructed with four D-flop and two XOR gate models like a real circuit.
- Up/down filter - Programmed as a counter with a variable filtering length and clock latency. This will be described in the sub-section 2.3.2.

- Controller - Similar to up/down filter. It is characterized with three parameters : bit-width of PI control word , bit-width of DCDB control word and clock latency.
- Phase Interpolator - Described as a weighted-sum of two input signals.
- Reference PLL - Substituted by a oscillator with quadrature sinusoidal outputs. Oscillation frequency is controllable to assume a certain frequency offset.

Except for several simulations with specified parameters, basic simulation setup was constructed as 1.25-Gb/s PRBS input, 200ppm frequency offset, latency of three clock cycles and 2-bit up/down filtering. Simulations are exercised to observe or verify four effects as listed below.

- Functionality of the CDR
- Effect of phase resolution
- Effect of filtering length
- Effect of latency



### 2.3.1 Effect of phase resolution

Phase resolution is directly related to a loop bandwidth, jitter generation and acquisition time of CDR. Intuitively, higher phase resolution means a narrower loop bandwidth, smaller jitter generation and longer acquisition time, vice versa.

Fig. 2.11 shows the simulated acquisition times. For a fair comparison, initial phase errors between data and reference clock should be equal for all cases. Initial phase error was set up as the worst case  $\pi$ . As expected, acquisition time of CDR is roughly doubled as phase resolution is doubled as shown in the figure. Since bandwidth of the loop is inversely proportional to acquisition time, it is confirmed in this result that loop bandwidth becomes narrower as phase resolution increases. Since digitally-controlled CDR is a non-linear system tracking input sinusoidal jitter with some distortions, measuring the loop bandwidth is not easy and ambiguous.

Also, Fig. 2.12 shows the simulation results of jitter generation versus phase resolution. Jitter levels are measured for CDRs having phase resolutions from 6-bit to 9-bit, respectively. In the figure, RMS jitter generation is inversely proportional to the phase resolution.

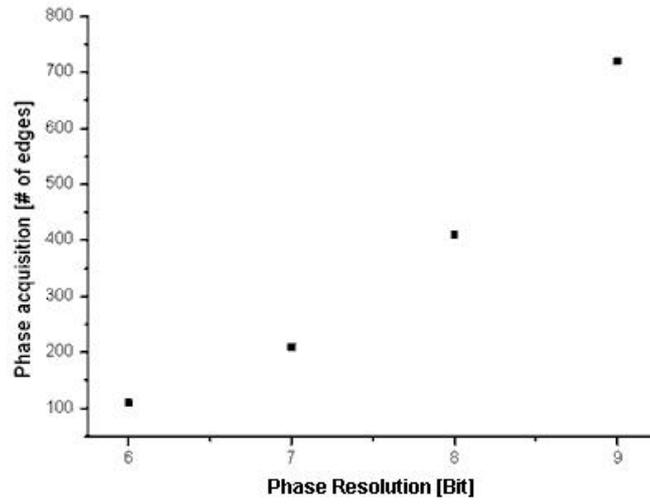


Figure 2.11: Acquisition time vs. phase resolution

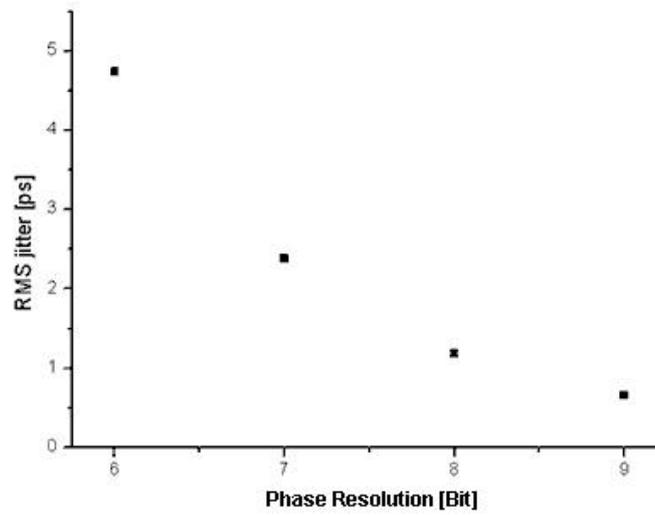


Figure 2.12: RMS jitter generation vs. phase resolution

### 2.3.2 Effect of counting number of up/down filter

The up/down filter disturbs unwanted phase movements by rejecting some up or down pulses from the BBPD. In this work,  $N$ -bit up/down filter is defined as a filter that generates up or down pulse only when  $N$ -consecutive up or down pulses are occurred by BBPD. For example, Fig. 2.13 (a) and (b) show movements of control code without filter and with filter respectively. By using a simple filter, dithering of control code is considerably rejected.

To investigate effect of counting number of filter, jitter generations of CDRs are measured for various cases. As shown in Fig. 2.14, three kinds of CDRs with six filters are simulated. It is noticed that up/down filter is especially useful for CDR having relatively low phase resolution. Also, although CDRs with longer counting number of filter generates less jitter, degree of enhancement is insufficient for relatively long counting as much as short counting.

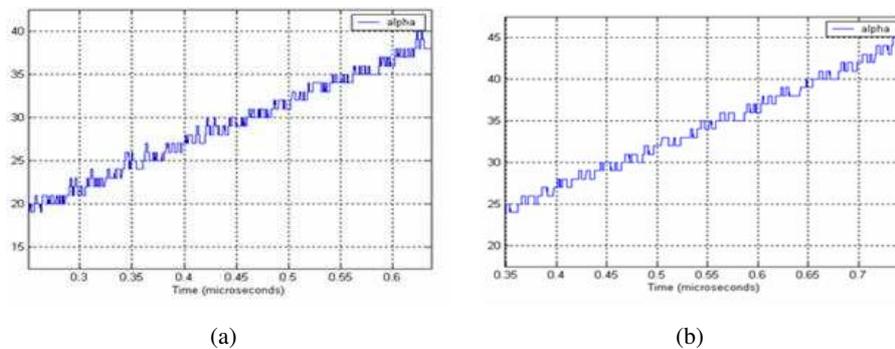


Figure 2.13: Movement of control code during tracking frequency offset in locked state (a) without up/down filter (b) with 2-bit length up/down filter

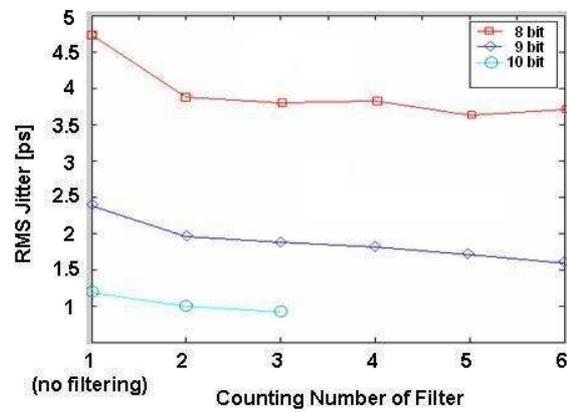


Figure 2.14: RMS jitter generation of CDRs for various combinations of phase resolution and filtering order

### 2.3.3 Effect of latency

Latency in BBPD and controller unstabilizes loop dynamics. Due to latency, the phase information generated by phase detector is lately updated and effects to the phase making circuits such as phase selection and interpolator circuits. As mentioned in Section 2.2.2, phase dithering is proportional to the number of cycles of latency. Fig. 2.15 shows jitter generation performance for various values of latency and filtering lengths. As clock latency increases, jitter performance is explicitly degraded in the case of no filtering represented as rectangular points. However, it is noticed that some filtering makes loop insensitive to the latency as well as improves jitter generation performance.

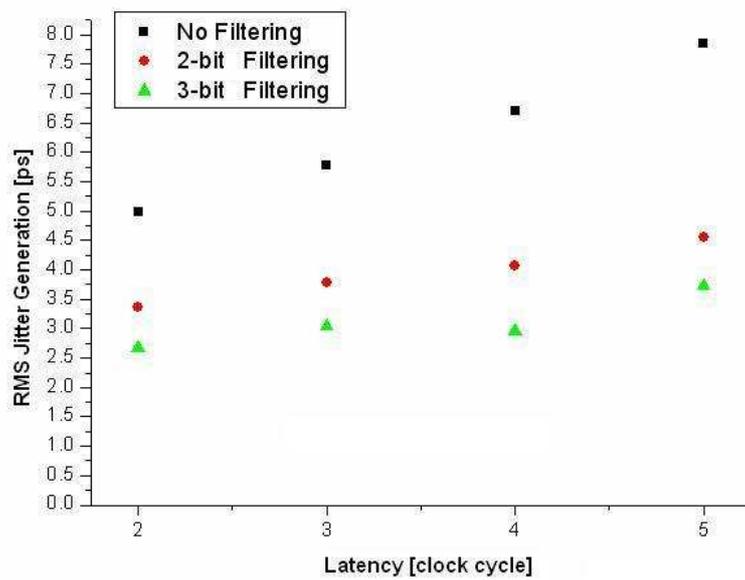


Figure 2.15: RMS jitter generation of CDR for various latencies and effects of filtering on it

## Chapter 3

# Proposed Digitally-Controlled Dual-Loop Clock and Data Recovery Circuit

As mentioned in the previous chapter, conventional PIs have two major problems: non-linear phase transfer and limited phase resolution. In this paper, new methods are proposed to overcome these problems as follows.

- Enhancing phase resolution of CDR by inserting digitally-controlled delay buffer to the loop
- Linearizing phase interpolator by compensating its bias current DACs

A new structure of digitally-controlled dual-loop CDR operating at 1.25-Gb/s is presented in Fig. 3.1. As a whole, the dual-loop system consists of reference PLL and CDR core. The CDR core receives two differential quadrature phase clocks from the reference PLL. Two 2:1 MUXs make up two adjacent phases that contain the desired phase by selecting inverted or non-inverted version of the reference clocks. The target phase is

selected, or mixed, by the combination of PI and DCDB. The PI having 16-level resolution is controlled by 15-bit thermometer code to avoid dynamic phase overshoot. It is too bulky to implement higher phase resolution than 16-level, i.e. 32-level or 64-level and so on, in the aspect of both die area and complexity in layout of PI and controller. Explanations about DCDB and its function will be presented in the next Section. The 2-bit or 3-bit up/down filter after the BBPD reduces unwanted phase dithering by generating output pulses only when two and three consecutive UP or DOWN pulses are occurred [11], respectively.

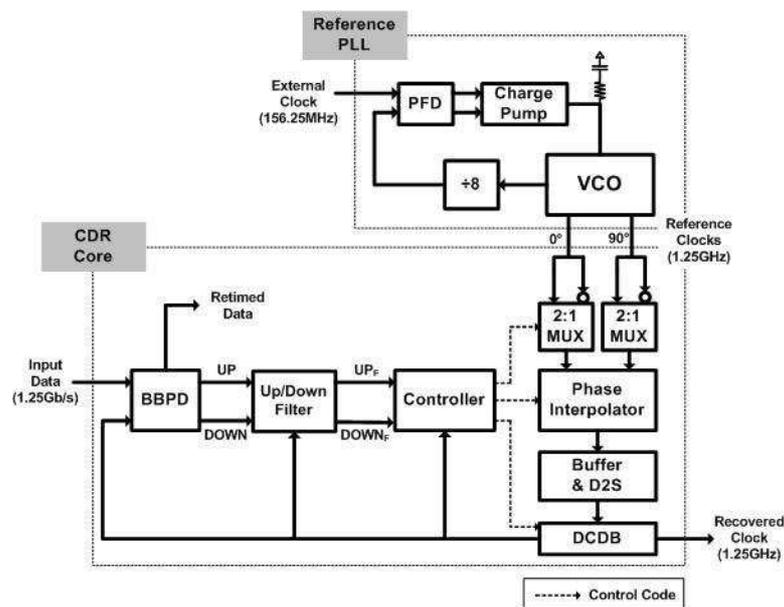


Figure 3.1: Block diagram of the proposed CDR

### 3.1 Phase Resolution Enhancement

A basic concept of the proposed method is shown in Fig. 3.2. By using DCDB having a variable delay of four levels, phase of interpolated clock is finely tuned again. As a result, total phase levels of CDR are increased from Eq. 2.1 to Eq. 3.1 by multiplying  $N_{DCDB \text{ resolution}}$ .

$$N_{totalphase} = N_{reference \ phase} \times N_{PI \ resolution} \times N_{DCDB \ resolution} \quad (3.1)$$

where  $N_{totalphase}$ ,  $N_{PI \ resolution}$  and  $N_{DCDB \ resolution}$  are available total phase levels of CDR, resolution of PI and DCDB, respectively, and  $N_{reference \ phase}$  is a number of reference phases from PLL.

By inserting DCDB having 4-level variable delay, 256 phase levels are achieved even with only 4-phase reference phases. In the past works, more reference phases have been used to make a higher resolution. For example, 12-phases reference clock was used to make 192 levels of phase in [13] and 8 phases was used to make 64 levels of phase in [14].

Unfortunately, it is not guaranteed that the DCDB provides the exact amount of desired delay due to variations of process, supply voltage, and temperature, or PVT variations. When the DCDB delay is different from the desired value, the combined phase transfer curve could be even non-monotonic as well as nonlinear as illustrated in Fig. 3.3. In the figure, large and small black circles correspond to the normal output phases of PI and DCDB, and crosses and diapers correspond to slipped output phases

of DCDB with +50% and -50% error, respectively. Shadowed regions are where the phase transfer curve suddenly changes. In this work, delay error of DCDB is defined as follows.

$$Err_{DCDB}(\%) = \frac{\Delta\phi_{Slip} - \Delta\phi_{Normal}}{\Delta\phi_{Normal}} \times 100 \quad (3.2)$$

where  $Err_{DCDB}$  is an defined delay error of DCDB,  $\Delta\phi_{Slip}$  is a slipped phase step due to variations, and  $\Delta\phi_{Normal}$  is a normal phase step.

Considering PVT variations, there may be delay errors in DCDB within  $\pm 30\%$  by careful circuit-design and layout.

Fig. 5.7 shows RMS jitter generation versus delay error of DCDB in behavioral simulation. Three horizontal lines correspond to output jitter of conventional 6-bit, 7-bit and 8-bit CDR using only 4-bit, 5-bit and 6-bit PI each. And, jitter generation of the proposed CDR are measured for delay error in the range from -50% to 100%. As a result, measured jitter performance of the proposed CDR is comparable to conventional 8-bit CDR for relatively wide range of DCDB error.

Although there can be sudden phase variations at the edge of two interpolated phases with DCDB errors shown in shadowed region in Fig. 3.3, the entire effective phase resolution is increased. In the inverse slope due to large positive errors, the phase will jumps to the opposite direction from input data phase. However, since the effect of increased phase resolution is more dominant than that of local phase fluctuation, the total jitter generation performance is improved.

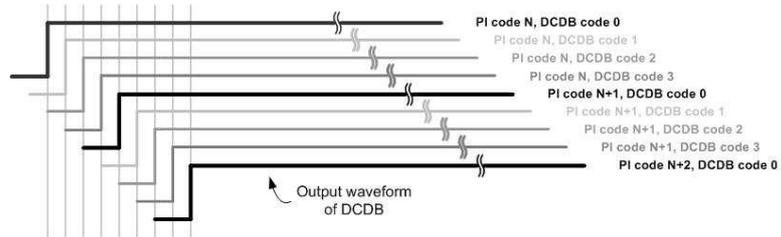


Figure 3.2: Concept of phase resolution enhancement using digitally-controlled delay buffer

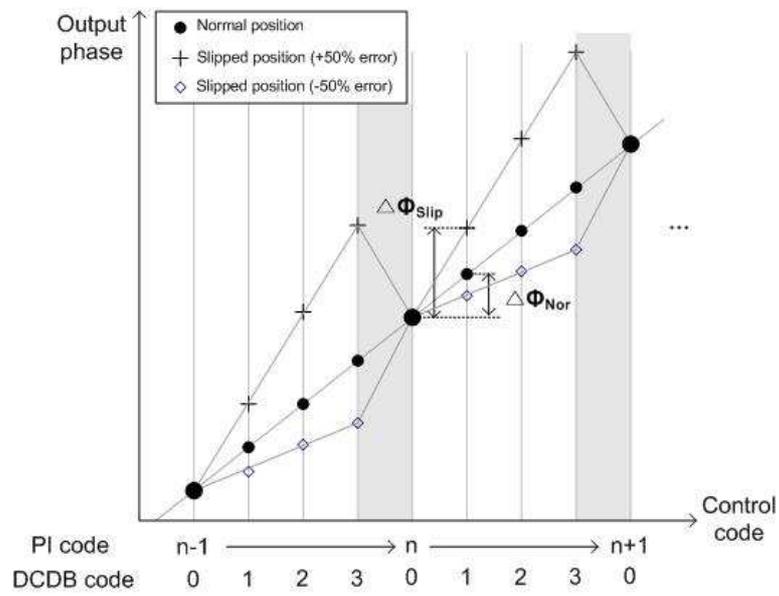


Figure 3.3: Concept of DCDB delay error

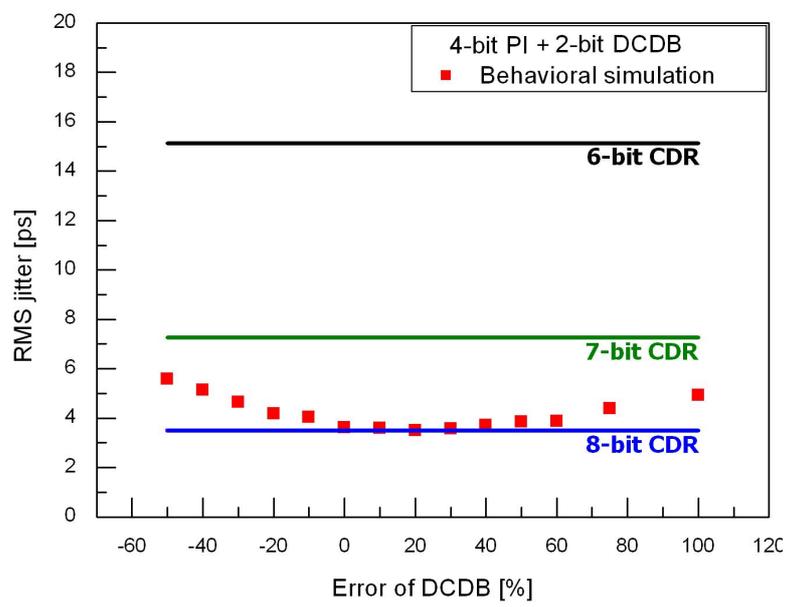


Figure 3.4: RMS Jitter generation vs. DCDB delay error

## 3.2 Phase Interpolator Linearization

By the linear control, transfer curve of PI becomes non-linear as shown in Fig. 3.5. As a slow and steep slope are repeated in each quadrant region, interpolated phases are fluctuated around the ideal straight line. This makes an effective phase resolution decrease and jitter performance degraded slightly because of some large phase steps in steep slopes. It is not a crucial problem in clock recovery applications since input jitter is more dominant than jitter generation of CDR loop. The proposed scheme in this thesis to enhance phase resolution, however, requires more linear transfer curve to reduce non-monotonic points. A new method is proposed to linearize transfer curve of PI by distorting the linearity of bias current DACs.

Fig. 3.6 (a) shows the simplified schematic of the conventional thermometer-type current bias DAC. It consists of  $N$  identical switches and current sources to represent  $N+1$  levels. As switches are turned on one by one, total bias current is linearly increased as illustrated in Fig. 3.6 (b).

To compensate the non-linear phase transfer curve, a nonlinear current DAC with gradually varying-sized current sources can be used. Assume two functions  $f(\alpha)$  and  $f(\beta)$  to compensate the transfer characteristic of PI. Then, they can be substituted in Eq. 2.7 instead of  $\alpha$  and  $\beta$ .

$$\frac{f(\beta)}{f(\alpha)} = \tan \frac{\pi}{2} \beta \quad (3.3)$$

For a convenience, assume that new compensating functions are also complementary.

$$\frac{f(\beta)}{1 - f(\beta)} = \tan \frac{\pi}{2} \beta \quad (3.4)$$

By solving Eq. 3.4, compensating function  $f(\beta)$  and  $f(\alpha)$  are obtained as follows.

$$f(\beta) = \frac{1}{1 + \cot \frac{\pi}{2} \beta}, \quad f(\alpha) = \frac{\cot \frac{\pi}{2} \beta}{1 + \cot \frac{\pi}{2} \beta} \quad (3.5)$$

In fact, the phase transfer of conventional PI having arc-tangent function with argument of fractional expression as written in Eq. 2.10 can be approximated to two piece-wised second-order curve. For a simplicity, second-order approximation is used in this work. Fig. 3.7 (b) shows the concept of second order code-to-current function with a linear increment.

Each curve in Fig. 3.8 is the simulated phase transfer of conventional and linearized PI by using HSPICE. The linearized PI is more linear than conventional one as expected. For quantitative comparison, Integral Non-Linearity(INL) and Differential Non-Linearity(DNL) are measured. INL means a distance function from ideal phases, so INL curve could be plotted by subtracting ideal phase function from transfer curve. On the other hand, since DNL means distortion of phase steps, DNL curve could be plotted by differentiating transfer curve. In the Fig. 3.9 (a) and (b), INL and DNL of two PIs are shown. In the plot (a), the difference of the maximum and minimum INL is reduced from 27 ps to 7.8 ps, or by 71.2%. Also, in the plot (b) DNL is improved from 11.6 ps to 5.2 ps, or by 55.2%.

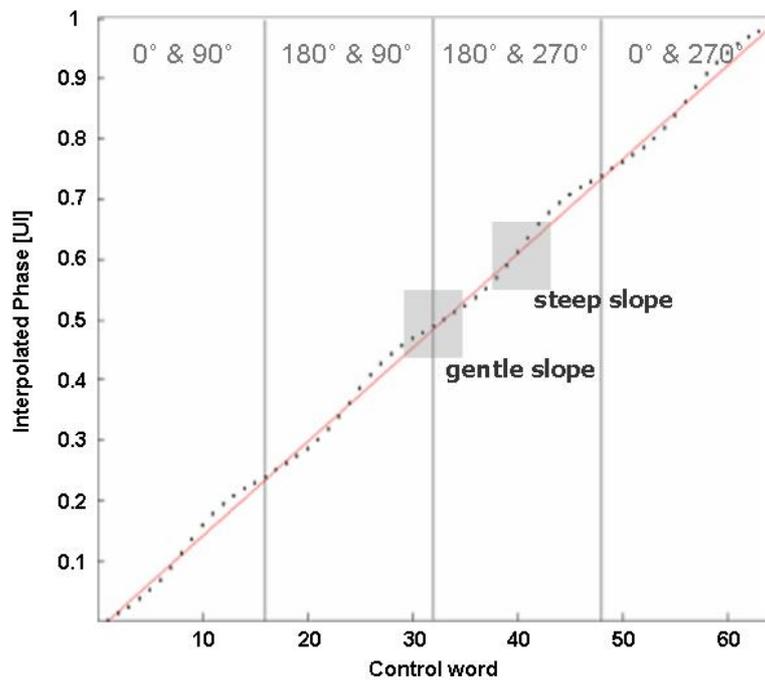
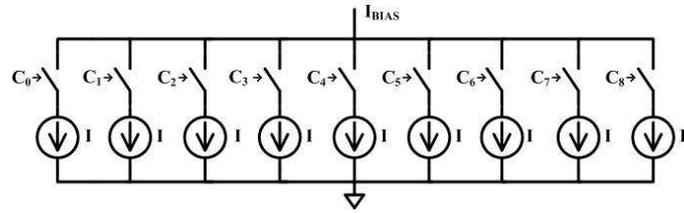
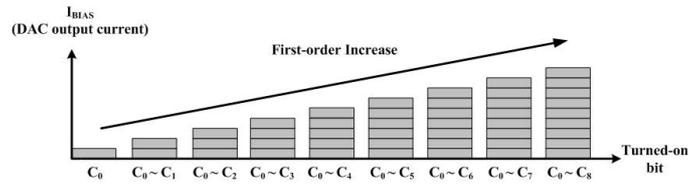


Figure 3.5: Non-linear transfer curve of conventional phase interpolator

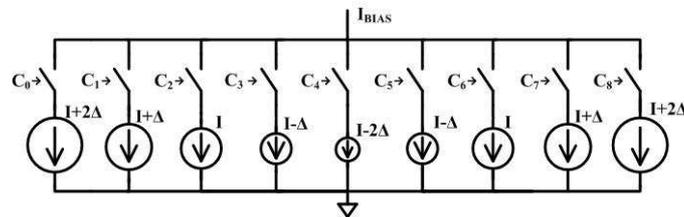


(a)

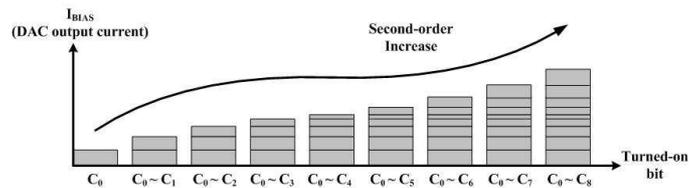


(b)

Figure 3.6: Linear DAC (a) schematic (b) First-order current increase



(a)



(b)

Figure 3.7: Non-linear DAC (a) schematic (b) Second-order current increase

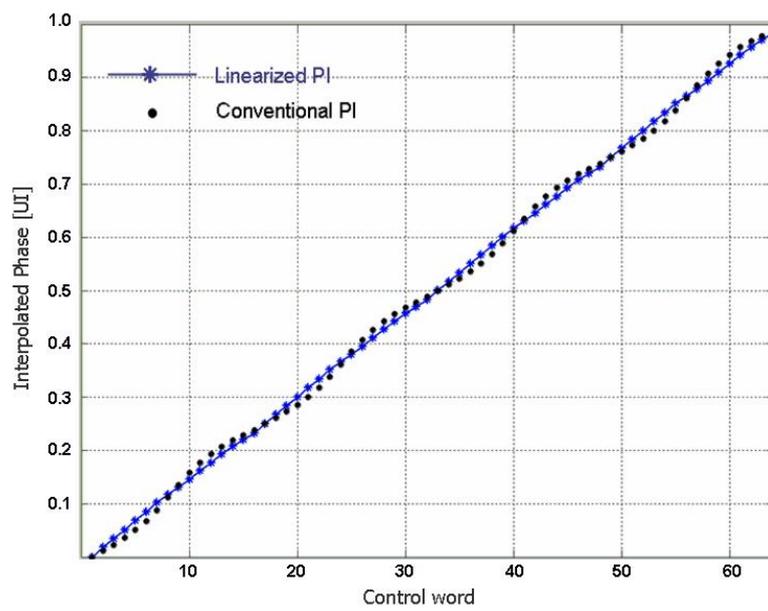
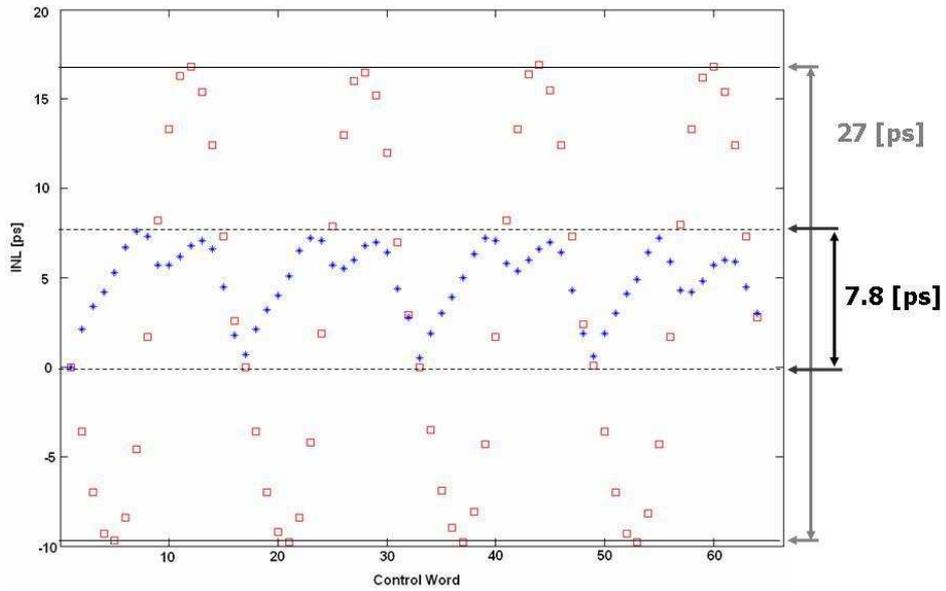
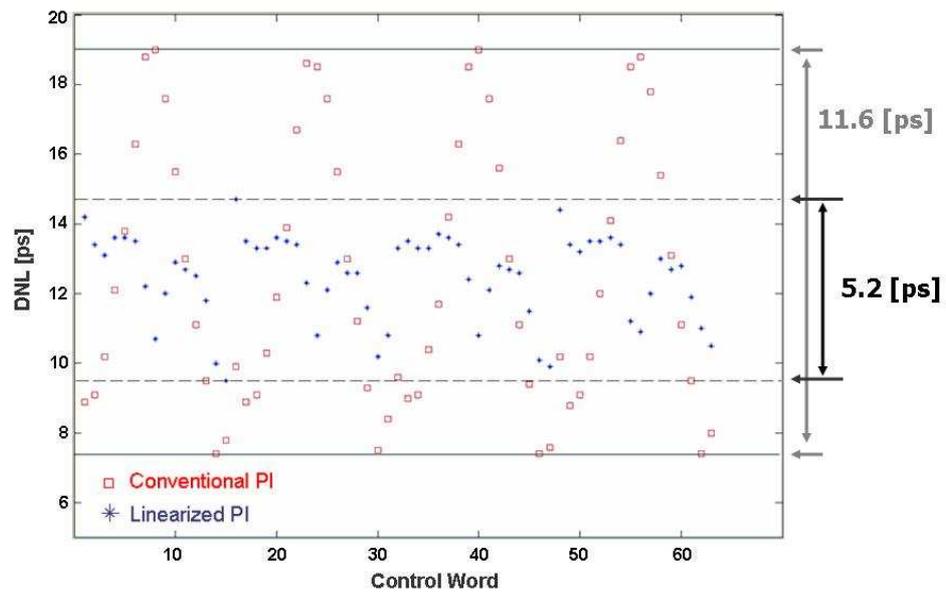


Figure 3.8: Comparison of transfer curve : Conventional vs. linearized phase interpolator



(a)



(b)

Figure 3.9: Comparison of nonlinearity : Conventional vs. linearized phase interpolator  
(a) Integral Nonlinearity (b) Differential Nonlinearity

## Chapter 4

# Circuit Design and Simulation Results

### 4.1 Circuit-level Design of Clock and Data Recovery circuit

The prototype circuit of the proposed CDR was designed by using  $0.18\mu m$  CMOS technology. A conventional CDR having 6-bit phase resolution with 4-bit PI alone is also designed for comparison. They have very similar structures except for the controller block and DCDB. Details for each blocks will be described in sub-sections followed.

#### 4.1.1 Phase Interpolator

Phase interpolator is the most important block in the dual-loop CDR. It is desired for PI to have three characteristics : limited slew rate of input clock, no phase overshoot, and linear phase transfer.

First, relatively large transistors are used for input differential pairs to limit slew rate of input clocks. Since the PI should be driven by MUXs in real circuits, the slew rate was observed by cooperating PI with MUX in the SPICE simulation. Second, bypass ca-

capacitors are attached to each bias current DACs to reduce current overshoot in switching process. Finally, bias current sources were designed as nonlinear DACs to compensate nonlinearity of PI as mentioned in Section 3.2. The current DAC consists of sixteen current sources and switches as shown in the schematic in Fig. 4.1. The channel width of current sources are increased step by step by  $0.3\mu m$  from the center. The increment was obtained by trial and error in simulation.

*offset* is single-bit code to generate an offset phase. The offset phase is necessary to avoid generating identical phases at the boundaries of quadrants. Two cases with and without *offset* are compared in Fig. 4.2.

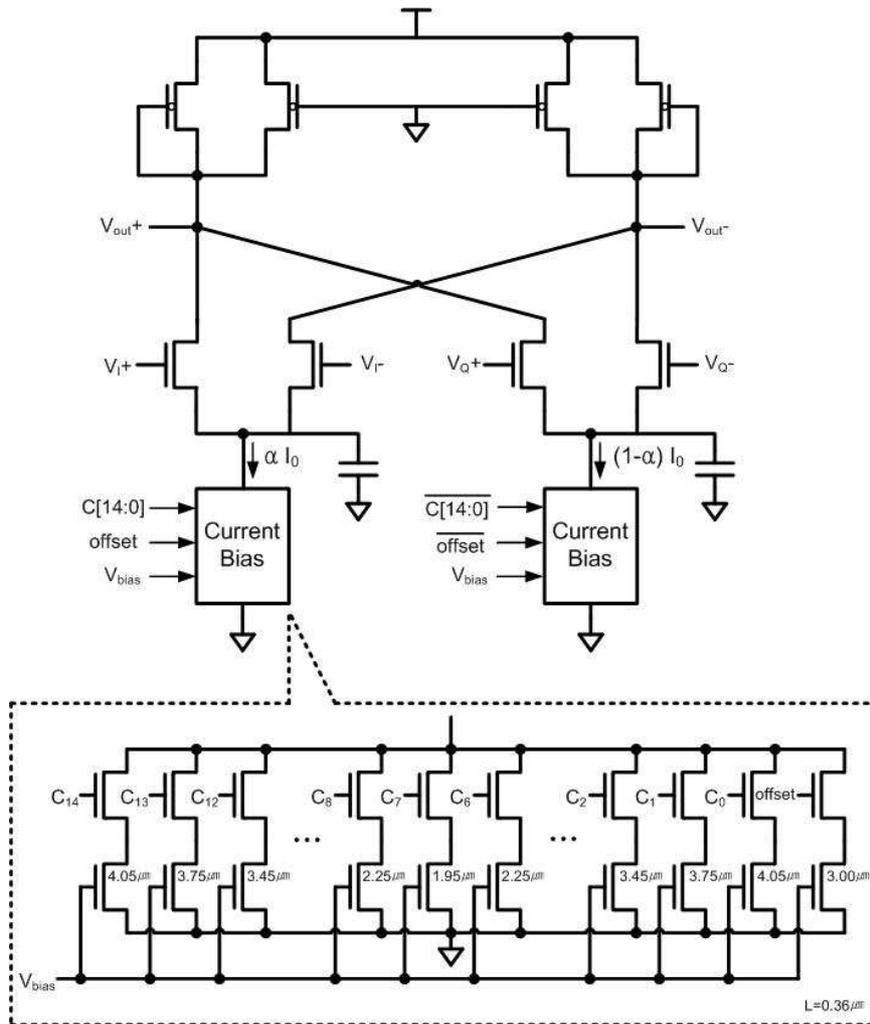


Figure 4.1: Schematic of linearized phase interpolator

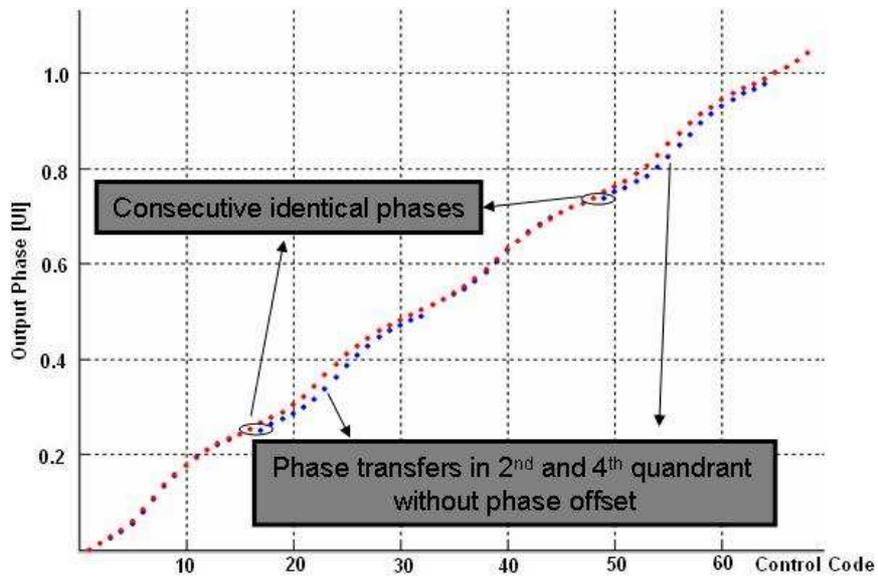


Figure 4.2: Phase transfer curve of PI with and without without phase offset

#### 4.1.2 Phase Controller

The phase controller generates the digital codes to make the target phase. It comprises the MUX controller, PI controller and DCDB controller. Three controllers are connected serially in order of the DCDB controller, the PI controller and the MUX controller, as shown in Fig. 4.3.

First, the DCDB controller is a 2-bits binary up / down counter. It receives a UP and DOWN pulse from the previous block, i.e. up/down filter, and takes a counting upward for a UP pulse or downward for a DOWN pulse as shown in Fig. 4.4. If there is no UP or DOWN pulse from the up/down filter, it is idle state and holds the previous state on. By using D-flip-flops with hold operation, the logics can be simplified since additional idle states could not be considered. In fact, total number of the states is reduced from

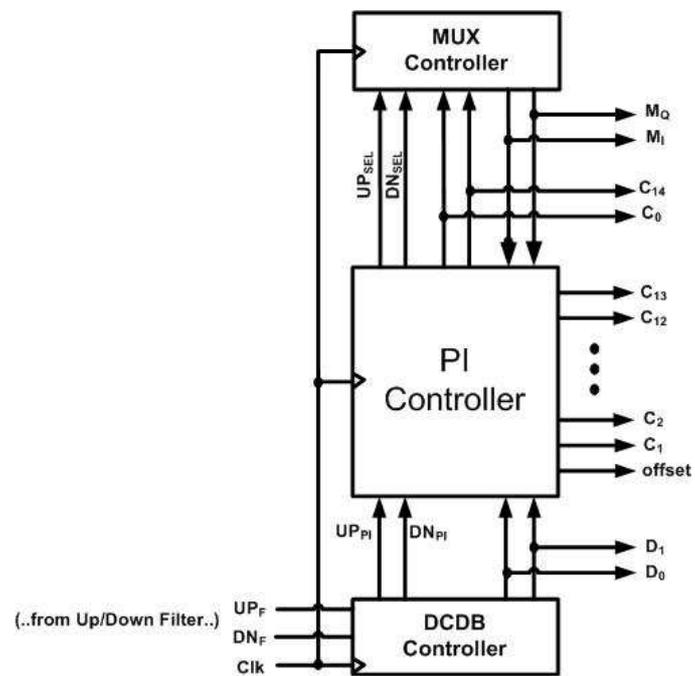


Figure 4.3: Configuration of three controllers

eight to four.

The PI controller is a 15-bit bidirectional shift register having a state-flow shown in Fig. 4.6. Related to MUX controller, it shifts words to the MSB-direction by inserting "1" to the LSB, or shifts words to the LSB-direction by inserting "0" to the MSB. The direction of shifting is changed by the presently selected quadrant as summarized in Fig. 4.7. To avoid the same phase in boundaries of two adjacent quadrants as gray-colored in the figure, the *offset* bit is activated and adds some offset coefficient, i.e. some bias current are added in the PI, in even-quadrants. The PI controller is implemented using serially chained fifteen D-flip-flops and some combinational CMOS logic gates. To reduce logic complexity, a simple CMOS gate that functions  $OUT = \sim (A \cdot B + C)$  is used instead of combining conventional logic gates. The schematic of PI controller is shown in Fig. 4.8.

The MUX controller is a kind of 2-bit up/down gray-coded counter. Gray-coded control word makes two MUXs determine the quadrant sequentially by selecting a inverted or non-inverted version of the reference phases. Its state-diagram, quadrant mapping table, and schematic are shown in Fig. 4.9 and Fig. 4.10 each.

Since the phase controller limits the operating speed of the system, design and verification are performed carefully in the aspect of timing margin. The simulation results of entire controller are presented in Fig. 4.11. The Fig. 4.11 (a) and (b) shows the operation of controller for consecutive UP and DOWN pulses, respectively. Function is verified for three variation corner of FF,TT and SS without any timing violations.

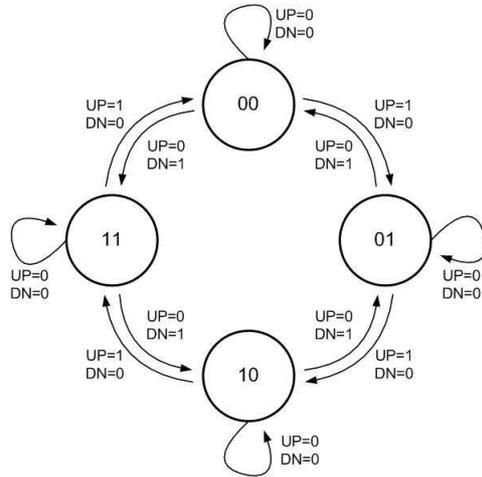


Figure 4.4: State-diagram of DCDB controller; 2-bit binary Up-Down counter

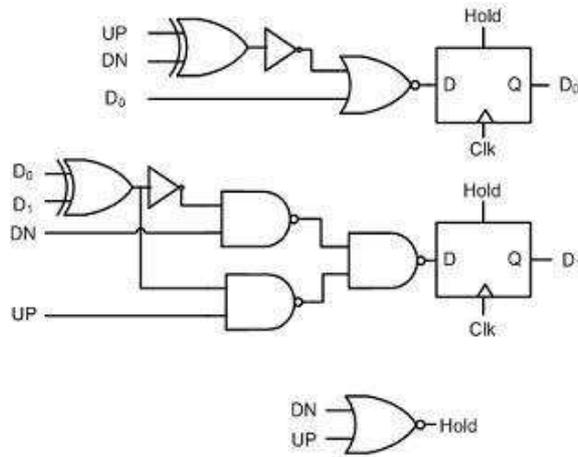


Figure 4.5: Schematic of DCDB controller

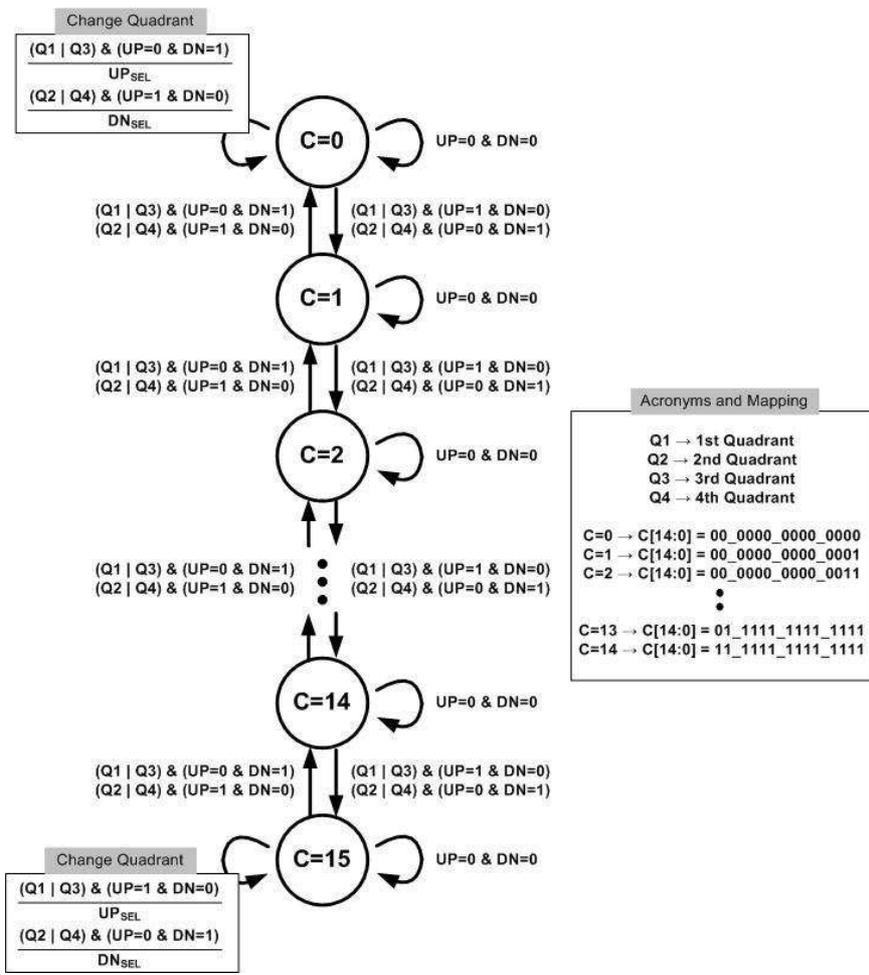


Figure 4.6: State-diagram of PI controller; 15-bit bidirectional shift register

Quadrant	M <sub>1</sub>	M <sub>0</sub>	C[0]	C[1]	C[2]	C[3]	C[4]	C[5]	C[6]	C[7]	C[8]	C[9]	C[10]	C[11]	C[12]	C[13]	C[14]	Offset	
Q1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
			.....																
			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
Q2	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	
			.....																
			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Q3	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
			.....																
			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	
Q4	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	
			.....																
			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Figure 4.7: Flow-table of PI controller; 15-bit bidirectional shift register

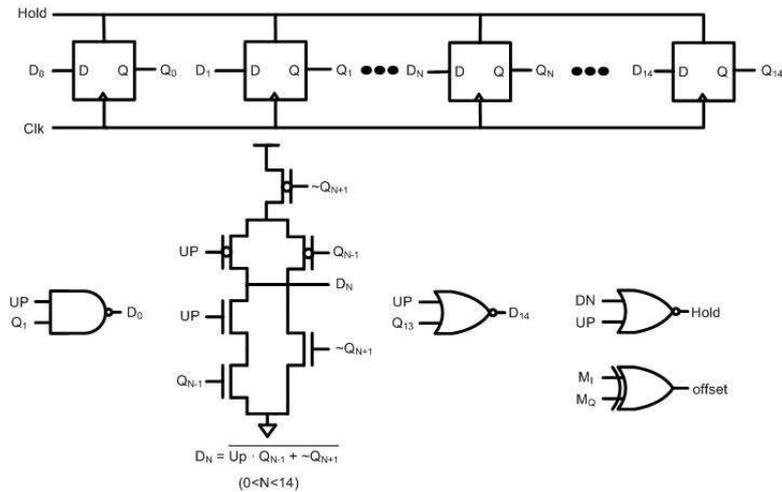
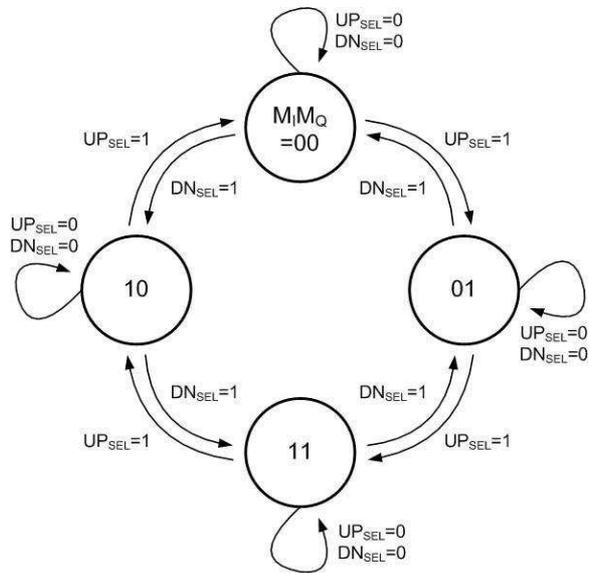


Figure 4.8: Schematic of PI controller



Quadrant Mapping				
M <sub>i</sub>	M <sub>Q</sub>	Selected phase by M <sub>i</sub>	Selected phase by M <sub>Q</sub>	Quadrant
0	0	0	90	1st
1	0	180	90	2nd
1	1	180	270	3rd
0	1	0	270	4th

Figure 4.9: State-diagram and Quadrant mapping of MUX controller

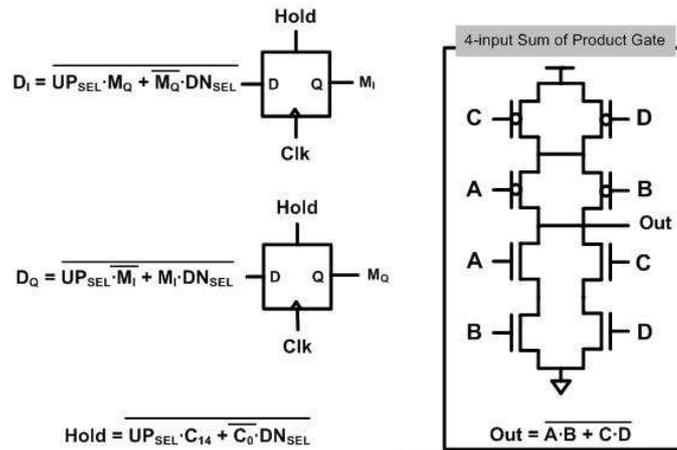
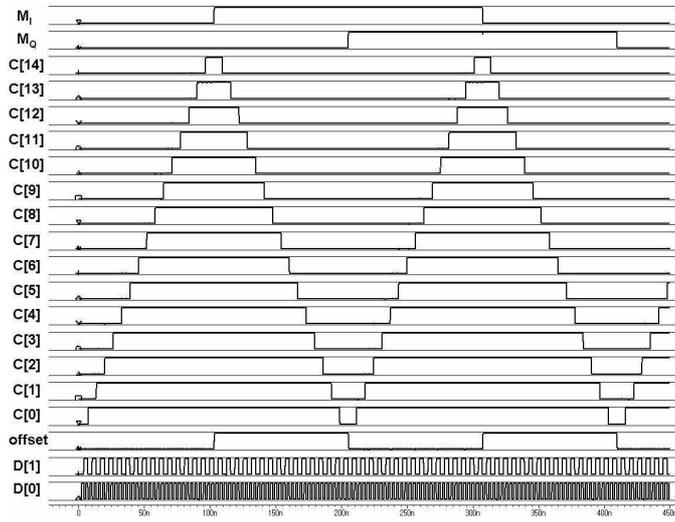
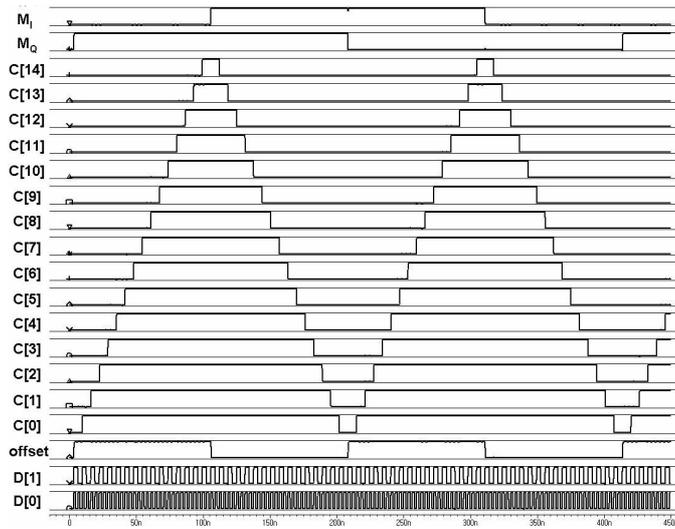


Figure 4.10: Schematic of MUX controller and 4-input Sum of Product gate



(a)



(b)

Figure 4.11: Operation of controller in  $2\pi$  range (a) for UP pulse (b) for DOWN pulse

### 4.1.3 Up/Down Filter

The up/down filter makes the controller free from burden of operating speed by eliminating consecutive operations as well as filters out some phase information. Two kinds of up/down filters are designed and simulated in this work, 2-bit and 3-bit up/down filters. They are 4-bit and 6-bit bidirectional shift registers each. For the UP pulse from BBPD, it shifts the word contained in register to the right by inserting "1" to the leftmost bit, vice versa for the DOWN pulse, as shown in Fig. 4.12(a) and (b). In the schematics of Fig. 4.13, six-input *sum of product* expressions such as  $OUT = \sim (A \cdot B \cdot C + D \cdot E \cdot F)$  are appeared frequently. For simplicity, they are implemented with a single gate as shown in Fig. 4.14.

Simulated waveforms of 2-bit and 3-bit up/down filter are presented in Fig. 4.15. It is observed that UP and DOWN pulses are randomly appeared with comparable frequencies in locked state of CDR. Also, less UP and DOWN pulses are survived in 3-bit filter than 2-bit filter due to larger counting number.

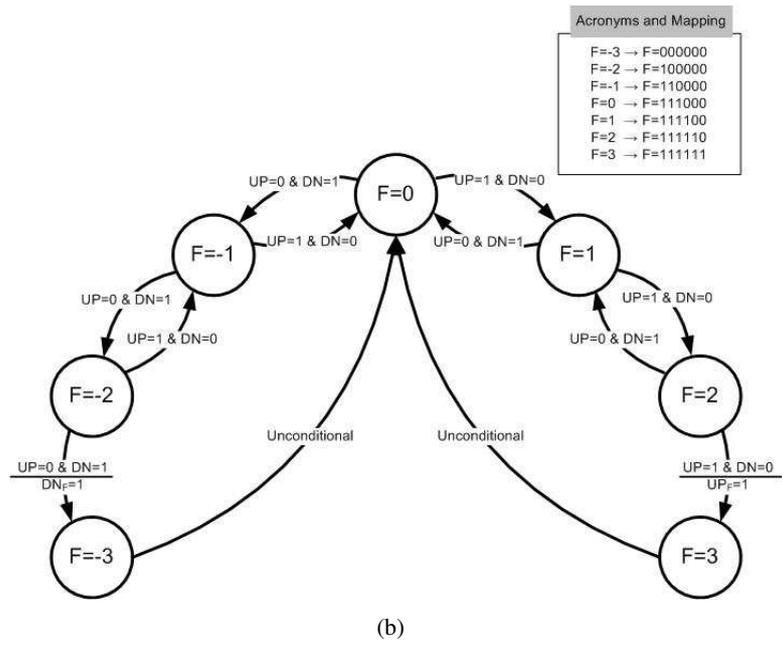
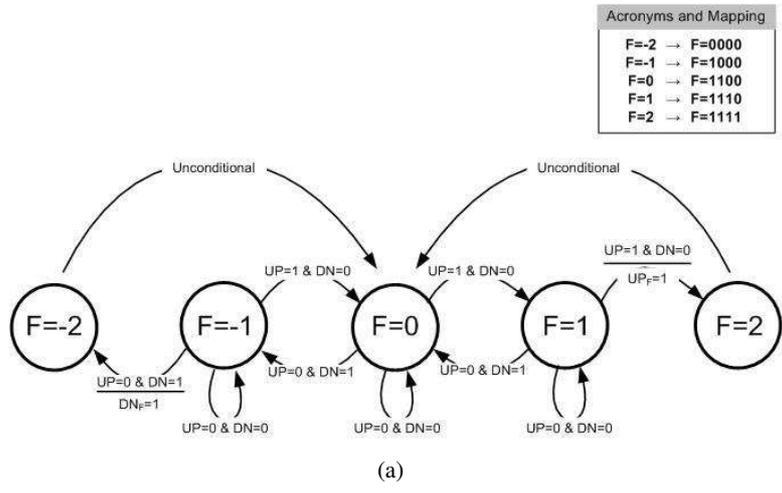
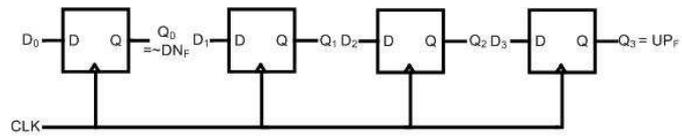


Figure 4.12: State-diagram of Up/Down filter (a) 2-bit filter (b) 3-bit filter



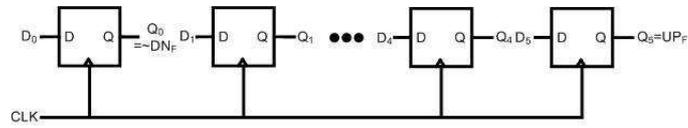
$$D_0 = \overline{DN \cdot Q_0 \cdot \sim Q_1} = \overline{DNF}$$

$$D_1 = \overline{DN \cdot Q_1 \cdot \sim Q_2 + \sim UP \cdot Q_0 \cdot \sim Q_1}$$

$$D_2 = \overline{\sim DN \cdot Q_2 \cdot \sim Q_3 + UP \cdot Q_1 \cdot \sim Q_2}$$

$$D_3 = \overline{UP \cdot Q_2 \cdot \sim Q_3} = \overline{UPF}$$

(a)



$$D_0 = \overline{DN \cdot Q_0 \cdot \sim Q_1} = \overline{DNF}$$

$$D_1 = \overline{DN \cdot Q_1 \cdot \sim Q_2 + \sim UP \cdot Q_0 \cdot \sim Q_1}$$

$$D_2 = \overline{DN \cdot Q_2 \cdot \sim Q_3 + \sim UP \cdot Q_1 \cdot \sim Q_2 + Q_0 \cdot \sim Q_1}$$

$$D_3 = \overline{\sim DN \cdot Q_3 \cdot \sim Q_4 + UP \cdot Q_2 \cdot \sim Q_3 + Q_1 \cdot \sim Q_2}$$

$$D_4 = \overline{\sim DN \cdot Q_4 \cdot \sim Q_5 + UP \cdot Q_3 \cdot \sim Q_4}$$

$$D_5 = \overline{UP \cdot Q_4 \cdot \sim Q_5} = \overline{UPF}$$

(b)

Figure 4.13: Schematic of Up/Down filter (a) 2-bit filter (b) 3-bit filter

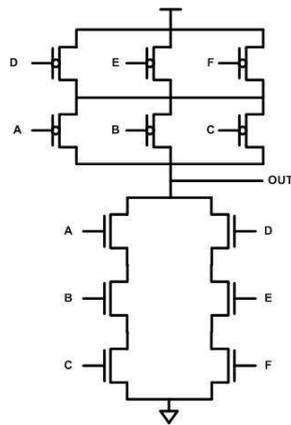
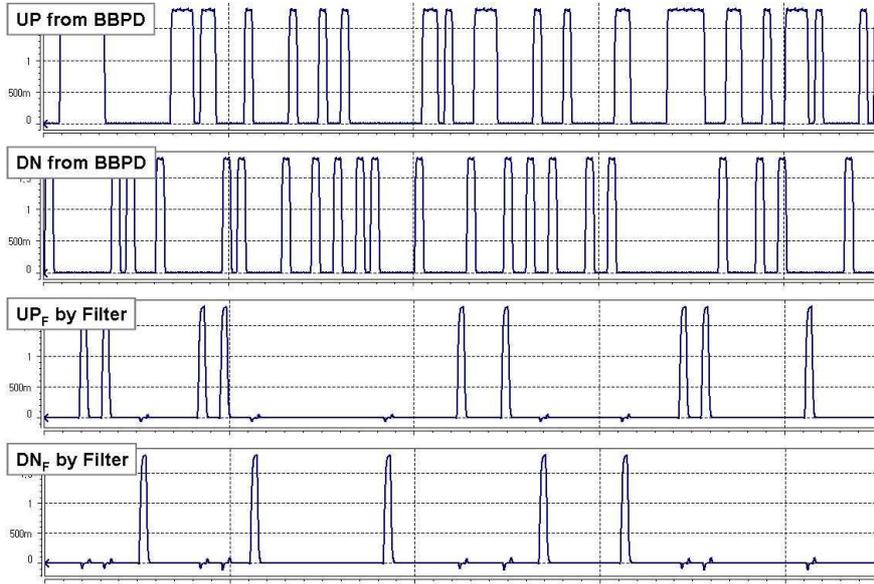
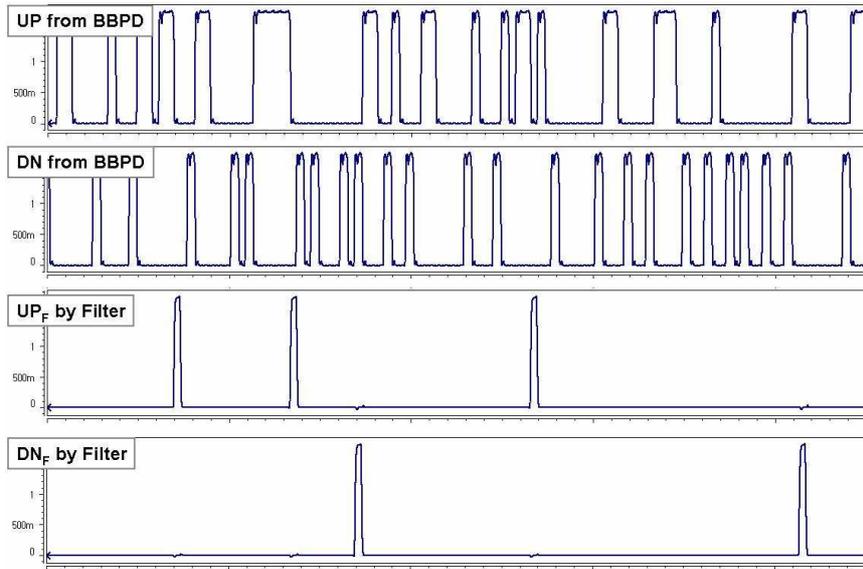


Figure 4.14: Schematic of 6-input *sum of product* gate :  $OUT = \sim (A \cdot B \cdot C + D \cdot E \cdot F)$



(a)



(b)

Figure 4.15: Simulation results of Up/Down filter in locked state using HSPICE (a) 2-bit filter (b) 3-bit filter



## 4.2 Circuit-level Design and Simulations of Reference Phase-Locked Loop

The reference PLL synthesizes 4-phase 1.25-GHz clock signals from the external 156.25MHz clock signal and provides them to the CDR as the reference clocks. The reference PLL includes a Phase and Frequency Detector(PFD), Charge Pump, loop filter, Voltage-to-Current Converter(VIC), Voltage-Controlled Oscillator(VCO) and Frequency Divider.

### 4.2.1 Voltage-Controlled Oscillator and Voltage-to-Current Converter

The VCO has a 4-stage ring structure using differential delay cells, as shown in the Fig. 4.18. It is considered a symmetry for each 4-phase clock signals since it should provide four clock signals having accurate quadrature relationship to the CDR. Four identical buffers are attached to the output nodes of each stage for this reason. The schematic of unit delay cell is shown in Fig. 4.19. The propagation delay of the cell is determined by two complementary control voltages, named  $V_{Cfast}$  and  $V_{Cslow}$ . The

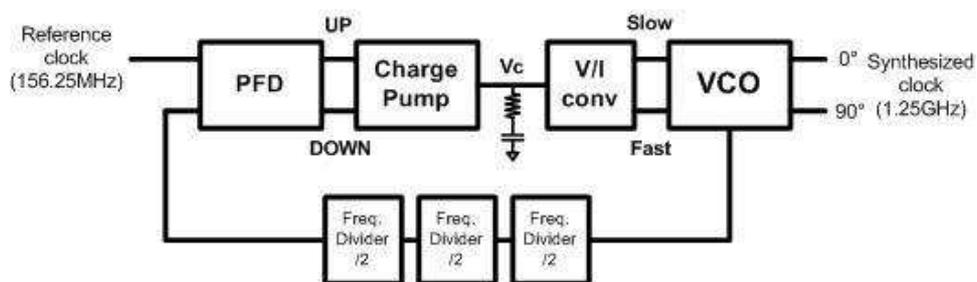


Figure 4.17: Block diagram of reference phase-locked loop

output voltage swing of the delay cell is linearly proportional to a bias current. Thus, the oscillation frequency range of the VCO is severely limited in a single-voltage control. In complement-voltage control, the output resistance of the PMOS load is decreased while the bias current is increased, vice versa. This leads wider oscillation frequency range as well as more constant voltage swing, which is expressed as a product of load resistance and bias current.

Complementary control voltage  $V_{Cslow}$  is provided by an additional block named VIC. It is a kind of inverting amplifier with a reduced voltage gain. The schematic is shown in Fig. 4.21.

In the SPICE simulation, the VCO oscillates in the whole range of control voltage: from 0V to 1.8V, and at 1.25GHz frequency for the all process corners: FF, TT and SS. The measured VCO gain around 1.25GHz is about 1.19GHz/V for the TT corner.

#### 4.2.2 Charge Pump

The charge pump transforms voltage pulses from the BBPD to current pulses linearly. Basically, the designed CP controls UP and DOWN current by switching current sources at the source nodes as shown in Fig. 4.22. It causes less current overshoot than switching at drain nodes [18]. Two transistors,  $M'_{swN}$  and  $M'_{swP}$ , are added to help current switching be faster. Pumping current of charge pump was set to  $100\mu A$ .

Because charge pumps suffer from many problems degrading the jitter performance of PLL such as a current mismatch, current leakage and charge sharing, it should be designed carefully. Especially, current mismatch induces a static phase error and ripples

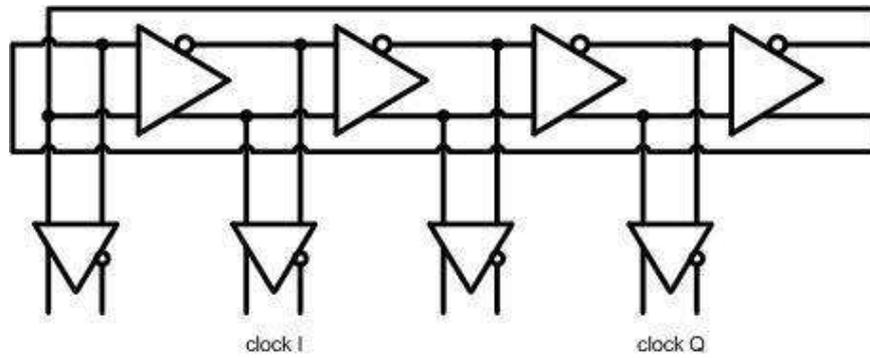


Figure 4.18: Block diagram of 4-stage ring oscillator

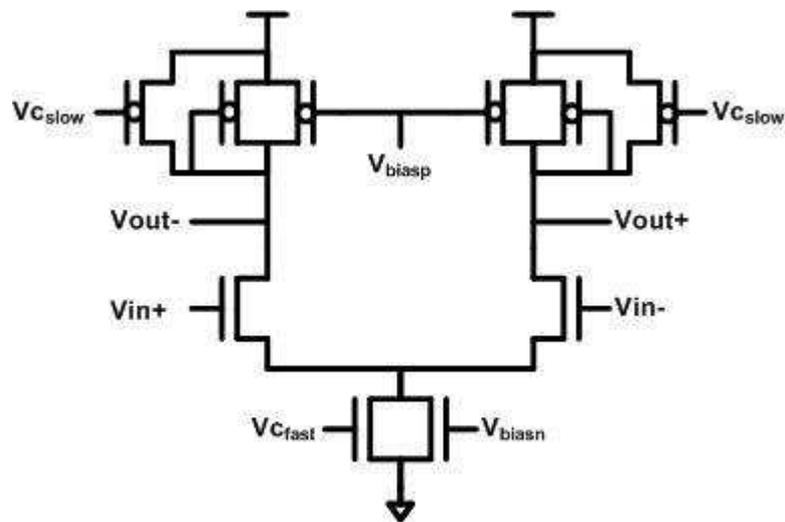


Figure 4.19: Schematic of unit delay cell in voltage-controlled oscillator

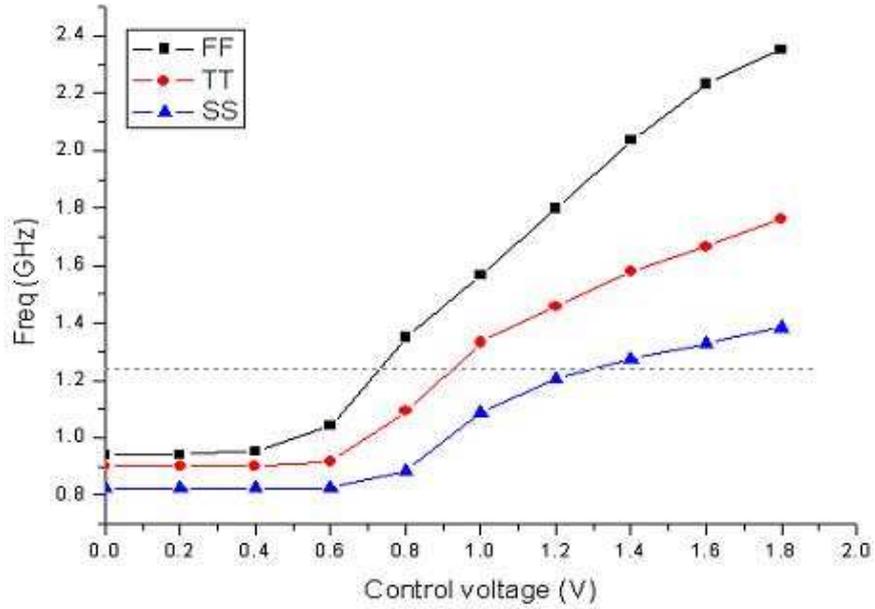


Figure 4.20: Transfer curve of the designed voltage-controlled oscillator: oscillation frequency vs. control voltage

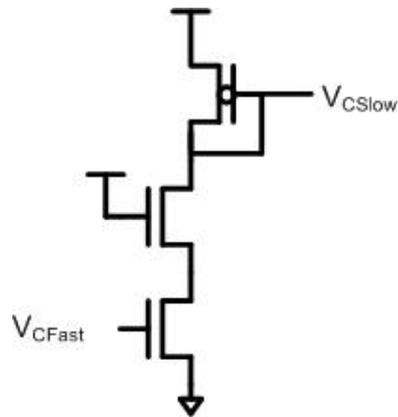
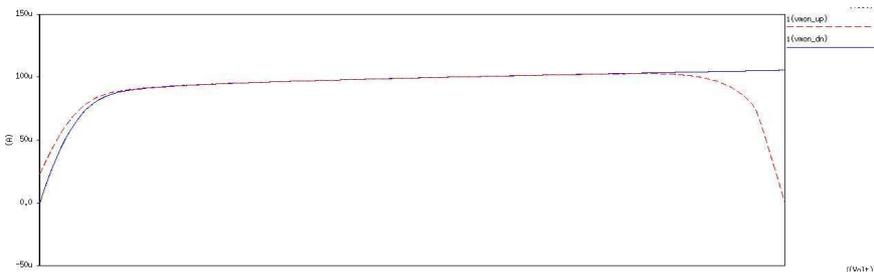
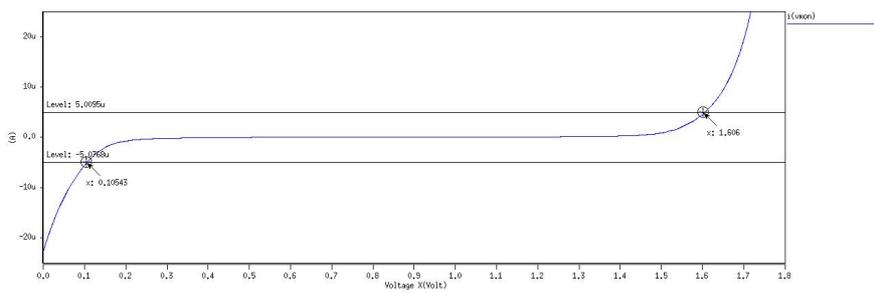


Figure 4.21: Schematic of voltage-to-current converter





(a)



(b)

Figure 4.23: Simulated current mismatch for whole range of control voltage (a) each of "UP" and "DOWN" current (b) difference of "UP" and "DOWN" currents: current mismatch

### 4.2.3 Loop Dynamics and Loop Filter

Undamped natural frequency  $\omega_N$  and damping ratio  $\zeta$  of second-order charge-pump PLL are determined by the following equations.

$$\omega_N = \sqrt{\frac{K_{VCO} I_{CP}}{M C}} \quad (4.1)$$

$$\zeta = \omega_N \frac{R C}{2} \quad (4.2)$$

where  $K_{VCO}$  is VCO gain in Hz/V,  $I_{CP}$  is charge pump current,  $M$  is dividing factor,  $R$  is stabilizing resistance, and  $C$  is capacitance.

By *Gardner's limit* referred in [15],  $\omega_N$  should be smaller than 1/20 of external reference frequency for stability. In this work, target  $\omega_N$  is to have about 1/40 of reference frequency, or 3.56 MHz, by using capacitor of 30 pF in loop filter.

Table 4.1: Parameters of reference PLL

	Value
$K_{vco}$	1.19 GHz/V
$I_{CP}$	100 $\mu A$
$C$	30 pF
$R$	3.1 K $\Omega$
$M$	8
$\omega_N$	3.56 MHz
$\zeta$	1.0

#### 4.2.4 Simulation of Reference Phase-Locked Loop

The designed PLL is simulated for three process corners. Fig. 4.24 shows transitions of control voltage during frequency acquisition. For all cases, it is observed that the PLL is successfully locked to the external reference clock having frequency of 156.25 MHz within  $2 \mu s$ .

Fig. 4.25 shows accumulated waveforms of synthesized clock having in-phase and quadrature-phase each. It is thought that intrinsic jitter of  $40 ps_{P-P}$  are caused by factors written below.

- Leakage current in charge pump and loop filter
- Delay mismatch between up and down pulse from PFD
- Noise coupling to bias voltage by noise coupling

First, leakage current in charge pump causes ripples on control voltage inducing the unwanted decaying of control voltage during no transition in both external reference clock and divided clock. Second, since the designed charge pump requires inverted version of up pulse and non-inverted version of down pulse as shown in Fig. 4.22, an inverter circuit for up pulse should be inserted between PFD and charge pump. Asymmetry between propagation delays of up and down pulse makes ripples on control voltage, too. Finally, noise is coupled from near circuits to bias voltage through parasitic capacitance.

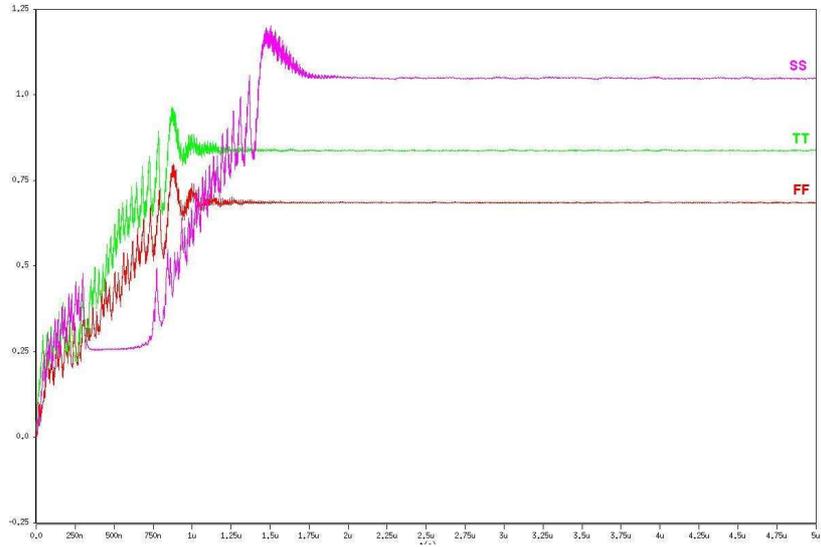


Figure 4.24: Acquisition process of reference PLL for three process corners

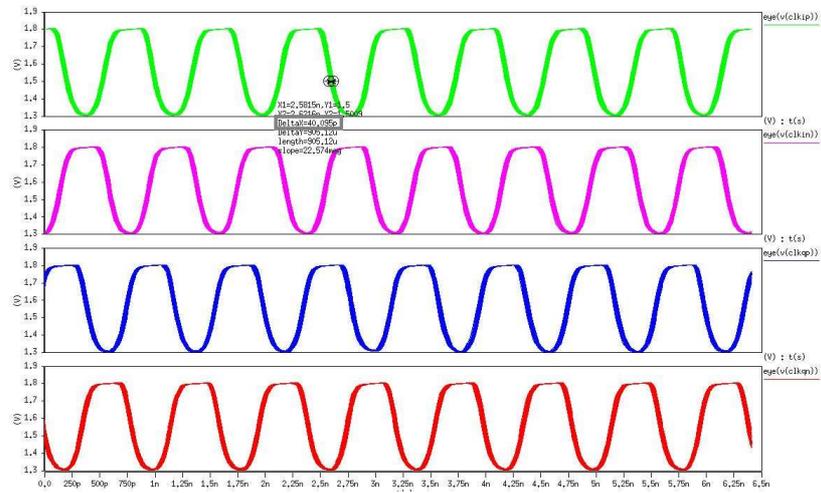


Figure 4.25: Accumulated waveform of the synthesized quadrature clocks

### 4.3 Circuit-level Simulation of Clock and Data Recovery circuit

To verify function and performance of the proposed CDR, simulations with four combinations of CDR as listed below are exercised.

- 6-bit CDR by using 4-bit PI with 2-bit up/down filter
- 6-bit CDR by using 4-bit PI with 3-bit up/down filter
- 8-bit CDR by using 4-bit PI and 2-bit DCDB with 2-bit up/down filter
- 8-bit CDR by using 4-bit PI and 2-bit DCDB with 3-bit up/down filter

Two kinds of simulation are performed for each CDR : jitter generation and jitter rejection. An ideally clean input data is given for jitter generation while jittery input for jitter rejection simulation. ISI through 350MHz bandlimited channel is induced to make input jitter having peak-to-peak value of 93 *ps* or 0.116 *UI*. Pattern of input data is  $2^{10} - 1$  PRBS and all simulations are carried out at 200 ppm frequency offset.

First, four CDRs are simulated and compared in jitter generation performance as shown in Fig. 4.26, 4.27, 4.28 and 4.29. Also, jitter rejection performance of them are simulated and compared as shown in Fig. 4.30, 4.31, 4.32, and 4.33. Their peak-to-peak and RMS jitter were measured and plotted in Fig. 4.34 (a) and (b), respectively. By extracting  $V_{DD}/2$ -crossing point of resulting clock presented as piece-wise linear form, RMS value is calculated using MATLAB.

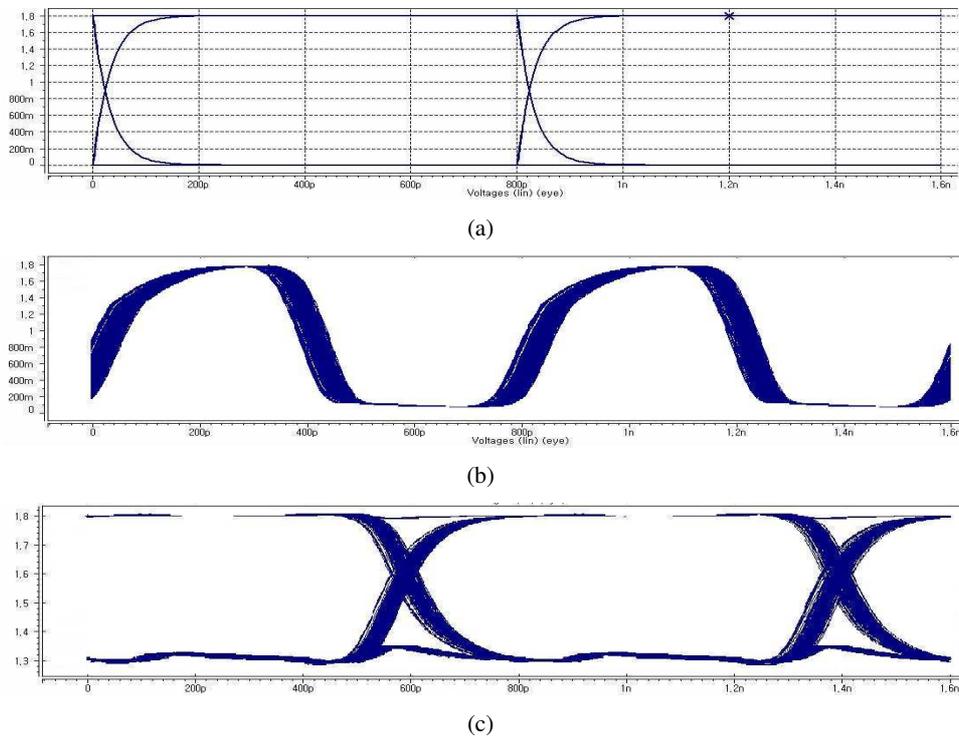


Figure 4.26: Jitter generation of the CDR using 6-bit PI with 2-bit filter (a) Input data without jitter (b) Recovered clock (c) Retimed data

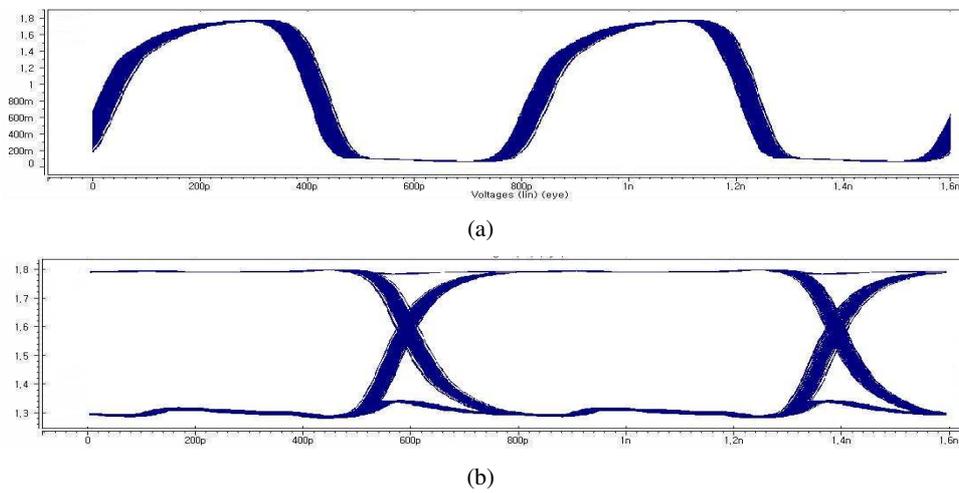
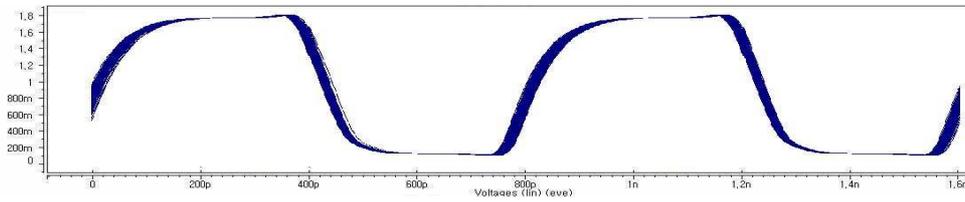
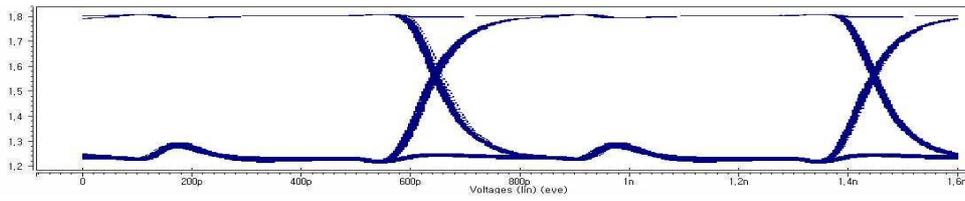


Figure 4.27: Jitter generation of the CDR using 6-bit PI with 3-bit filter (a) Recovered clock (b) Retimed data

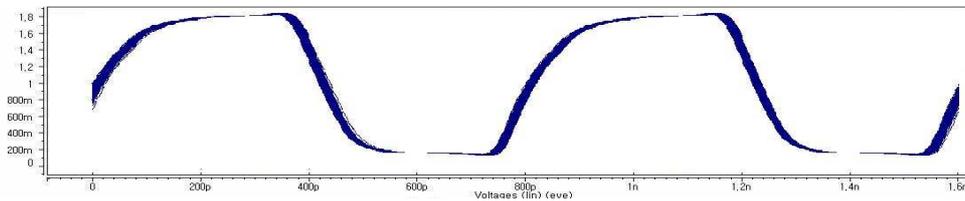


(a)

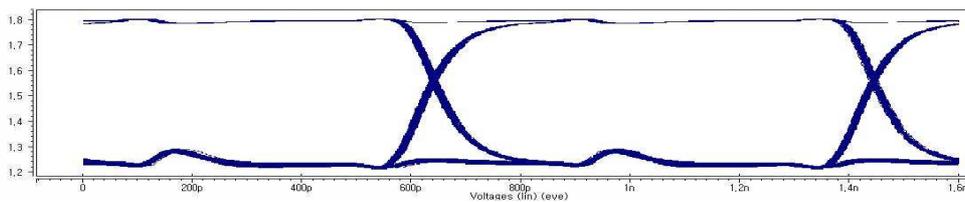


(b)

Figure 4.28: Jitter generation of the CDR using both 6-bit PI and 2-bit DCDB with 2-bit filter (a) Recovered clock (b) Retimed data

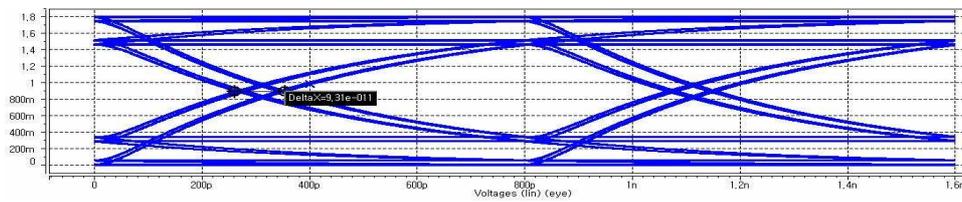


(a)

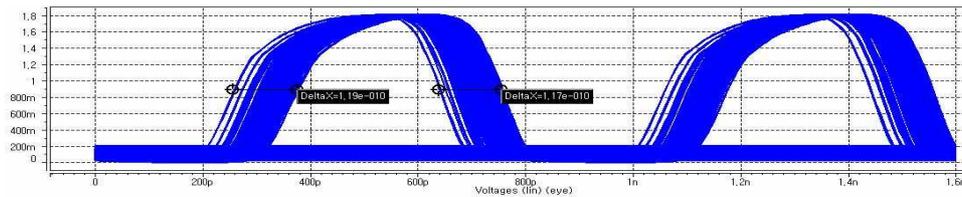


(b)

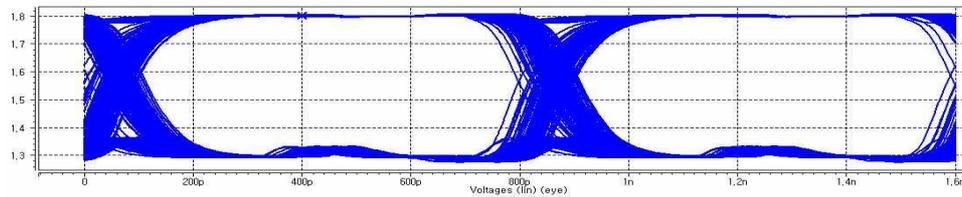
Figure 4.29: Jitter generation of the CDR using both 6-bit PI and 2-bit DCDB with 3-bit filter (a) Recovered clock (b) Retimed data



(a)

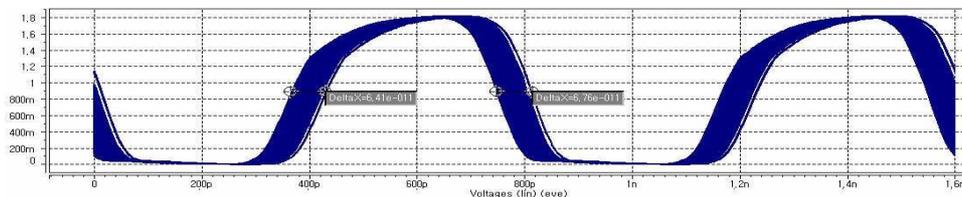


(b)

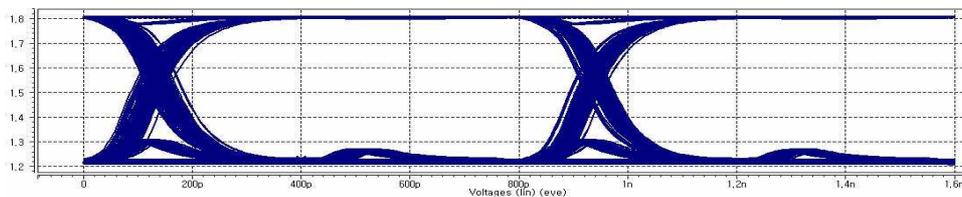


(c)

Figure 4.30: Jitter rejection of the CDR using 6-bit PI with 2-bit filter (a) Input data with ISI (b) Recovered clock (c) Retimed data

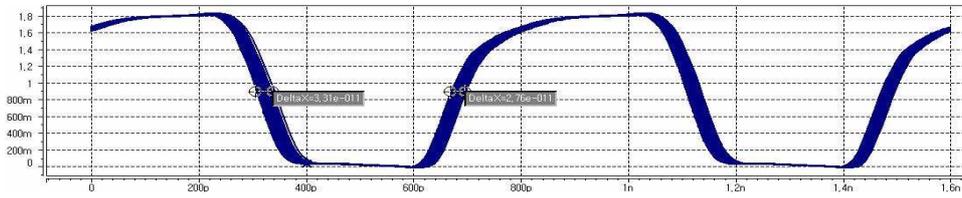


(a)

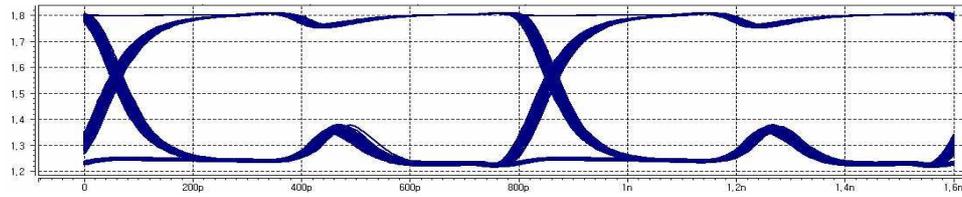


(b)

Figure 4.31: Jitter rejection of the CDR using 6-bit PI with 3-bit filter (a) Recovered clock (b) Retimed data

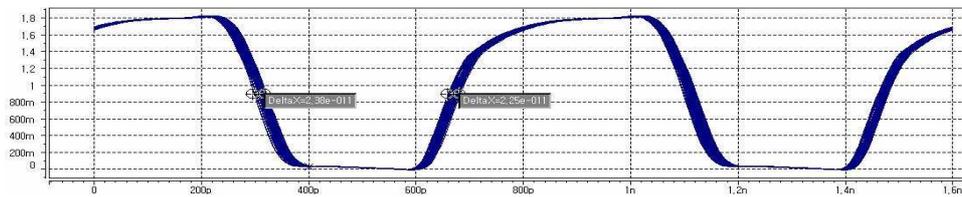


(a)

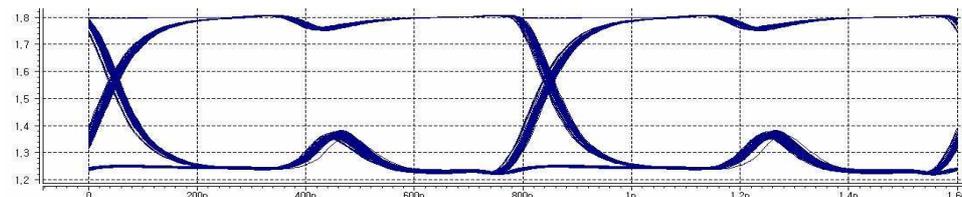


(b)

Figure 4.32: Jitter rejection of the CDR using both 6-bit PI and 2-bit DCDB with 2-bit filter (a) Recovered clock (b) Retimed data

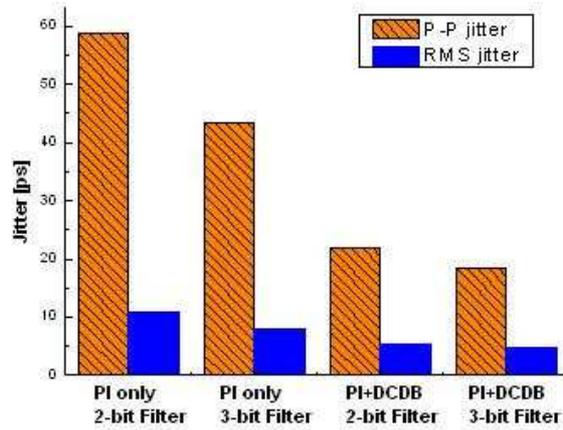


(a)

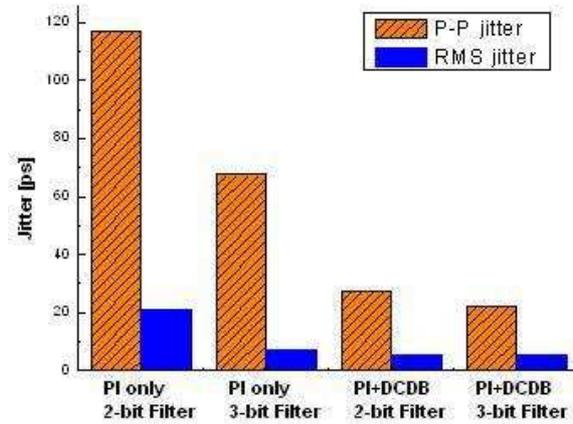


(b)

Figure 4.33: Jitter rejection of the CDR using both 6-bit PI and 2-bit DCDB with 3-bit filter (a) Recovered clock (b) Retimed data



(a)



(b)

Figure 4.34: Comparison of four CDRs in jitter performance (a) jitter generation, no input jitter (b) jitter rejection,  $93ps_{P-P}$  input jitter

As shown in Fig. 4.34 and Table. 4.2, both jitter generation and rejection is significantly improved by inserting DCDB to loop as expected. Although output jitter of the proposed CDR should be  $1/4$  of conventional one, it is reduced by about  $1/3$  in simulations. Dead-zone in BBPD and noise coupling appeared in transistor-level would result it.

In the aspect of filter, 3-bit up/down filter is relatively effective in the CDR using only PI, i.e. 6-bit CDR, compared to the case of 2-bit filter. However, as phase resolution is increased to 8-bit in the proposed CDR, 3-bit filtering does not greatly improve performance comparing to 2-bit filtering while it causes longer acquisition time. Therefore, 3-bit filter is not so useful in the proposed CDR.

Table. 4.3 summarizes power consumption of four CDRs. The proposed CDR does not have no more dissipation compared to conventional one. As known, power consumption of CMOS logic circuit is linearly proportional to the operating frequency. DCDB and its controller evidently consume additional power. However, since PI and MUX controller operates only when overflow is happened in DCDB controller, its operating frequency would be reduced to  $1/4$  statistically. In the same manner, CDRs with 3-bit filter consumes only negligible additional power compared to CDRs with 2-bit filter.

Table 4.2: Comparison of four CDRs in jitter performance

	No input jitter	Input jitter by ISI ( $93ps_{P-P}$ )
PI only, 2-bit filter	$58.8ps_{P-P}$ $10.8ps_{RMS}$	$117ps_{P-P}$ $21.08ps_{RMS}$
PI only, 3-bit filter	$43.5ps_{P-P}$ $8.07ps_{RMS}$	$67.6ps_{P-P}$ $7.18ps_{RMS}$
PI and DCDB, 2-bit filter	$21.9ps_{P-P}$ $5.45 ps_{RMS}$	$27.6ps_{P-P}$ $5.48ps_{RMS}$
PI and DCDB, 3-bit filter	$18.3ps_{P-P}$ $4.76ps_{RMS}$	$22.5ps_{P-P}$ $5.47ps_{RMS}$

Table 4.3: Comparison of four CDRs in power consumption

Filter length	PI only	PI and DCDB
2	$17.820 mW$	$17.820 mW$
3	$17.822 mW$	$17.822 mW$

## Chapter 5

# Experimental Results

The chip microphoto is shown in Fig. 5.1. The layouts of CDR core and reference PLL in prototype chip are presented in Fig. 5.2 and Fig. 5.3, respectively. They occupies  $165 \times 255 \mu\text{m}^2$  and  $195 \times 240 \mu\text{m}^2$  each.

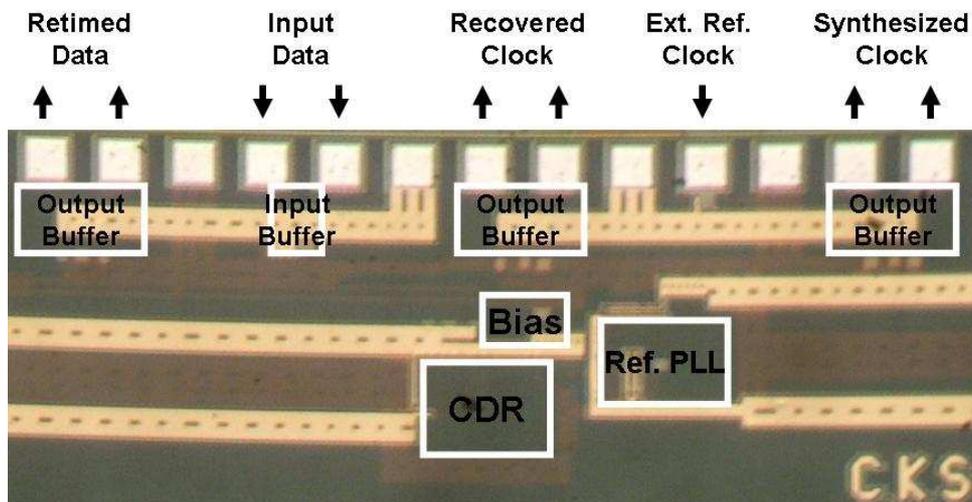


Figure 5.1: Chip microphoto

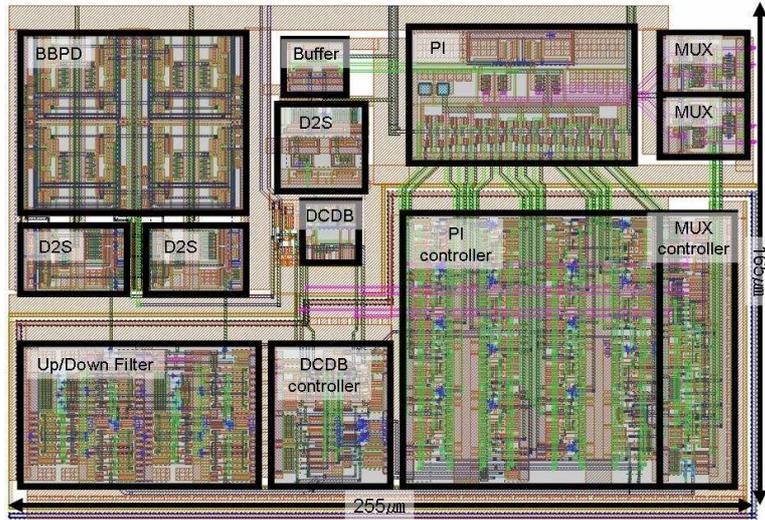


Figure 5.2: Layout of CDR core

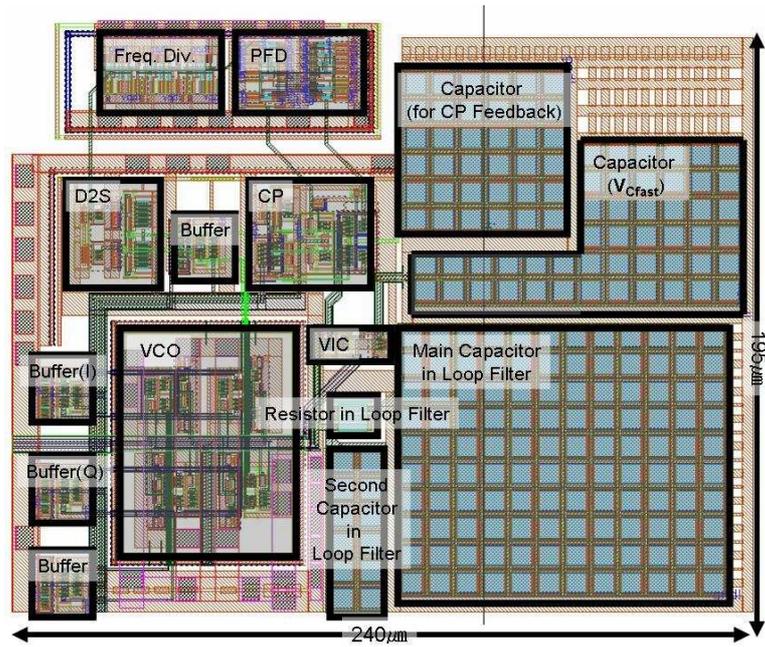


Figure 5.3: Layout of reference PLL

## 5.1 Measurement Setup

Fig. 5.4 shows an experimental setup. RF source provides the external reference clock to the PLL and Programmable Pattern Generator (PPG) provides input data to the CDR. Spectrum analyzer is used to check whether the PLL is locked or not and to measure the frequency offset between input data and synthesized clock. All waveforms are observed by an oscilloscope.

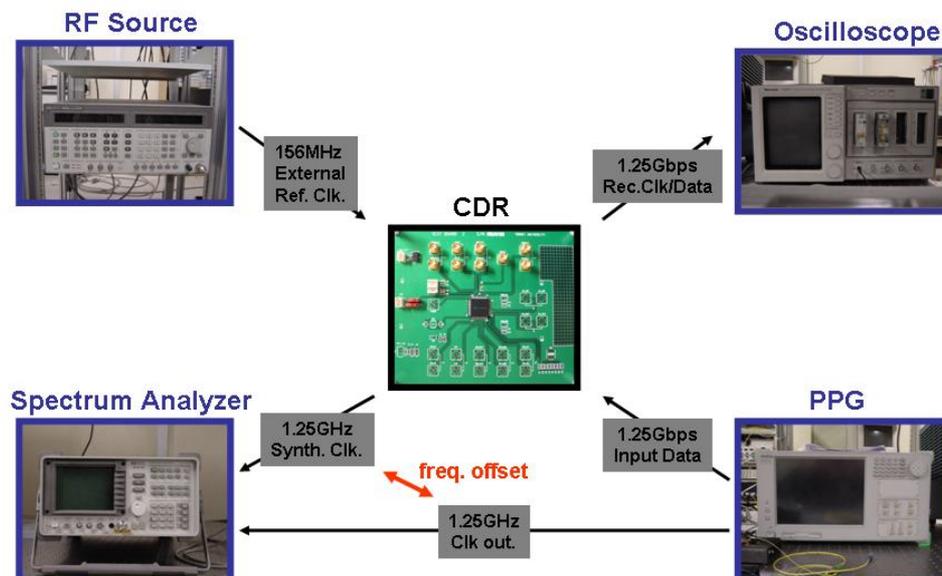


Figure 5.4: Measurement setup

The flow of experiments are listed below.

- Check whether the reference PLL is locked to the external reference clock or not.
- Check the frequency offset between output clock of PPG and synthesized clock of

PLL by observing spectrum analyzer. Tune the external reference clock finely to make frequency offset the desired value.

- Generate  $2^7 - 1$  PRBS and measure peak-to-peak and RMS jitter of input data using oscilloscope.
- Input PRBS to the CDR and check whether both recovered clock and retimed data are synchronized to input data.
- Measure output jitter for various  $V_{tuning}$  and frequency offset

## 5.2 Measurement Results

In the chip measurement, both CDR core and PLL operated at 1.25-Gb/s in 2.0V supply voltage, not 1.8V. The reason is thought that parasitic resistance and capacitance not considered in circuit-level simulation limit operating speed of the circuit.

The operation range of the reference PLL was from 960MHz to 1.38GHz in 2.0V supply. Fig. 5.5 shows the measured jitter of reference PLL. RMS jitter and peak-to-peak jitter are 15.36ps and 105.2ps, respectively.

The output jitter was measured at frequency offset of 200ppm for various delay errors by tuning  $V_{tuning}$ . Three tuning voltages, 0V, 0.2V and 0.4V, correspond to -50%, 0%, 50% DCDB error, respectively. Six waveforms in Fig. 5.6 are measured output clock of CDR for various values of tuning voltage.

Measured peak-to-peak and RMS jitter are plotted in Fig. 5.7. Flat jitter performance for DCDB was verified for DCDB errors from -50% to 50%. Beyond 50% error, however, output jitter increases since phase transfer of PI and DCDB becomes severely nonlinear. Overall jitter performance is degraded comparing to that in simulation results. It is thought that jitter or noise from PLL and output buffer cause more jitter.

To evaluate jitter rejection ability, input jitter was added by transmitting input data with 200ppm frequency offset through 2m PCB trace with 3.5m cable. As shown in Fig. 5.8, the CDR successfully recovered data from eye-closed data. There was no bit-error during 30 minutes corresponding to about  $2 \times 10^{12}$  bits.

The CDR covered frequency offset within  $\pm 400ppm$ , which is wide enough range for Giga-bit Ethernet and SGMII applications specifying frequency precision within

$\pm 100ppm$ .

Table. 5.1 shows a summary of the fabricated chip.

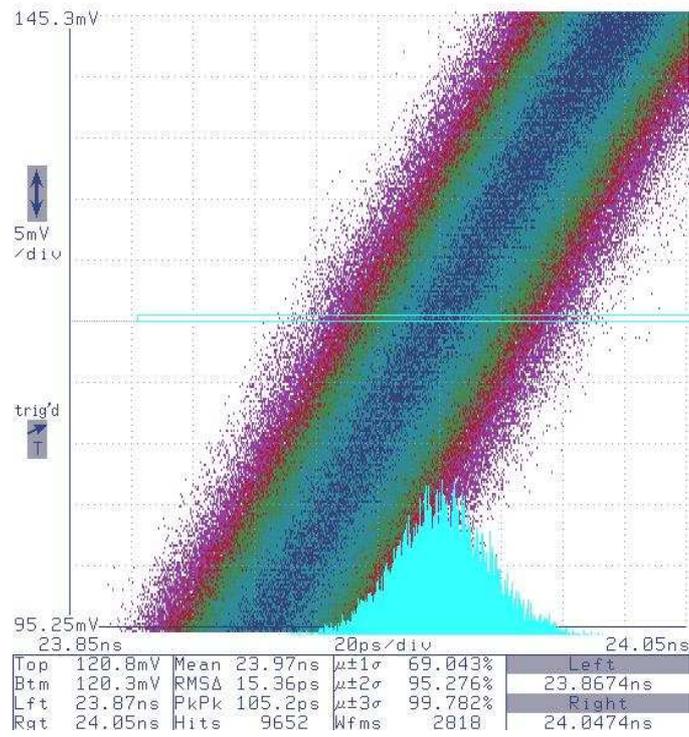


Figure 5.5: Measured jitter of the reference PLL

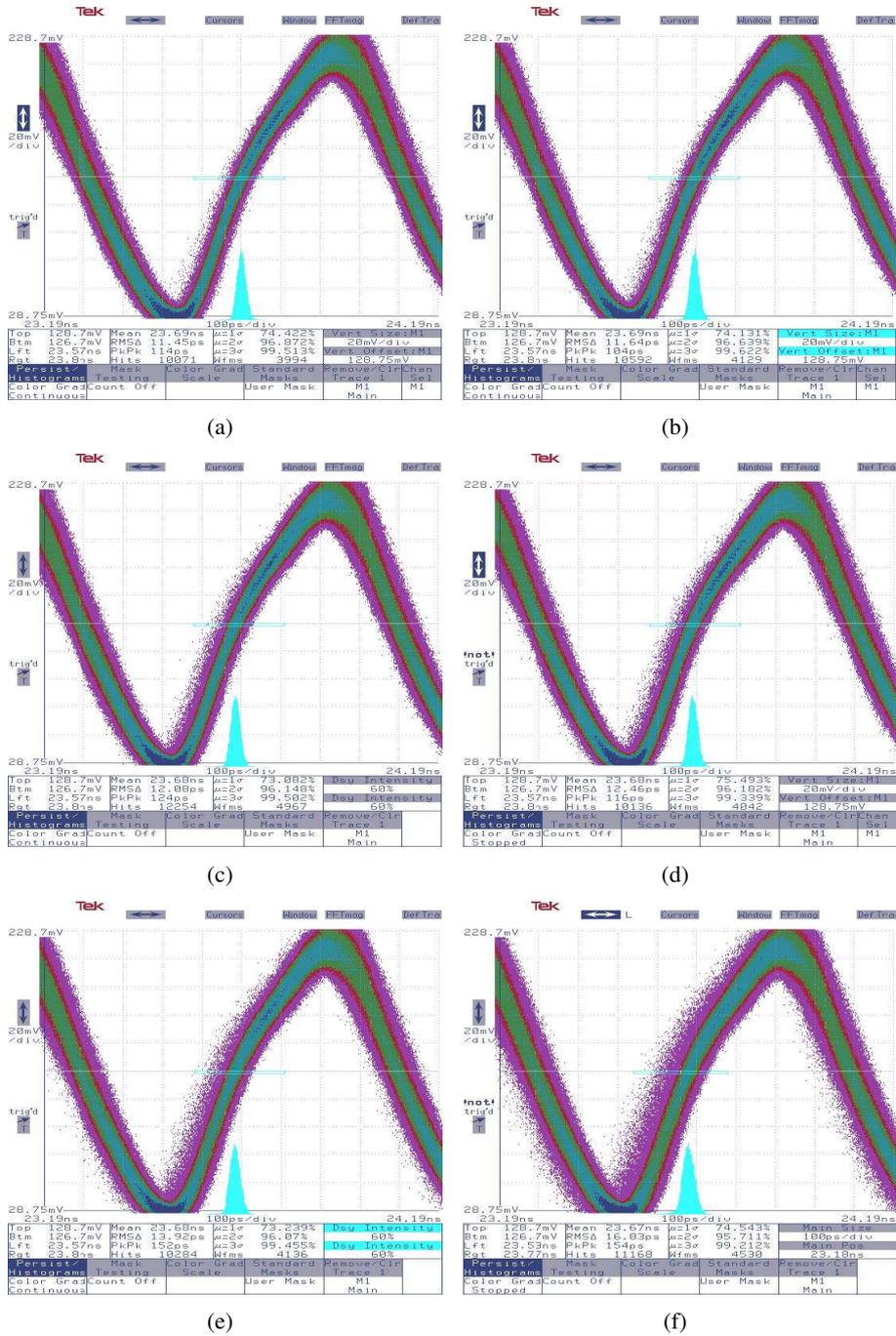


Figure 5.6: Measured waveform of output clock (a)  $V_{tuning}=0V$  (-50% DCDB error) (b)  $V_{tuning}=0.2V$  (0% DCDB error) (c)  $V_{tuning}=0.4V$  (50% DCDB error) (d)  $V_{tuning}=0.6V$  (e)  $V_{tuning}=0.8V$  (f)  $V_{tuning}=0.9V$



Table 5.1: Chip summary

Process	Dongbu-Anam 0.18 $\mu$ m CMOS
Supply voltage	2.0V
Operating range of reference PLL	960MHz ~ 1382MHz
Power consumption	17.8mW (CDR core)
Die area	CDR core : 165 $\times$ 255 $\mu$ m <sup>2</sup> PLL : 195 $\times$ 240 $\mu$ m <sup>2</sup>
Frequency tolerance	$\pm$ 400ppm
Bit Error Rate	$< 2 \times 10^{-12}$

## Chapter 6

# Conclusion

Dual-loop is the most widely used structure in multi-channel applications. Although digitally-controlled dual-loop CDR has advantages over analog-controlled one, such as smaller area, robustness and almost infinite loop gain, its limited phase resolution is the critical problem. To overcome problems, a new structure containing linearized phase interpolator and digitally-controlled delay buffer is proposed. By inserting DCDB, CDR could have 256 levels of phase using only 4-phase reference clock with little additional power consumption and die area. Also, by linearizing technique, both INL and DNL of phase transfer are improved.

Function and jitter performance of the CDR were verified in both behavioral and circuit-level simulation. Although DCDB has an uncertainty of delay, jitter generation performance was improved in relatively wide range of DCDB error.

The prototype chip was fabricated in 1-poly 6-metal  $0.18\mu m$  mixed-mode CMOS process. The CDR core and reference PLL occupy  $255 \times 165 \mu m$  and  $240 \times 190 \mu m$ , respectively. In the chip measurement, the CDR successfully operated and covered frequency offset within  $\pm 400$  ppm, wide enough range for SGMII application.

In the future, data rate of digitally-controlled dual-loop CDR can be increased by using frequency dividing scheme for speed-limiting block, i.e. controller circuit. Also, lower-power circuit can be implemented by reducing the power consumption in the BBPD.

# References

- [1] Cisco Systems, “Serial-GMII Specification v1.7”, Jul. 2001.
- [2] Tae-Sik Jeong, “Ethernet physical layer interface”, ETRI technical report, Nov. 2003.
- [3] Rainer Kreienkamp, Ulrich Langmann, Christoph Zimmermann, and Takuma Aoyama, “A 10-Gb/s CMOS Clock and Data Recovery Circuit with an Analog Phase Interpolator”, *IEEE Custom Intergrated Circuits Conference*, 2003.
- [4] Fuji Yang, Jay H. O’Neill, David Inglis, and Joseph Othmer, “A CMOS Low-Power Multiple 2.5-3.125Gb/s Serial Link Macrocell for High IO Bandwidth Network ICs”, *IEEE Journal of Solid-State Circuits*, vol. 37, pp.1813-1821, Dec. 2002.
- [5] Dong Zheng *et al.*, “A Quad 3.125Gb/s/Channel Transceiver with Analog Phase Rotators”, *IEEE International Solid-State Circuit Conference*, 2004.
- [6] Behzad Razavi, “Design of intergrated circuits for optical communications,” McGraw Hill, 2003.
- [7] Patrik Larsson, “Measurements and analysis of PLL jitter caused by digital switching noise”, *IEEE Journal of Solid-State Circuits*, vol. 36, pp.113-119, Jul. 2001.

- [8] Jeongsik Yang, Jinwook Kim, Sangjin Byun, Cormac Conroy, and Beomsup Kim, "A quad-channel 3.125Gb/s/ch serial-link transceiver with mixed-mode adaptive equalizer in 0.18 $\mu$ m CMOS," *IEEE International Solid-State Circuit Conference*, 2004.
- [9] Ali Hajimiri, Sotirios Limotyrakis, and Thomas H. Lee, "Jitter and phase noise in Ring Oscillators," *IEEE Journal of Solid-State Circuits*, vol.34, No. 6, pp.790-804, Jun. 1999.
- [10] Ali Hajimiri, and Thomas H. LEE, "A general theory of phase noise in electrical oscillators," *IEEE Journal of Solid-State Circuits*, vol.33, No. 2, pp.179-194, Feb. 1998.
- [11] Muneo Fukaishi, et al., "A 20-Gb/s CMOS multichannel transmitter and receiver chip set for ultra-high-resolution digital displays," *IEEE Journal of Solid-State Circuits*, vol.35, No. 11, pp.1611-1618, Jun. 2000.
- [12] M. H. Perrott, "Fast and accurate behavioral simulation of fractional-N synthesizers and other PLL/DLL circuits," *Design Automation Conference*, pp.498-503, Jun. 2002.
- [13] Stefanos Sidiropoulos, and Mark A. Horowitz, "A semidigital dual delay-locked loop," *IEEE Journal of Solid-State Circuits*, vol.32, No. 11, pp.1683-1692, Nov. 2000.
- [14] Adrian Maxim, "A 0.16-2.55-GHz CMOS active clock deskewing PLL using

- analog phase interpolator," *IEEE Journal of Solid-State Circuits*, vol. 40, No. 1, pp.110-131, Jan. 2005.
- [15] Floyd M. Gardner, "Charge-pump phase-lock loops," *IEEE Transactions on Communicatios*, vol. COM-28, No. 11, pp.1849-1858, Nov. 1980.
- [16] Yuan Ji-Ren, Ingemar Karlsson, and Christer Svensson, "A True Single-Phase-Clock Dynamic CMOS Circuit Technique," *IEEE Journal of Solid-State Circuits*, vol. sc-22, No. 5, pp.899-901, Oct. 1987.
- [17] Pyung-Su Han, "A new burst mode clock and data recovery circuit using two loop switching technique," Master thesis in Yonsei University, 2004
- [18] Woogeun Rhee, "Design of high-performance CMOS charge pumps in phase-locked loops," *IEEE International Symposium on Circuits and Systems*, 1999.
- [19] Jae Shin Lee, *et al.*, "A wide range PLL for 64X speed CD-ROMS & 10X speed DVD-ROMS," *IEEE Transactions on Consumer Electronics*, vol. 46, No.3, pp.487-493, Aug. 2000.

## 국 문 요 약

### 향상된 해상도를 가지는

### 디지털 제어 방식의 듀얼 루프 클럭 및 데이터 복원 회로

본 논문에서는 새로운 구조를 갖는 1.25-Gb/s 급 디지털-제어 방식의 듀얼 루프 클럭 및 데이터 복원 회로를 제안하였다. 제안된 구조에서는 4 단계의 가변 지연을 가지는 디지털-제어 방식의 지연 버퍼(digitally-controlled delay buffer)를 이용하여 출력 위상을 미세 조정함으로써 향상된 위상 해상도를 얻는다.

또한 위상 보간기의 위상 전달 특성을 선형화하기 위한 보상 방법을 제안하였다. 이 방법을 통하여 디지털-제어 방식의 위상 보간기의 INL과 DNL이 각각 71.2%와 55.2% 향상되었음을 확인하였다.

설계된 회로의 동작과 지터 성능은 동작적 시뮬레이션과 회로 수준 시뮬레이션을 통하여 검증하였다. 평가 칩은 동부아남 1-poly 6-metal 0.18  $\mu\text{m}$  CMOS 공정을 통해 제작되었으며, 측정 결과 성공적인 동작을 확인하였다. 설계된 회로의 전력 소모는 1.8V 전원 전압에서 17.8mW이고, 면적은  $255 \times 165 \mu\text{m}^2$  이다. 이 회

로는 기가 비트 이더넷 스위치 등의 멀티-채널 데이터 통신 시스템에 널리 사용  
될 수 있을 것이다.