Time-to-Digital Converter Architectures

Based on Mathematics

DISSERTATION

Submitted by

CONGBING LI

In partial fulfillment of the requirements for the award of the Degree of

DOCTOR OF PHILOSOPHY
IN
ELECTRONICS & INFORMATICS ENGINEERING

Under the guidance of

PROFESSOR HARUO KOBAYASHI, Ph. D. Eng.

DIVISION OF ELECTRONICS & INFORMATICS
GRADUATE SCHOOL OF SCIENCE & TECHNOLOGY
GUNMA UNIVERSITY
JAPAN

March 2016

Acknowledgement

I would like to express my deepest appreciation to all those who provided me the possibility to complete this dissertation.

Special thanks to my supervisor Professor Dr. Haruo KOBAYASHI for his great guidance and support. To me, he is an academic advisor rather than a college supervisor: he promotes my internal (mental) motivation on research and encourages me to achieve the research goals with the right set of planning and measured steps. He not only enhances my knowledge on circuit design, but also develops my skills to research more effectively with intrinsic motivation. Without his support this study could not have been done properly.

I am gratefully acknowledges the research support from members of Semiconductor Technology Academic Research Center (STARC) group for their priceless advice and support during my study. My great appreciation also goes to Associate Professor Kentaroh Katoh from Tsuruoka National College of Technology for his valuable discussion and advice. Additionally, I would like to thank Mr. Nobuyoshi ISHIKAWA for his valuable support during my present in Kobayashi laboratory.

I would like to thank my review committee members, Professor Sadao ADACHI, Kuniyuki MOTOJIMA, Yasushi YUMINAKA and Tadashi ITO.

I am thankful to the members of our Laboratory. Thanks to Junshan WANG, Richen JIANG, Taifeng WANG, Rui WANG and Jianlong WANG for their valuable help on my research and daily life. I would also like to thank Yutaro KOBAYASHI, Takeshi CHUJO and Masataka KAMIYAMA for their kind cooperation on my research.

Furthermore, I would like to thank Yoshiharu DOI, Makoto NAGAI, Ken NAKAOKA and Jiani YE who are my previous manager and team members in Sanyo Electric Co., Ltd. They taught me Japanese culture and inspired me to become enthusiastic about research.

Finally, sincerely thank to my parent Weilun LAI and Xinzhong LI. Thanks for their patience, support and endless love that give me courage.

Declaration

I hereby declare about this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person, nor material which has been accepted for the award of any other degree of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Signature:

Name: CONGBING LI

Student No.: 13802483

Date:

Table of Contents

List of Figures

List of Tables

Abbreviations

ADC	Analog-to-Digital Converter
AMS	Analog/Mixed-Signal
CMOS	Complementary Metal-Oxide-Semiconductor
DAC	Digital-to-Analog Converter
DFF	D-type Flip-Flop
DNL	Differential Nonlinearity
FF	Flip-Flop
FPGA	Field Programmable Gate Array
INL	Integrated Nonlinearity
LED	Light Emitting Diode
LMS	Least Mean Square
LSB	Least Significant Bit
MOS	Metal-Oxide-Semiconductor
MOSFET	Metal-Oxide-Semiconductor Field-Effect Transistor
PLL	Phase Locked Loop
RTL	Register Transfer Level
SAR	Successive Approximation Register
SNR	Signal-to-Noise Ratio
SoC	System on Chip
SPICE	Simulation Program with Integrated Circuit Emphasis
TDC	Time-to-Digital Converter

Abstract

The purpose of this dissertation is to design some new TDC architectures suitable for implementation with fine digital CMOS. These architectures can either reduce circuit complexity significantly compared to conventional TDCs while keeping comparable performance and glitch-free characteristics, or achieve fine time resolution, high linearity, and self-calibration features.

Mathematics is the language of electronic and electrical engineering. Mathematical concepts and methods are used in all areas of electronic and electrical engineering. Especially, number theory and coding theory in mathematics are found to be suitable for circuit design, as number theory and coding theory combine mathematical elegance and some engineering problems to an unusual degree.

To attain the main objective to design some new TDC architectures suitable for implementation with fine digital CMOS, mathematical methods such as number theory and coding theory are introduced into the design of TDC architectures. In detail, residue number system, Gray code, and cyclic code are applied in parallel ring oscillator TDC architectures, in order to reduce hardware, power consumption, as well as chip area significantly compared to a flash type TDC, while keeping comparable performance and glitch-free characteristics. Furthermore, stochastic process theory is also applied in TDC architecture to utilize the large variation in circuit characteristics of fine CMOS. The stochastic architecture with self-calibration feature can realize a linear TDC with fine (sub-picosecond) time-resolution.

Chapter 1 introduces the background, the motivation, and the objectives of this research and the proposed approaches. Chapter 2 discusses the basic TDC specifications and its architectures. Chapter 3 presents proposed parallel ring oscillator TDCs based on residue number system, Gray code, and cyclic code. Their RTL simulation waveforms, FPGA implementation and verification results are presented and discussed. Chapter 4 presents proposed stochastic TDC based on stochastic process theory. The RTL simulation waveforms, FPGA implementation and

verification results are presented and discussed. Chapter 5 summarizes conclusions obtained through this research.

Applying number theory, coding theory and stochastic process theory in TDC circuit design is showing the beauty of the particular combination of mathematics and engineering. We make fully digital FPGA implementation (design, simulation, verification, and testing) of these TDCs. The design work uses only RTL (without SPICE) simulation and FPGA (instead of full custom CMOS) implementation, which would be suitable for mixed-signal SoC design in nano-CMOS era.

Chapter 1

INTRODUCTION

In this chapter, the research background and motivation are introduced in section 1.1. Then the research objective and approach are discussed in section 1.2 and section 1.3 respectively. Finally, the outline of the thesis is described in section 1.4.

1.1 Research Background and Motivation

With the integrated circuit fabrication technologies (e.g., CMOS) reach the deep sub-micrometer regime, circuits that process analog voltage signals encounter scaling impediments. As the supply voltage decreases in accordance with the scale shrinkage of semiconductor processes, there is often not enough voltage headroom for any sophisticated analog architectures, e.g., cascades [1]. And while voltage levels decrease continuously noise does not scale, the signal-to-noise ratio (SNR) degrades and the noise would cause a meta-stability issue. Since most applications have stringent performance requirements in terms of SNR and distortion ratio, i.e., resolution, it is getting more difficult to design the analog circuit part in the voltage-domain that configures the mixed-signal chip [2].

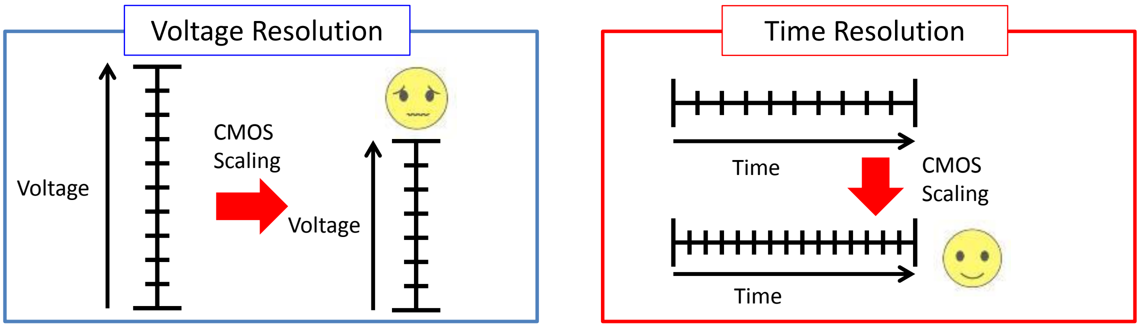


Fig. 1. 1 The way to the time domain

On the other hand, the positive side of scaling, the switching characteristics of MOS transistors offer excellent timing accuracy at high frequencies. Thus, a new design paradigm with deep sub-micrometer CMOS technologies is possible, in which the time-domain resolution of a digital signal edge transition is superior to the voltage resolution of an analog signal [3].

In electronics, a Time-to-Digital-Converter (TDC) is a device measuring the time interval between two signals and converting it into digital (binary) output. Advanced CMOS process enables TDC time resolution of several pico-seconds, and the resolution improves as switching speed increases. TDC is playing an increasingly important role in the nano-CMOS era, as it is well suited to implementation with fine digital CMOS processes [4][5][6][7].

1.2 Research Objective

In current Systems-on-a-Chip (SoC), digital technology is dominant although there are usually some analog circuits present. Most devices are targeted for digital circuit improvement; if the design of analog/mixed-signal (AMS) circuits were taken into account, the manufacturing cost would increase. The role of AMS circuit designers, who have to be very adaptive, is to design high-performance AMS circuits utilizing digital-friendly devices. Nano CMOS processes are digital device oriented and AMS designers face challenges of low voltage supply, small intrinsic gain and large device parameter variation as well as reliability and testing problems. The solution to this problem has been suggested as digitally-assisted analog technology, which utilizes digital technology extensively for AMS circuit performance improvement.

As CMOS processes scale down, design and implementation of a full custom SoC becomes more difficult technically and economically. On the other hand, a field programmable gate array (FPGA) is attractive due to its flexibility, and it can be used for so-called disruptive innovation. We consider that FPGA implementation (design, simulation, verification, and testing) of all AMS, logic and memory would be one of the goals for the digitally-assisted analog technology.

In nano CMOS era, a TDC plays an important role. Varieties of TDC architectures are proposed, such as flash-type TDC, Vernier delay line TDC, and so on. However, conventional TDC architectures have disadvantages on circuit complexity, and non-linearity due to buffer delay mismatches [6][9].

Based on above consideration, the main objective of this study is to design some new TDC architectures suitable for implementation with fine digital CMOS. These architectures can either reduce circuit complexity significantly compared to conventional TDCs while keeping comparable performance and glitch-free characteristics, or achieve fine time resolution, high linearity, and self-calibration features. We make fully digital FPGA implementation (design, simulation, verification, and testing) of these TDCs. The design work uses only RTL (without SPICE) simulation and FPGA (instead of full custom CMOS) implementation, which would be suitable for mixed-signal SoC design in nano-CMOS era.

1.3 Approach

Mathematics is the language of electronic and electrical engineering. A number of electrical laws (e.g., Maxwell's equations for electromagnetics, Kirchhoff's Rules for circuit analysis) are mathematical expressions. Mathematical concepts and methods are used in all areas of electronic and electrical engineering. For example, Circuit analysis is the study of methods of solving generally linear systems for unknown variables such as the voltage at a certain node or the current through a certain branch of a network.

Especially, number theory and coding theory in mathematics are found to be suitable for circuit design, as number theory and coding theory combine mathematical elegance and some engineering problems to an unusual degree. Number theory, sometimes called "higher arithmetic", is a branch of pure mathematics devoted primarily to the study of the natural numbers and the integers. It is sometimes called "The Queen of Mathematics" because of its foundational place in the discipline. Coding theory, sometimes called algebraic coding theory, is the study of the properties of codes and their fitness for a specific application. It makes use of classical and modern algebraic techniques involving finite fields, group theory, and polynomial algebra. It has connections with other areas of discrete mathematics, especially number theory.

Recently, several novel circuit design schemes have been proposed based on number theory and coding theory. The major advantage of applying number theory and coding theory is the beauty of this particular combination of mathematics and engineering. For example, Fibonacci sequence is applied in SAR ADC algorithm design to realize reliable and high-speed SAR AD conversion [10]. Magic square is applied in segmented DAC layout design to reduce the linear gradient error effects and improve the linearity effectively [11].

To attain the main objective to design some new TDC architectures suitable for implementation with fine digital CMOS, mathematical methods such as number theory and coding theory are introduced into the design of TDC architectures. In detail, residue number system, Gray code, and cyclic code are applied in parallel ring oscillator TDC architectures, in order to reduce hardware, power consumption, as well as chip area significantly compared to a flash type TDC, while keeping comparable performance and glitch-free characteristics. Furthermore, stochastic process theory is also applied in TDC architecture to utilize the large variation in circuit characteristics of fine CMOS. The stochastic architecture with self-calibration feature can realize a linear TDC with fine (sub-picosecond) time-resolution.

1.4 Outline of the Thesis

The outline of this dissertation is as follows:

Chapter 1

This chapter introduces the background, the motivation, and the objectives of this study and the proposed approaches.

Chapter 2

The chapter discusses the basic TDC specifications and its architectures.

Chapter 3

This chapter presents proposed parallel ring oscillator TDCs based on residue number system, Gray code, and cyclic code. Their RTL simulation waveforms, FPGA implementation and verification results are presented and discussed.

Chapter 4

The chapter presents proposed stochastic TDC based on stochastic process theory. The RTL simulation waveforms, FPGA implementation and verification results are presented and discussed.

Chapter 5

This chapter summarizes conclusions obtained through this study.

References

[1] Stephan Henzler, “Time-to-Digital Converters”, Springer Series in Advanced Microelectronics.

[2] Stephan Henzler, Siegmar Koeppe, Dominik Lorenz, Winfried Kamp, Ronald Kuenemund, Doris Schmitt-Landsiedel, “A Local Passive Time Interpolation Concept for Variation-Tolerant High-Resolution Time-to-Digital Conversion”, IEEE Journal of Solid-State Circuits, vol.43, no.7, July 2008.

[3] Guansheng Li, Yahya M. Tousi, Arjang Hassibi, Ehsan Afshari, “Delay-Line-Based Analog-to-Digital Converters”, IEEE Transactions on Circuits and System II: Express Briefs, vol.56, no.6, June 2009.

[4] Robert Bogdan Staszewski, Poras T. Balsara, “All-Digital Frequency Synthesizer in Deep-Submicron CMOS”, Wiley-Interscience (2006).

[5] T. Komuro, R. Jochen, K. Shimizu, M. Kono, H. Kobayashi, “ADC Architecture Using Time-to-Digital Converter,” IEICE Transactions on Electronics, vol. J90-C, no.2, pp.125–133, 2007.

[6] S. Uemori, M. Ishii, H. Kobayashi, D. Hirabayashi, Y. Arakawa, Y. Doi, O. Kobayashi, T. Matsuura, K. Niitsu, Y. Yano, T. Gake, T. Yamaguchi, N. Takai, “Multi-bit Sigma-Delta TDC Architecture with Improved Linearity,” Journal of Electronic Testing : Theory and Applications, Springer, vol.29, no.6, pp.879-892, December 2013.

[7] K. Katoh, Y. Kobayashi , T. Chujyo , J. Wang, E. Li, C. Li, H. Kobayashi, “A Small Chip Area Stochastic Calibration for TDC Using Ring Oscillator,” Journal of Electronic Testing: Theory and Applications, Springer, vol.30, no.6, pp.653-663,

December 2014.

[8] H. Kobayashi, H. Aoki, K. Katoh, C. Li, “Analog/Mixed-Signal Circuit Design in Nano CMOS Era”, IEICE Electronics Express, vol.11 no.3, pp.1-15, 2014.

[9] Y. Osawa, D. Hirabayashi, N. Harigai, H. Kobayashi, O. Kobayashi, M. Tsuji, S. Umeda, R. Shiota, N. Dobashi, M. Watanabe, T. Matsuura, K. Niitsu, T. J. Yamaguchi, N. Takai, and I. Shimizu, "Phase Noise Measurement and Testing with Delta-Sigma TDC," in Proc. International Conference on Integrated Circuits, Design, and Verification (ICDV 2013), pp.105-109, November 2013.

[10] Yutaro Kobayashi, Haruo Kobayashi, “Redundant SAR ADC Algorithm Based on Fibonacci Sequence”, Advanced Micro-Device Engineering VI, Key Engineering Materials (2016).

[11] Masashi Higashino, Haruo Kobayashi, “Segmented DAC Linearity Improvement with Layout Technique Using Magic Square”, 1st International Symposium of Gunma University Medical Innovation and 6th International Conference on Advanced Micro‐Device Engineering (GUMI&AMDE 2014).

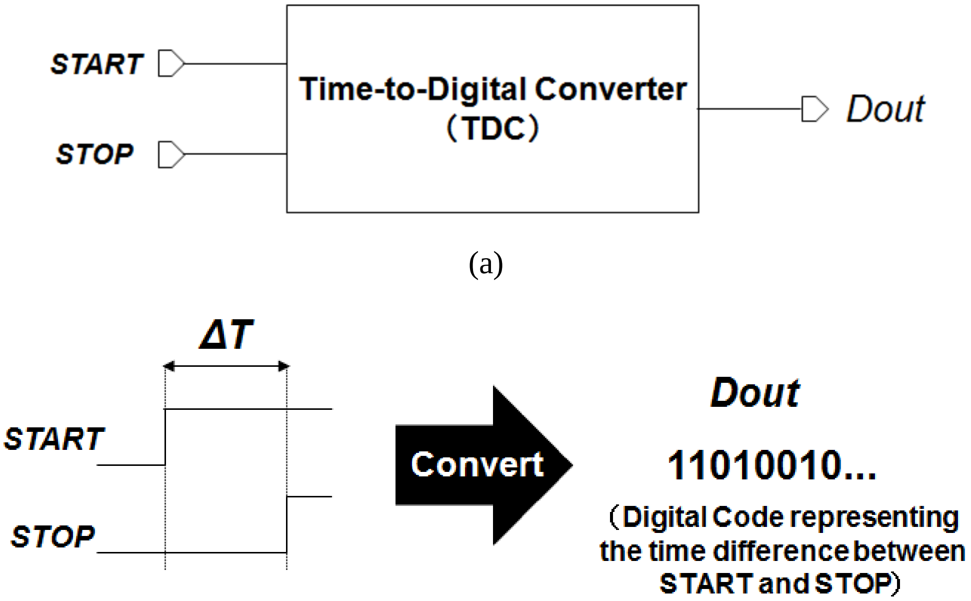
Chapter 2

FUNDAMENTAL OF TIME-TO-DIGITAL CONVERTER

In this chapter, section 2.1 introduces TDC definition and its applications. Then, the important TDC specification terms are described in section 2.2. The conventional TDC architectures are discussed in section 2.3. Finally, the summary is presented in the last section.

2.1 TDC Definition

A TDC quantizes the time difference between the START pulse and the STOP pulse and converts this time interval into a digital representation, as illustrated in Fig.2.1 (a). Measurement is started and stopped, when either the rising or the falling edge of a signal pulse crosses a set threshold, as shown in Fig.2.1 (b).


START
STOP
Time-to-Digital Converter
(TDC)
Dout
ΔT
START
STOP
Convert
Dout
11010010...
(Digital Code representing
the time difference between
START and STOP)

(a)

(b)

Fig. 2. 1 Concept of time-to-digital converter

TDC circuit consists of mostly digital circuits, and its applications include time-of-flight measurements, all-digital PLLs, TDC-based ADCs, and so on.

2.2 TDC Performance Parameters

The important performance parameters to evaluate a TDC includes time resolution, measurement range, nonlinearity, conversion time, and stability [1].

2.2.1 Time Resolution

The term time resolution is used for the minimum time interval that can theoretically be resolved by the TDC in a single measurement, that is, the quantization step (LSB).

2.2.2 Measurement Range

The measurement range of the TDC defines the maximum time interval that can be measured [2]. If the time resolution of a TDC is T_{LSB}, then the measurement range can be denoted by

$$DR = 2^N \cdot T_{LSB} \quad (2-1)$$

where DR represents the measurement range, and N refers to the number of bits of TDC outputs [1].

2.2.3 Nonlinearity

The nonlinearity performances include differential nonlinearity (DNL) and integrated nonlinearity (INL). DNL is the deviation of a single quantization step from the ideal value of 1 LSB.

$$DNL_i = T_i - T_{LSB} \quad (2-2)$$

where DNL_i is the i^{th} value of the differential nonlinearity. T_i is the width of the i^{th} step in real transfer curve, T_{LSB} is the ideal 1 LSB step width.

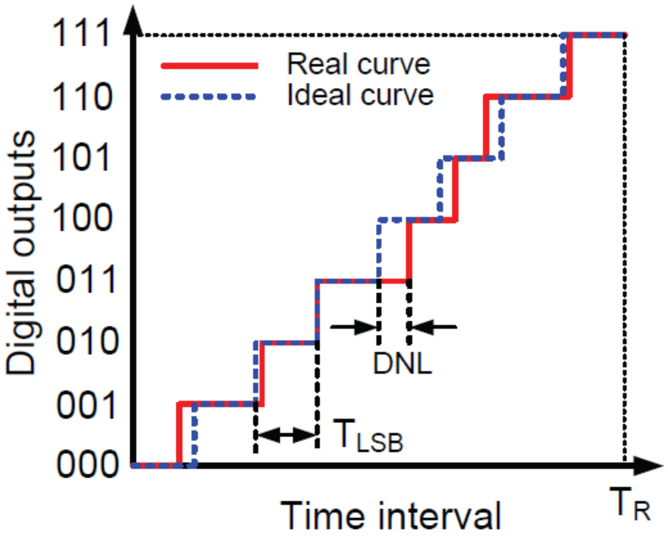


Fig. 2. 2 Definition of differential nonlinearity (DNL) [2]

INL is the deviation of the input-output characteristics from the ideal, straight-line input-output characteristics. It is defined as the deviation of the step position from its ideal value normalized to one T_{LSB}. The calculation of INL is given as

$$INL_i = \sum_{n=0}^{i-1} DNL_i \qquad (2-3)$$

2.2.4 Conversion Time

Conversion time, also known as conversion speed, refers to the time between the end mark of the input time interval and the moment when the measurement result is ready. It evaluates the speed of signal processing and device delay at each conversion time window in a TDC.

2.2.5 Stability

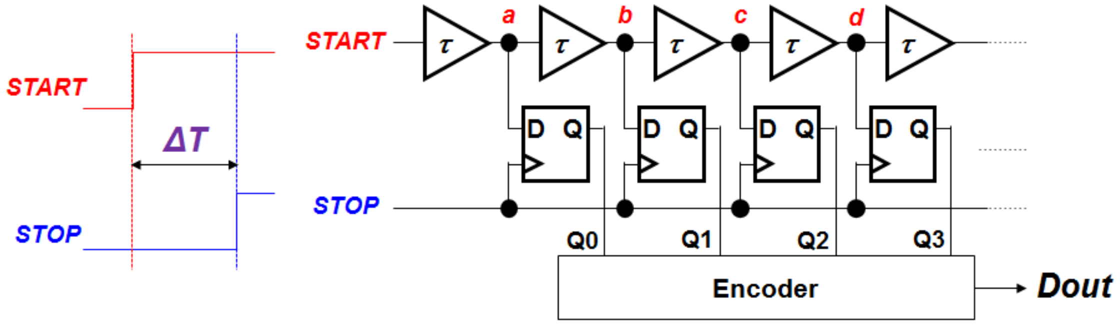
Stability of a TDC is defined as the sensitivity of its characteristics with PVT (process, voltage, temperature) variations, and the like.

2.3 Basic TDC Structures

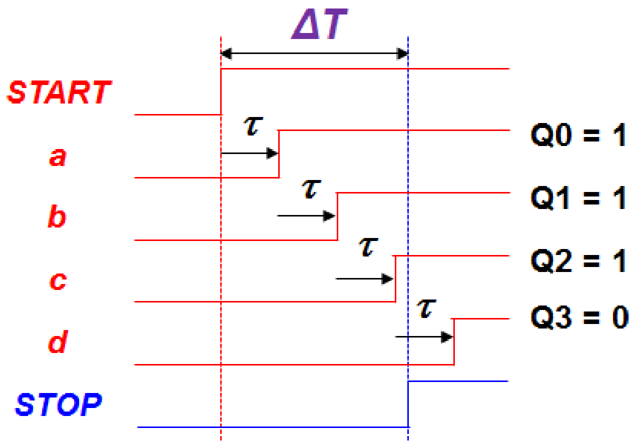
2.3.1 Flash-type TDC

Fig.2.3 illustrates the architecture of a basic flash-type TDC. The flash-type TDC uses a delay line which consists of CMOS inverter buffer delays and D Flip-Flops (DFFs). The START signal is connected to the delay line input and passes through the serial delay elements. The output array of the delay buffers are connected to the D input terminals in the D flip-flop array. The START signal is delayed by an integer multiple of the buffer delay τ. When the STOP signal is on the rising edge, the D flip-flop array are triggered. The outputs of the D flip-flop array are transferred to the thermometer-code-to-binary encoder. The encoder produces a digital output D_{out} representing the time interval between the START signal and the STOP signal [3]. The measured time interval between the START and STOP signals is equal to a certain number of steps of buffer delay.

Though the flash-type TDC has the advantages such as high-speed timing measurement, single-event timing measurement, and all digital implementation, it has a disadvantage on circuit complexity: for a conventional n-bit delay-chain TDC with 2^n quantization levels, at least a total of $2^n - 1$ delay elements and $2^n - 1$ DFFs are required, leading to large power and chip area.



(a) Block diagram of flash-type TDC



(b) Timing chart of flash-type TDC

Fig. 2. 3 Flash-type TDC

2.3.2 Vernier-type TDC

The time resolution of the flash-type TDC can be improved significantly by using the gate delay as the basic time unit. The fundamental concept of the delay Vernier technique is that the timing resolution is determined by the difference between two propagation delay values. The Vernier structure consists of a pair of tapped delay lines with a flip–flop at each corresponding pair of taps.

As presented in Fig.2.4, two buffer delay lines are applied in Vernier-type TDC. The buffer delay in the upper delay line is slightly greater than the buffer delay in the lower delay line. The START and STOP signal propagate through the upper delay line and the lower delay line respectively. The START signal is delayed by an integer multiple of the buffer delay $\tau 1$, and the STOP signal is delayed by an integer multiple of the buffer delay $\tau 2$. The output array of the upper delay line are connected to the D input terminals in the D flip-flop array. When output array of the lower delay line are on the rising edge, the D flip-flop array are triggered. The outputs of the D flip-flop array are transferred to the thermometer-code-to-binary encoder. The encoder produces a digital output representing the time interval between the START signal and the STOP signal.

As the buffer delay $\tau 2$ is slightly smaller than the buffer delay $\tau 1$, the time difference between the START and STOP signals is decreased in each Vernier stage by the buffer delay difference $(\tau 1-\tau 2)$. The position in the delay line, where the START signal is caught up by the STOP signal, indicates the information about the time difference between START and STOP, with the resolution equal to the buffer delay difference, i.e. $T_{LSB}=\tau 1-\tau 2$.

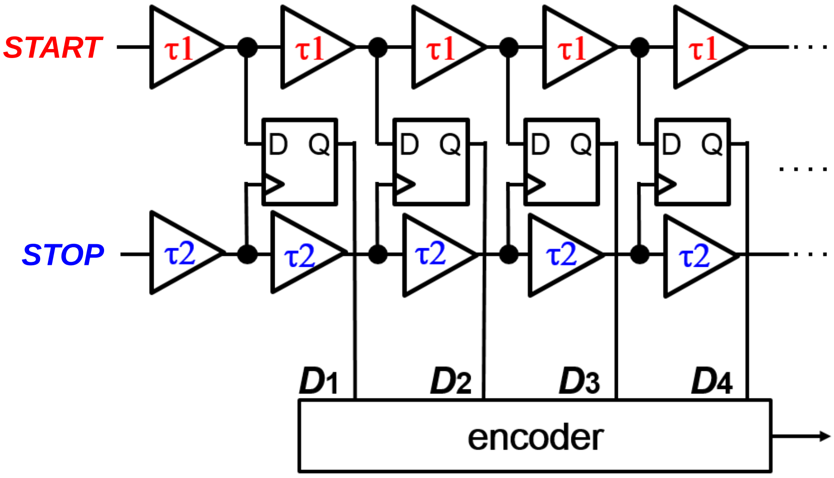


Fig. 2. 4 Typical Vernier delay line

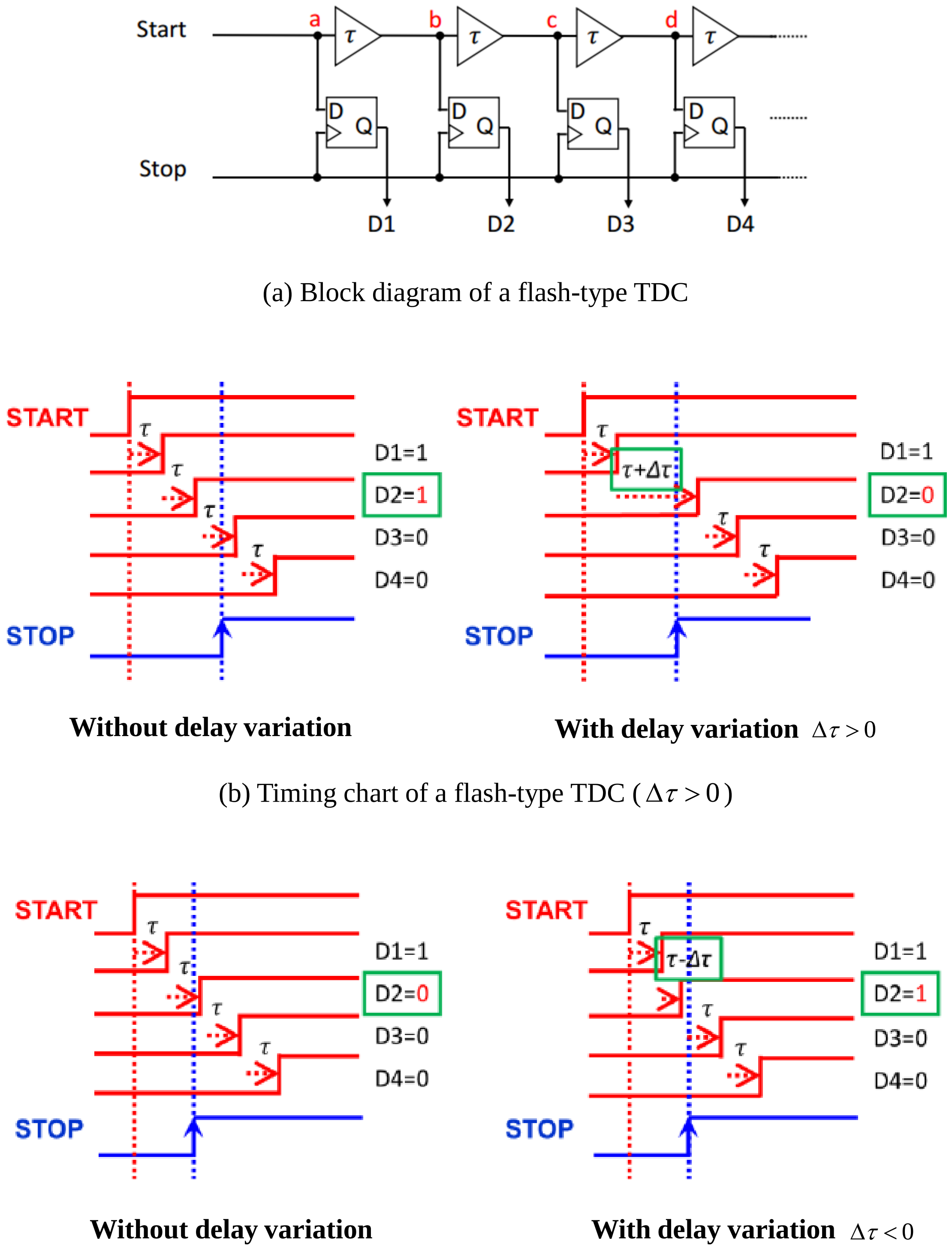
The measurement range of the TDC based on Vernier delay technique, i.e. the maximum time that can be measured, can be denoted by $t_{DR}=n\cdot(\tau 1-\tau 2)$, where n is the number of Vernier stages [4].

Although the Vernier delay line TDC improves the resolution effectively, the area and power consumption is increased dramatically as the dynamic range becomes larger due to that each stage costs two buffers and one flip-flop [5].

2.3.3 Self-Calibration Technology

Though the flash-type TDC has the advantages such as high-speed timing measurement, single-event timing measurement, and all digital implementation, its overall linearity will degrade when there are mismatches among the delay stages due to device and circuit characteristics variation.

For example, Fig.2.5 illustrates a timing chart for the flash-type TDC, where τ is average delay, $\Delta\tau$ is deviation from τ, and D is TDC digital output.



(a) Block diagram of a flash-type TDC

Without delay variation

With delay variation $\Delta\tau > 0$

(b) Timing chart of a flash-type TDC ($\Delta\tau > 0$)

Without delay variation

With delay variation $\Delta\tau < 0$

(c) Timing chart of a flash-type TDC ($\Delta\tau < 0$)

Fig. 2. 5 Timing diagram of TDC circuit

In Fig.2.5 (b), the left figure shows the timing chart when there is no mismatch among the delay stages and all delay elements have the same delay τ. As the time interval of START and STOP signal is between 2τ and 3τ, the output D1, D2, D3, D4 are 1, 1, 0, 0 respectively. On the other hand, the right figure illustrates the timing chart when there are mismatches among the delay stages: the second delay element has a delay variation of $\Delta\tau > 0$. The output D2 becomes 0 and the TDC linearity is degraded.

In Fig.2.5 (c), the left figure shows the timing chart when there is no mismatch among the delay stages and all delay elements have the same delay τ. As the time interval of START and STOP signal is between τ and 2τ, the output D1, D2, D3, D4 are 1, 0, 0, 0 respectively. On the other hand, the right figure illustrates the timing chart when there are mismatches among the delay stages: the second delay element has a delay variation of $\Delta\tau < 0$. The output D2 becomes 1 and the TDC linearity is degraded.

Therefore, a self-calibration technique is proposed to compensate for this nonlinearity [3][8][9].

The flash-type TDC with self-calibration technology is illustrated in Fig.2.6 [10][11]. In Fig.2.6, the self-calibration circuit consists of an upper ring oscillator, D flip-flop array, and histogram engine/digital error correction circuit. The upper ring oscillator consists of several delay buffers (in Fig.2.6, the number of delay buffers is 24) and one inverter. The select signal of the multiplexer determines the handover between calibration mode and measurement mode (normal operation mode). In calibration mode, under the control of the multiplexer select signal, the delay line output is connected to its input and the upper ring oscillator is configured. The histogram engine is used to acquire histogram data. On the other hand, in measurement mode, under the control of the multiplexer select signal, the START signal is connected to the delay line input and the upper ring oscillator closed-loop is open. The digital error correction circuit is applied to conduct digital error correction operation based on the obtained histogram data in calibration mode.

The flow chart of self-calibration and digital error correction is illustrated in Fig.2.7. In calibration mode, the histogram engine measures the relative variation (ratio) among the delay elements [10][11][12]. In measurement mode, the digital error correction circuit corrects the TDC digital output and compensates for the nonlinearity, i.e. improves the TDC linearity correspondingly.

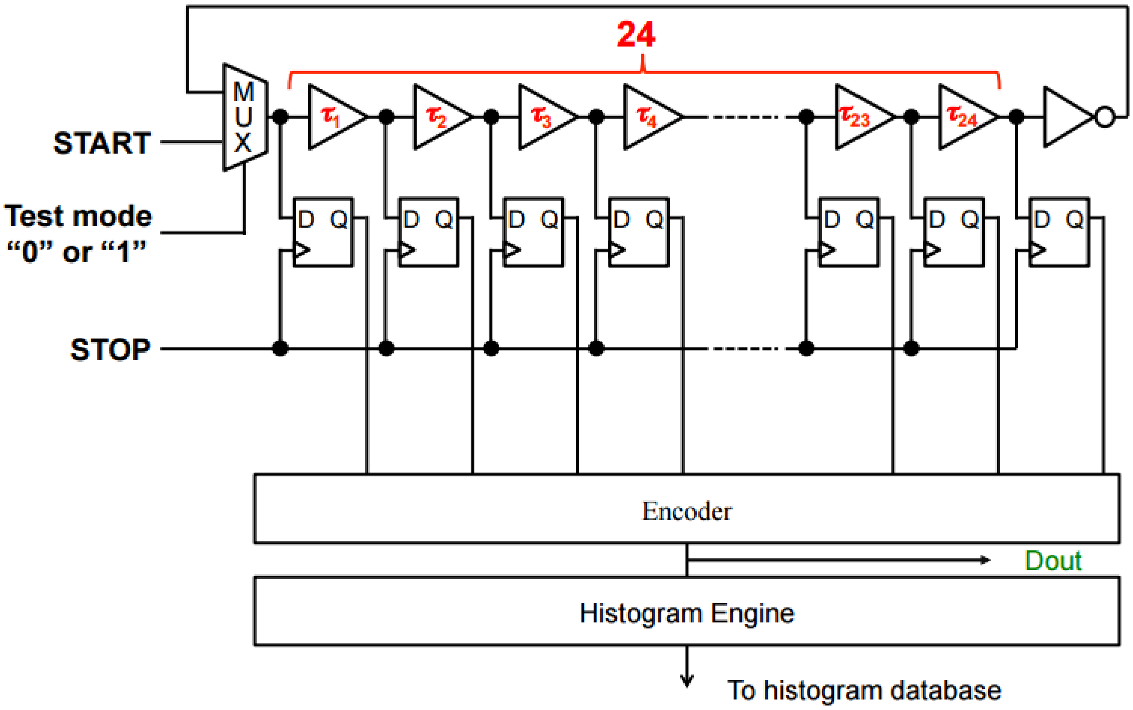


Fig. 2. 6 The self-calibration technology

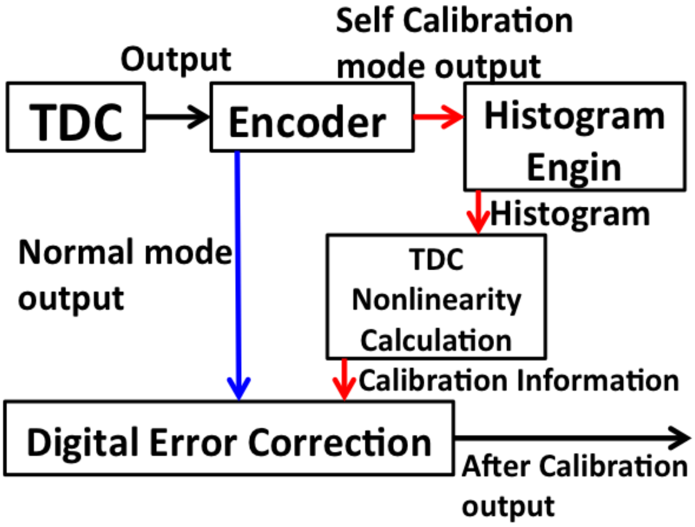


Fig. 2. 7 Flow chart of calibration and digital error correction

(1) Calibration Mode (Histogram Data Acquisition)

In calibration mode, under the control of the multiplexer select signal, the delay line output is connected to its input and the upper ring oscillator is configured. The upper ring oscillator runs freely with an oscillating frequency f_1, while an external clock with frequency f_2 is inputted as the STOP signal. The ring oscillator and the external clock are asynchronous (without correlation), i.e. $f_1 \neq f_2$. When the external clock signal is on the positive edge, the D flip-flop array are triggered. The outputs of the D flip-flop array are transferred to the encoder, and the histogram for each bin (digital output) is computed.

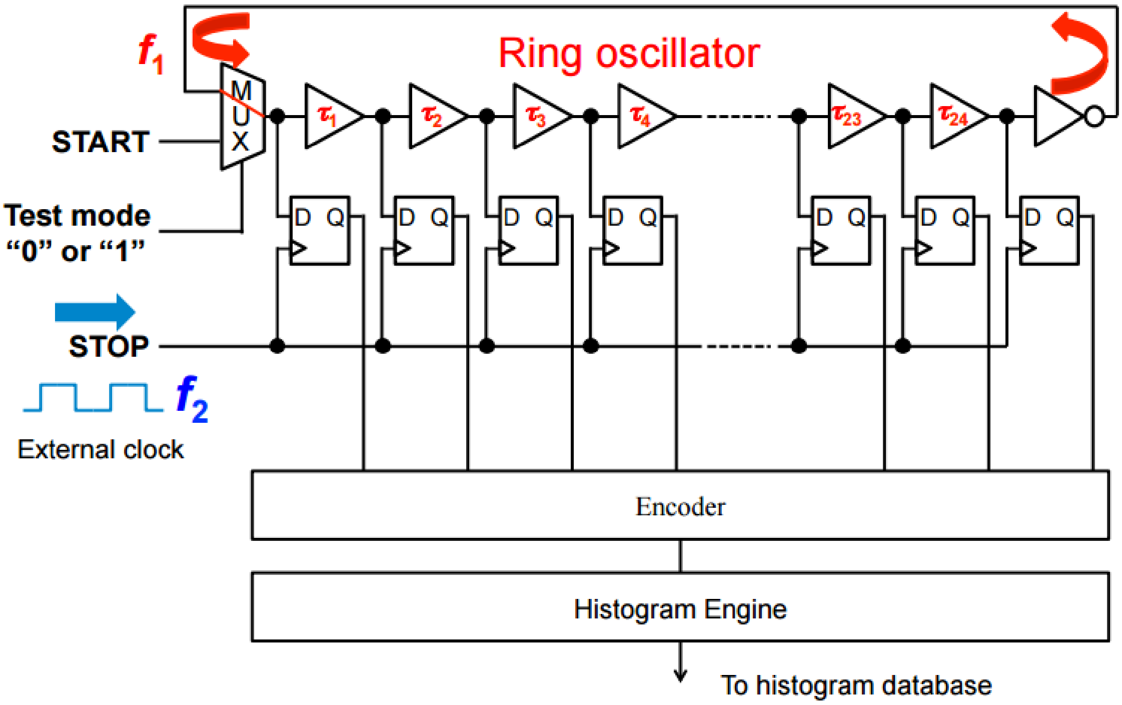


Fig. 2. 8 Self-calibration mode

If the TDC has perfect linearity, i.e. there is no mismatch among the delay elements, the histogram for each bin (digital output) would be equal, after sufficiently large measurement times. However, in practice, variations among delay elements exist. When the upper ring oscillator runs freely, the external clock signal rising edge triggers a digital output. As the probability of digital code for large delay is high, while the probability of digital code for small delay is low. The digital output is with the probability proportional to the delay value of the corresponding delay element [10][11][12]. Thus the each delay (relative) value can be measured by the histogram engine. After large enough measurement cycles, each bin of histogram varies with corresponding delay value.

(2) Measurement Mode (Normal Operation Mode)

In measurement mode, under the control of the multiplexer select signal, the START signal is connected to the delay line input and the upper ring oscillator closed-loop is open. In this situation, the START signal and the STOP signal are inputted as a normal flash-type TDC, and the thermometer code corresponding to the rising-edge timing interval between them is outputted.

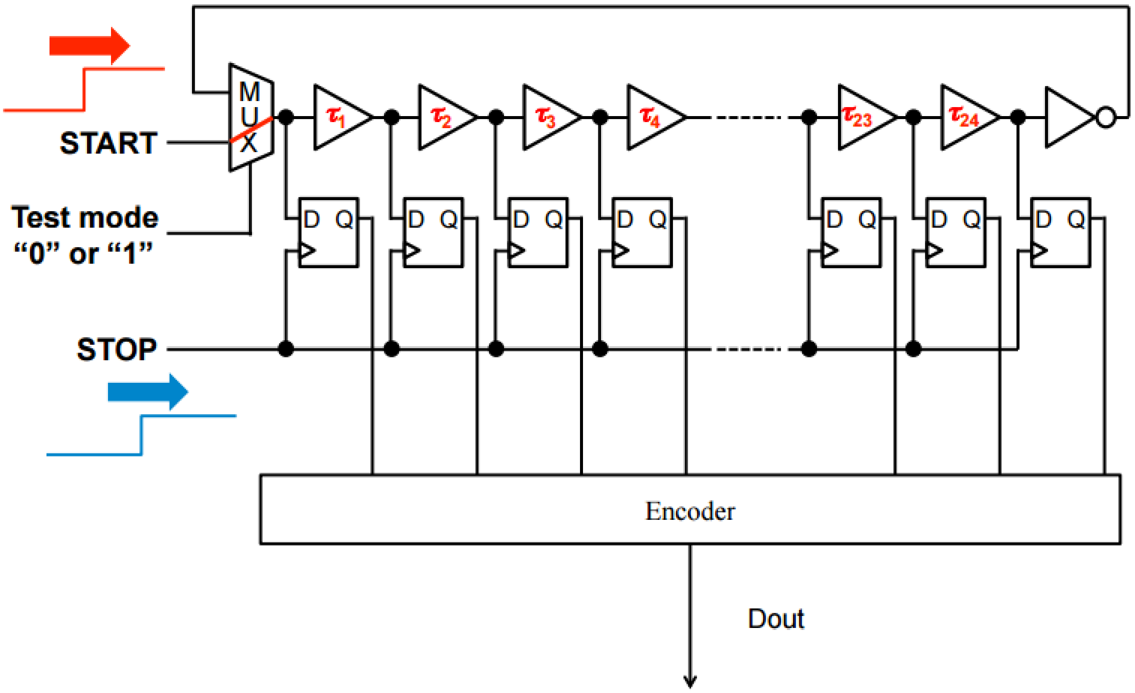


Fig. 2. 9 Measurement mode (normal operation mode)

(3) Digital Error Correction Operation

The histogram data of each (relative) delay value in the delay elements is acquired in calibration mode. In measurement mode, the digital error correction operation is conducted based on the acquired histogram data, as shown in equation (2-6).

$$Dout(N) = \frac{\sum_{i=1}^{N} Pin(i)}{\sum_{i=1}^{FS} Pin(i)} \times FS \quad (2-6)$$

N : digital output to be corrected

$D_{out}(N)$: corrected digital output for raw TDC output N

$Pin(i)$: histogram for raw TDC output i

FS : maximum TDC digital output value

The principle of self-calibration is illustrated in Fig.2.10.

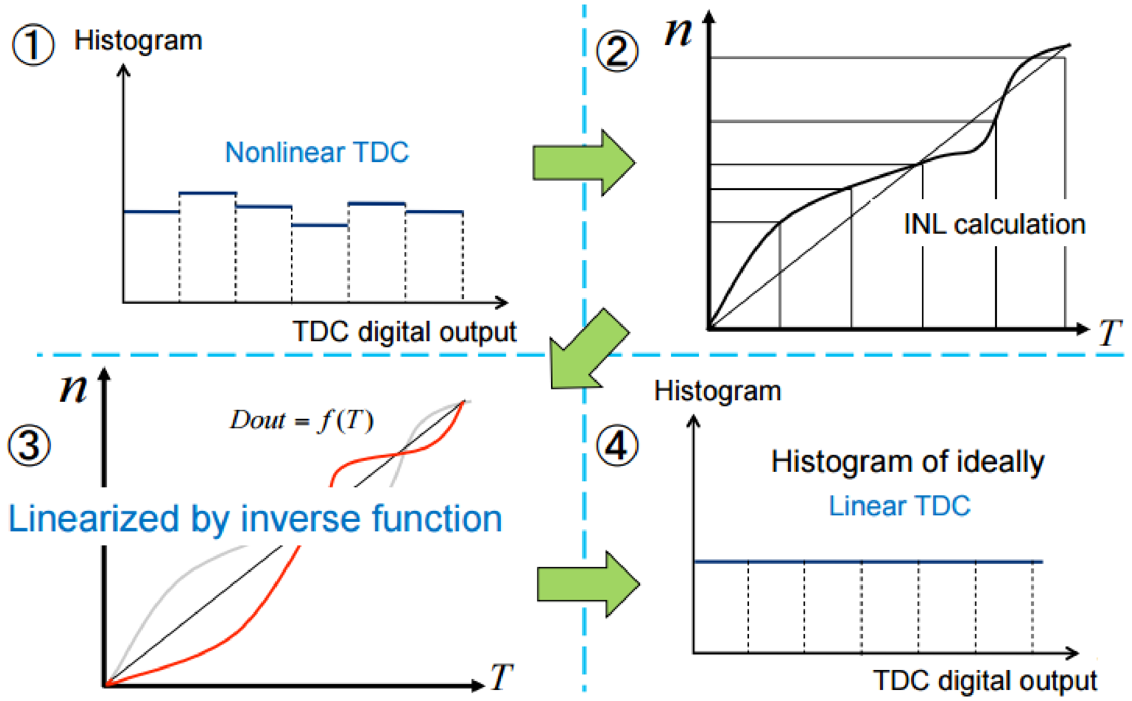


Fig. 2. 10 Principle of self-calibration

As START (ring oscillator) and STOP signals are asynchronous (without correlation), if the TDC has perfect linearity, the histogram for each bin (digital output) would be equal, after sufficiently large measurement times. However, in practice, histogram data in each bin will vary and be different from each other as variations among delays exist. Probability of digital code for large delay is high, while probability of digital code for small delay is low. Each bin of histogram varies with corresponding delay value. According delay variations, DNL can be easily measured, and INL can be calculated from DNL data. Non-linearity correction can be made by applying inverse function, and the TDC linearity calibration can be achieved.

2.4 Summary

Conventional TDC architectures have been described in this chapter.

The flash-type TDC uses a delay line which consists of CMOS inverter buffer delays and D Flip-Flops (DFFs). The START signal is connected to the delay line input and passes through the serial delay elements. The output array of the delay buffers are used as the data input for the D flip-flop array. The STOP signal is used as the clock signal of the D flip-flop array. The outputs of the D flip-flop array are transferred to the thermometer-code-to-binary encoder. The encoder produces a digital output representing the time interval between the START signal and the STOP signal. The measured time interval between the START and STOP signals is equal to a certain number of steps of buffer delay. However, the flash-type has disadvantages on circuit complexity, as well as non-linearity due to buffer delay mismatches.

The Vernier structure consists of a pair of tapped delay lines with a flip–flop at each corresponding pair of taps. The buffer delay $\tau 1$ in the upper delay line is slightly greater than the buffer delay $\tau 2$ in the lower delay line. The START and STOP signal propagate through the upper delay line and the lower delay line respectively. The time resolution is equal to the buffer delay difference $(\tau 1-\tau 2)$, which can be smaller than that of the flash-type TDC. However, for a measurement range from 0 to $N(\tau 1-\tau 2)$, the Vernier structure requires 2N buffers (N buffers of $\tau 1$ and N buffers of $\tau 2$), which leads to large chip area and power consumption. Furthermore, its monotonicity is not guaranteed and it may show some non-linearity due to buffer delay mismatches.

The buffer delay relative mismatches due to device and circuit characteristics variation can degrade the linearity of the output. The self-calibration technique is applied to compensate for this linearity degradation. The calibration circuit can be realized by using a delay-locked loop based on ring oscillator architecture.

References

[1] Wu Gao, Deyuan Gao, Christine Hu-Guo, Yann Hu, “Integrated High-Resolution Multi-Channel Time-to-Digital Converters (TDCs) for PET Imaging”, InTechOpen.

[2] Stephen A. Dyer, “Wiley Survey of Instrumentation and Measurement”, March 2004, Wiley-IEEE Press.

[3] Takeshi Chujo, Junshan Wang, Daiki Hirabayashi, Congbing Li, Yutaro Kobayashi, Kentaroh Katoh, Haruo Kobayashi, Masanobu Tsuji, Koshi Sato, “FPGA Evaluation of Flash-type TDC With Histogram Method for Linearity Self-Calibration”, Advanced Micro-Device Engineering VI, Key Engineering Materials (2016).

[4] G. S. Jovanovic, M. K. Stojcev, “Vernier’s Delay Line Time-to-Digital Converter”, Scientific Publications of The State University of Novi Pazar Series A: Applied Mathematics Information and Mechanical, vol.1, no.1, pp.11-20, 2009.

[5] Chen Yao, Stockholm, “Time to Digital Converter used in ALL digital PLL”, Master of Science Thesis, In System-on-Chip Design, August, 2011.

[6] K. Blutman, J. Angevare, A. Zjajo, N.P. van der Meijs, “A 0.1 pJ Freeze Vernier time-to-digital converter in 65nm CMOS”, IEEE International Symposium on Circuits and Systems (ISCAS), pp.85-88, June 2014.

[7] R. Rashidzadeh, M. Ahmadi, and W. C. Miller, “An all-digital selfcalibration method for a vernier-based time-to-digital converter”, IEEE Transactions on Instrumentation and Measurement, vol.59, no.2, pp.463–469, 2010.

[8] Junshan Wang, Kentaroh Katoh, Congbing Li, Ensi Li, Yutaro Kobayashi, Takeshi Chujo, Daiki Hirabayashi, Haruo Kobayashi, “Digital FPGA Implementation of TDC With Self-Calibration”, The 3rd Solid State Systems Symposium-VLSIs and Semiconductor Related Technologies & The 17th International Conference on Analog VLSI Circuits, Ho Chi Minh City, Vietnam (Oct. 22-24, 2014).

[9] Takeshi Chujo, Daiki Hirabayashi, Kentaroh Katoh, Congbing Li, Yutaro Kobayashi, Junshan Wang, Koshi Sato, Haruo Kobayashi, “FPGA Evaluation of Flash-type TDC With Histogram Method Self-Calibration”, The 3rd Solid State Systems Symposium-VLSIs and Semiconductor Related Technologies & The 17th

International Conference on Analog VLSI Circuits, Ho Chi Minh City, Vietnam (Oct. 22-24, 2014).

[10] Kentaroh Katoh, Yutaro Kobayashi, Takeshi Chujyo, Junshan Wang, Ensi Li, Congbing Li, Haruo Kobayashi, “A Small Chip Area Stochastic Calibration for TDC Using Ring Oscillator”, Journal of Electronic Testing: Theory and Applications, vol.30, issue 6, pp.653-663, Springer (Dec. 2014).

[11] Satoshi Ito, Shigeyuki Nishimura, Haruo Kobayashi, Satoshi Uemori, Yohei Tan, Nobukazu Takai, Takahiro J. Yamaguchi, and Kiichi Niitsu, “Stochastic TDC Architecture with Self-Calibration”, IEEE Asia Pacific Conference on Circuits and Systems (APCCAS 2010), pp.1027-1030, December 2010.

[12] J. Rivoir, “Fully-Digital Time-to-Digital Converter for ATE with Autonomous Calibration”, IEEE International Test Conference, pp.1–10, Santa Clara, CA (Oct. 2006).

Chapter 3
PARALLEL RING OSCILLATOR TDC

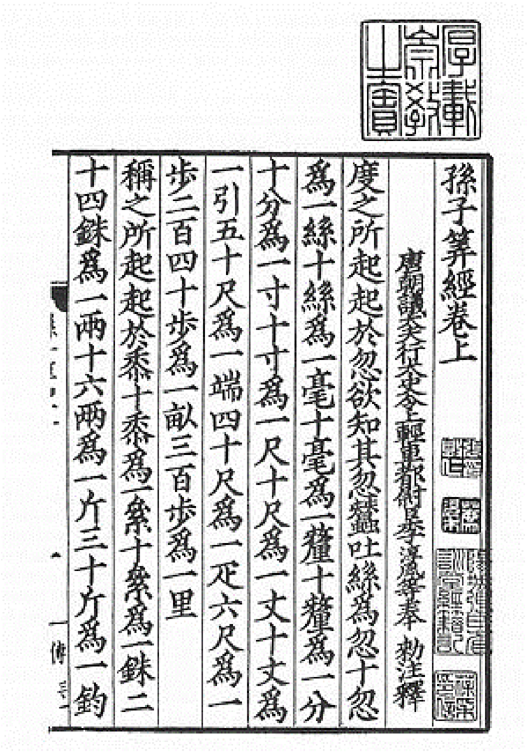
In this chapter, residue number system, Gray code, and cyclic code are applied in parallel ring oscillator TDC architectures. Section 3.1 introduces a residue number system based TDC architecture, which represents a TDC circuit with large measurement range using a set of smaller ring oscillator TDCs. It can reduce the number of delay cells and flip-flops significantly compared to a flash-type TDC, while keeping a comparable performance. Then, a TDC architecture based on Gray code is proposed in section 3.2. The proposed Gray code TDC can provide a glitch-free binary code sequence, i.e. no out-of-sequence code, even there are some amounts of mismatches among the delay stages. Then, a Gray code based TDC with cyclic code is presented in section 3.3. Cyclic code can be applied to generate the lower bits of Gray code, which can reduce the frequency of the outputs, i.e. to reduce the parasitic capacitance among circuit elements. Finally, the summary is provided in the last section.

3.1 Residue Number System Based TDC

3.1.1 Residue Number System

As the original form of the theorem appears in the 5th-century book Sunzi's Mathematical Classic by the Chinese mathematician Sun Tzu, residue numeral system is called the "Chinese remainder theorem" in the West. It deals with the remainder from the division of one number by another number. Sunzi's Mathematical Classic includes the following problem: “A number is divisible by 3 with remainder 2, by 5 with remainder 3, and by 7 with remainder 2. What is the number?” The book solves this problem as follows: “Numbers that are divisible by 3 with remainder 2 include 140, those that are divisible by 5 with remainder 3 include 63, and those that are divisible by 7 with remainder 2 include 30. Adding these three numbers together produces 233. Subtracting 105 from this 233 and continuing to subtract 105 from the previous remainder produces the numbers 128, 23, ... Among the results, 23 is the minimum number that satisfies the conditions.” The number 105 is derived by product of the divisors 3, 5, and 7.

As at the last stage for seeking the minimum positive answer one should subtract by 105, in Japan, Seki Takakazu, a Japanese mathematician in the Edo period, named the remainder problem “Hyakugo Genzan” [1].



孫子算經卷上
唐朝議大夫行太史令上輕車都尉臣李淳風等奉 勅注釋
度之所起起於忽欲知其忽蠶吐絲爲忽十忽
爲一絲十絲爲一毫十毫爲一氂十氂爲一分
十分爲一寸十寸爲一尺十尺爲一丈十丈爲
一引五十尺爲一端四十尺爲一疋六尺爲一
步二百四十步爲一畝三百步爲一里
稱之所起起於黍十黍爲一絫十絫爲一銖二
十四銖爲一兩十六兩爲一斤三十斤爲一鈞

(a) Sunzi's Mathematical Classic

(b) Seki Takakazu

Fig. 3.1 The ancient study of the residue number system

Residue number system is a result about congruences in number theory and its generalizations in abstract algebra. It represents a large integer using a set of smaller integers, so that computation may be performed more efficiently. Residue number system have applications in the field of digital computer arithmetic. By decomposing in this a large integer into a set of smaller integers, a large calculation can be performed as a series of smaller calculations that can be performed independently and in parallel.

In general, suppose that $m_1, m_2, ..., m_r$ are positive integers and coprime each other. Then there is unique positive integer x for given integers ($a_1, a_2, ..., a_r$) which satisfies the following:

$$x \equiv a_k (\mathrm{mod}\, m_k), \quad k = 1,2,...,r \quad (3-1)$$

where $0 \leqq a_k < m_k$, $0 \le x < N$ $(N = m_1 \cdot m_2 \cdots m_r)$. Table 3.1 shows the case of $m_1 = 2$, $m_2 = 3$, $m_3 = 5$ and $N = 2 \times 3 \times 5 = 30$, and we see that each x is mapped to residues of (m_1, m_2, m_3) one to one [2].

Table 3.1 An integer x and residues of (m_1, m_2, m_3)

m_1	m_2	m_3	x
0	0	0	0
1	1	1	1
0	2	2	2
1	0	3	3
0	1	4	4
1	2	0	5
0	0	1	6
1	1	2	7
0	2	3	8
1	0	4	9
0	1	0	10
1	2	1	11
0	0	2	12
1	1	3	13
0	2	4	14
1	0	0	15
0	1	1	16
1	2	2	17
0	0	3	18
1	1	4	19
0	2	0	20
1	0	1	21
0	1	2	22
1	2	3	23
0	0	4	24
1	1	0	25
0	2	1	26
1	0	2	27
0	1	3	28
1	2	4	29

3.1.2 Natural Time-Domain Residue Generator

In TDC, the signal is treated as “time” instead of “voltage”. Ring oscillators can be applied to obtain the residue easily. As illustrated in Fig.3.2, in a ring oscillators, the output array of the delay buffers since oscillating is time-dependent, and cycles every the ring oscillator period. The output array of the delay buffers can be regarded as the residue, and the elapsed time since oscillating can be back-calculated from this residue. So a ring oscillator can be regarded as a natural time-domain residue generator.

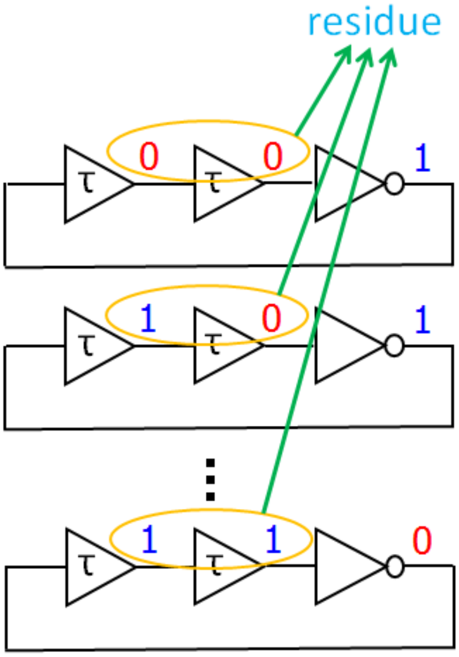


Fig. 3.2 Ring oscillator: natural time-domain residue generator

3.1.3 Residue Number System Based TDC Architecture

For a conventional n-bit flash-type TDC with 2^n quantization levels, at least a total of $2^n - 1$ delay elements and $2^n - 1$ DFFs are required. For a measurement range, i.e. n is large, this circuit complexity leads to large chip area and power consumption. Residue number system represents a large integer using a set of smaller integers, so that computation may be performed more efficiently. In the TDC circuit, the signal is “time” instead of “voltage”, and the residue can be easily obtained with a ring oscillator. Applying residue number system concept at circuit design, we can represent a TDC circuit with large measurement range using a set of smaller ring oscillator TDCs, so the hardware, power consumption, as well as chip area can be reduced correspondingly.

A residue number system based TDC architecture can be conceived by grouping a few ring oscillators to operate on the same input. Fig.3.3 shows the proposed residue number system based TDC in the case of $m_1 = 2\tau$, $m_2 = 3\tau$, $m_3 = 5\tau$. The residues $a_1 \pmod{2\tau}$, $a_2 \pmod{3\tau}$, $a_3 \pmod{5\tau}$ are obtained with three different ring oscillators with the same input.

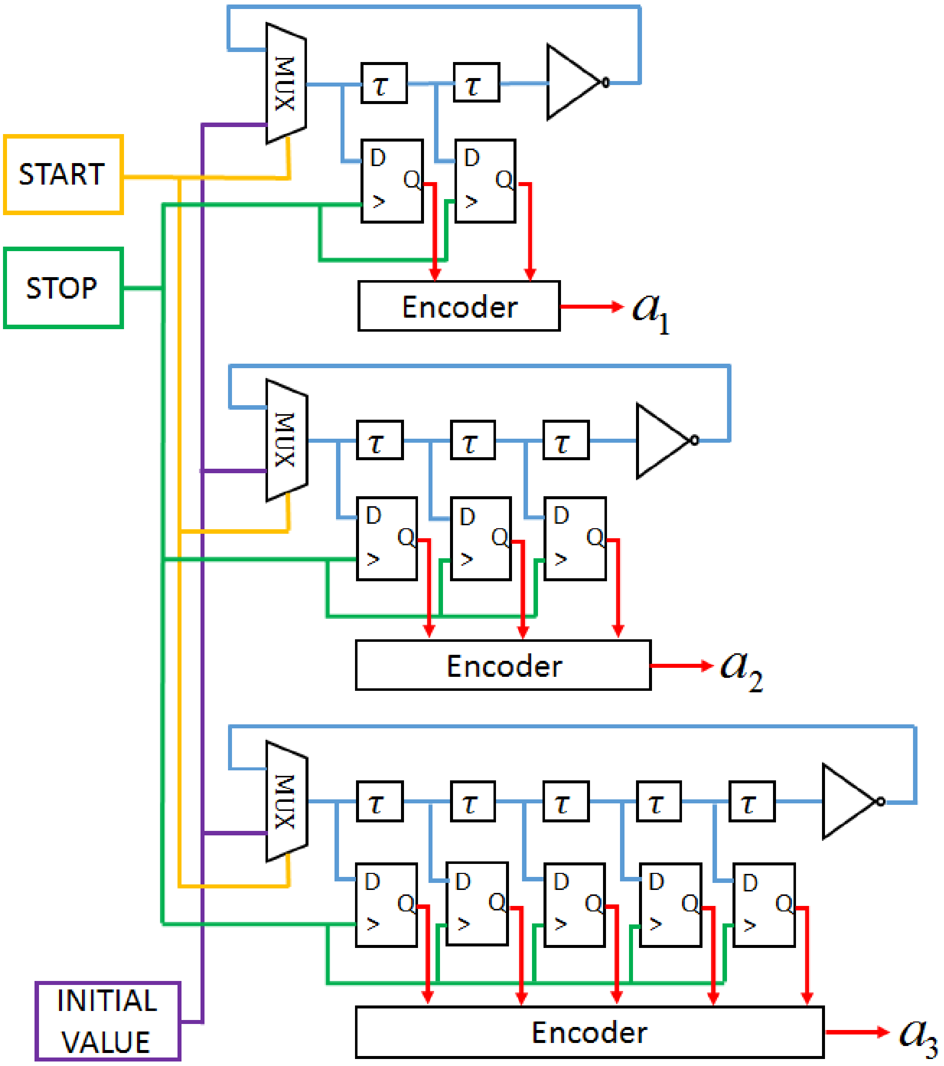


Fig. 3.3 Proposed residue number system based TDC architecture

This TDC can measure the time elapsed between START and STOP incoming pulses and output the residues a_1, a_2, a_3 as follows:

(1) When START signal is in LOW state, three ring oscillators are initialized by Initial Value.

(2) When START signal goes from LOW to HIGH, three ring oscillators begin to oscillate.

(3) When STOP signal is on the rising edge, i.e. LOW-to-HIGH transition, the DFFs are triggered and the value of D is transferred to the output Q.

(4) The Q values are transferred into the residues $a_1 \pmod{2\tau}$, $a_2 \pmod{3\tau}$, $a_3 \pmod{5\tau}$ by the encoders.

(5) The corresponding x can be achieved from "a_1, a_2, a_3" based on the residue number system. So the time interval between START and STOP is equal to $x \cdot \tau$.

RTL simulation was conducted to verify the characteristics of the residue number system based TDC in Fig.3.3. The delay of each buffer is equal to 10ns, and START signal goes from LOW to HIGH at 100ns. RTL simulation waveforms are illustrated in Fig.3.4. We can see that the proposed residue number system based TDC works as expected in the time domain.

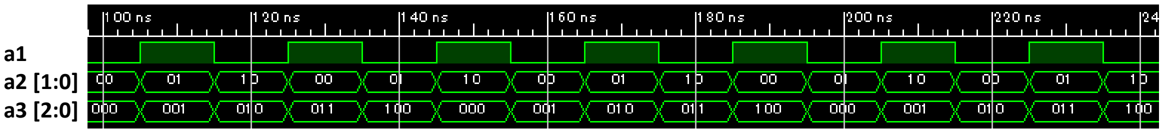


Fig. 3.4 RTL simulation waveform of residue number system based TDC

Note that the proposed TDC uses only 10 delay cells and 10 Flip-Flops while the corresponding flash-type TDC requires 30 delay cells and 30 Flip-Flops. In general, the proposed TDC uses M delay cells and M Flip-Flops (where $M = m_1 + m_2 + \cdots + m_r$, additive increase) while the corresponding flash-type TDC uses N delay cells and N Flip-Flops (where $N = m_1 \cdot m_2 \cdots m_r$, multiplicative increase).

Table 3.2 Residue number system based TDC vs. Flash-type TDC

	Number of delay cells	Number of DFFs	Maximum stage
Residue number system based TDC	$m_1+m_2+\cdots+m_r$	$m_1+m_2+\cdots+m_r$	Maximum of （$m_1, m_2,..., m_r$）
Flash-type TDC	$m_1 \cdot m_2 \cdots m_r$	$m_1 \cdot m_2 \cdots m_r$	$m_1 \cdot m_2 \cdots m_r$

For example, a residue number system based TDC with moduli {5, 7, 9} can realize 5×7×9 = 315 quantization levels with only 5+7+9=21 delay cells and 21 Flip-Flops, while the corresponding flash-type TDC needs 315 delay cells and 315 Flip-Flops. So the proposed residue number system based TDC is especially suitable for TDC architecture with large time measurement range requirement: the number of delay cells and Flip-Flops in the proposed TDC decreases rapidly (M<<N) compared with flash-type TDC, which reduces the hardware and chip area significantly.

Furthermore, the maximum stage of TDC architecture is also reduced rapidly in the proposed TDC. The use of shorter delay lines reduces the integral non-linearity caused by mismatches between the delay stages.

3.1.4 FPGA Implementation

A proof-of-concept residue number system based TDC is implemented on Xilinx Virtex-6 FPGA ML605 Evaluation Kit. Input “START” and “Initial Value” and connected to user push buttons, and “STOP” is connected to 200MHz FPGA clock. Each buffer is realized by a delay Flip-Flop with 100MHz clock frequency, i.e. the delay of each buffer is equal to 10ns. Encoder output “ a_1 , a_2 (consists of $a_2[0]$ and $a_2[1]$), a_3 (consists of $a_3[0]$, $a_3[1]$ and $a_3[2]$)” are delivered to user LEDs.

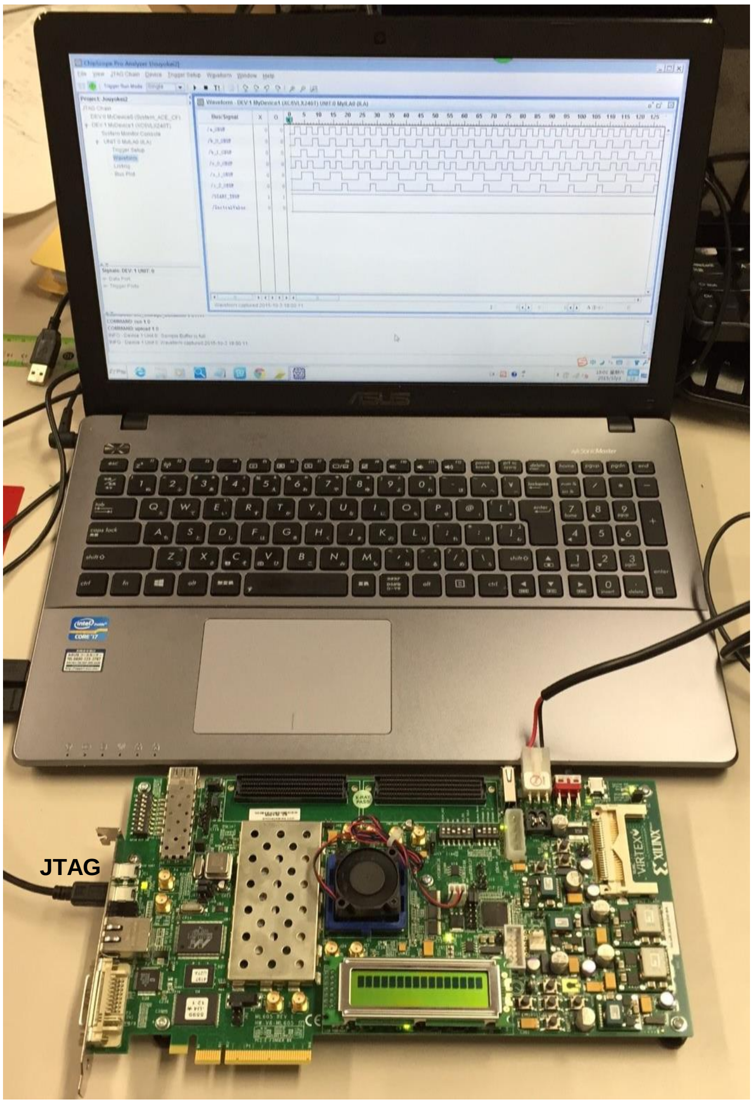


Fig. 3.5 Residue number system based TDC implementation on FPGA

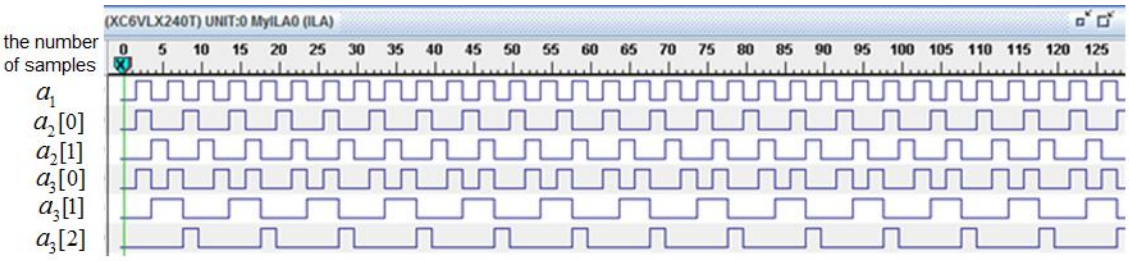


Fig. 3.6 FPGA output waveforms of a_1, a_2, a_3

The measurement starts by pushing the START push button, which produces a rising edge “START” signal. Chipscope is applied to probe the internal signal of FPGA. The

FPGA output waveforms of a_1, a_2 (consists of $a_2[0]$ and $a_2[1]$), a_3 (consists of $a_3[0]$, $a_3[1]$ and $a_3[2]$) are illustrated in Fig.4. The number of samples represents for the ID of Chipscope output waveform samples. Chipscope sampling rate is 200MHz, and the time interval between two waveform samples is 5ns. The corresponding x can be achieved from "a_1, a_2, a_3" based on the residue number system. So the elapsed time between START and STOP is equal to $x \cdot 10ns$.

In Fig.3.7, the horizontal axis stands for the time interval between START and STOP, while the vertical axis represents the output of the proposed TDC. We can see that the proposed residue number system based TDC works with good linearity as expected.

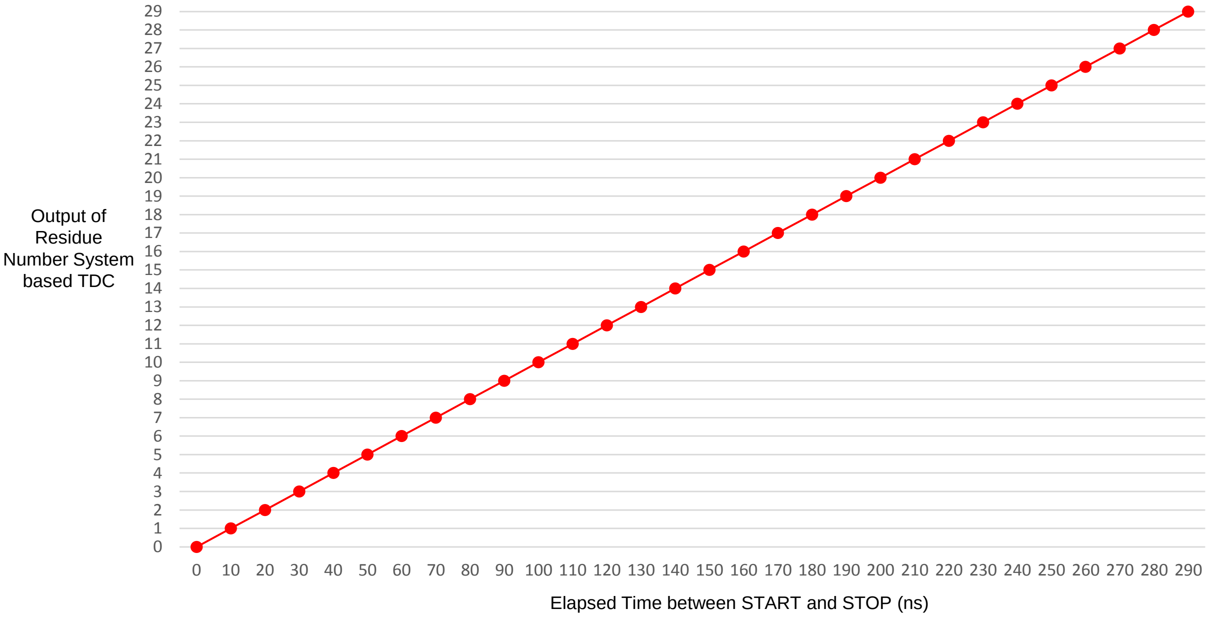


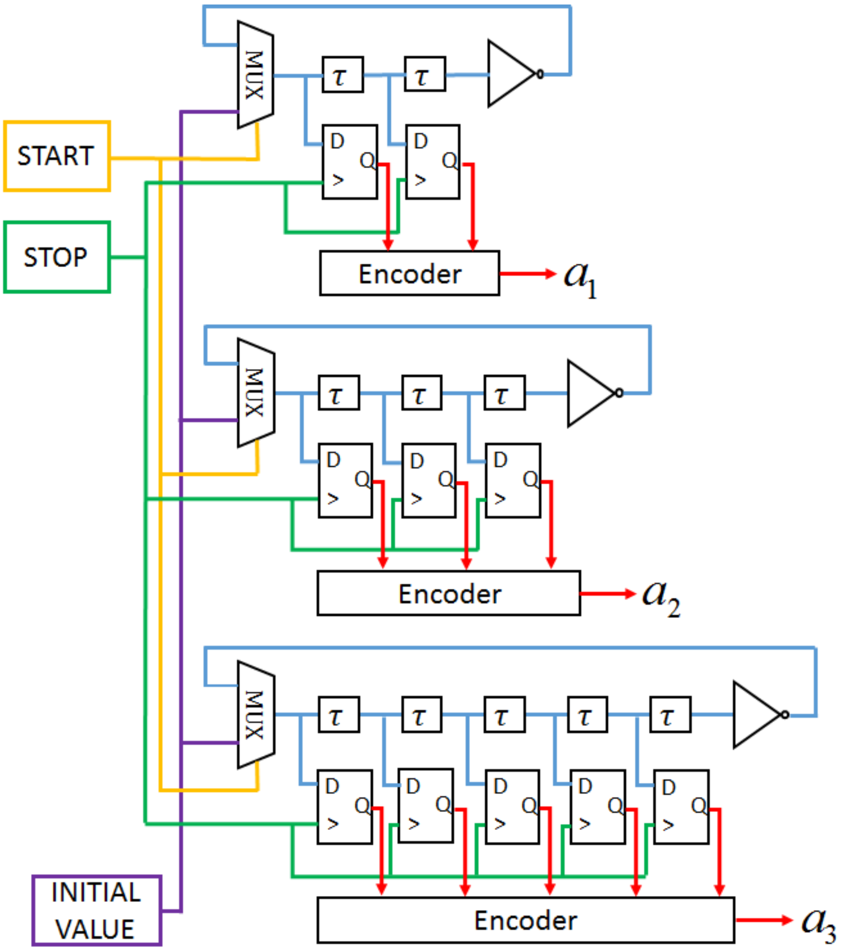
Fig. 3.7 Measured characteristics of residue number system based TDC

3.1.5 Drawbacks of Residue Number System Based TDC

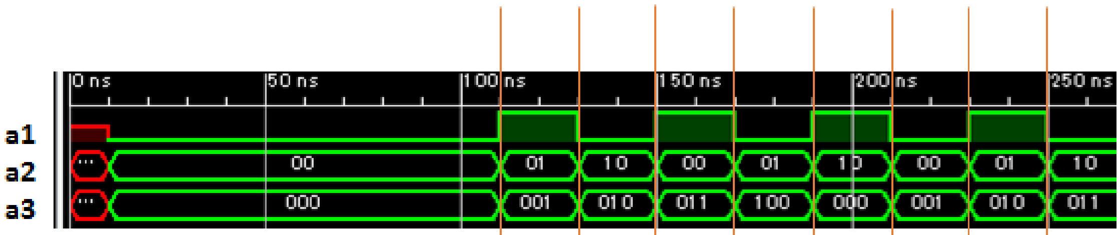
Although with the residue number system based TDC architecture, small chip area and low power consumption can potentially be achieved, glitches (i.e. out-of-sequence codes) may occur when there are mismatches between the delay stages. This glitch problem is due to the fact that when x changes from d to $d+1$, all of $a_1, a_2, \ldots, a_r$ values have to change simultaneously.

However if there are delay mismatches among the ring oscillators, i.e. some of $a_1, a_2, \ldots, a_r$ may change faster than the others, it may produce a glitch on the boundary when x changes from d to $d+1$.

For example, Fig.3.8 (a) shows the residue number system based TDC where three ring oscillators have the same buffer delay $\tau = 20ns$, and START signal goes from LOW to HIGH at 100ns. Fig.3.8 (b) shows the RTL simulation waveforms and Fig.3.8 (c) shows the output digital codes achieved when there are no mismatches among the delay stages. In this case, the outputs of (a_1, a_2, a_3) always change at the same time. The derived x is what we expected.

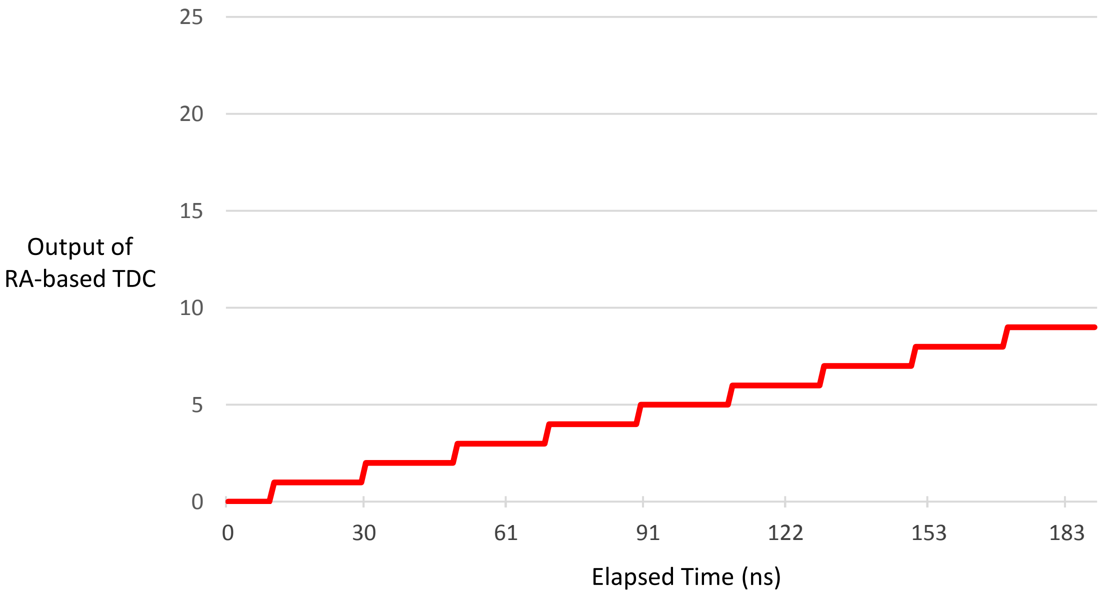


(a) Residue number system based TDC without mismatches among delay cells



a1	0	1	0	1	0	1	0	1
a2	0	1	2	0	1	2	0	1
a3	0	1	2	3	4	0	1	2
x	0	1	2	3	4	5	6	7

(b) RTL simulation waveforms

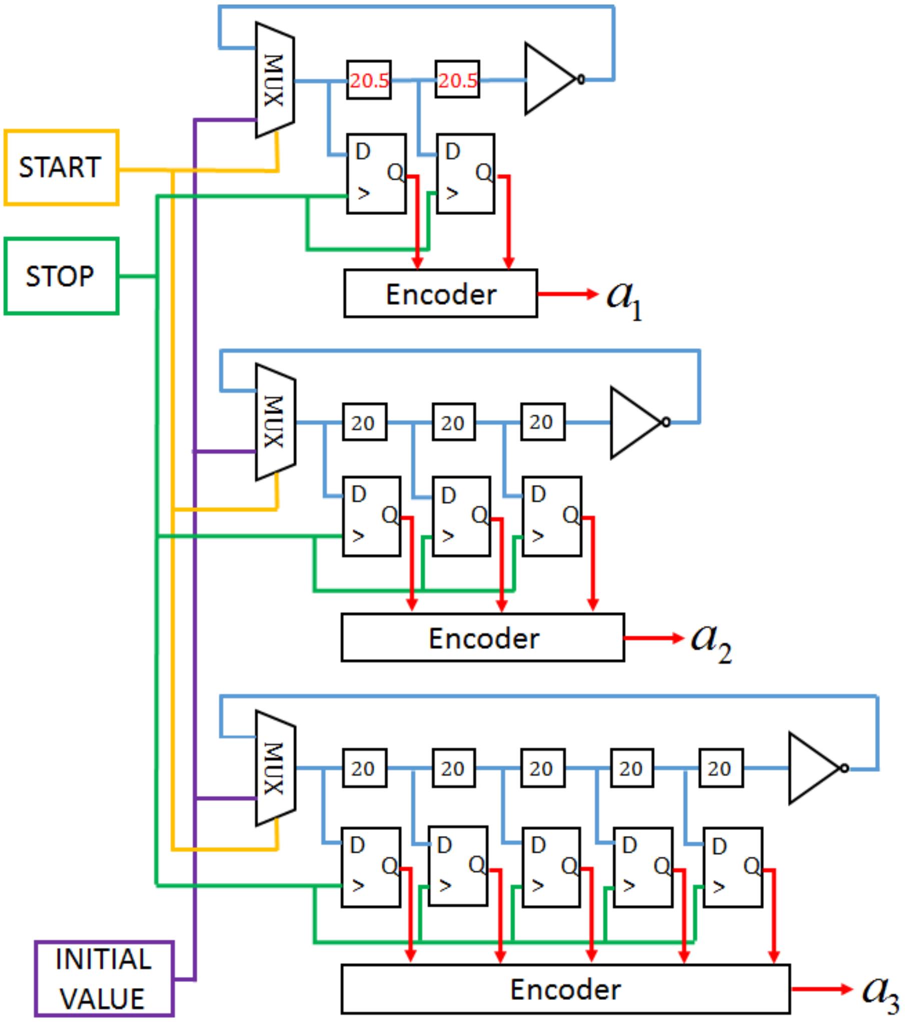


(c) Input-output characteristics: no mismatches among the delay stages

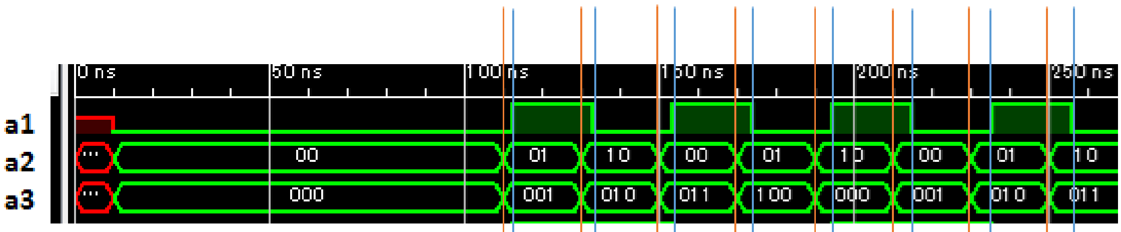
Fig. 3.8 Residue number system based TDC without mismatches among delay cells in ring oscillators

Fig.3.9 (a) illustrates the residue number system based TDC where three ring oscillators have different buffer delays, i.e. the buffer delay in a_1 generation ring oscillator is 20.5ns, while the buffer delay in a_2 and a_3 generation ring oscillators is 20ns. The START signal goes from LOW to HIGH at 100ns. Fig.3.9 (b) illustrates the RTL simulation waveforms and Fig.3.9 (c) illustrates the output digital codes generated when mismatches exist among the delay stages. In this case, and the outputs of

(a_1, a_2, a_3) do not change at the same time. This causes some glitches in the output of x, which is marked in red in Fig.3.9 (b). As the unsynchronization error accumulates, the width of glitches increases, which is shown in Fig.3.9 (c).



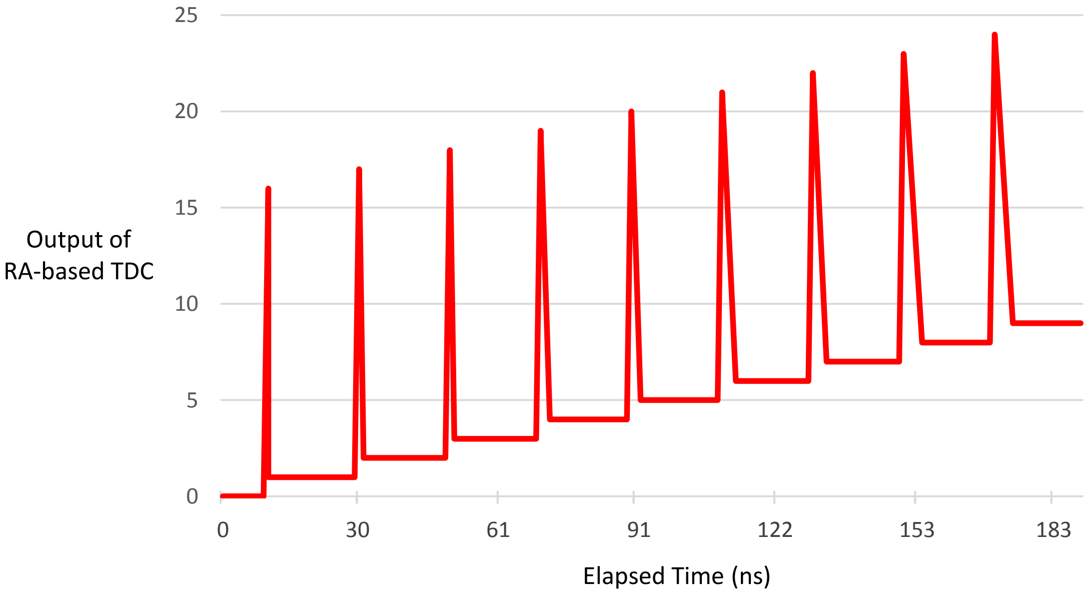
(a) Residue number system based TDC with mismatches among delay cells



a1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a2	0	1	1	2	2	0	0	1	1	2	2	0	0	1	1	2
a3	0	1	1	2	2	3	3	4	4	0	0	1	1	2	2	3
x	0	16	1	17	2	18	3	19	4	20	5	21	6	22	7	23

glitches caused by unsynchronization

(b) RTL simulation waveforms



(c) Input-output characteristics: mismatches exist among the delay stages (large glitches are observed)

Fig. 3.9 Residue number system based TDC with mismatches among delay cells in ring oscillators

We can see that glitches occur in the time-domain when there are mismatches among the delay stages, which triggers an instability in the output digital codes. Due to this, a parallel ring oscillator TDC architecture based on Gray code is proposed to reduce the hardware and chip power consumption significantly and remove the glitches effectively compared to the flash-type TDC and residue number system based TDC, while keeping comparable performance.

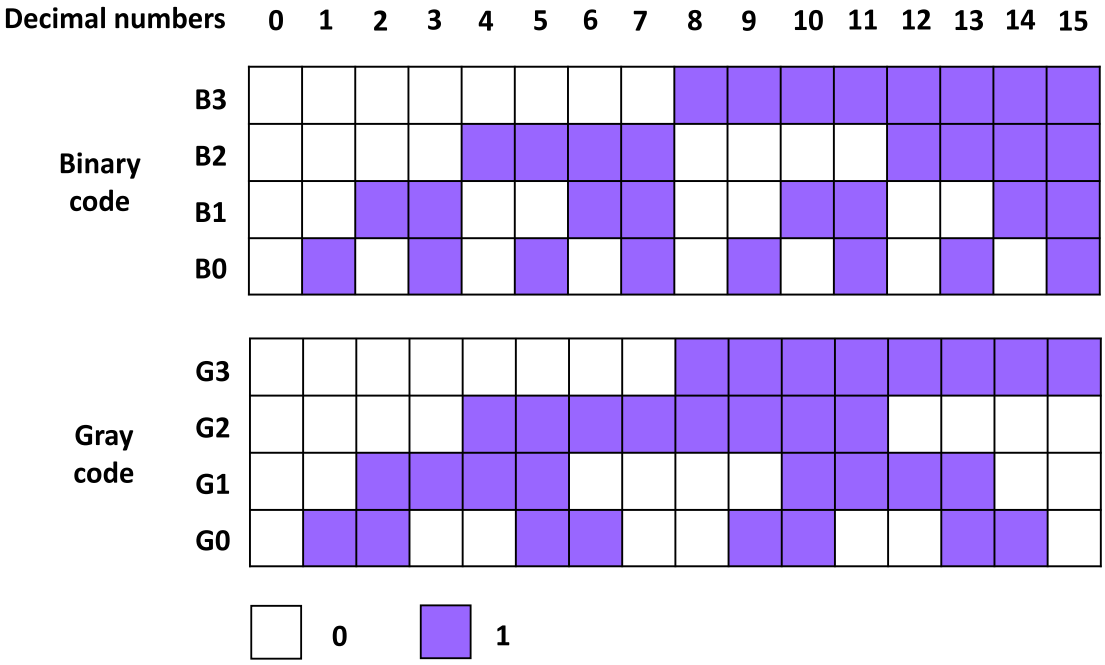
3.2 Gray Code TDC

3.2.1 Gray code

A Gray code, also known as the reflected binary code, is a binary numeral system where two successive values differ in only one bit (binary digit). Gray code was originally designed to prevent spurious output from electromechanical switches. Table 3.3 illustrates the difference between Natural Binary and Gray codes. Fig.3.10 shows the binary and Gray code count sequences, and Fig.3.11 illustrates the binary and Gray code waveforms.

Table 3.3 4-bit Gray code vs. 4-bit Natural binary code

Decimal numbers	Natural binary code	4-bit Gray code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000



Decimal numbers		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Binary code	B3	□	□	□	□	□	□	□	□	■	■	■	■	■	■	■	■
	B2	□	□	□	□	■	■	■	■	□	□	□	□	■	■	■	■
	B1	□	□	■	■	□	□	■	■	□	□	■	■	□	□	■	■
	B0	□	■	□	■	□	■	□	■	□	■	□	■	□	■	□	■
Gray code	G3	□	□	□	□	□	□	□	□	■	■	■	■	■	■	■	■
	G2	□	□	□	□	■	■	■	■	■	■	■	■	□	□	□	□
	G1	□	□	■	■	■	■	□	□	□	□	■	■	■	■	□	□
	G0	□	■	■	□	□	■	■	□	□	■	■	□	□	■	■	□

□ 0 ■ 1

Fig. 3.10 Binary vs. Gray code count sequences

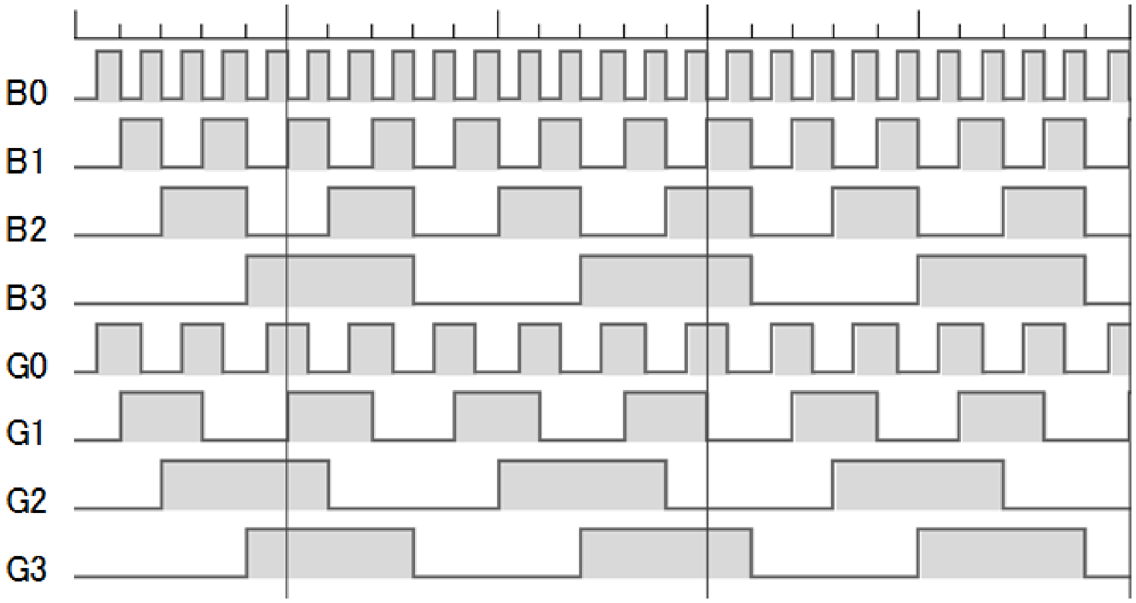


Fig. 3.11 Binary vs. Gray code waveforms

With Gray code, only one of G3, G2, G1, G0 in 4-bit case changes state from one position (decimal k) to another (decimal $k+1$). This effect can be seen clearly in Table 3.3 and Fig.3.10.

For example, look at the natural binary code sequence in Table 3.3, the number 7 represented as 0111. As it changes to 8, every bit changes state, to 1000. Every value changes from either 1 to 0, or 0 to 1. If all of these bits changed instantaneously, and synchronously, from one state to the other, there would never be any problem. However, in a highly capacitive circuit (or sluggish system response), some bits would flip before others. Since every bit is changing in this example, depending on the order they change, it might produce an output of any value from this collection of bits. In other words, if for the bits that should be changed for a small amount of the TDC input change, some are changed and the others are not, the TDC output error can be very big in natural binary code.

In the 4-bit Gray code sequence in Table 3.3, between any two adjacent numbers, only one bit changes at a time. Even from position 7 to 8, Gray code only changes one bit state. The sort of error mentioned above is not possible with Gray code, so the data is more reliable. In other words, even if the bit that should be changed does not change for a small amount of the TDC input change, the error is only 1 LSB in Gray code.

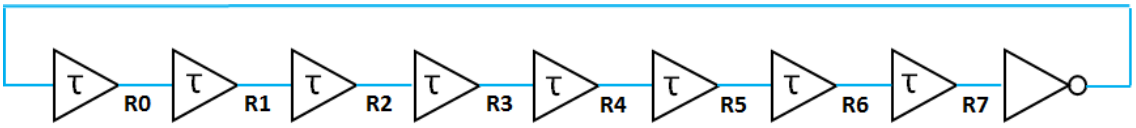
The Gray code is designed so that only one bit will change state for each state for each count transition, unlike the binary code where multiple bits change at certain count transitions. For the Gray code, the uncertainty during a transition is only one count, unlike with the binary code, where the uncertainty could be multiple counts [10]. So the Gray code provides data with the least uncertainty. This feature prevents certain data errors which can occur with natural binary code during state changes. Moreover, this characteristic allows a circuit to perform some error checking, i.e. if more than one bit changes for a small amount of the input change, the data must be incorrect.

3.2.2 Natural Time-Domain Gray Code Bit Generator

In a ring oscillator, between any two adjacent states, only one output changes at a time. This characteristic is very similar to Gray code.

For example, as illustrated in Fig.3.12, two successive states of the 8-stage ring oscillator differ in only one output. The output R3 is the same as the Gray code bit G2, and R7 is the same as G3.

Ring oscillator is a natural time-domain Gray code bit generator. For any given Gray code, its each bit can be generated by a certain ring oscillator.



State	8-stage ring oscillator output								4-bit Gray code			
	R0	R1	R2	R3	R4	R5	R6	R7	G3	G2	G1	G0
0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	1
2	1	1	0	0	0	0	0	0	0	0	1	1
3	1	1	1	0	0	0	0	0	0	0	1	0
4	1	1	1	1	0	0	0	0	0	1	1	0
5	1	1	1	1	1	0	0	0	0	1	1	1
6	1	1	1	1	1	1	0	0	0	1	0	1
7	1	1	1	1	1	1	1	0	0	1	0	0
8	1	1	1	1	1	1	1	1	1	1	0	0
9	0	1	1	1	1	1	1	1	1	1	0	1
10	0	0	1	1	1	1	1	1	1	1	1	1
11	0	0	0	1	1	1	1	1	1	1	1	0
12	0	0	0	0	1	1	1	1	1	0	1	0
13	0	0	0	0	0	1	1	1	1	0	1	1
14	0	0	0	0	0	0	1	1	1	0	0	1
15	0	0	0	0	0	0	0	1	1	0	0	0

Fig. 3.12 8-stage ring oscillator and 4-bit Gray code

3.2.3 Gray Code Based TDC Architecture

As a ring oscillator is a natural time-domain Gray bit code generator [2], a Gray code TDC architecture can be conceived by grouping a few ring oscillators to operate on the same input.

Fig.3.13 shows the proposed Gray code TDC in 4-bit case. The Gray code bit G0 is generated by a 2-stage ring oscillator, and G1 is generated by a 4-stage ring oscillator, and G2, G3 are generated by 8-stage ring oscillators.

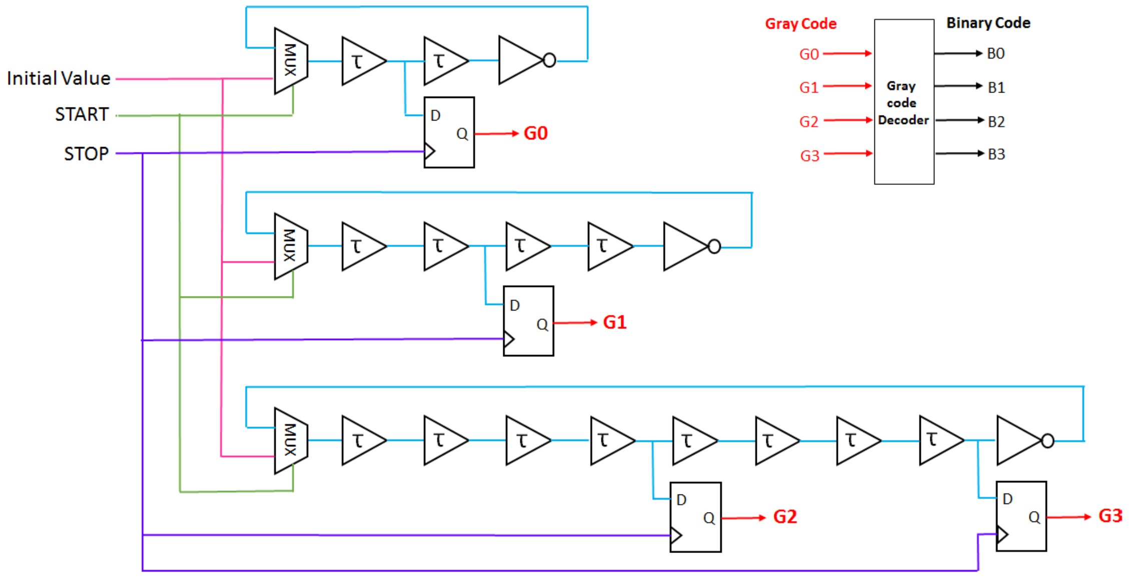


Fig. 3.13 Proposed Gray code TDC architecture in 4-bit case

This TDC can measure the time elapsed between START and STOP incoming pulses and output the binary digital representation of the time interval as follows:

(1) When START signal is in LOW state, three ring oscillators are initialized by Initial Value.

(2) When START signal goes from LOW to HIGH, three ring oscillators begin to oscillate.

(3) When STOP signal is on the rising edge, i.e. LOW-to-HIGH transition, the DFFs are triggered and the value of D is transferred to the output Q. Each Q stands for a certain Gray code bit.

(4) All the generated Gray code bits are delivered to Gray code decoder, as illustrated in Fig.3.14, and transferred into binary code, which represents the time interval between START and STOP.

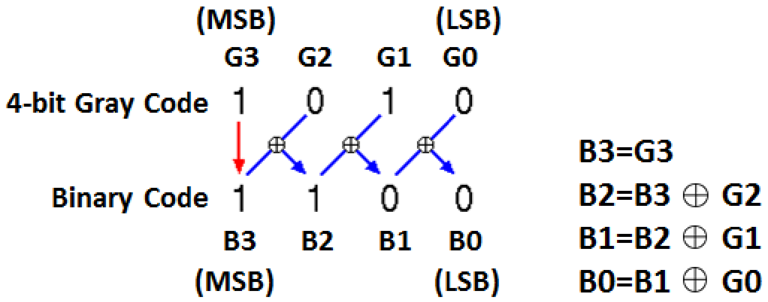


Fig. 3.14 4-bit Gray code decoder

RTL simulation was conducted to verify the characteristics of Gray code TDC in Fig.3.13. The delay of each buffer is equal to 10ns, and START signal goes from LOW to HIGH at 100ns. RTL simulation waveforms are illustrated in Fig.3.15. We can see that the proposed 4-bit Gray code TDC works as expected in the time domain.

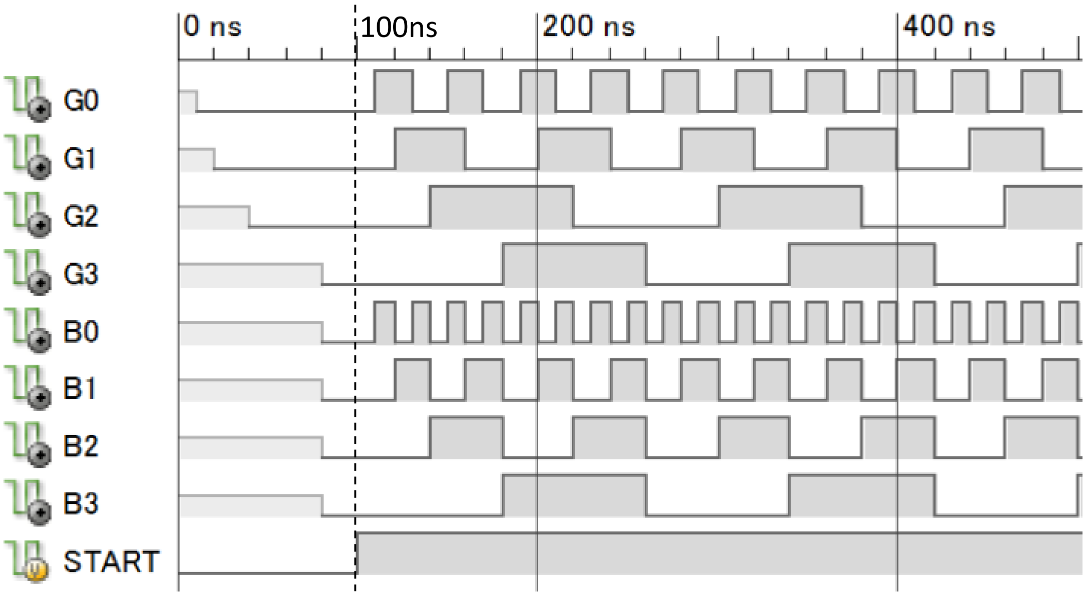


Fig. 3.15 RTL simulation waveforms of 4-bit Gray code TDC

Note that the proposed Gray code TDC uses only 14 delay cells and 4 flip-flops, and the maximum stage of the ring oscillator is 8, while the corresponding flash TDC requires 16 delay cells and 16 flip-flops, and the maximum stage of the ring oscillator is 16.

In general, for a measurement range of 2^n, the proposed Gray code TDC uses $2^n - 2$

delay cells and n flip-flops (linear growth), and the maximum stage of the ring oscillator is 2^{n-1}, while the corresponding flash-type TDC uses 2^n delay cells and 2^n flip-flops (exponential growth), and the maximum stage of the ring oscillator is 2^n.

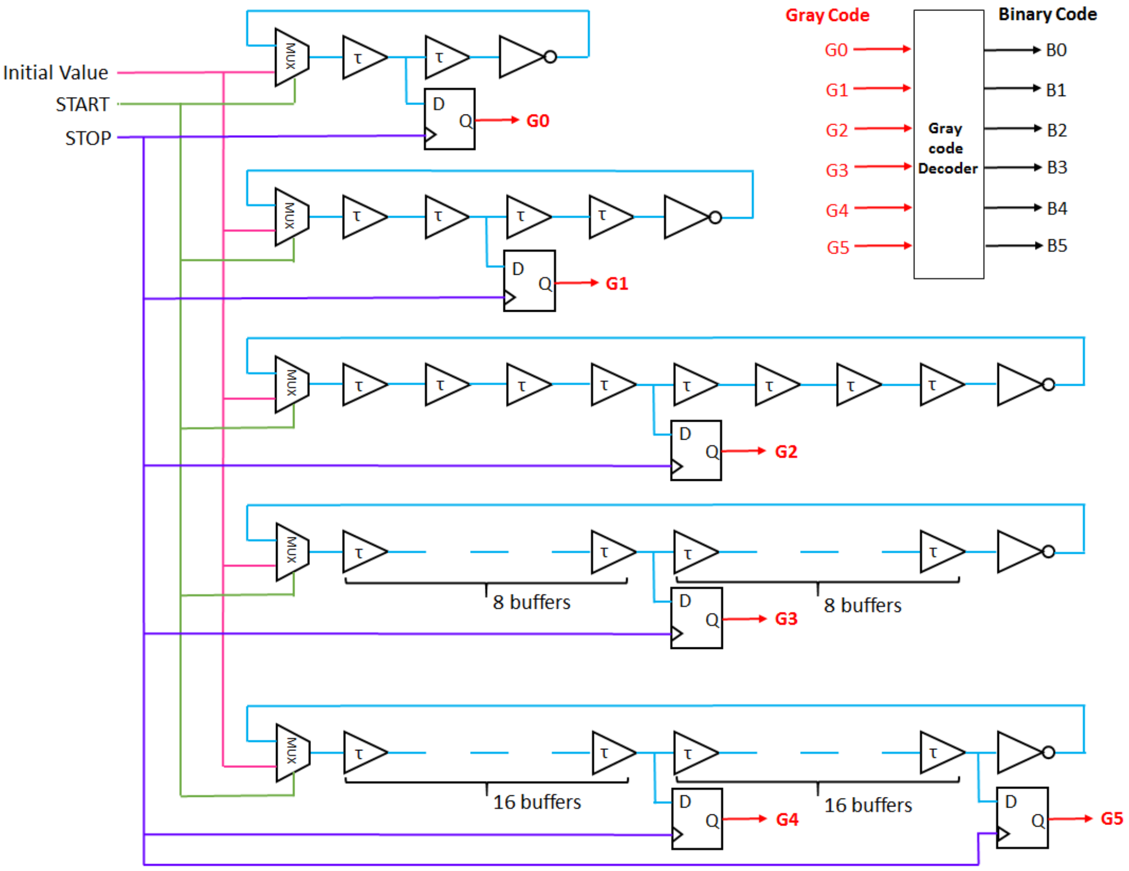
For large measurement range, the number of flip-flops in the proposed TDC decreases rapidly ($n << 2^n$) compared with flash-type TDC, which reduces the hardware and chip area consumption significantly.

Furthermore, the maximum stage of the ring oscillator is also reduced by half in the proposed TDC. The use of shorter delay lines reduces also the integral non-linearity caused by mismatches between the delay stages.

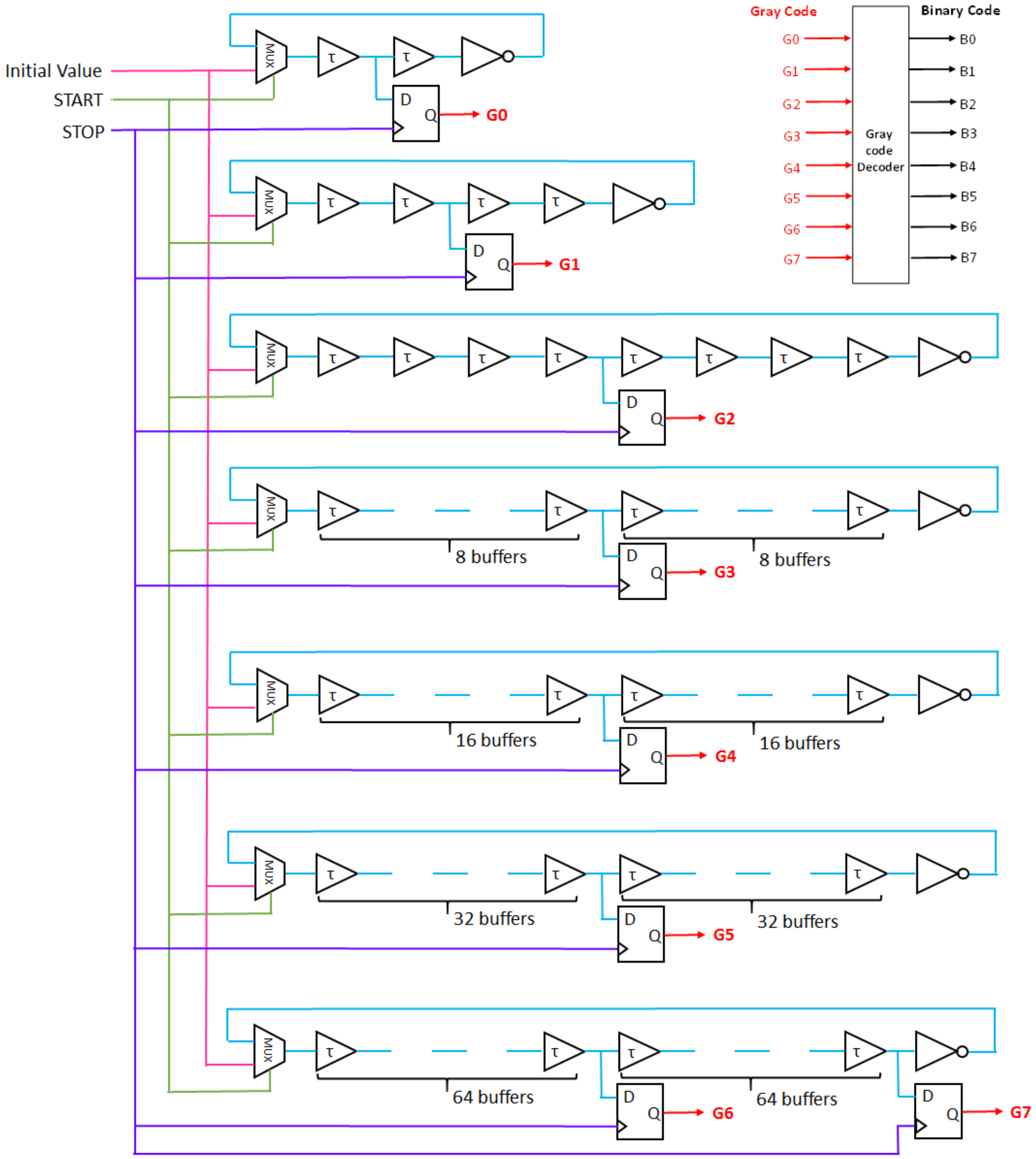
Table 3.4 Gray code TDC vs. Flash-type TDC

	Number of delay cells	Number of DFFs	Maximum stage of RO
Gray code TDC	$2^n - 2$	n	2^{n-1}
Flash-type TDC	2^n	2^n	2^n

Similarly, 6-bit and 8-bit Gray code TDC architectures can be conceived by a group of ring oscillators (Fig.3.16).



(a) Proposed 6-bit Gray code TDC architecture



(b) Proposed 8-bit Gray code TDC architecture

Fig. 3.16 Proposed 6-bit and 8-bit Gray code TDC architectures

3.2.4 FPGA Implementation

A proof-of-concept 4-bit Gray code TDC in Fig.3.13 is implemented on FPGA (Fig.3.17). For the 4-bit Gray code TDC, inputs “START” and “Initial Value” are connected to user push buttons, and “STOP” is connected to 200MHz FPGA clock.

Outputs “B3 B2 B1 B0” are delivered to user LED. Each buffer is realized by a delay flip-flop with 100MHz clock frequency, i.e. the delay of each buffer is equal to 10ns.

The measurement starts by pushing the START push button, which produces a rising edge “START” signal. ChipScope is applied to probe the internal signal of FPGA [12]. We can see from Fig.3.18 (a) that the proposed 4-bit Gray code TDC works with good linearity as expected. Similarly, proof-of-concept 6-bit and 8-bit Gray code TDC architectures were implemented on FPGA and measured (Fig.3.18 (b), (c)).

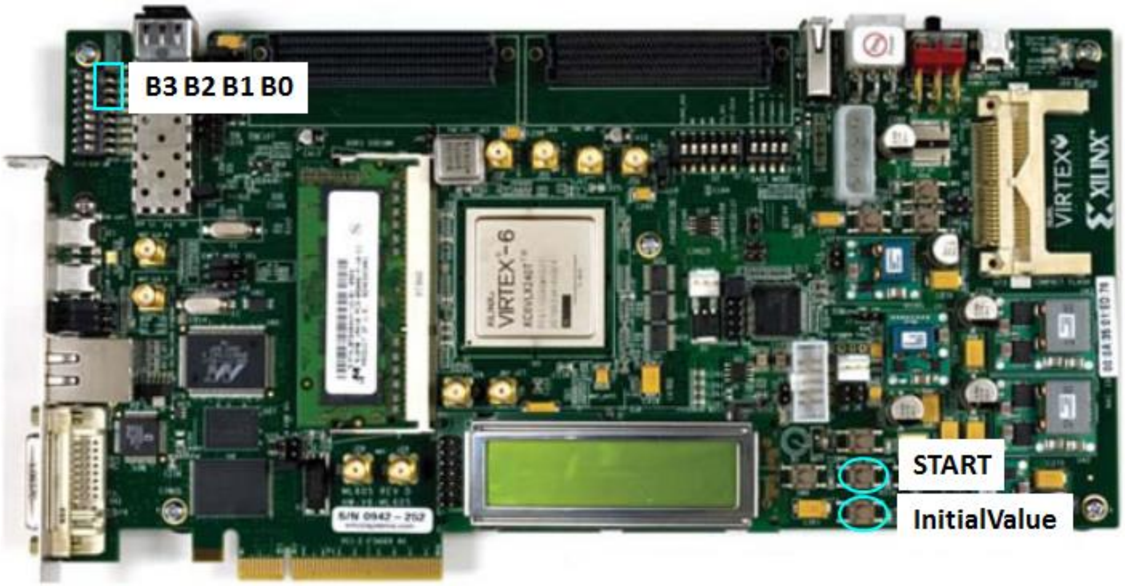
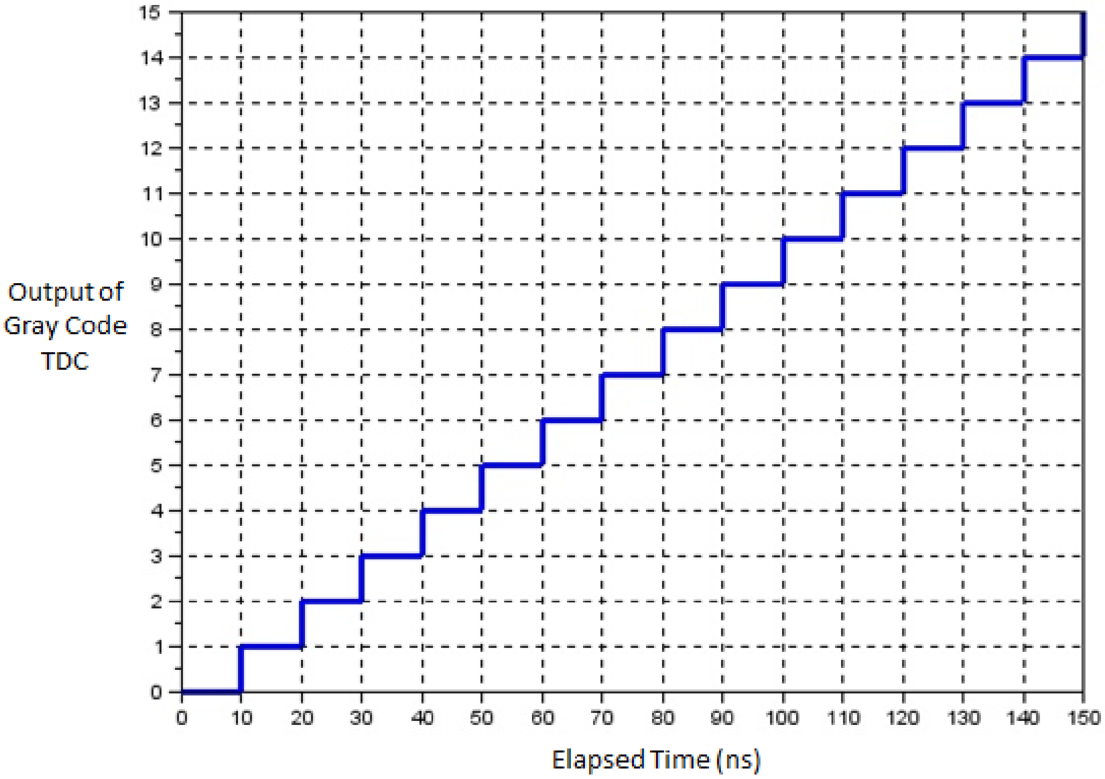
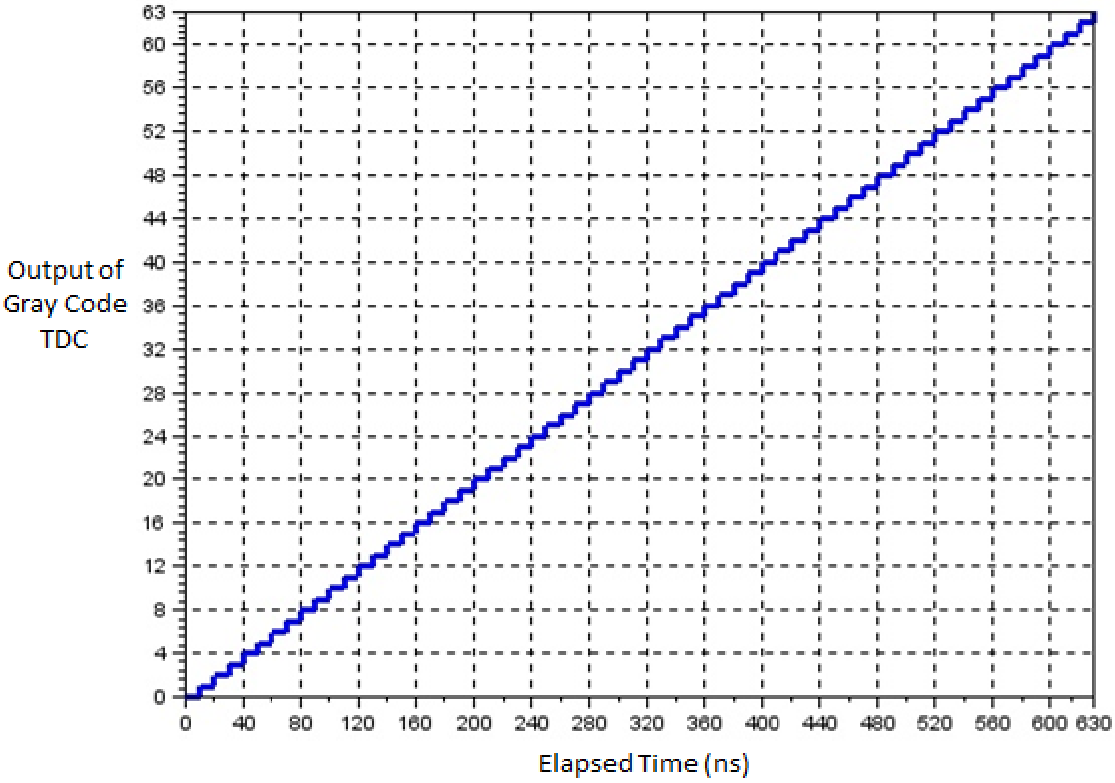


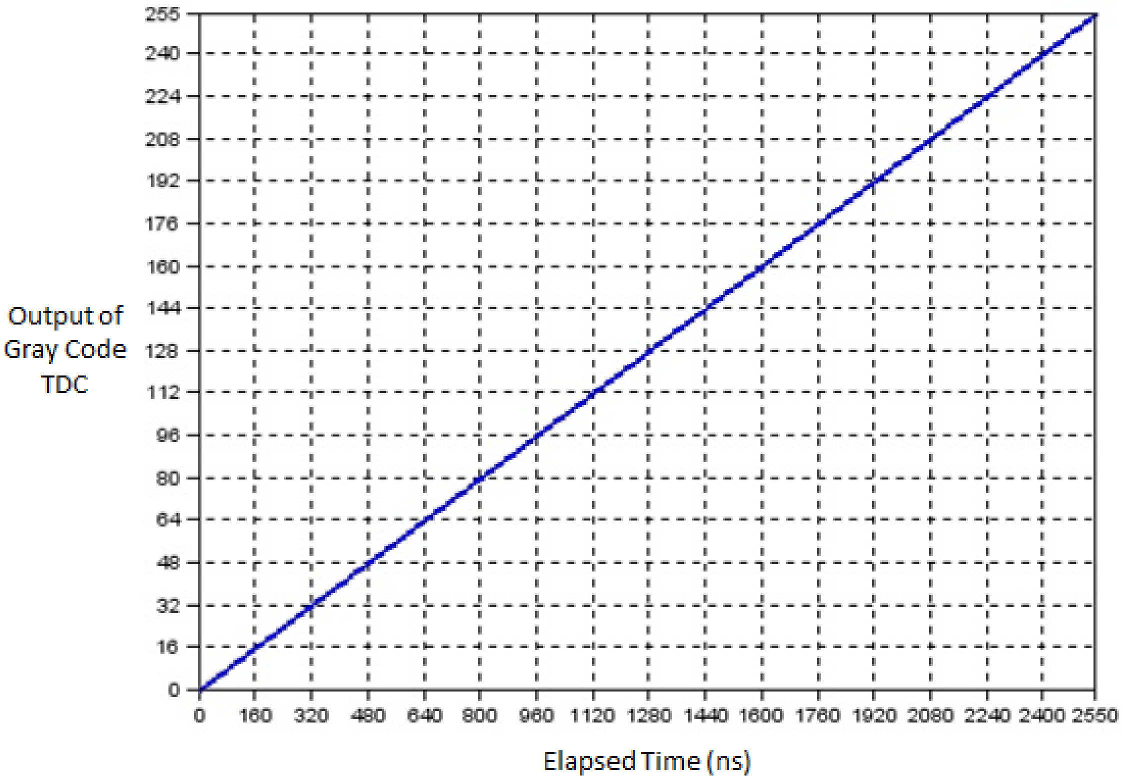
Fig. 3.17 FPGA implementation of 4-bit Gray code TDC



(a) 4-bit TDC case



(b) 6-bit TDC case



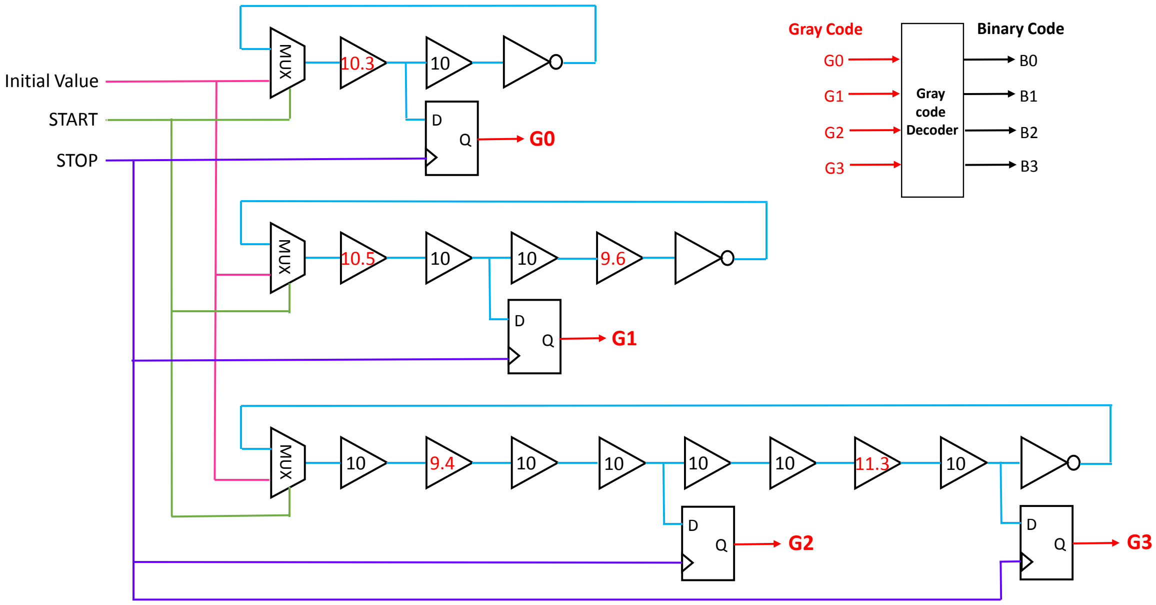
(c) 8-bit TDC case

Fig. 3.18 FPGA measurement results of 4-bit, 6-bit and 8-bit Gray code TDCs

3.2.5 Glitch-Free Characteristics

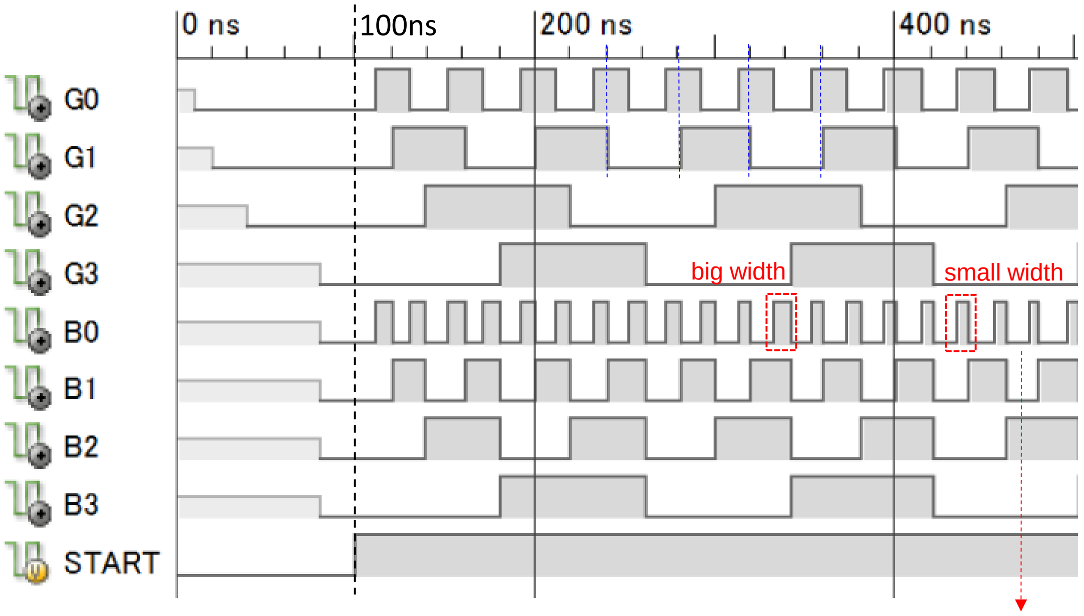
The proposed Gray code TDC has a unique characteristic where only one output of the DFFs changes state with each clock pulse, so it can provide a glitch-free binary code sequence, i.e. no out-of-sequence code, even there are some amounts of mismatches among the delay stages.

RTL simulation was conducted to verify this characteristic. In Fig.3.19 (a), mismatches happen among the delay stages in 4-bit Gray code TDC. Fig.3.19 (b) shows the RTL simulation waveforms. From Fig.3.15, we can see that in no mismatch condition, for Gray code, the waveform G_{n+1} always has signal edges at the center of the rectangular wave of G_n, and for binary code, the widths of B_n rectangular waves are always the same. Comparatively, from Fig.3.19 (b), we can see that under mismatch condition, for Gray code, the signal edges of G_{n+1} are not at the center of the rectangular wave of G_n, and for binary code waveforms, the widths of rectangular waves varies due to delay mismatches.



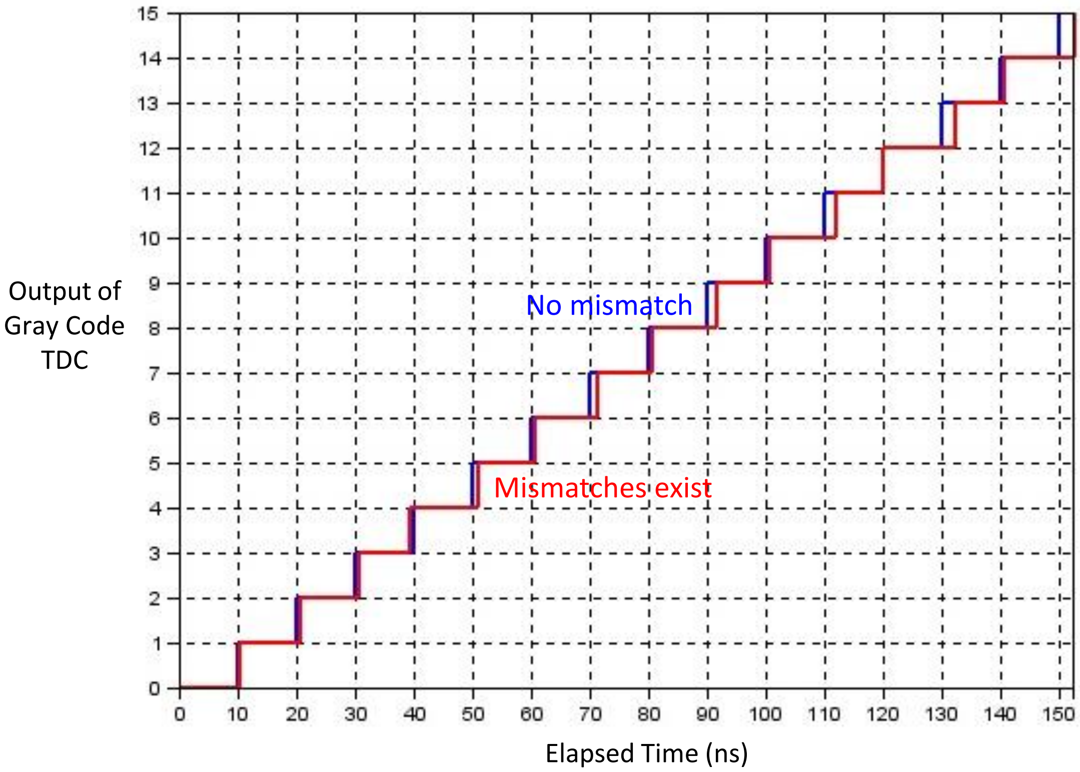
(a) 4-bit Gray code TDC with delay mismatches

For gray code waveforms, the signal edges of G_{n+1} are not at the center of the rectangular wave of G_n due to delay mismatches



For binary code waveforms, B_n rectangular waves have varying widths due to delay mismatches

(b) RTL simulation waveforms



(c) FPGA measurement results

Fig. 3.19 4-bit Gray code TDC with delay mismatches

Similarly, a proof-of-concept 4-bit Gray code TDC in Fig.3.19 (a) is implemented on FPGA. Buffers are realized by combinations of delay flip-flops with 100MHz/200MHz clock frequency and IODELAY blocks with 200MHz REFCLK. We can see from Fig.3.19 (c) that even though the code width varies due to the mismatches, the proposed Gray code TDC can still output a glitch-free binary code sequence.

As this glitch-free advantage is achieved from the unique characteristics of Gray code, where only one output changes at a time between any two adjacent states, for Gray code waveforms, the signal edges of $G_n(n>0)$ should be within the range of the corresponding rectangular wave of G_0, as illustrated in Fig.3.20. Otherwise, the unique characteristics of Gray code will no longer exist, i.e. glitches will appear.

The allowable maximum delay mismatch can be calculated based on above analysis. For example, for 4-bit Gray code TDC in Fig.3.19 (a), the allowable maximum delay mismatch between G_0 ring oscillator stage and G_1 ring oscillator stage is equal to

10ns.

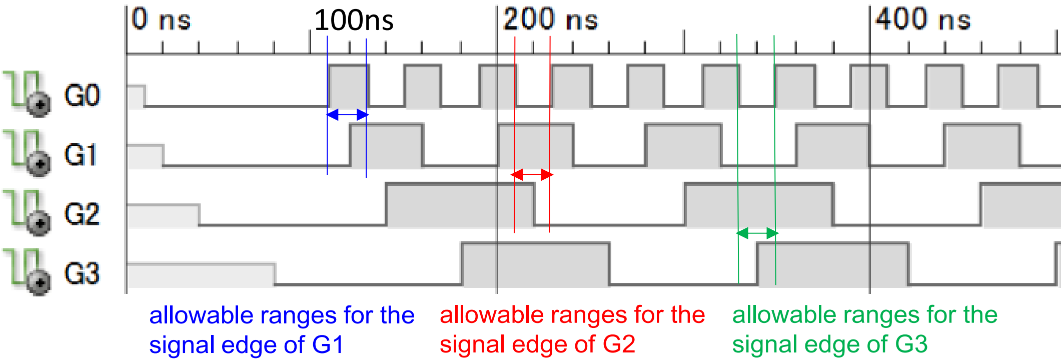


Fig. 3.20 Allowable ranges for the signal edges of G1/G2/G3 in 4-bit Gray code TDC

It is worth to mention that though the multiplexer delay and the inverter delay are not considered in RTL simulation and FPGA verification, these delays can be equivalently convert into certain mismatches among the delay stages. That is to say, the multiplexer delay and the inverter delay do not influence Gray TDC providing a glitch-free output sequence.

3.3 Gray Code Based TDC with Cyclic code

3.3.1 Cyclic code

In coding theory, a cyclic code is a block code, where the circular shifts of each code-word gives another word that belongs to the code. They are error-correcting codes that have algebraic properties that are convenient for efficient error detection and correction.

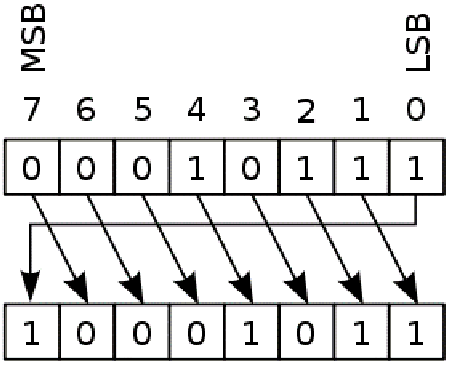


Fig. 3.21 Cyclic code [17]

As illustrated in Fig.3.21, if 00010111 is a valid code-word, applying a right circular shift gives the string 10001011. If the code is cyclic, then 10001011 is again a valid code-word. In general, applying a right circular shift moves the least significant bit (LSB) to the leftmost position, so that it becomes the most significant bit (MSB); the other positions are shifted by 1 to the right.

Cyclic code bits are very similar to Gray code bits. For example, cyclic code with code-word 00001111 is illustrated in Table 3.5 (b). The output C1 is the same as the Gray code bit G1 in Table 3.5 (a), and $C0 \oplus C2$ is equal to G0. If the frequency of G1, C0, C1, and C2 are f_1, then the frequency of G0 is $2f_1$. The value of G0 changes very quickly, which leads to a high frequency output in circuit design, while the value of C0 and C2 change comparably slowly, which produce a comparable low frequency output in circuit design. So cyclic code can be applied to generate the lower bits of Gray code, which can reduce the frequency of the outputs.

Table 3.5 Gray code & Cyclic code

(a) 4-bit Gray code

State	4-bit Gray code			
	G3	G2	G1	G0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	1

3	0	0	1	0
4	0	1	1	0
5	0	1	1	1
6	0	1	0	1
7	0	1	0	0
8	1	1	0	0
9	1	1	0	1
10	1	1	1	1
11	1	1	1	0
12	1	0	1	0
13	1	0	1	1
14	1	0	0	1
15	1	0	0	0

(b) Cyclic code with code-word 00001111

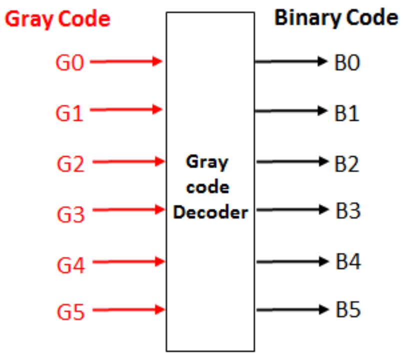
State	Cyclic code with code-word 00001111							
	C0	C1	C2	C3	C4	C5	C6	C7
0	0	0	0	0	1	1	1	1
1	1	0	0	0	0	1	1	1
2	1	1	0	0	0	0	1	1
3	1	1	1	0	0	0	0	1
4	1	1	1	1	0	0	0	0
5	0	1	1	1	1	0	0	0
6	0	0	1	1	1	1	0	0
7	0	0	0	1	1	1	1	0

In electrical circuits, parasitic capacitance, or stray capacitance is an unavoidable and usually unwanted capacitance that exists between the parts of an electronic component or circuit simply because of their proximity to each other. All actual circuit elements have internal capacitance, which can cause their behavior to depart from that of 'ideal' circuit elements. At low frequencies parasitic capacitance can usually be ignored, but in high frequency circuits it can be a major problem. Applying the cyclic code generator to achieve the lower Gray code bits can reduce the frequency of the output, i.e. to reduce the parasitic capacitance among circuit elements.

3.3.2 Gray Code Based TDC with Cyclic code

A novel TDC architecture can be conceived based on Gray code and cyclic code: the upper Gray code bits are generated by grouping a few ring oscillators, and the lower Gray code bits are conceived by cyclic code generators, in order to reduce the frequency of the output.

Fig.3.22 shows the proposed coding theory based TDC. The Gray code bit G0, G1 are generated by an 8-bit cyclic code generator, and G2, G3 are generated by 8-stage, 16-stage ring oscillator with buffer delay τ separately, and G4, G5 are generated by 32-stage ring oscillator with buffer delay τ.



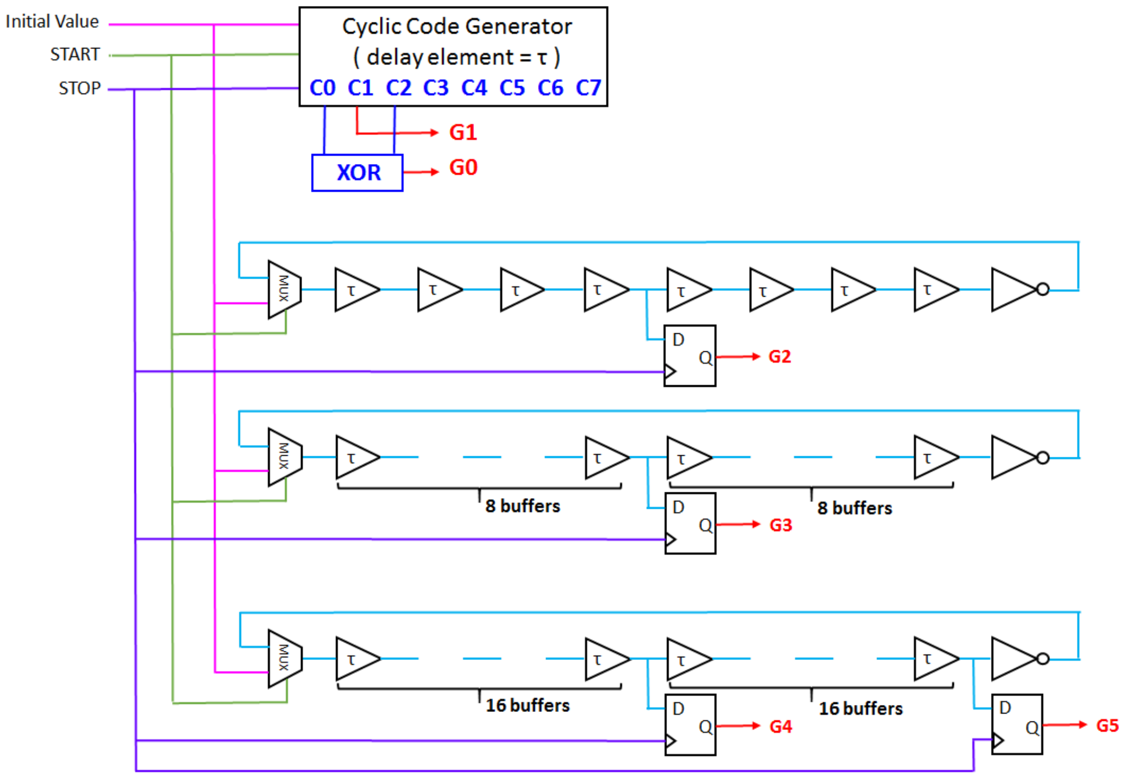
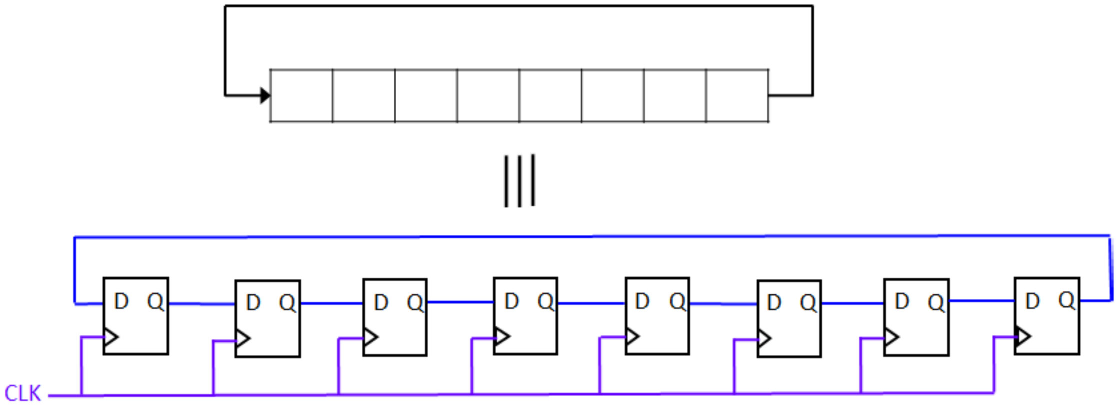
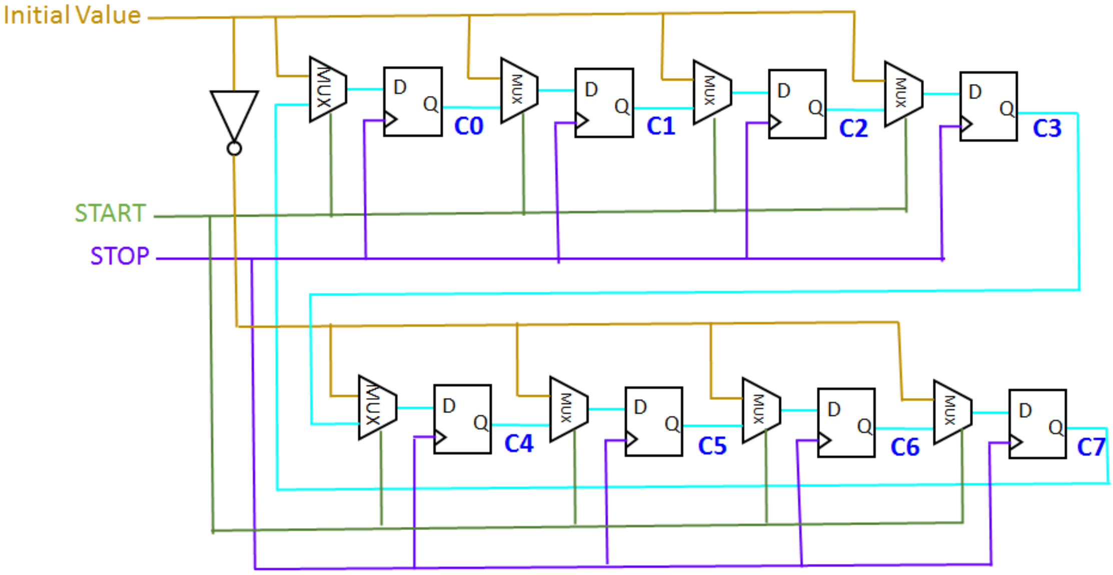


Fig. 3.22 Proposed 6-bit coding theory based TDC



(a) Generator structure for 8-bit cyclic code



(b) Circuit diagram of 8-bit cyclic code generator

Fig. 3.23 Cyclic code generator

As illustrated in Fig.3.23 (a), 8-bit cyclic code generator can be implemented by 8 flip-flops. Fig.3.23 (b) shows the circuit diagram applied in proposed 6-bit coding theory based TDC. When START signal is in LOW state, C0 C1 C2 C3 flip-flops are initialized by (*Initial Value*), and C4 C5 C6 C7 flip-flops are initialized by ($\overline{Initial\ Value}$). If *Initial Value* = 0, then the initialization state, i.e. code-word of the cyclic generator is 00001111. When START signal goes from LOW to HIGH, the output of C7 flip-flop is connected to the input of C0 flip-flop, resulting in a closed loop. When STOP signal is on the rising edge, the cyclic generator starts to work and produces one shift.

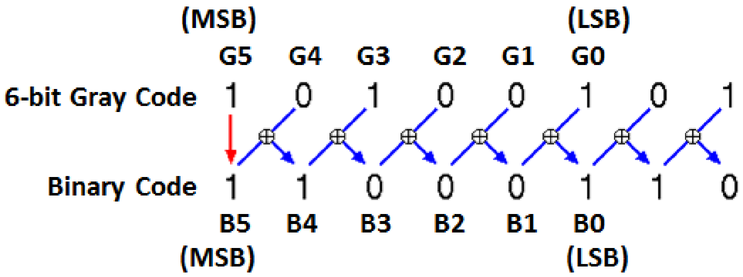
This TDC can measure the time elapsed between START and STOP incoming pulses and output the binary digital representation of the time interval as follows:

(1) When START signal is in LOW state, the cyclic generator and the ring oscillators are initialized by Initial Value.

(2) When START signal goes from LOW to HIGH, the cyclic generator starts to work, and the ring oscillators begin to oscillate.

(3) When STOP signal is on the rising edge, the cyclic generator produces one shift, and $C0 \oplus C2$, C1 are transferred to G0, G1 simultaneously. Meanwhile, the DFFs of the ring oscillators are triggered and the value of D is transferred to the output G2, G3, G4, and G5.

(4) All the generated Gray code bits are delivered to Gray code decoder, as illustrated in Fig.3.24, and transferred into binary code, which represents the time interval between START and STOP.



B5=G5

B4=B5 ⊕ G4

B3=B4 ⊕ G3

B2=B3 ⊕ G2

B1=B2 ⊕ G1

B0=B1 ⊕ G0

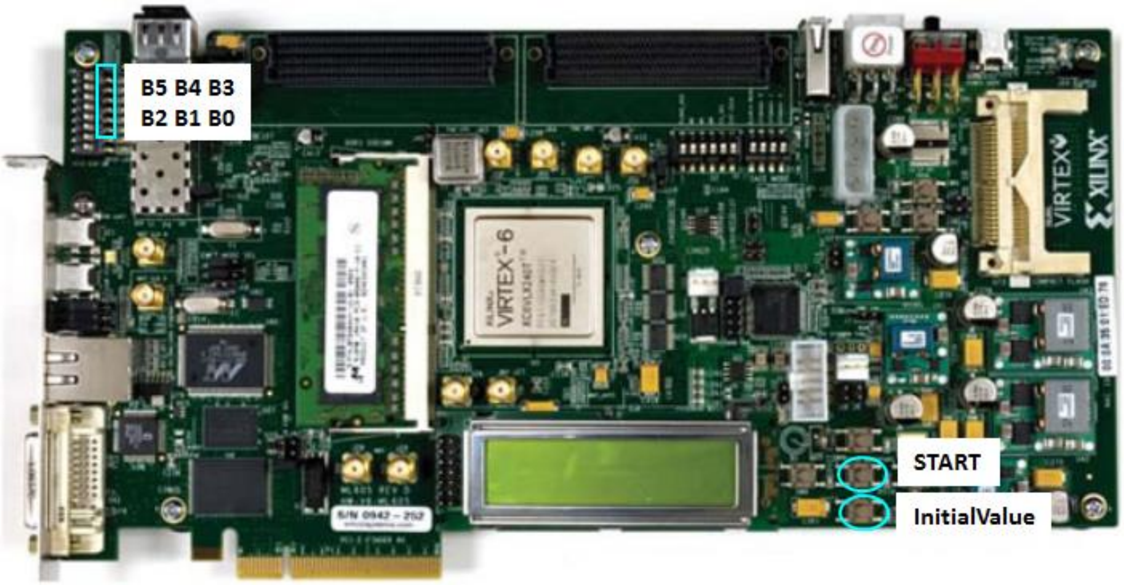
Fig. 3.24 6-bit Gray code decoder

Note that the proposed Gray code based TDC with cyclic code uses only 12 flip-flops, and the maximum stage of the ring oscillator is 32, while the corresponding flash TDC requires 64 flip-flops, and the maximum stage of the ring oscillator is 64.

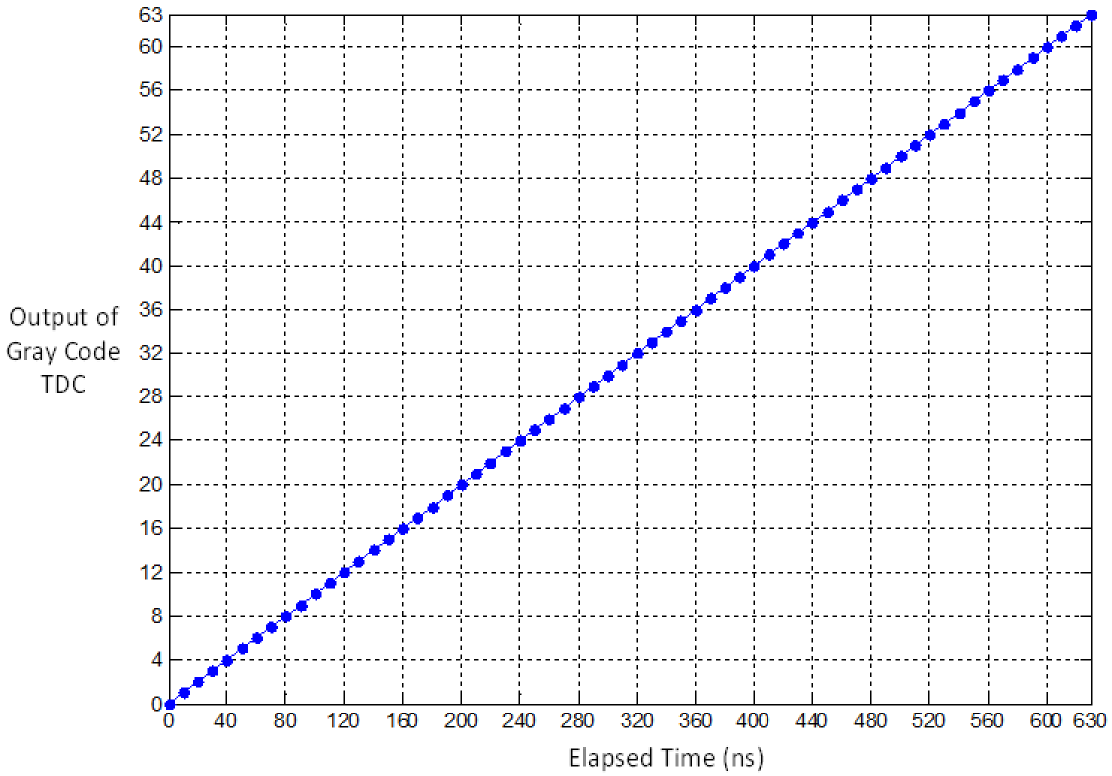
3.3.3 FPGA Implementation

Proof-of-concept 6-bit Gray code based TDC with cyclic code is implemented on FPGA. Inputs “START” and “Initial Value” and connected to user push buttons, and “STOP” is connected to 100MHz FPGA clock. Outputs “B5 B4 B3 B2 B1 B0” are delivered to user LED. The delay element of cyclic code generator is 10ns. Each buffer is realized by a delay flip-flop with 50MHz clock frequency, i.e. the delay of each buffer is equal to 20ns.

The measurement starts by pushing the START push button, which produces a rising edge “START” signal. Chipscope is applied to probe the internal signal of FPGA. Fig.3.25 shows that the proposed 6-bit Gray code based TDC with cyclic code works with good linearity as expected.



(a) FPGA implementation



(b) FPGA measurement result (6-bit case)

Fig. 3.25 FPGA implementation and measurement result of a 6-bit Gray code based TDC with cyclic code

3.4 Summary

In this chapter, residue number system, Gray code, and cyclic code are applied in parallel ring oscillator TDC architectures.

Residue number system represents a large integer using a set of smaller integers, so that a large calculation can be performed as a series of smaller calculations that can be performed independently and in parallel. In TDC, the signal is treated as “time” instead of “voltage”. Ring oscillators can be applied to obtain the residue easily. Applying residue number system concept at circuit design, we can represent a TDC circuit with large measurement range using a set of smaller ring oscillator TDCs. This architecture reduces drastically the number of delay cells and flip-flops while keeping a comparable performance. FPGA experimentation and simulation results validated the effectiveness.

However, for residue number system based TDC, glitches (i.e. out-of-sequence codes) may occur when there are mismatches between the delay stages, which triggers an instability in the output digital codes. Due to this, Gray code based TDC architecture is proposed to remove the glitches effectively.

The Gray code is designed so that only one bit will change state for each state for each count transition. As the uncertainty during a transition is only one count, Gray code can provide data with the least uncertainty. This feature prevents certain data errors which can occur with natural binary code during state changes. Ring oscillator is a natural time-domain Gray code bit generator. For any given Gray code, its each bit can be generated by a certain ring oscillator. A Gray code TDC architecture can be conceived by grouping a few ring oscillators to operate on the same input. Corresponding RTL simulation and FPGA implementation are conducted, and the operation principle is verified by the measurement results.

Furthermore, the proposed Gray code TDC can provide a glitch-free binary code sequence, i.e. no out-of-sequence code, even there are some amounts of mismatches among the delay stages. RTL simulation is conducted to verify this characteristic and the allowable maximum delay mismatch is analyzed. A proof-of-concept prototype is implemented on FPGA. The measurement results verify that even though there are mismatches among the delay stages, the proposed Gray code TDC can still output a glitch-free binary code sequence.

Parasitic capacitance, in electrical circuits, is the extra effect of conductors that serve as plates between a dielectric, which is usually air. It becomes a problem with higher frequencies because the very small distributed capacitances that exist will have lower impedances at these frequencies. Taking this issue into consideration, Gray code based TDC with cyclic code is presented. Cyclic code can be applied to generate the lower bits of Gray code, which can reduce the frequency of the outputs, i.e. to reduce the parasitic capacitance among circuit elements. FPGA implementation and measurement results verify the operation principle.

The required number of delay cells and D flip-flops in the proposed parallel ring oscillator TDCs and flash-type TDC is compared. For example, for a TDC with 2^{13} quantization levels, the number of delay cells and Flip-Flops applied in the proposed parallel ring oscillator TDCs and corresponding flash-type TDC are:

Table 3. 6 Proposed parallel ring oscillator TDCs vs. Flash-type TDC

	Number of delay cells	Number of DFFs	Maximum stage
Residue number system based TDC	40	40	13
Gray code TDC	8190	13	4096
Gray code TDC with cyclic code	8184	19	4096
Flash-type TDC	8192	8192	8192

Compared with flash-type TDC, the number of delay cells and D flip-flops declines significantly in residue number system based TDC, and the number of D flip-flops decreases rapidly in Gray code TDC and Gray code TDC with cyclic code. The proposed parallel ring oscillator TDCs can reduce hardware, power consumption, as well as chip area significantly.

Furthermore, the maximum stage is also reduced rapidly in the proposed parallel ring oscillator TDCs. The use of shorter delay lines reduces the integral non-linearity caused by mismatches between the delay stages.

That is to say, Gray code TDC with cyclic code architecture can be applied to measurement applications where large detectable range is required, e.g. digital frequency synthesizers used in wireless applications, divider-assisted digital phase locked loop, etc.

References

[1] Eberhard Knobloch, Hikosaburo Komatsu, Dun Liu, “Seki, Founder of Modern Mathematics in Japan”, Springer Proceedings in Mathematics & Statistics.

[2] William A. Chren Jr., “Low-Area Edge Sampler Using Chinese Remainder Theorem”, IEEE Transactions on Instrumentation and Measurement, vol.48, no.4, pp.793-797 (1999).

[3] Y. Arai, T. Baba, “A CMOS Time to Digital Converter VLSI for High-Energy Physics,” IEEE Symposium on VLSI Circuits, pp.121-122, Tokyo (1988).

[4] Robert Bogdan Staszewski, Poras T. Balsara, "All-Digital Frequency Synthesizer in Deep-Submicron CMOS", Wiley-Interscience (2006).

[5] Takeshi Chujo, Daiki Hirabayashi, Kentaroh Kentaroh, Congbing Li, Yutaro Kobayashi, Junshan Wang, Koshi Sato, Haruo Kobayashi, "Experimental Verification of Timing Measurement Circuit With Self-Calibration", IEEE International Mixed-Signals, Sensors and Systems Test Workshop (IMS3TW'14), Porto Alegre, Brazil (Sept.17-19, 2014).

[6] Kentaroh Katoh, Yutaro Kobayashi , Takeshi Chujyo, Junshan Wang, Ensi Li, Congbing Li, Haruo Kobayashi, "A Small Chip Area Stochastic Calibration for TDC Using Ring Oscillator", Journal of Electronic Testing: Theory and Applications, vol.30, issue 6, pp.653-663, Springer (Dec. 2014).

[7] T. Komuro, J. Rivoir, K. Shimizu, M. Kono, H. Kobayashi, “ADC Architecture Using Time-to-Digital Converter”, IEICE Trans. vol.J90-C, pp.126-133 (April 2007).

[8] Congbing Li, Kentaroh Katoh, Haruo Kobayashi, Junshan Wang, Shu Wu, Shaiful Nizam Mohyar, “Time-to-Digital Converter Architecture with Residue Arithmetic and its FPGA Implementation”, 11th International SoC Design Conference, Paper A2-5, Jeju, Korea (Nov. 2-6, 2014).

[9] Divya Aggarwal, “Alignment of Single Mirror in Closed Loop”, Amity University.

[10] Y. Arai, T. Baba, “A CMOS Time to Digital Converter VLSI for High-Energy Physics”, IEEE Symposium on VLSI Circuits (1988).

[11] Y. Osawa, D. Hirabayashi, N. Harigai, H. Kobayashi, K. Niitsu, O. Kobayashi,

"Phase Noise Measurement Techniques Using Delta-Sigma TDC", IEEE International Mixed-Signals, Sensors and Systems Test Workshop, Porto Alegre, Brazil (Sept. 2014).

[12] Xilinx, "Using Xilinx ChipScope Pro ILA Core with Project Navigator to Debug FPGA Applications". [Online]. Available: www.xilinx.com.

[13] S. Uemori, M. Ishii, H. Kobayashi, D. Hirabayashi, Y. Arakawa, Y. Doi, O. Kobayashi, T. Matsuura, K. Niitsu, Y. Yano, T. Gake, T. Yamaguchi, N. Takai, "Multi-bit Sigma-Delta TDC Architecture with Improved Linearity," Journal of Electronic Testing : Theory and Applications, Springer, vol.29, no.6, pp.879-892, December 2013.

[14] B. Wu, S. Zhu, Y. Zhou, and Y. Chiu, “A 9-bit 215-MS/s Folding-Flash Time-to-Digital Converter Based on Redundant Remainder Number System”, IEEE Custom Integrated Circuits Conference, San Jose, CA (Sept. 2014).

[15] H. Kobayashi, H. Aoki, K. Katoh, C. Li, "Analog/Mixed-Signal Circuit Design in Nano CMOS Era", IEICE Electronics Express, vol.11 no.3, pp.1-15, 2014.

Chapter 4
STOCHASTIC TDC

In this chapter, section 4.1 introduces the concept of stochastic TDC architecture. Then, the stochastic TDC architecture with self-calibration feature is described in section 4.2. RTL verification on stochastic TDC is presented in section 4.3. Finally, the summary is provided in the section 4.4.

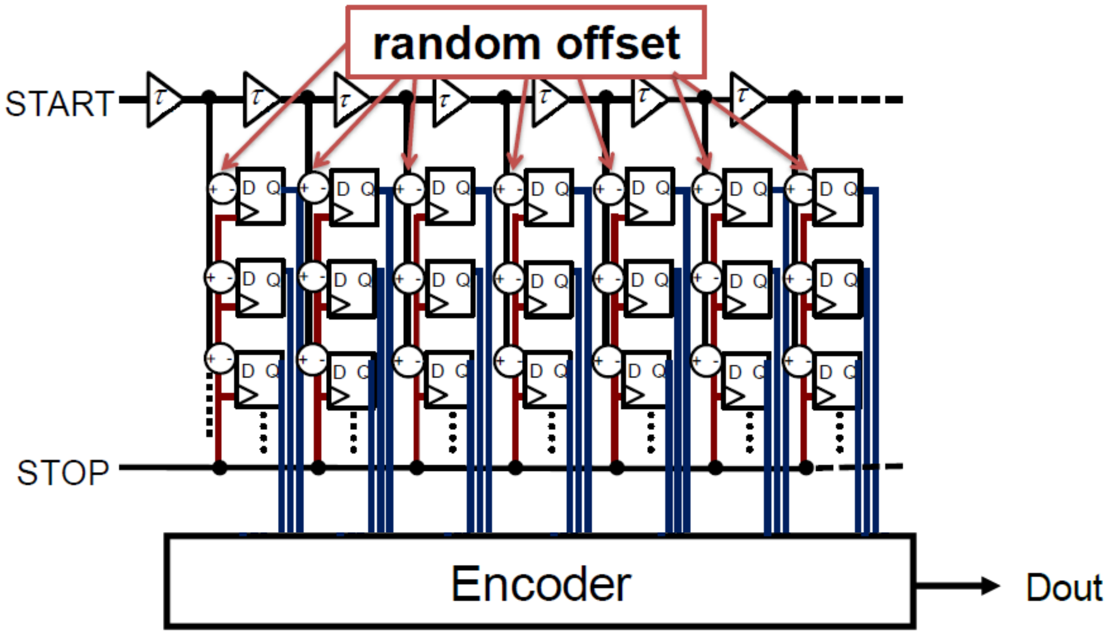
4.1 Concept of Stochastic TDC Architecture

The term stochastic occurs in a wide variety of professional or academic fields to describe events or systems that are unpredictable due to the influence of a random variable. In mathematics, specifically in probability theory, the field of stochastic processes has been a major area of research.

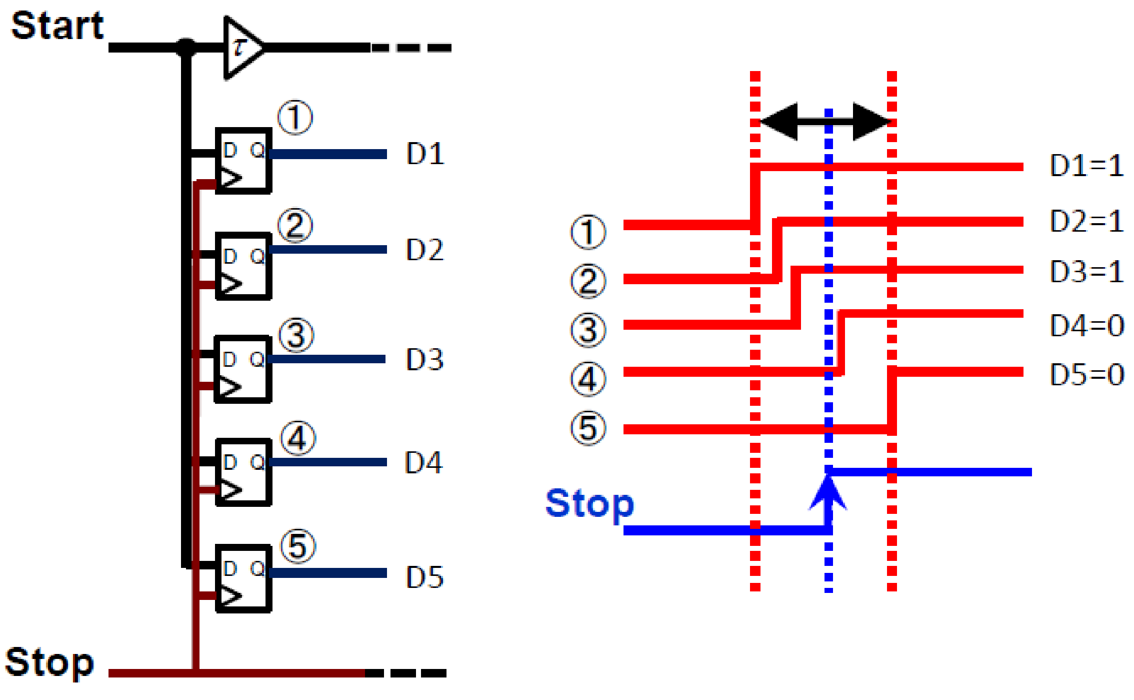
Manufacturing process variations have emerged as the primary challenge to the technology scaling of CMOS devices and circuits in the nano-CMOS regime. In the presence of process variations, device and interconnect parameters such as channel length, wire parasitics, etc., are modeled as random variables or spatial stochastic processes that vary across the sample space of the manufactured chips. Under such conditions, circuit performance characteristics like the voltage, delay, slew and power are also stochastic processes.

The stochastic variation in circuit characteristics of CMOS circuitry, which is caused by the integrated circuit manufacturing process uncertainty, could be utilized to obtain effective fine time resolution. Moreover, since the stochastic architecture utilizes the variation in characteristics positively, each MOSFET in delay line buffers and DFFs can be implemented with minimum channel length and width, which reduces power consumption and is advantageous in a high-switching-speed fine CMOS process.

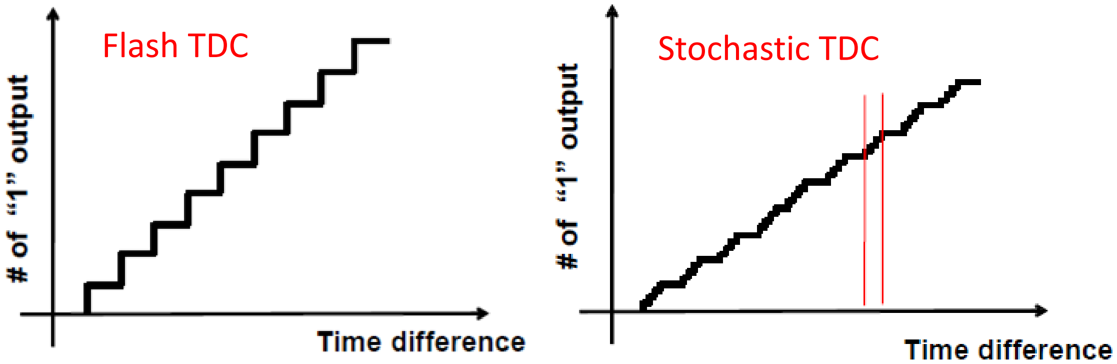
As illustrated in Fig.4.1 (a), the output of each delay buffer is connected to the data inputs of several DFFs. Since the setup and hold times of each DFF varies from each other due to process variations, the edge timing when the DFF changes the outputs from 0 to 1 can be different among these DFFs (Fig.4.1 (b)). This statistical variation among the DFFs can be utilized as the TDC's effective time resolution, which is much finer than the order of magnitude of a buffer delay τ (Fig.4.1 (c)).



(a) Concept of stochastic TDC architecture
(The setup and hold times of each DFF is assumed to be random.)



(b) Operation of stochastic TDC



(c) Flash-type TDC's coarse time resolution vs. Stochastic TDC's fine time resolution

Fig. 4. 1 Concept of stochastic TDC architecture

4.2 Stochastic TDC Architecture with Self-Calibration Feature

The stochastic architecture utilizes the stochastic process variation in deep-submicron CMOS positively to improve the resolution effectively. The stochastic TDC may be highly nonlinear, but its nonlinearity can be compensated by the self-calibration technology.

Stochastic TDC architecture with self-calibration feature is illustrated in Fig.4.2. In Fig.4.2 one ring oscillator with buffer delay τ_1 is employed on the upper row and the other ring oscillator with buffer delay τ_2 ($\tau_1 \neq \tau_2$) is applied on the lower row. The select signal of the multiplexer determines the handover between calibration mode and measurement mode (normal operation mode). In calibration mode, the delay line outputs are connected to their inputs, and the upper and lower ring oscillators are configured. The histogram engine is used to acquire histogram data. On the other hand, in measurement mode, the START and STOP signals are connected to the delay line inputs, and the upper and lower ring oscillator closed-loops are open. The digital error correction circuit is applied to conduct digital error correction operation based on the obtained histogram data in calibration mode.

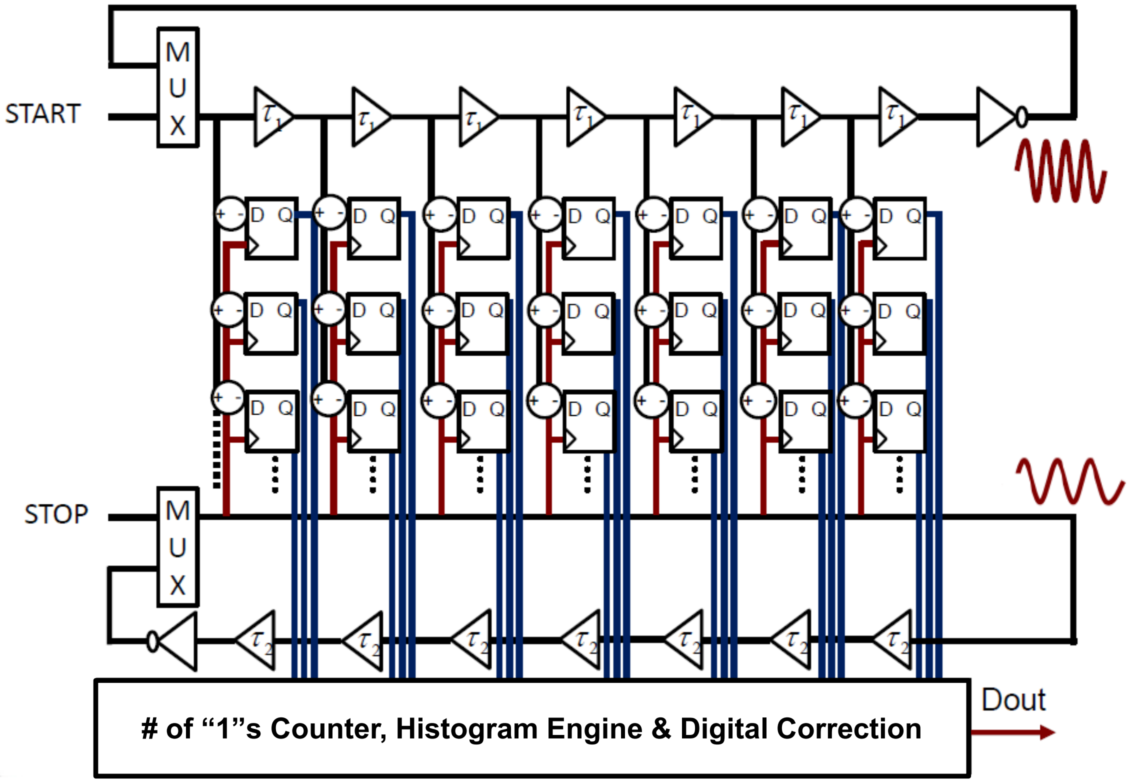


Fig. 4. 2 Stochastic TDC with self-calibration

(1) Calibration Mode (Histogram Data Acquisition)

In calibration mode, under the control of the multiplexer select signal, the delay line outputs are connected to their inputs, and the upper and lower ring oscillators are configured. As the buffer delay of the two ring oscillators are different, the two ring oscillators run freely with different oscillating frequencies, i.e. the upper ring oscillator and the lower ring oscillator are asynchronous (without correlation). When the STOP signal is on the rising edge, the D flip-flop array are triggered. The outputs of the D flip-flop array are transferred to the # of “1”s counter, and the histogram for each bin (digital output) is computed.

If there is no stochastic variation among the DFFs, i.e. all the setup and hold times of the DFFs were identical, the histogram for each bin (digital output) would be equal, after sufficiently large measurement times.

However, in practice, the setup and hold times of each DFF can vary and be different from each other due to process variations. When the upper ring oscillator runs freely, the lower ring oscillator produces rising edges to trigger the DFFs to generate digital outputs. As the probability of digital code for large delay is high, while the probability of digital code for small delay is low. The digital output is with the probability proportional to the setup and hold time of the corresponding DFF. Thus the relative variation (ratio) among DFFs can be measured by the histogram engine. After large enough measurement cycles, each bin of histogram varies with corresponding DFF’s stochastic variation.

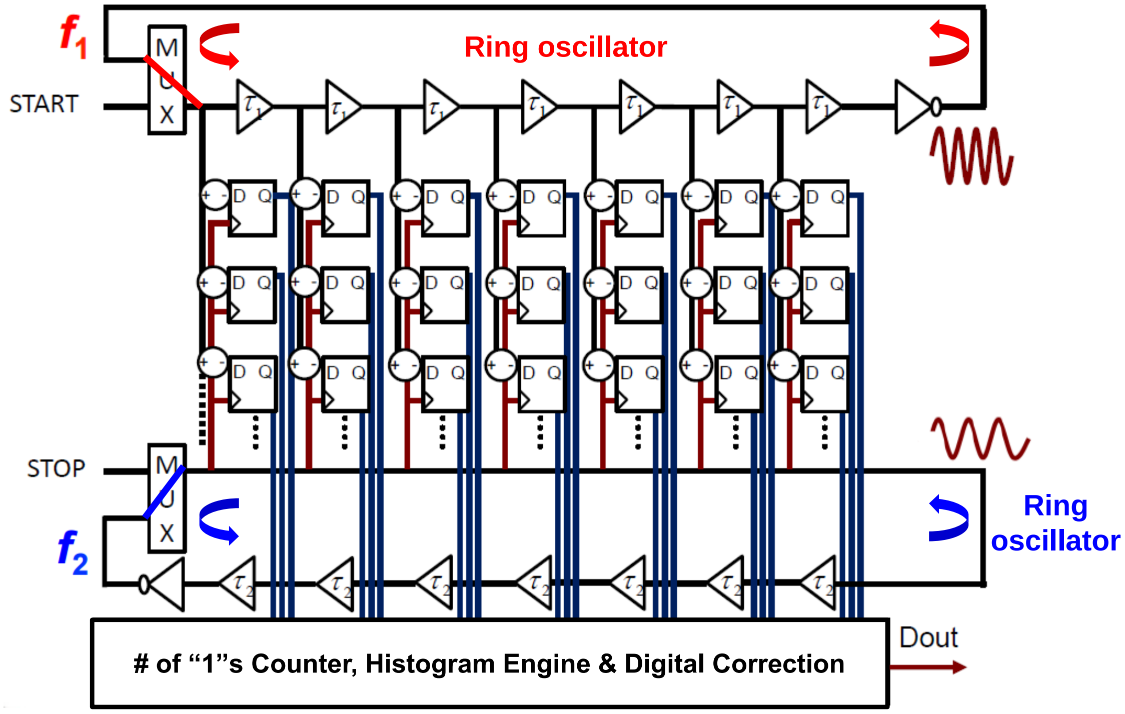


Fig. 4. 3 Self-calibration mode

(2) Measurement Mode

In measurement mode, under the control of the multiplexer select signal, the START and STOP signals are connected to the delay line inputs, and the upper and lower ring oscillator closed-loops are open. The START signal and STOP signal are inputted as a normal flash-type TDC, and the thermometer code corresponding to the rising-edge timing interval between them is outputted.

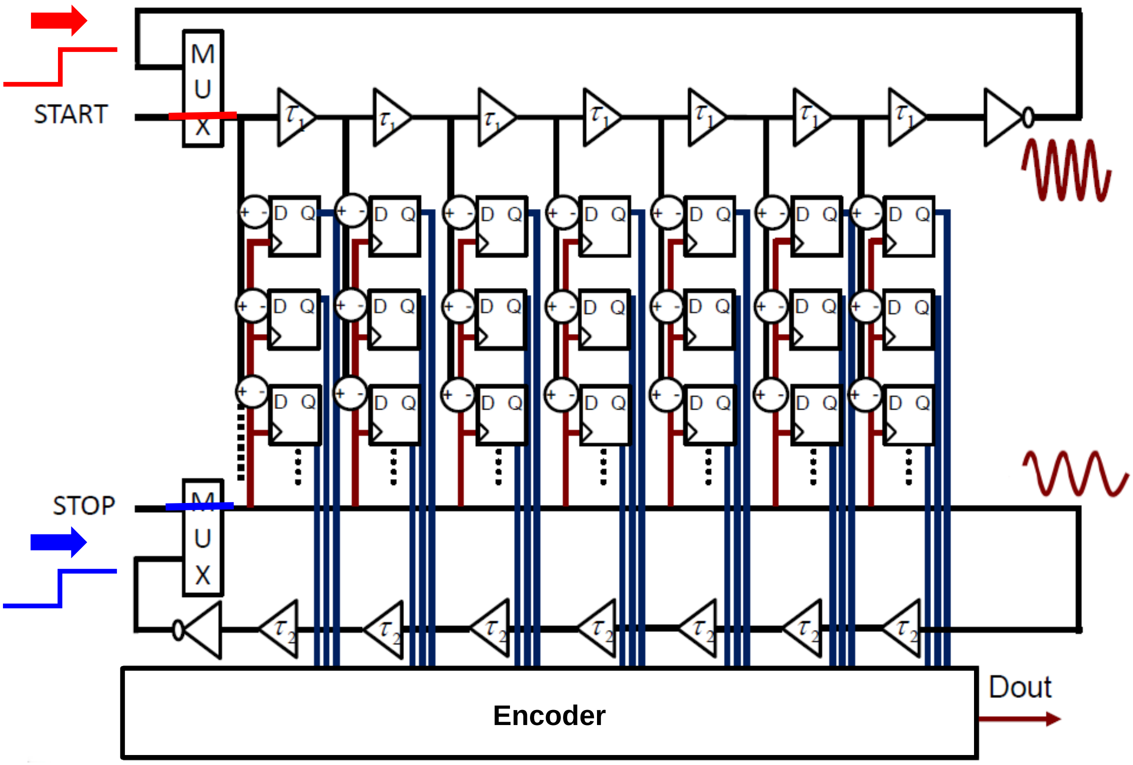


Fig. 4. 4 Measurement mode (normal operation mode)

(3) Digital Error Correction Operation

The histogram data of each (relative) setup and hold time among DFFs is acquired in calibration mode. In measurement mode, the digital error correction operation is conducted based on the acquired histogram data, as shown in equation (4-1).

$$Dout(N) = \frac{\sum_{i=1}^{N} Pin(i)}{\sum_{i=1}^{FS} Pin(i)} \times FS \qquad (4-1)$$

N : digital output to be corrected

$D_{out}(N)$: corrected digital output for raw TDC output N

$Pin(i)$: histogram for raw TDC output i

FS : maximum TDC digital output value

Non-linearity correction can be made by applying inverse function of $D_{out}(N)$, and the TDC linearity calibration can be achieved.

(4) Easy Calibration for Time-Domain Signals

Stochastic architecture is originally generated from stochastic flash ADC (SFADC) design, which uses offset voltage variations in comparators as voltage references of A-to-D conversion. As the comparator offset voltages are change randomly, the SFADC is usually highly nonlinear. However, as the signal processed in ADC is “voltage”, it is very difficult to apply self-calibration technology to ADC [6][7].

On the other hand, in TDC, the signal processed is “time” instead of “voltage”. Self-calibration technology can be easily applied to stochastic TDC to compensate the nonlinearity. Moreover, the digital-like constructions of histogram data acquisition and digital error correction enable compatibility with digital LSI process. Stochastic TDC architecture with self-calibration feature can be easily implemented in an all-digital way.

4.3 RTL Verification on Stochastic TDC

4.3.1 Simulation Model

RTL simulation was conducted to verify the operation and effectiveness of stochastic TDC. The stochastic TDC modeled in RTL simulation is illustrated in Fig.4.5.

As illustrated in Fig.4.5 (a), in measurement mode, the START and STOP signals are connected to the delay line inputs as a normal flash-type TDC, all the generated output Q are delivered to the thermal-to-binary encoder, and transferred into a binary representation of the time interval between START and STOP.

And as shown in Fig.4.5 (b), in calibration mode, one ring oscillator with two dummy buffers is employed on the upper row and the frequencies of the upper ring oscillator and the STOP signal are different, i.e. the upper ring oscillator and the STOP signal are asynchronous (without correlation). The histogram data acquisition and digital error correction can be achieved as follows:

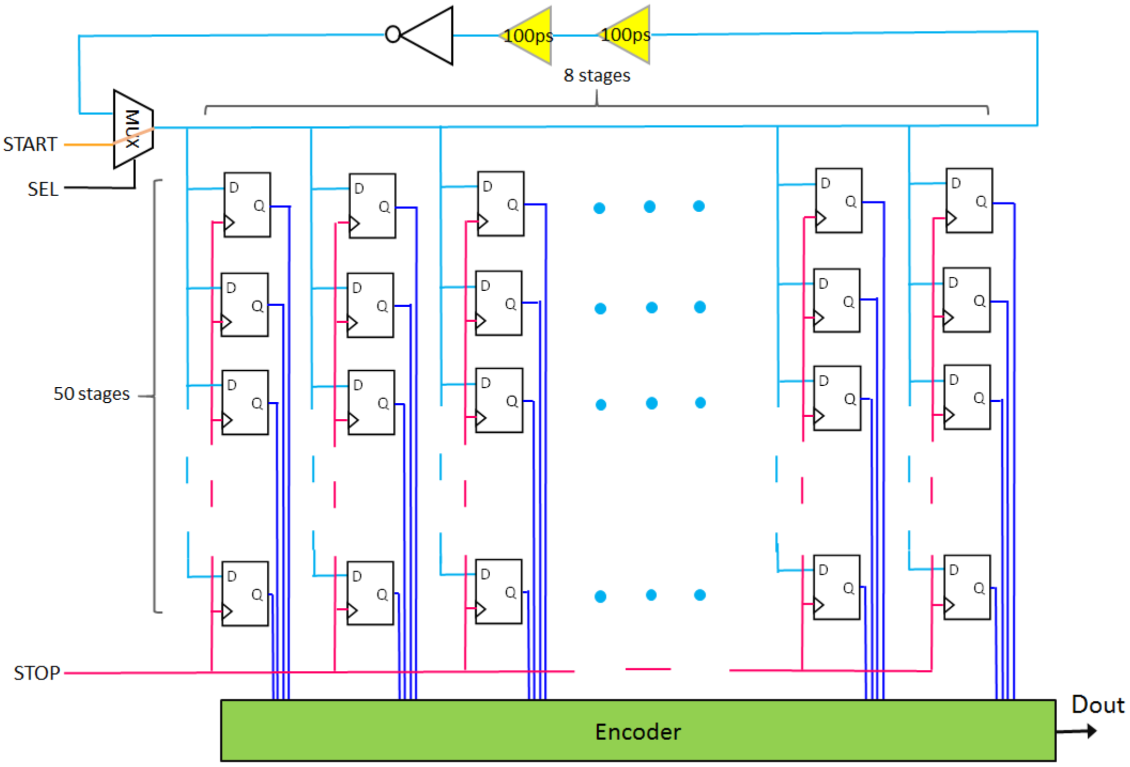
(1) When SEL signal is in LOW state, the upper ring oscillator is initialized by the START signal.

(2) When SEL signal goes from LOW to HIGH, the upper ring oscillator begins to oscillate.

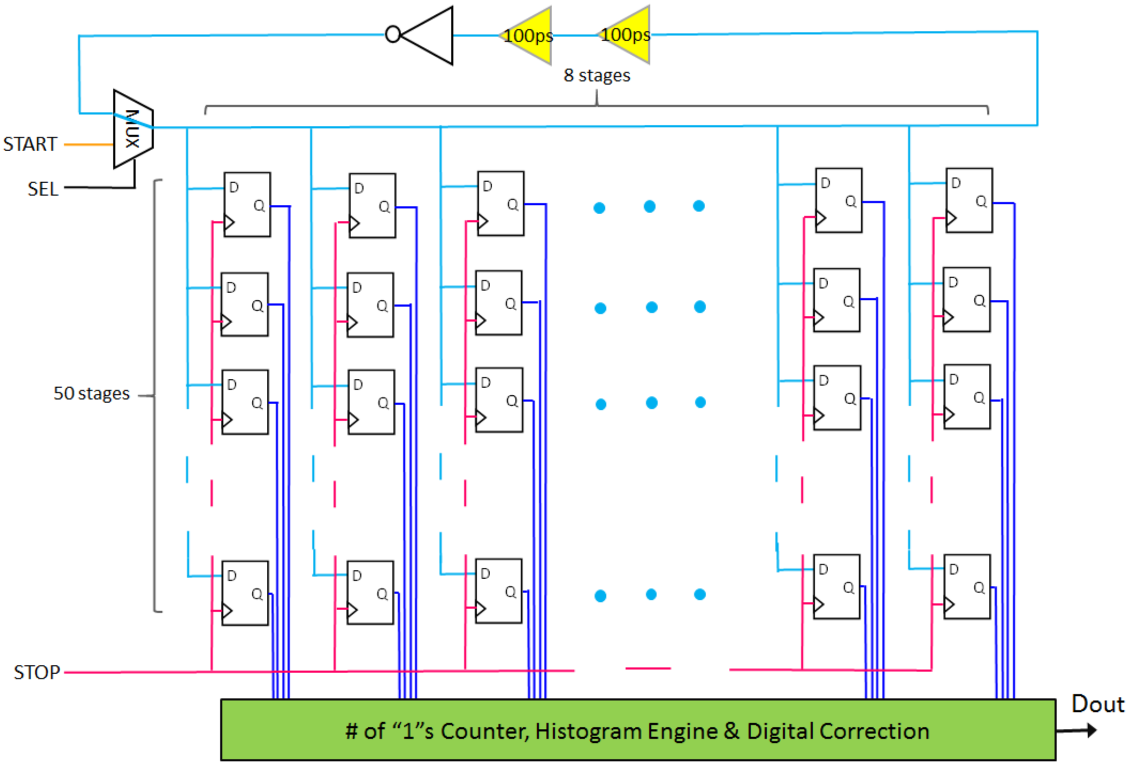
(3) When STOP signal is on the rising edge, i.e. LOW-to-HIGH transition, the DFFs are triggered and the value of D is transferred to the output Q. Each Q is with probability proportional to the setup and hold time of the corresponding DFF.

(4) All the generated output Q are delivered to the # of "1"s counter, and the histogram for each bin (digital output) is computed. The relative variation (ratio) among DFFs is measured by the histogram engine. After large enough measurement cycles, each bin of histogram varies with corresponding DFF's stochastic variation.

(5) As the histogram data of each (relative) setup and hold time among DFFs is acquired in calibration mode. In measurement mode, the digital error correction operation is conducted based on the acquired histogram data.



(a) Measurement mode



(b) Calibration mode

Fig. 4. 5 Simulation model of stochastic TDC

In practice, the setup and hold times of each DFF can vary and be different from each other due to process variations. To model this uncertainty in RTL simulation, we model the setup and hold times of the DFFs as a normally distributed stochastic variable τ_{DFF}, with a τ mean and a standard deviation σ.

The index of the DFFs is defined by its column address and row address. For example, DFF11 stands for the DFF in column stage 1 and row stage 1. DFF_Q422 represents for the DFF in column stage 4 and row stage 22. The setup and hold times of each DFF (case #1) is listed in Table 4.1.

Table 4. 1 The setup and hold times of each DFF (case #1)

(Normal distribution, mean = 20ps, standard deviation = 6, unit: ps)

Column Row	1	2	3	4	5	6	7	8
1	29.98	26.06	8.94	9.45	21.94	21.53	20.50	14.58

2	23.28	23.50	12.39	24.67	27.11	18.22	12.38	26.56
3	24.48	19.28	22.86	23.98	30.90	29.14	3.71	15.03
4	15.13	28.56	12.75	18.37	15.68	10.23	23.66	24.34
5	21.72	25.84	13.99	15.72	14.28	18.26	26.12	12.65
6	16.26	14.95	22.71	24.04	15.80	27.79	23.89	21.03
7	17.60	17.37	19.91	6.91	22.41	15.45	22.19	17.51
8	21.34	3.97	25.61	16.77	29.67	19.50	22.36	16.27
9	25.83	30.94	16.37	29.24	29.45	19.35	17.56	23.40
10	17.13	21.00	24.56	23.41	23.53	23.72	14.33	17.50
11	20.14	35.19	18.24	29.43	25.66	20.42	17.10	15.84
12	20.41	21.98	21.31	24.74	17.90	11.71	19.59	23.93
13	12.09	25.10	25.55	17.31	16.44	13.21	19.74	15.73
14	26.18	20.33	13.98	33.21	19.65	19.80	26.64	24.87
15	27.57	15.41	14.64	12.22	22.43	16.89	16.46	19.05
16	21.30	25.16	27.43	7.84	26.07	21.95	20.78	22.48
17	18.93	25.33	16.76	21.05	22.56	16.19	35.35	4.09
18	18.43	18.78	22.44	20.75	20.98	10.36	25.68	19.94
19	17.28	25.81	15.25	26.01	15.64	24.89	21.35	17.01
20	22.84	20.74	21.28	21.01	14.45	26.68	20.35	28.14
21	16.50	25.27	15.19	7.46	23.31	20.08	27.74	17.29
22	26.49	19.93	34.95	23.44	24.40	12.25	21.29	22.37
23	23.59	17.55	9.49	9.28	9.69	16.83	27.73	30.38
24	14.91	15.91	21.58	19.56	28.00	23.74	17.93	15.74
25	34.83	5.96	4.91	17.11	21.52	13.78	21.99	17.54
26	15.01	28.57	14.85	13.60	16.87	23.57	13.86	24.86
27	17.82	28.06	28.95	22.16	18.61	25.02	10.20	22.35
28	23.25	23.63	20.23	8.79	27.47	16.90	10.60	22.69
29	15.20	27.36	27.63	24.18	18.02	21.13	13.84	22.70
30	14.42	19.48	18.84	13.96	26.23	15.11	10.31	22.49
31	35.90	29.79	18.54	18.01	15.62	26.86	19.73	28.66

32	22.87	15.18	22.15	30.55	26.76	20.95	23.82	13.64
33	14.16	20.31	25.12	23.21	17.57	21.27	26.43	16.48
34	14.41	20.79	26.72	24.49	22.83	14.78	14.31	14.26
35	24.15	13.29	30.89	19.29	18.27	15.16	22.23	25.11
36	18.58	22.85	18.36	18.04	38.67	24.64	14.03	19.13
37	1.82	11.04	15.94	19.30	20.05	23.67	21.04	15.95
38	22.73	9.60	21.63	10.39	20.10	16.69	26.97	17.48
39	26.11	7.76	19.32	10.92	13.59	17.88	18.69	19.58
40	18.05	27.83	23.14	16.34	23.69	24.99	16.33	19.33
41	11.54	25.85	26.99	10.94	15.92	19.01	24.57	18.48
42	11.97	30.59	19.71	11.41	23.91	33.20	17.52	22.13
43	18.23	29.40	31.01	12.59	9.68	22.30	20.97	29.54
44	27.42	20.73	22.42	28.43	20.04	22.57	18.75	18.72
45	18.64	19.21	19.17	27.37	26.92	26.98	13.36	33.02
46	21.12	13.65	9.57	24.55	21.57	26.80	20.99	23.65
47	24.85	29.25	13.01	16.08	13.14	22.79	18.51	19.52
48	15.15	13.25	27.28	13.97	17.85	21.40	15.10	20.06
49	24.39	24.47	23.29	22.20	18.83	11.70	26.89	17.07
50	19.22	26.42	21.25	16.71	18.71	18.25	17.58	15.96

The probability density function of DFFs’ setup and hold times in case #1 is illustrated in Fig.4.6.

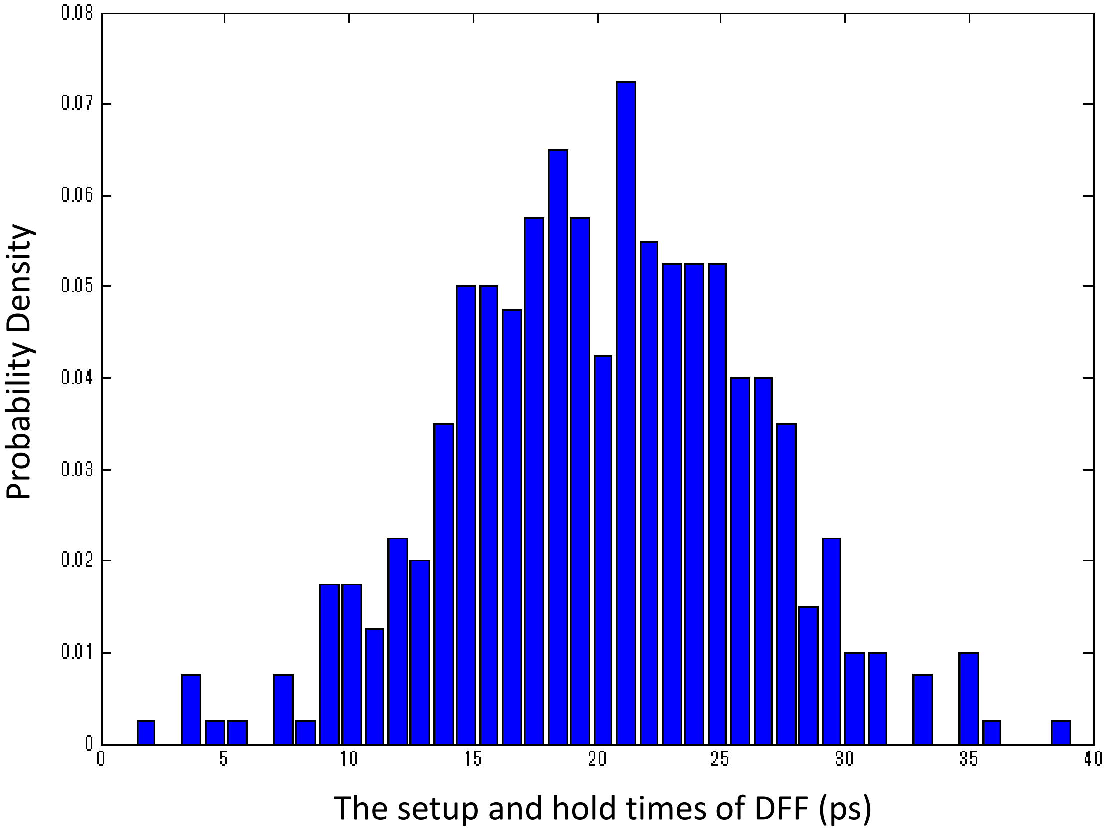


Fig. 4. 6 The probability density function of DFFs' setup and hold times (case #1)

Three different stochastic TDC patterns are simulated separately in order to make a comparison.

- Pattern 1: the number of the DFFs exploited in the stochastic TDC is 100 (2 columns × 50 rows). The setup and hold times of each DFF is the same as the value listed in column 1~2 in Table 4.1.

- Pattern 2: the number of the DFFs applied in the stochastic TDC is 200 (4 columns × 50 rows). The setup and hold times of each DFF is the same as the value listed in column 1~4 in Table 4.1.

- Pattern 3: the number of the DFFs used in the stochastic TDC is 400 (8 columns × 50 rows). The setup and hold times of each DFF is the same as the value listed in column 1~8 in Table 4.1.

4.3.2 Simulation Results

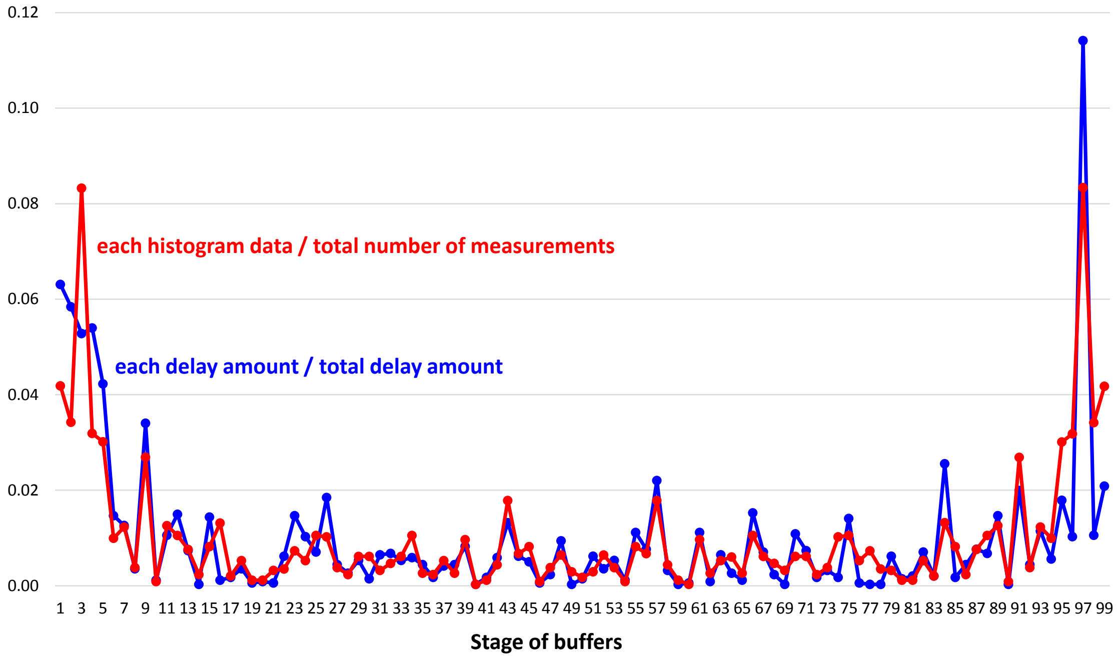
This RTL simulation results of the above three patterns are as follows:

(1) Correlation between the Histogram and the DFF's Stochastic Variation

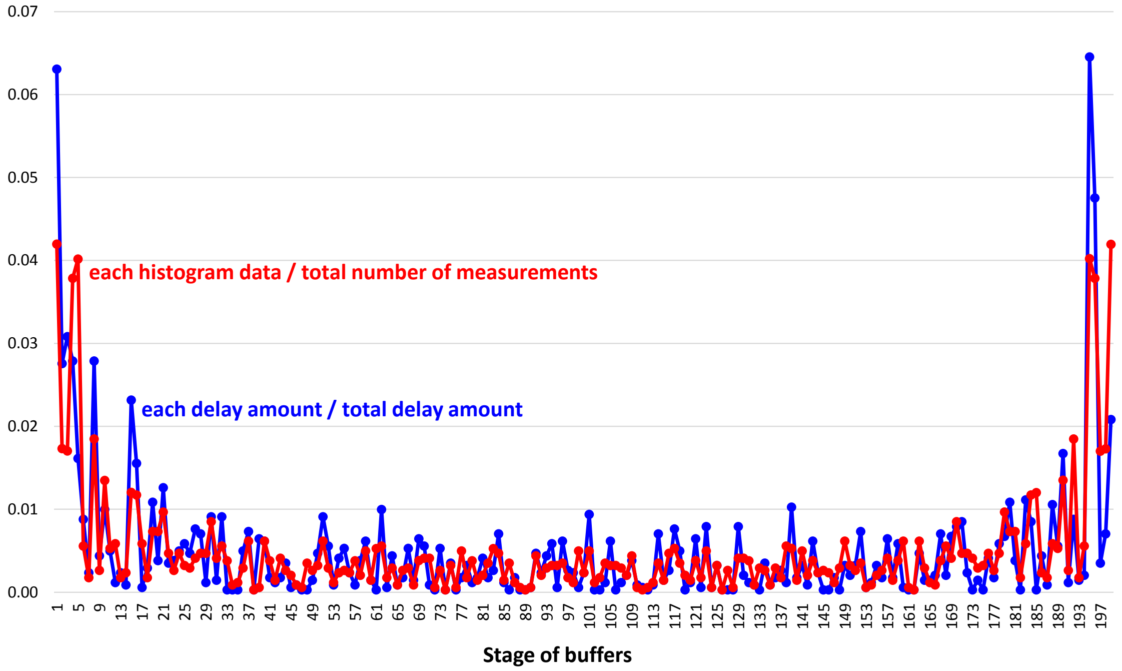
The setup and hold time of each DFF is normalized with dividing by the total delay amount, i.e. (each delay amount / total delay amount), which represents the relative variation (ratio) among the DFFs.

Measured histogram data for each bin of the stochastic TDC with the self-calibration is normalized with dividing by the total number of measurements, i.e. (each histogram data / total number of measurements), which stands for the relative variation (ratio) among the histogram bins.

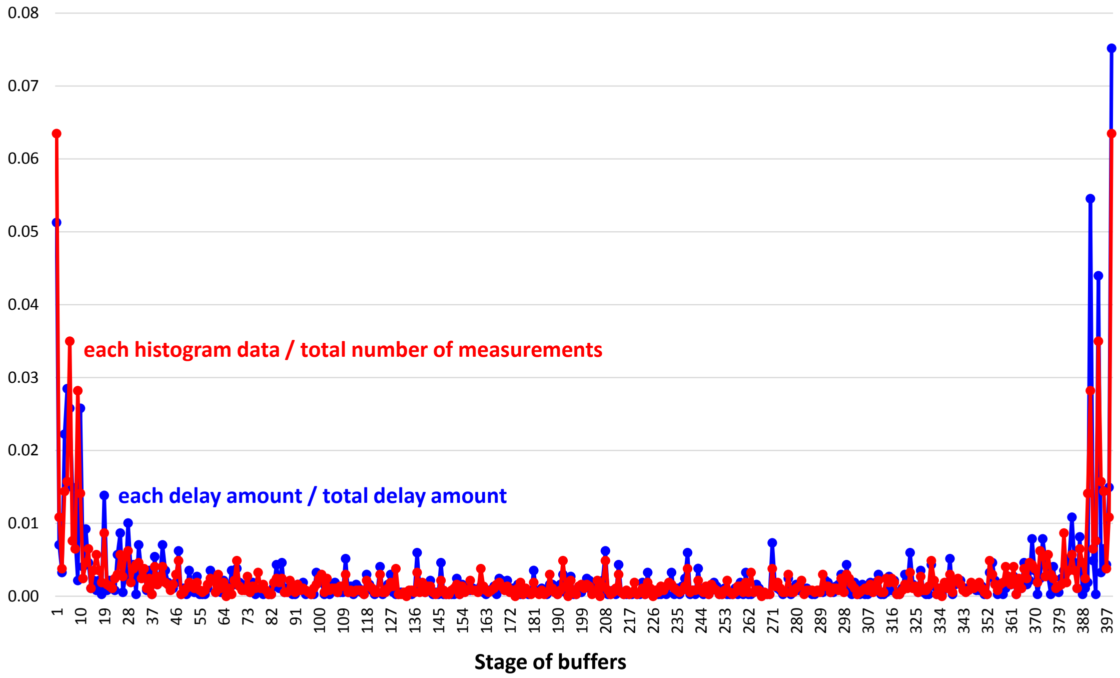
The relative variation among the DFFs and the relative ratio among the histogram bins are illustrated in Fig.4.7. The simulation results indicate that there is strong correlation between the histogram and the corresponding DFF’s stochastic variation: the histogram for each bin (digital output) is with probability proportional to the setup and hold time of the corresponding DFF.



(a) The number of the DFFs exploited in the stochastic TDC is 100



(b) The number of the DFFs exploited in the stochastic TDC is 200



(c) The number of the DFFs exploited in the stochastic TDC is 400

Fig. 4. 7　Correlation between histogram and DFF's stochastic variation (case #1)

(2) Input-output Characteristics before and after Calibration

The linearity is compensated based on the correlation between the histogram data and the DFF's stochastic variation, as shown in equation (4−2).

$$Dout(N)=\frac{\sum_{i=1}^{N} Pin(i)}{\sum_{i=1}^{FS} Pin(i)} \times FS \qquad (4-2)$$

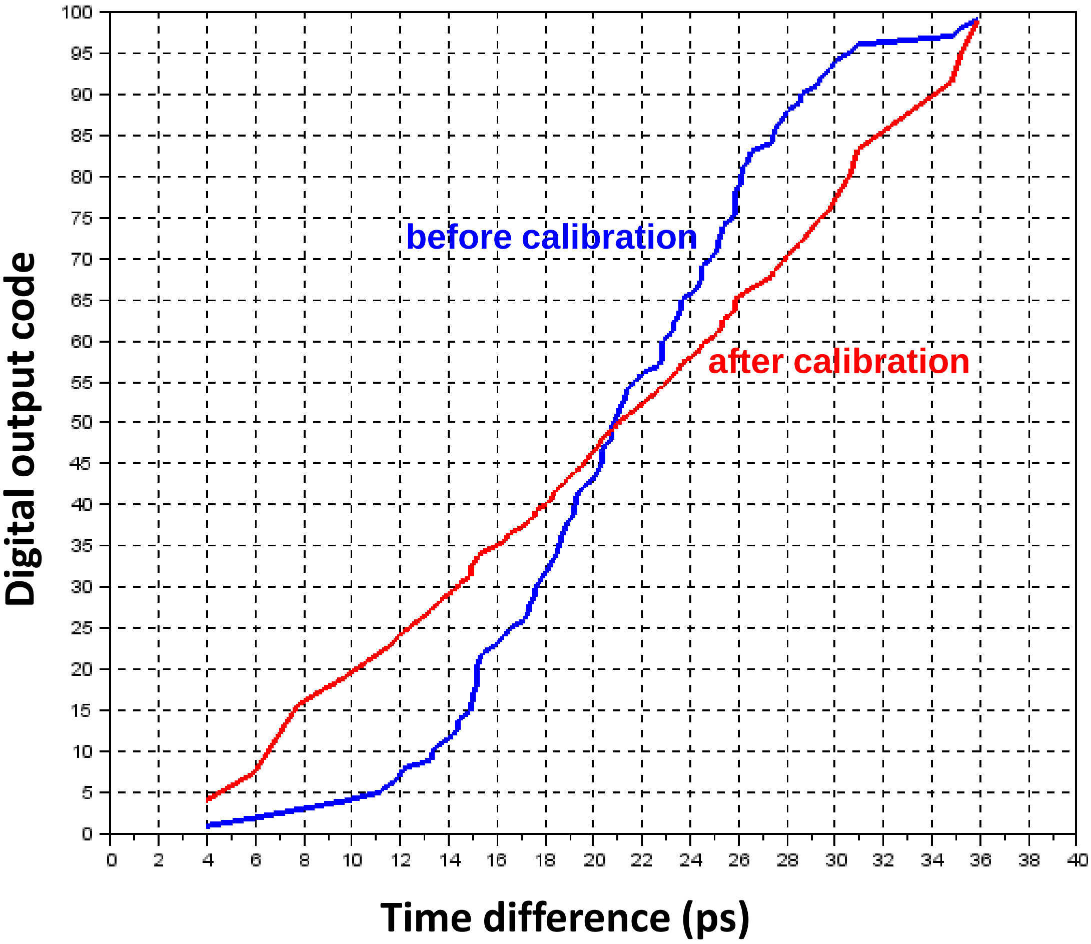
N : digital output to be corrected

$D_{out}(N)$: corrected digital output for raw TDC output N

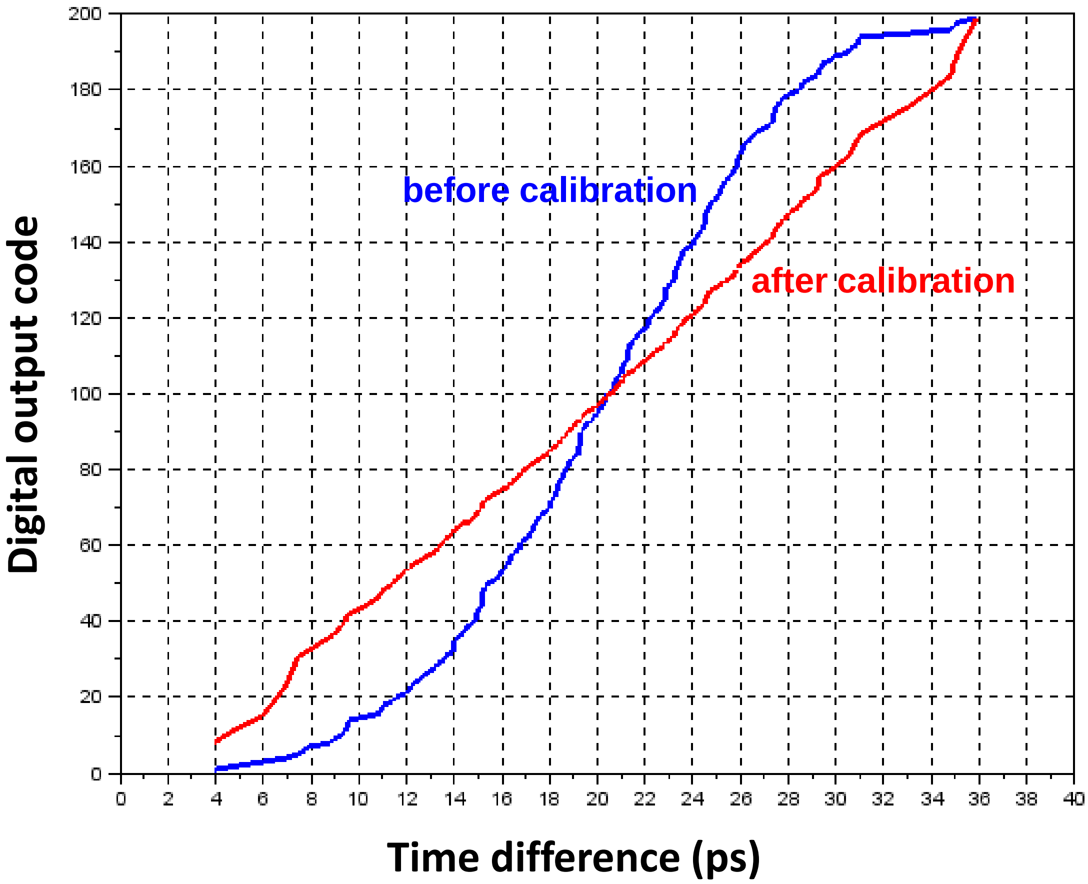
$Pin(i)$: histogram for raw TDC output i

FS : maximum TDC digital output value

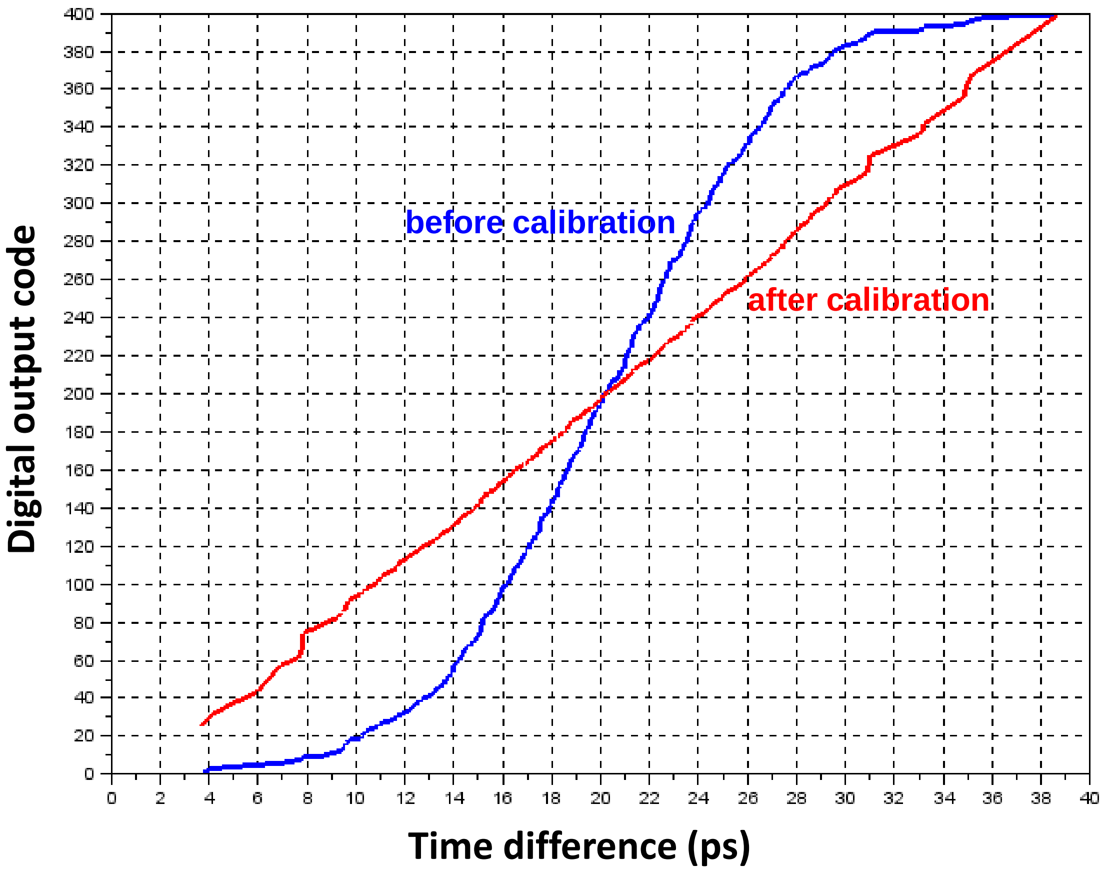
The input-output characteristics of the stochastic TDC before and after calibration are illustrated in Fig.4.8. The simulation results show that the TDC linearity is improved after calibration.



(a) The number of the DFFs exploited in the stochastic TDC is 100



(b) The number of the DFFs exploited in the stochastic TDC is 200



(c) The number of the DFFs exploited in the stochastic TDC is 400

Fig. 4. 8 Input-output characteristics before and after calibration (case #1)

(3) Measured INL before and after Calibration

The calibration results of the stochastic TDC are evaluated quantitatively. A least-mean-square (LMS) method is applied to find a linear approximation to the measurement results. The integral nonlinearity (INL), i.e. the deviation from the straight line is calculated. INL is an index of the cumulative error between the measurement result and the linear approximation straight line. For INL, the more close to 0 the better. The gain of the linear approximation straight line and offset can be calculated by the following formulas:

$$gain = \frac{N \cdot K_4 - K_1 \cdot K_2}{N \cdot K_3 - K_1^2} \qquad (4-3)$$

$$offset=\frac{K_2}{N}-gain\cdot\frac{K_1}{N} \quad (4-4)$$

when the number of the DFFs exploited in the stochastic TDC is 100, 200, and 400, N is equal to 100, 200, and 400 respectively. K_1 to K_4 are defined as follows:

$$K_1=\sum_{i=1}^{N} i \quad (4-5)$$

$$K_2=\sum_{i=1}^{N} S(i) \quad (4-6)$$

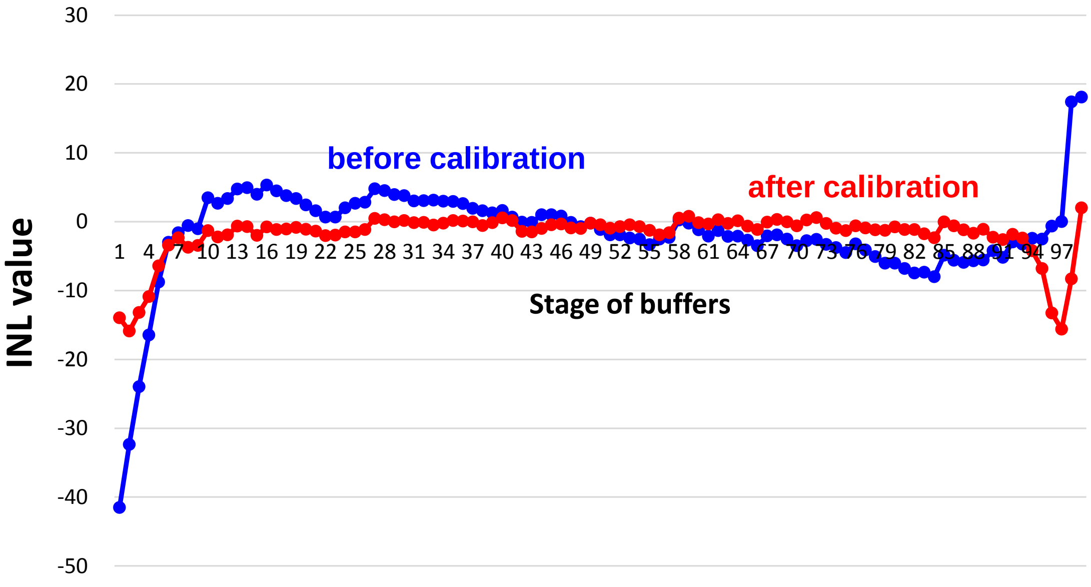
$$K_3=\sum_{i=1}^{N} i^2 \quad (4-7)$$

$$K_4=\sum_{i=1}^{N} i\cdot S(i) \quad (4-8)$$

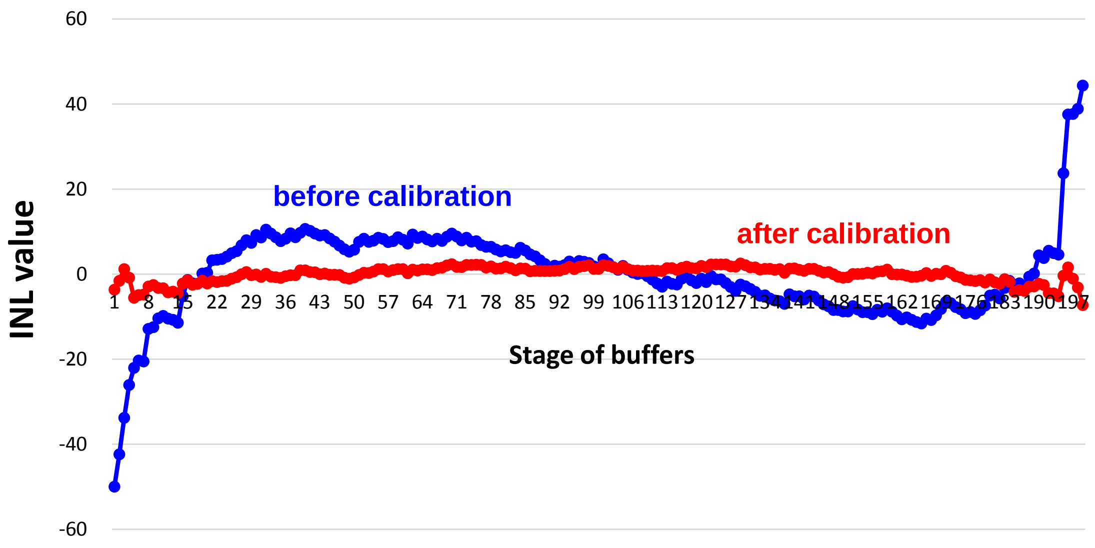
here i indicates a normalized input time difference given by the reference clock division and $S(i)$ is the corresponding TDC output histogram. The INL can be denoted by

$$INL(i)=\frac{S(i)-(gain\cdot i+offset)}{gain} \quad (4-9)$$

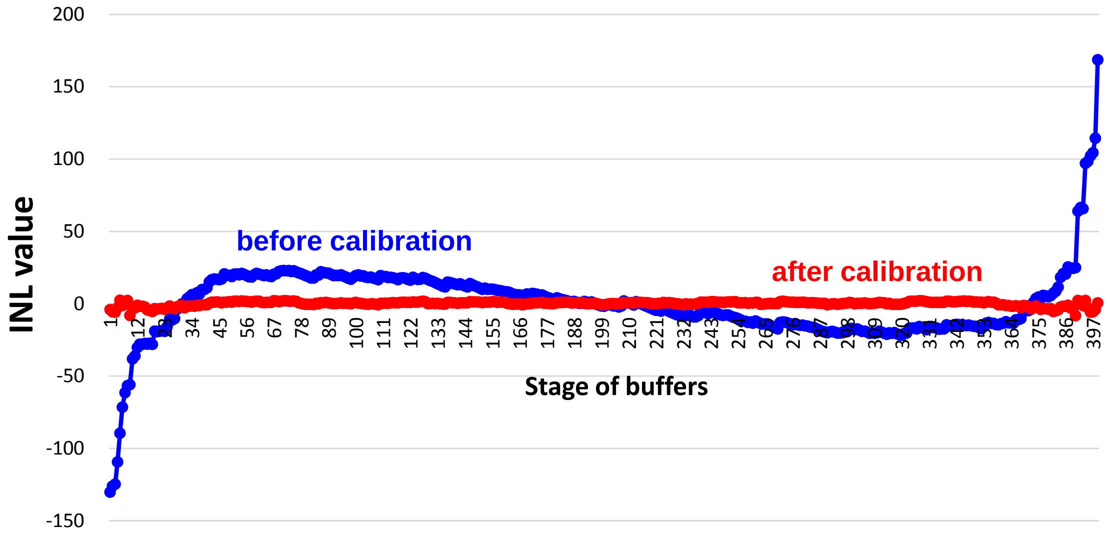
The INL calculated from the above formulas are illustrated in Fig.4.9. The simulation results indicate that INL is reduced after the calibration.



(a) The number of the DFFs exploited in the stochastic TDC is 100



(b) The number of the DFFs exploited in the stochastic TDC is 200



(c) The number of the DFFs exploited in the stochastic TDC is 400

Fig. 4. 9 Measured INL before and after calibration (case #1)

(4) Time Resolution

Time resolution represents the minimum time interval that can theoretically be resolved by the TDC in a single measurement, that is, the quantization step (LSB). It can be calculated by the following formula:

$$Time\_resolution = \frac{maximum\_output\_code - minimum\_output\_code}{the\_number\_of\_levels} \quad (4-10)$$

Table 4. 2 Time resolution after calibration (case #1)

Pattern	Time resolution
Pattern 1: the number of the DFFs exploited in the stochastic TDC is 100	0.3258ps
Pattern 2: the number of the DFFs exploited in the stochastic TDC is 200	0.1613ps
Pattern 3: the number of the DFFs exploited in the stochastic TDC is 400	0.0876ps

Table 4. 3 Comparison with other TDC architectures

TDC architecture	Time resolution
This work	0.0876ps
Freeze Vernier [11]	4.88ps
Vernier gated ring oscillator [12]	3.2ps
Delay line [13]	6.25ps
2D Vernier [14]	4.8ps
Local passive interpolation [15]	4.7ps
Inverter-chain [16]	80.0ps
Two-step [17]	3.75ps

We see that the stochastic TDC may be highly nonlinear before calibration, but its nonlinearity can be compensated by the self-calibration method, and its time resolution after calibration can reach sub-picosecond level.

It is worth to mention that though the multiplexer delay and the inverter delay are not considered in RTL simulation, these delays can be equivalently convert into certain stochastic variations among the DFFs. That is to say, the multiplexer delay and the inverter delay do not influence stochastic TDC providing a linear output with sub-picosecond time-resolution.

4.3.3 The Influence of the Number of DFFs

The influence of the number of DFFs on the performance of stochastic TDC is obvious from the simulation results.

For the input-output characteristics and measured INL before calibration, there is no significant difference among the three patterns. However, for these performance parameters after calibration, there are obvious difference among the three patterns.

From pattern 1 to pattern 2 then pattern 3, the number of the DFFs exploited in the stochastic TDC increases from 100 to 200 then 400. Meanwhile, the improvement of the linearity of the input-output characteristics and the reduction of measured INL after calibration increases obviously.

This is because the stochastic architecture utilizes the stochastic variation in deep-submicron CMOS. The setup and hold time of the DFFs are usually modeled as a normally distributed stochastic variable τ_{DFF}, with a τ mean and a standard deviation σ.

Though the setup and hold times of each DFF can vary and be different from each other due to manufacturing process variations, the stochastic variation among them is very small compared to the measurement range. In this situation, in order to make the self-calibration result accurate, adequate DFFs are required. The minimum required number of DFFs can be calculated by:

$$\begin{aligned}&The\_minimum\_required\_number\_of\_DFFs\\&=\frac{measurement\_range}{the\_order\_of\_magnitude\_of\_the\_stochastic\_variation\_among\_DFFs}\end{aligned} \qquad (4-11)$$

For example, in above simulation, the measurement range is about 40ps and the order of magnitude of the stochastic variation among DFFs is 10^{-1} ps. In this case, the required number of DFFs for accurate calibration is about $40ps \div 10^{-1} ps = 400$.

4.3.4 Other Delay Variation Cases

Beside delay variation case #1, other delay variation cases for stochastic TDC are simulated to verify the operation principle. The detail information of the cases is listed in Table 4.4.

Table 4. 4 Delay Variation Cases
(The setup and hold times of each DFF for case #2 ~ #6 are listed in the appendix.)

Case #	Mean of normal distribution	Standard deviation of normal distribution	The number of DFFs
#1	20ps	6	100, 200, 400
#2	20ps	6	400
#3	20ps	6	400
#4	20ps	1	400
#5	20ps	2	400
#6	20ps	3	400

The input-output characteristics and measured INL before and after calibration for case #2 ~ #6 are presented below. The simulation results indicate that the TDC linearity is improved after calibration, and the INL is reduced after the calibration.

(1) Case #2 (Normal Distribution, Mean = 20ps, Standard Deviation = 6)

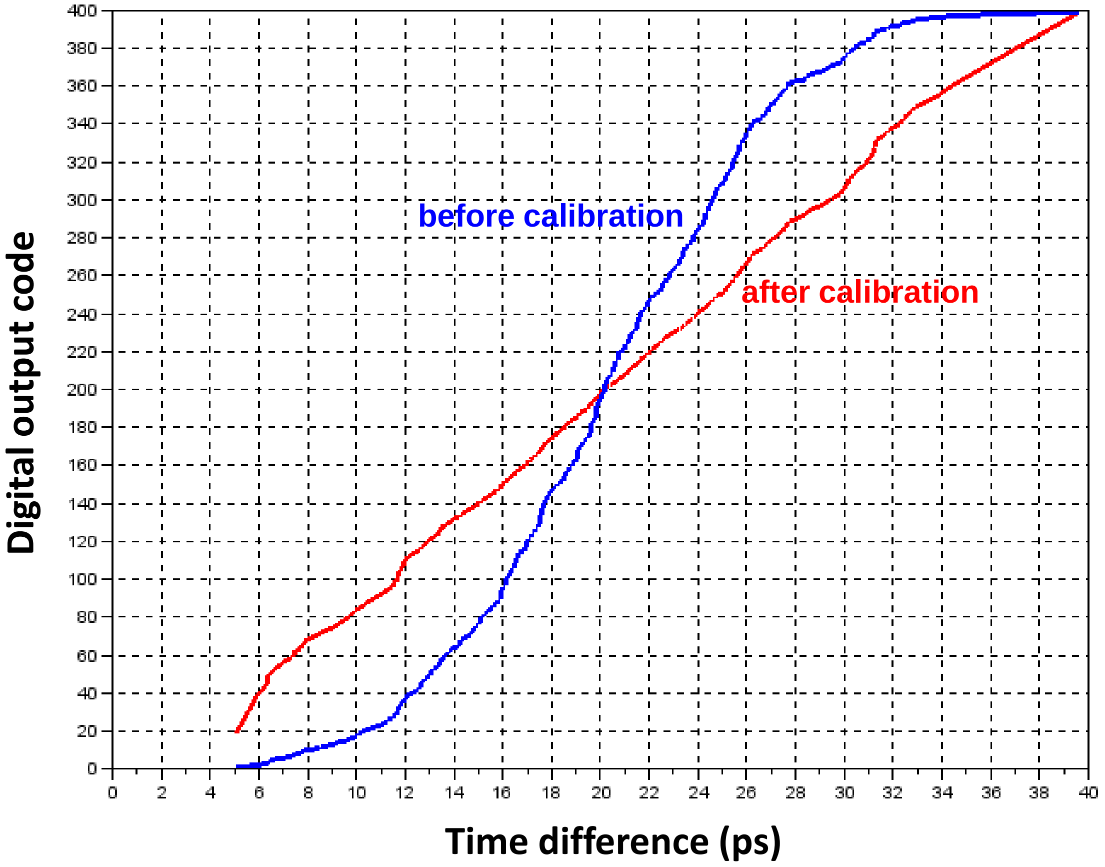


Fig. 4. 10 Input-output characteristics before and after calibration (case #2)

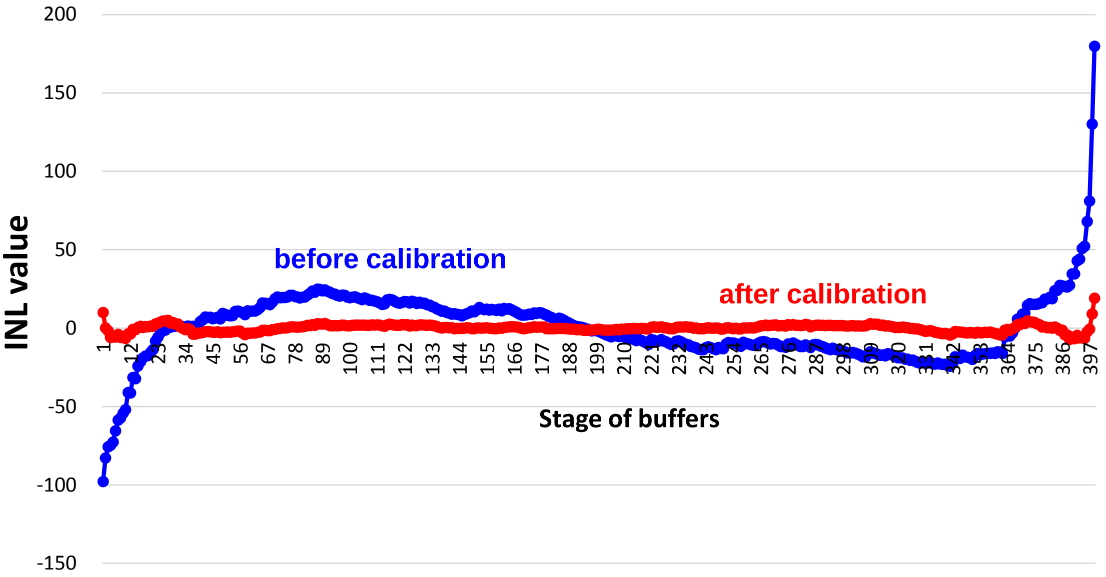


Fig. 4. 11 Measured INL before and after calibration (case #2)

(2) Case #3 (Normal Distribution, Mean = 20ps, Standard Deviation = 6)

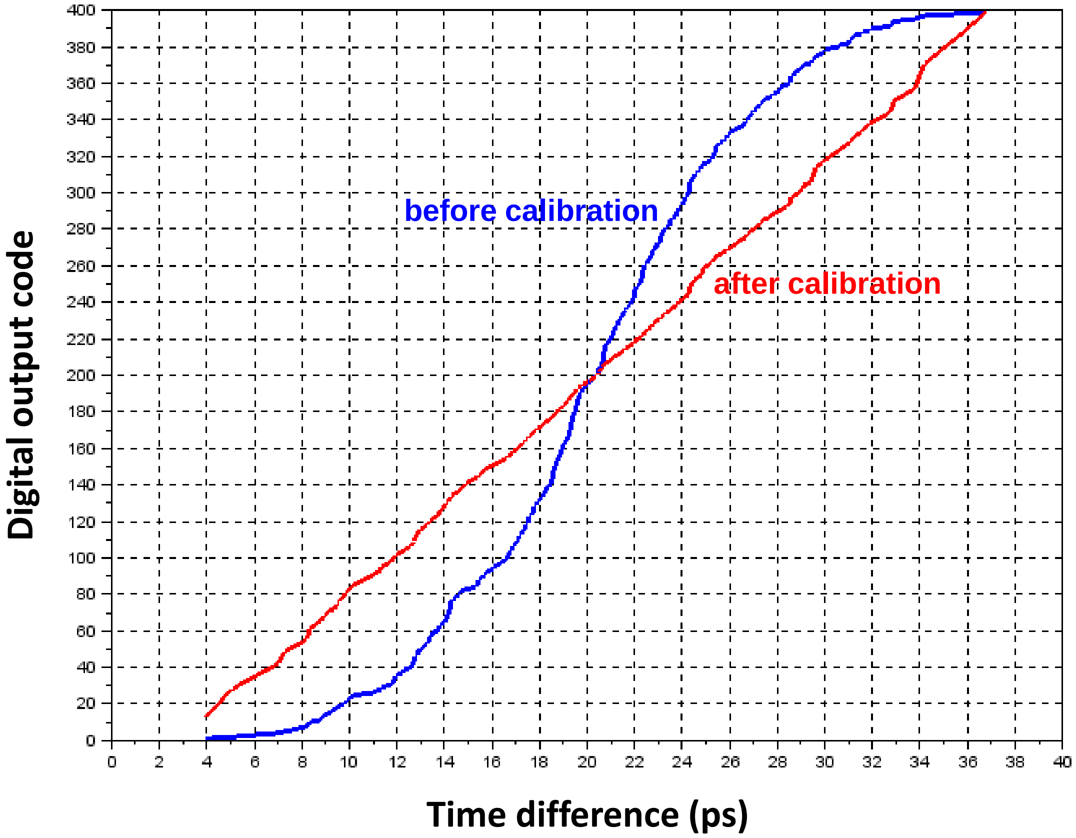


Fig. 4. 12 Input-output characteristics before and after calibration (case #3)

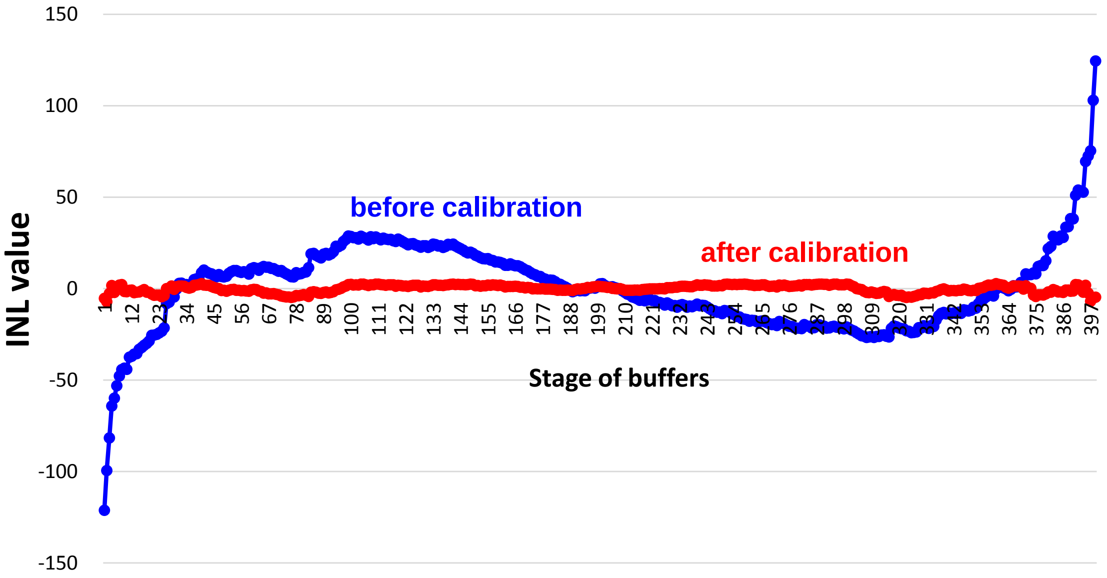


Fig. 4. 13 Measured INL before and after calibration (case #3)

(2) Case #4 (Normal Distribution, Mean = 20ps, Standard Deviation = 1)

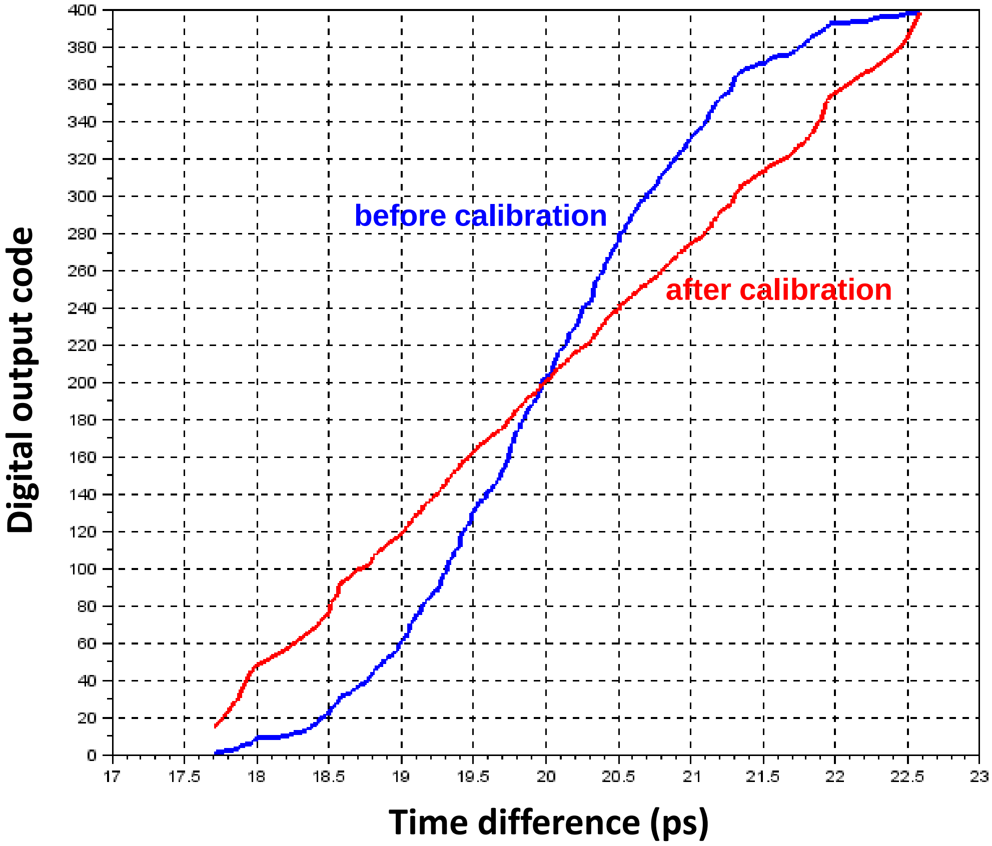


Fig. 4. 14 Input-output characteristics before and after calibration (case #4)

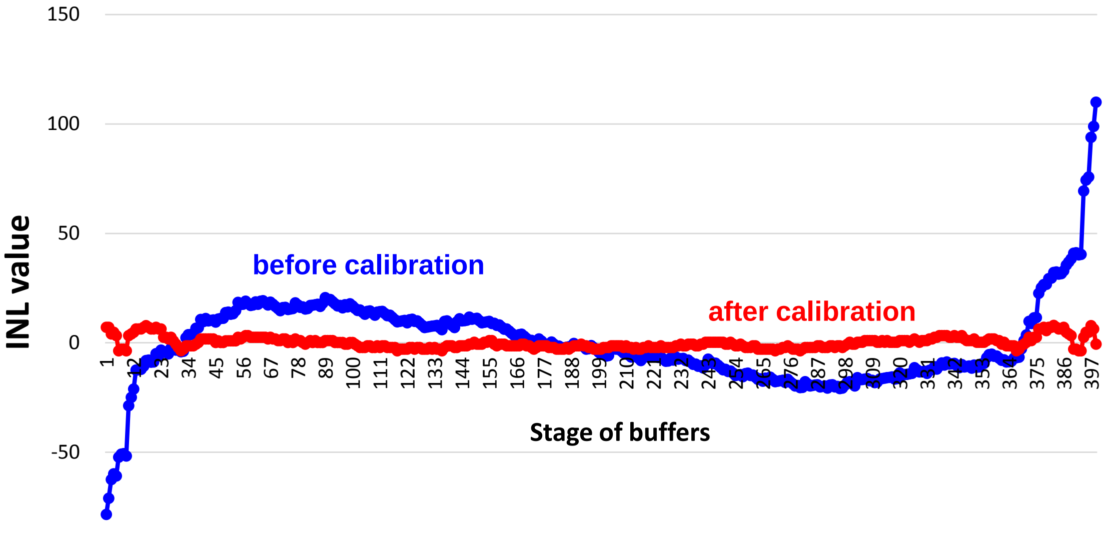


Fig. 4. 15 Measured INL before and after calibration (case #4)

(2) Case #5 (Normal Distribution, Mean = 20ps, Standard Deviation = 2)

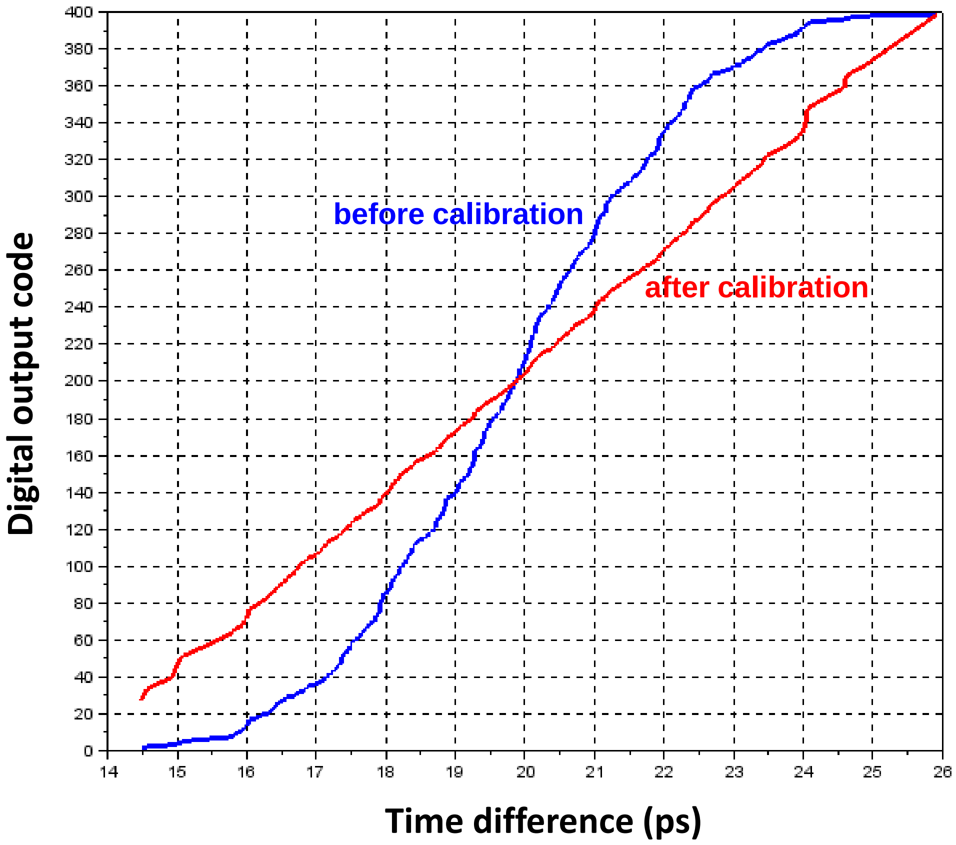


Fig. 4. 16 Input-output characteristics before and after calibration (case #5)

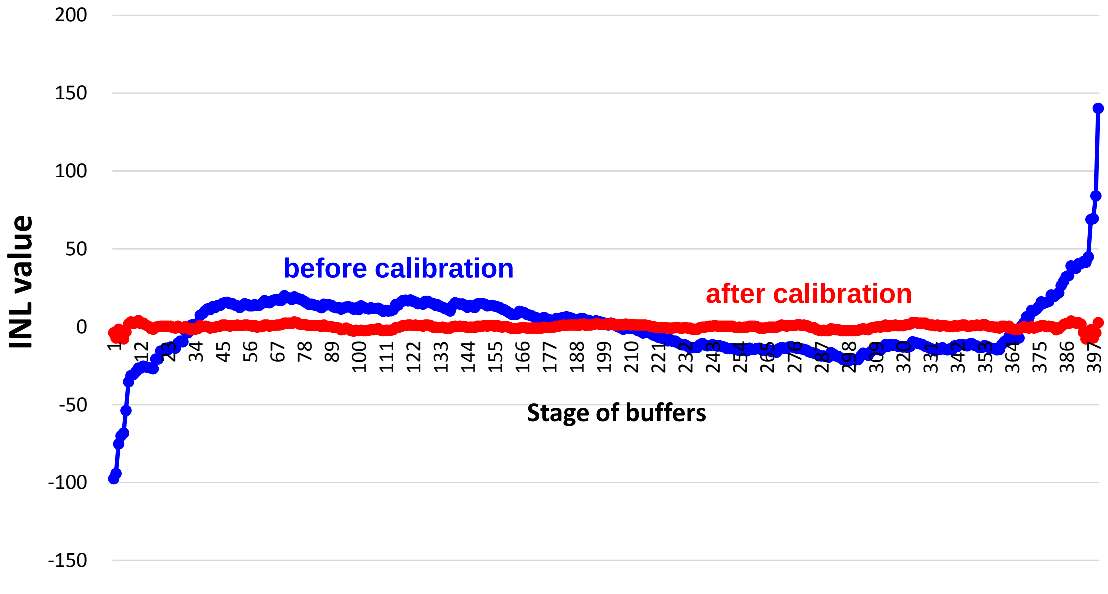


Fig. 4. 17 Measured INL before and after calibration (case #5)

(2) Case #6 (Normal Distribution, Mean = 20ps, Standard Deviation = 3)

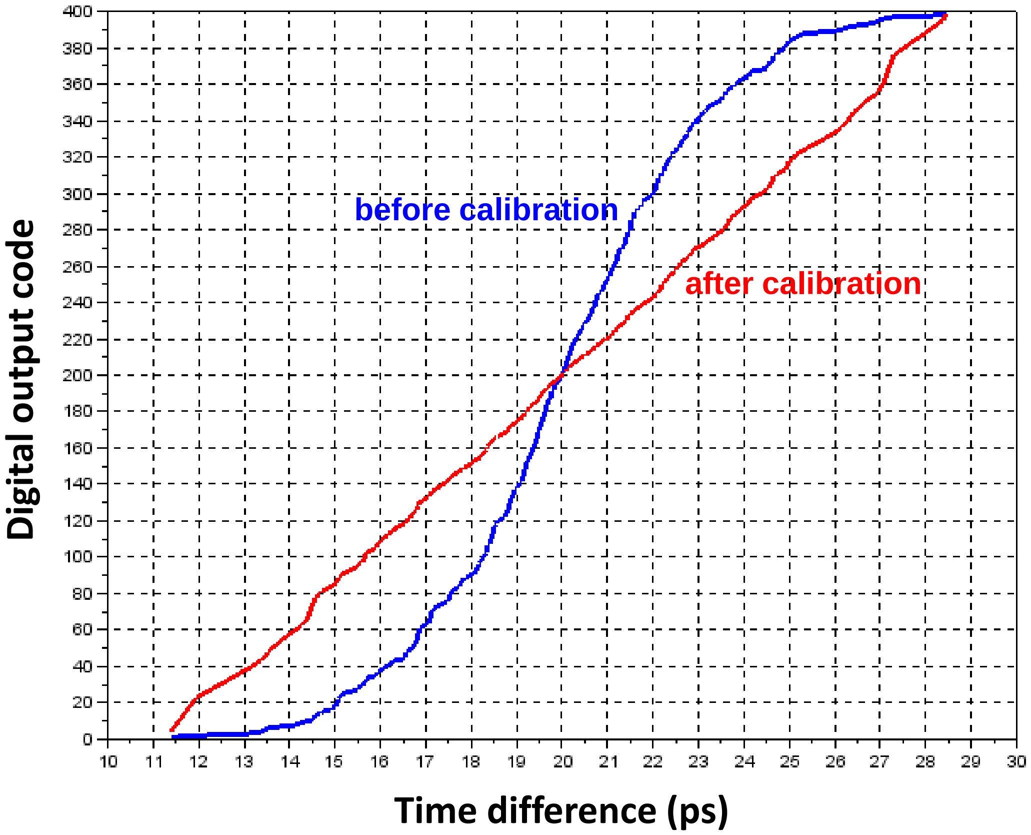


Fig. 4. 18 Input-output characteristics before and after calibration (case #6)

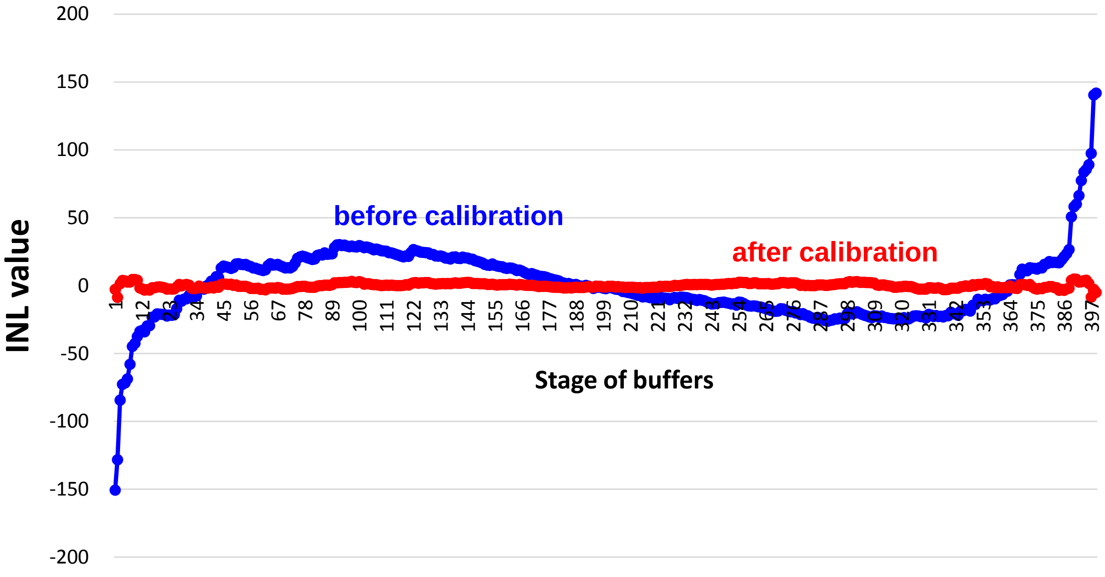


Fig. 4. 19 Measured INL before and after calibration (case #6)

4.4 Stochastic Architecture and Vernier Architecture

Both Vernier-type TDC and stochastic TDC can achieve sub-gate time resolution. But the principles of these two architectures are different.

The fundamental concept of Vernier-type TDC is that the timing resolution is determined by the delay difference of the start and stop line stages, i.e. buffer delay difference, thus it can theoretically achieve an arbitrarily small value. However, this architecture still suffers from the buffer delay mismatches which are caused by device and circuit characteristics variations. Practical limits will arise when increasing resolution are typically due to the time capture elements, process spread and drive/load mismatch. Moreover, its monotonicity is not guaranteed and it may show some non-linearity due to buffer delay mismatches.

The principle of stochastic TDC is quite different from Vernier-type TDC. Though the delay mismatches still exist in stochastic TDC due to manufacturing process variations, they are positively utilized to achieve fine time resolution. The resolution of Vernier-type TDC is limited by the delay mismatches, but this does not happen in stochastic TDC, as the resolution of stochastic TDC is achieved by utilizing the delay mismatches. Moreover, the non-linearity of stochastic TDC can be compensated by self-calibration.

Vernier architecture and stochastic architecture can be combined to realize stochastic Vernier-type TDC, which can achieve an even finer time resolution than conventional Vernier-type TDC. Self-calibration technique can also be applied on stochastic Vernier-type TDC to improve the linearity.

4.5 Summary

In this chapter, stochastic process theory is applied in TDC architecture to positively utilize the stochastic variation, which is caused by the integrated circuit manufacturing process uncertainty, to improve the time resolution to sub-picosecond level.

This stochastic TDC may be highly nonlinear, but its nonlinearity can be compensated by self-calibration method. In calibration mode, the relative variation (ratio) among DFFs can be measured by the histogram engine. After large enough measurement cycles, each bin of histogram is with the probability proportional to the setup and hold time of the corresponding DFF.

In measurement mode, the digital error correction operation is conducted based on the acquired histogram data. The TDC performance before and after self-calibration can be evaluated quantitatively by exploiting a least-mean-square method. In addition, in order to make the self-calibration result accurate, there is a minimum requirement on the number of DFFs.

Corresponding RTL simulation is conducted, and the operation principle of stochastic TDC is verified by the simulation results.

Stochastic TDC with self-calibration can be applied to many physical experiments with sub-picosecond resolution requirement, such as time-of-flight and lifetime measurements in atomic and high energy physics, experiments that involve laser ranging and electronic research involving the testing of integrated circuits and high-speed data transfer.

References

[1] Satoshi Ito, Shigeyuki Nishimura, Haruo Kobayashi, Satoshi Uemori, Yohei Tan, Nobukazu Takai, Takahiro J. Yamaguchi, Kiichi Niitsu, “Stochastic TDC Architecture with Self-Calibration”, IEEE Asia Pacific Conference on Circuits and Systems, Kuala Lumpur, Malaysia (Dec. 2010).

[2] Kentaroh Katoh, Yuta Doi, Satoshi Ito, Haruo Kobayashi, Ensi Li, Nobukazu Takai, Osamu Kobayashi, “An Analysis of Stochastic TDC architecture with Self-Calibration”, 5th International Conference on Advanced Micro-Device Engineering (AMDE2013), Kiryu, Japan (Dec. 19, 2013).

[3] Yuta Doi, Satoshi Ito, Shigeyuki Nishimura, Haruo Kobayashi, Nobukazu Takai, “Vernier Stochastic TDC Architecture with Self-Calibration”, 4th International Conference on Advanced Micro-Device Engineering, Kiryu, Japan (Dec. 7, 2012).

[4] Y. Park, D. D. Wentzloff, “A Cyclic Vernier TDC for ADPLLs Synthesized from a Standard Cell Library”, IEEE Transactions on Circuits and Systems, vol.58, no.7, pp. 1511-1517 (July 2011).

[5] Satoshi Uemori, Masamichi Ishii, Haruo Kobayashi, Yuta Doi, Osamu Kobayashi, Tatsuji Matsuura,Kiichi Niitsu, Fumitaka Abe, Daiki Hirabayashi, “Multi-bit Sigma-Delta TDC Architecture for Digital Signal Timing Measurement”, IEEE International Mixed-Signals, Sensors, and Systems Test Workshop, Taipei, Taiwan (May 2012).

[6] Hisato Takehata, Toshiki Sugimoto, Hiroshi Tanimoto, Shingo Yoshizawa, “FPGA Implementation of Stochastic Flash A-to-D Converter and Its Evaluation”, IEEE International Symposium on Communications and Information Technologies (ISCIT), Oct. 2015.

[7] Toshiki Sugimoto, Hiroshi Tanimoto, Shingo Yoshizawa, “A Stochastic Flash A-to-D Converter with Dynamic Element Matching Technique”, IEEE International Symposium on Communications and Information Technologies (ISCIT), Oct. 2015.

[8] Jianjun Yu, Fa Foster Dai, Richard C. Jaeger, “A 12bit Vernia Ring Time-to-Digital Converter in 0.13μm Technology”, IEEE JSSC, vol. 45, no. 4 (April 2010).

[9] M Zanuso, P Madoglio, S Levantino, C Samori, AL Lacaita, “Time-to-Digital Converter for Frequency Synthesis Based on a Digital Bang-Bang PLL”, IEEE Trans. CAS, (March 2010).

[10] S. Henzler, S. Koeppe, W. Kamp, H. Mulatz, D. Schmitt-Landsiedel, “90nm 4.7ps-Resolution 0.7-LSB Single-Shot Precision and 19pJ-per-Shot Local Passive Interpolation Time-to-Digital Converter with On-Chip Characterization”, ISSCC (Feb. 2008).

[11] K. Blutman, J. Angevare, A. Zjajo, N.P. van der Meijs, “A 0.1 pJ Freeze Vernier time-to-digital converter in 65nm CMOS”, IEEE International Symposium on Circuits and Systems (ISCAS), pp.85-88, June 2014.

[12] P. Lu, A. Liscidini, P. Andreani, “A 3.6 mW, 90 nm CMOS gated vernier time-to-digital converter with an equivalent resolution of 3.2ps”, IEEE Journal of

Solid-State Circuits, vol. 47, no. 7, pp. 1626–1635, 2012.

[13] K. Townsend, A. Macpherson, J. Haslett, “A fine-resolution time-to-digital converter for a 5GS/s ADC”, Proceedings of IEEE International Symposium on Circuits and Systems, 2010, pp. 3024–3027.

[14] A. Liscidini, L. Vercesi, R. Castello, “Time to digital converter based on a 2-dimensions vernier architecture”, Proceedings of IEEE Custom Integrated Circuits Conference, 2009, pp. 45–48.

[15] S. Henzler, S. Koeppe, D. Lorenz, W. Kamp, R. Kuenemund, D. Schmitt-Landsiedel, “A local passive time interpolation concept for variation-tolerant high-resolution time-to-digital conversion”, IEEE Journal of Solid-State Circuits, vol. 43, no. 7, pp. 1666–1676, 2008.

[16] M. Elsayed, V. Dhanasekaran, M. Gambhir, J. Silva-Martinez, E. Sanchez-Sinencio, “A 0.8ps DNL time-to-digital converter with 250MHz event rate in 65nm CMOS for time-mode-based ΣΔ modulator”, IEEE Journal of Solid-State Circuits, vol. 46, no. 9, pp. 2084–2098, Sept 2011.

[17] K. Kim, Y. Kim, W. Yu, and S. Cho, “A 7b, 3.75ps resolution two step time-to-digital converter in 65nm CMOS using pulse-train time amplifier”, Proceedings of IEEE International Symposium on VLSI Circuits, June 2012, pp. 192–193.

Chapter 5
CONCLUSION

5.1 Conclusion

In this study, mathematical methods including number theory, coding theory and stochastic process theory are applied in the design of TDC architecture.

Though conventional flash-type TDC has the advantage of being able to measure a single-event input, however, it has a disadvantage on circuit complexity, which leading to large power consumption and chip area. In number theory, residue number system represents a large integer using a set of smaller integers, so that computation may be performed more efficiently. In the TDC circuit, the signal is “time” instead of “voltage”, and the residue can be easily obtained with a ring oscillator. So I applied the residue number system concept on TDC circuit design. By utilizing residue number system, a large Flash-type TDC can be converted into a set of smaller Flash-type TDC performed independently and in parallel, in order to reduce power consumption and chip area significantly. RTL simulation and FPGA implementation are conducted to verify this idea. The verification results show that the proposed residue number system based TDC works with good linearity as expected.

However, this residue number system based TDC is easily to be affected by the mismatches among the delay stages. Glitches occurs in the time-domain when there are mismatches among the delay stages, which triggers an instability in the output digital codes. Due to this, a parallel ring oscillator TDC architecture based on Gray code is proposed. For the Gray code, the uncertainty during a transition is only one count, unlike with the binary code used in residue number system, where the uncertainty could be multiple counts. So the Gray code provides data with the least uncertainty. This feature prevents certain data errors which can occur with natural binary code during state changes. So I applied Gray code concept on TDC circuit design as an improvement of residue number system based TDC. As a ring oscillator is a natural time-domain Gray bit code generator, a Gray code TDC architecture can be easily conceived by grouping a few ring oscillators. The proposed Gray code TDC has a unique characteristic where only one output of the DFFs changes state with each clock pulse. RTL simulation and FPGA implementation are conducted to verify this idea. The verification results show that the proposed Gray code TDC can provide a glitch-free binary code sequence, i.e. no out-of-sequence code, even there are some amounts of mismatches among the delay stages.

Parasitic capacitance, in electrical circuits, is the extra effect of conductors that serve as plates between a dielectric, which is usually air. It becomes a problem with higher frequencies because the very small distributed capacitances that exist will have lower impedances at these frequencies. Taking this issue into consideration, I applied cyclic code to generate the lower bits of Gray code, in order to reduce the frequency of the outputs, i.e. to reduce the parasitic capacitance among circuit elements. FPGA implementation and measurement results verify the operation principle.

In manufacturing process, device and interconnect parameters such as channel length, wire parasitics, etc. can vary across the sample space of the manufactured chips. Under such conditions, circuit performance characteristics like the voltage, delay, slew and power are also stochastic processes. I applied stochastic process theory on TDC circuit design, in order to positively utilize the stochastic process variation in circuit characteristics to obtain effective fine time resolution. Based on conventional flash-type TDC, we connect each delay buffer output to the data inputs of several DFFs. Since the setup and hold times of the DFFs are not identical due to process variations, the edge timing which changes DFF output from 0 to 1 can be different among these DFFs. Then their statistical variation becomes the effective time resolution of the TDC, which is much finer than the order of magnitude of the buffer delay. This stochastic TDC may be highly nonlinear, but its nonlinearity can be compensated by the above self-calibration method. RTL simulation is conducted to verify this idea. The verification results show that the proposed stochastic TDC with self-calibration is practical for realizing a linear TDC with sub-picosecond time-resolution.

The proposed residue number system based TDC, Gray code TDC, Gray code TDC with cyclic code, and stochastic TDC architectures combine mathematical elegance and TDC circuit design to an unusual degree, and show the beauty of this particular combination of mathematics and engineering.

Furthermore, we make fully digital FPGA implementation (design, simulation, verification, and testing) of these TDCs. The design work uses only RTL (without SPICE) simulation and FPGA (instead of full custom CMOS) implementation, which would be suitable for mixed-signal SoC design in nano-CMOS era.

5.2 Future Work

Stochastic process theory can also be combined with Vernier-type TDC, which can achieve an even finer time resolution than current stochastic TDC. In the future, stochastic Vernier-type TDC with self-calibration feature will be highly considered.

Publications

Journal Papers

[1] **Congbing Li**, Haruo Kobayashi, "A Glitch-Free Time-to-Digital Converter Architecture Based on Gray Code", 電気学会論文誌 (和文誌 C) 「電子回路関連技術」特集, vol.136 , no.1 (2016 年 1 月).

[2] **Congbing Li**, Haruo Kobayashi, "A Residue Number System Based Time-to-Digital Converter Architecture and its FPGA Implementation", Advanced Micro-Device Engineering VI, Key Engineering Materials (2016).

[3] Masataka Kamiyama, Daiki Oki, Satoru Kawauchi, **Congbing Li**, Nobuo Takahashi, Seiichi Banba, Toru Dan, Haruo Kobayashi, "Triple-Band CMOS Low Noise Amplifier Design Utilizing Transformer Couplings", Advanced Micro-Device Engineering VI, Key Engineering Materials (2016).

[4] Takeshi Chujo, Junshan Wang, Daiki Hirabayashi, **Congbing Li**, Yutaro Kobayashi, Kentaroh Katoh,Haruo Kobayashi, Masanobu Tsuji, Koshi Sato, "FPGA Evaluation of Flash-type TDC With Histogram Methodfor Linearity Self-Calibration", Advanced Micro-Device Engineering VI, Key Engineering Materials (2016).

[5] Daiki Oki, Satoru Kawauchi, **Li CongBing**, Masataka Kamiyama, Seiichi Banba, Toru Dan, Nobuo Takahashi, Haruo Kobayashi, "A Power-Efficient Noise Canceling Technique Using Signal-Suppression Feed-forward for Wideband LNAs", pp.109-116, Advanced Micro-Device Engineering V, Key Engineering Materials (2015).

[6] Kentaroh Katoh, Yutaro Kobayashi, Takeshi Chujyo, Junshan Wang, Ensi Li, **Congbing Li**, Haruo Kobayashi, "A Small Chip Area Stochastic Calibration for TDC Using Ring Oscillator", Journal of Electronic Testing: Theory and Applications, vol.30, issue 6, pp.653-663, Springer (Dec. 2014).

[7] Haruo Kobayashi, Hitoshi Aoki, Kentaroh Katoh, **Congbing Li**, "Analog/Mixed-Signal Circuit Design in Nano CMOS Era", IEICE Electronics

Express, vol.11 no.3, pp.1-15 (2014).

International Conference Papers

[1] **Congbing Li**, Haruo Kobayashi, “A Gray Code Based Time-to-Digital Converter Architecture and its FPGA Implementation”, IEEE International Symposium on Radio-Frequency Integration Technology (RFIT), Sendai, Japan (Aug. 26-28, 2015).

[2] **Congbing Li**, Kentaroh Katoh, Haruo Kobayashi, Junshan Wang, Shu Wu, Shaiful Nizam Mohyar, “Time-to-Digital Converter Architecture with Residue Arithmetic and its FPGA Implementation”, 11th International SoC Design Conference, Jeju, Korea (Nov. 2-6, 2014).

[3] Takeshi Chujo, Daiki Hirabayashi, Kentaroh Kentaroh, **Congbing Li**, Yutaro Kobayashi, Junshan Wang, Koshi Sato, Haruo Kobayashi, “Experimental Verification of Timing Measurement Circuit With Self-Calibration”, IEEE International Mixed-Signals, Sensors and Systems Test Workshop (IMS3TW'14), Porto Alegre, Brazil (Sept. 17-19, 2014).

[4] Takeshi Chujo, Daiki Hirabayashi, Kentaroh Katoh, **Congbing Li**, Yutaro Kobayashi, Junshan Wang, Koshi Sato, Haruo Kobayashi, “FPGA Evaluation of Flash-type TDC With Histogram Method Self-Calibration”, The 3rd Solid State Systems Symposium-VLSIs and Semiconductor Related Technologies & The 17th International Conference on Analog VLSI Circuits, Ho Chi Minh City, Vietnam (Oct. 22-24, 2014).

[5] Junshan Wang, Kentaroh Katoh, **Congbing Li**, Ensi Li, Yutaro Kobayashi, Takeshi Chujo, Daiki Hirabayashi, Haruo Kobayashi, “Digital FPGA Implementation of TDC With Self-Calibration”, The 3rd Solid State Systems Symposium-VLSIs and Semiconductor Related Technologies & The 17th International Conference on Analog VLSI Circuits, Ho Chi Minh City, Vietnam (Oct. 22-24, 2014).

[6] Masataka Kamiyama, Oki Daiki, Kawauchi Satoru, Banba Seiichi, Takahashi Nobuo, Dan Toru, **Congbing Li**, Haruo Kobayashi, “Multi-Band CMOS Low Noise Amplifiers Utilizing Transformers”, The 3rd Solid State Systems Symposium-VLSIs and Semiconductor Related Technologies & The 17th

International Conference on Analog VLSI Circuits, Ho Chi Minh City, Vietnam (Oct. 22-24, 2014).

[7] Satoru Kawauchi, Daiki Oki, **Congbing Li**, Masataka Kamiyama, Seiichi Banba, Toru Dan, Nobuo Takahashi, Koji Sakata, Haruo Kobayashi, Nobukazu Takai, “A Power-Efficient Noise Canceling Technique Using Signal-Suppression Feed-forward for Wideband LNAs”, The 4th IEICE International Conference on Integrated Circuits Design and Verification, Ho Chi Minh City, Vietnam (Nov. 15-16, 2013).

[8] John Kenney, Saurabh Barve , Vinuth Rai, Yuichi Kanai, **Congbing Lee**, Masashi Horri, "Comparing Communication Performance of DSRC OBEs from Multiple Suppliers", ITS World Congress, Vienna Austria (Oct. 2012).

Domestic Conferences / Seminars

[1] **李 从兵**、王 俊善、小林 春夫　「確率的時間デジタイザとその自己校正、RTL 検証の検討」　電子回路研究会　電子回路一般、東京都市大学（2016 年 3 月 7 日－8 日）

[2] **李 从兵**、小林春夫　「符号理論アプローチを用いた時間デジタル変換回路アーキテクチャ」　第 61 回システム LSI 合同ゼミ、東工大 大岡山キャンパス（2015 年 10 月 17 日）

[3] **李　从兵**、小林春夫　「符号理論アプローチを用いた時間デジタル変換回路アーキテクチャ」　電気学会　電子回路研究会　ECT-15-064　横須賀（2015 年 7 月 3 日）

[4] **李 从兵**、加藤健太郎、王俊善、小林春夫　「剰余系を用いたタイミング測定用回路の検討」　第 71 回　FTC 研究会　東京（2014 年 7 月 17 日－19 日）

[5] **李从兵**、加藤健太郎、小林春夫　「剰余系を用いた TDC 回路の FPGA 実現の検討」　第 4 回 電気学会 東京支部 栃木・群馬支所 合同研究発表会（2014 年 3 月 3 日－4 日）

[6] 興　大樹、河内　智、**李从兵**、神山雅貴、馬場清一、壇　徹、高橋伸夫、小林春夫 「信号抑制フィードフォワードを用いた広帯域 LNA の低消費電力ノイズキャンセル技術」 電気学会　電子回路研究会　ECT-15-008　高

知（2015 年 1 月 22 日）

[7] Junshan Wang, Kentaroh Katoh, **Congbing Li**, Ensi Li, Yutaro Kobayashi, Takeshi Chujo, Daiki Hirabayashi, Haruo Kobayashi, “Digital FPGA Implementation of TDC With Self –Calibration”, 1st International Symposium of Gunma University Medical Innovation and 6th International Conference on Advanced Micro-Device Engineering, Dec. 5, 2014 Kiryu City Performing Art Center.

[8] Masataka Kamiyama, Daiki Oki, Satoru Kawauchi, **Congbing Li**, Nobuo Takahashi, Seiichi Banba, Toru Dan, Haruo Kobayashi, “Triple-Band CMOS Low Noise Amplifier Design Utilizing Transformers”, 1st International Symposium of Gunma University Medical Innovation and 6th International Conference on Advanced Micro-Device Engineering, Dec. 5, 2014 Kiryu City Performing Art Center.

[9] 王俊善、加藤健太郎、**李从兵**、李恩思、小林佑太朗、中條剛志、平林大樹、小林春夫　「タイミング測定用回路のデジタル自己校正とＦＰＧＡ実現」　第 71 回　FTC 研究会　東京（2014 年 7 月 17 日－19 日）

[10] 中條剛志、平林大樹、加藤健太郎、**李从兵**、李恩思、小林佑太朗、王俊善、佐藤幸志、小林春夫　「フラッシュ型タイムデジタイザ回路の線形性自己校正の実験検証」　第 4 回 電気学会 東京支部 栃木・群馬支所 合同研究発表会（2014 年 3 月 3 日－4 日）

[11] 王俊善、加藤健太郎、**李从兵**、李恩思、小林佑太朗、中條剛志、平林大樹、小林春夫　「時間ディジタイザ回路の自己校正技術の研究」　第 4 回 電気学会 東京支部 栃木・群馬支所 合同研究発表会（2014 年 3 月 3 日－4 日）

[12] 加藤健太郎、**李从兵**、李恩思、王俊善、小林佑太朗、小林春夫　「時間ディジタイザのストカスティックキャリブレーションのシミュレーション解析」　電気学会　電子回路研究会　ECT-14-005　金沢（2014 年 1 月 23 日）

[13] 中條剛志、平林大樹、加藤健太郎、**李从兵**、李恩思、小林佑太朗、王俊善、佐藤幸志、小林春夫　「フラッシュ型タイムデジタイザ回路のヒストグラム法による自己校正の実験検証」　電気学会　電子回路研究会 ECT-14-006 金沢（2014 年 1 月 23 日）

[14] Daiki Oki, Satoru Kawauchi, **Li CongBing**, Masataka Kamiyama, Seiichi Banba, Toru Dan, Nobuo Takahashi, Koji Sakata, Haruo Kobayashi, Nobukazu Takai, “A

Power-Efficient Noise Canceling Technique Using Signal-Suppression Feed-forward for Wideband LNAs", 5th International Conference on Advanced Micro-Device Engineering (AMDE2013) Kiryu, Japan (Dec. 19, 2013).

[15] Kentaroh Katoh, Ensi Li, Junshan Wang, **Congbing Li**, Haruo Kobayashi, "A Reduction Technique of Volume of Input Sequences for Time-Multiplexed Delay Measurement Using Embedded Delay Measurement Circuit", 5th International Conference on Advanced Micro-Device Engineering (AMDE2013) Kiryu, Japan (Dec. 19, 2013).

[16] Kentaroh Katoh, Ensi Li, Junshan Wang, **Congbing Li**, Haruo Kobayashi, "An On-Chip Delay Measurement Using Adjacency Testable Scan Design", 5th International Conference on Advanced Micro-Device Engineering (AMDE2013) Kiryu, Japan (Dec. 19, 2013).

[17] 輿大樹、河内智、**Li CongBing**、神山雅貴、馬場清一、壇徹、高橋伸夫、坂田浩司、小林春夫、高井伸和　「Signal-Suppression Feed-forward を用いた広帯域 LNA の低消費電力ノイズキャンセル技術」 第 33 回 シリコンアナログ RF 研究会、湯河原（2013 年 8 月 28 日）

[18] 樋口啓介、永井真、**Li Congbing**、中岡謙　「車車・路車共用通信のための TDMA 型アクセス制御方式に関する検討」 電子情報通信学会大会講演 論文集, 頁:149, 基礎・境界ソサイエティ.(2010 年)

[19] 中岡 謙、永井真琴、樋口啓介、**Congbing Li**「車車間・路車間共用通信のための時分割アクセス制御方式」 電子情報通信学会大会 講演論文集, 頁:S.25-S.26, 基礎・境界ソサイエティ (2009 年)

Patents

[1] He Zhibin, **Li Congbing**, Guo Gengsheng, Wang Min, "Method of Implementing Source Control Rate of AMR Speech Sound Coding", application #: CN200510086745, publish date: March 29, 2006.

Appendix

Table I The setup and hold times of each DFF (case #2)

(Normal distribution, mean = 20ps, standard deviation = 6, unit: ps)

Column Row	1	2	3	4	5	6	7	8
1	26.15	20.75	19.54	28.32	19.90	30.13	16.94	17.80
2	18.22	26.65	19.29	20.02	18.48	17.06	11.26	24.26
3	14.22	19.82	19.22	28.80	9.10	17.45	25.49	26.92
4	33.77	21.91	16.14	17.15	27.72	17.54	11.74	16.84
5	10.40	26.86	22.46	17.30	23.80	31.17	21.21	15.11
6	21.53	22.79	31.33	24.69	16.28	18.84	16.33	29.24
7	21.51	6.31	12.30	13.44	25.70	31.13	20.74	23.97
8	20.38	31.76	21.17	25.44	16.96	22.55	15.76	23.65
9	21.07	21.49	13.76	23.98	12.00	24.23	21.00	16.12
10	9.09	32.29	11.99	29.50	37.05	24.42	25.40	17.65
11	22.94	9.76	24.43	16.11	30.00	20.24	17.19	21.01
12	27.73	20.32	18.58	24.75	20.03	17.66	22.15	22.85
13	12.79	20.09	26.66	22.50	21.29	30.05	23.05	16.59
14	23.45	15.92	22.36	25.43	24.12	23.77	17.85	21.95
15	16.46	24.46	15.05	14.99	23.34	17.54	15.80	23.48
16	17.59	23.81	22.82	14.82	17.84	25.07	23.21	15.04
17	17.71	32.80	18.68	17.33	15.35	13.02	19.47	21.51
18	13.65	17.50	11.30	13.62	27.68	25.11	25.24	16.91
19	25.14	19.80	18.71	21.56	21.17	26.85	31.19	27.43
20	16.99	17.43	21.93	39.64	13.09	12.12	15.89	12.40
21	13.86	11.48	30.54	14.41	18.11	23.24	27.10	28.45
22	22.11	20.48	25.67	10.04	22.09	19.62	24.48	29.95
23	24.30	25.09	14.88	11.50	21.59	19.08	18.52	25.96
24	15.89	29.81	26.38	28.60	17.57	24.75	34.49	20.52

25	13.06	10.73	31.80	21.56	22.58	12.58	25.68	25.88
26	32.39	21.71	8.53	12.54	17.57	11.73	7.39	20.75
27	25.72	11.89	14.68	25.37	16.44	16.54	16.65	25.16
28	7.51	20.75	13.04	8.49	25.61	11.79	19.67	7.71
29	5.08	30.53	4.26	19.58	27.36	14.28	19.77	23.26
30	12.79	25.42	21.83	18.24	27.58	30.37	19.09	20.59
31	21.22	12.72	6.57	26.21	9.56	16.47	7.88	9.93
32	18.49	16.95	17.77	19.64	10.91	19.33	19.65	15.46
33	21.42	22.58	26.69	30.43	19.11	19.99	19.43	27.54
34	18.03	11.63	20.49	21.97	23.40	30.20	23.39	24.53
35	17.69	22.59	21.62	16.30	17.33	19.59	20.20	25.82
36	24.32	15.91	20.97	25.95	6.99	12.64	17.50	15.99
37	24.74	24.27	18.61	15.95	6.41	21.66	17.20	17.59
38	11.62	24.63	11.11	18.35	12.64	13.43	23.14	13.40
39	15.24	14.18	20.57	14.71	23.28	30.92	15.59	13.15
40	31.23	15.18	23.98	15.09	27.22	20.22	19.80	26.92
41	19.09	25.60	14.57	27.30	20.14	20.61	16.21	13.34
42	23.86	19.06	16.59	19.15	19.80	23.64	10.26	20.16
43	24.93	30.86	20.19	24.18	23.78	26.11	22.77	19.61
44	32.92	20.73	19.07	19.96	24.80	16.57	14.77	29.92
45	16.32	20.17	28.44	21.63	26.20	20.40	5.90	25.54
46	19.93	13.98	29.15	13.44	24.71	24.22	18.84	25.53
47	18.75	25.84	27.22	24.57	11.92	16.07	19.84	23.39
48	21.20	25.99	20.63	27.47	26.09	18.96	15.60	24.55
49	16.51	22.87	13.71	15.96	26.23	25.23	14.25	17.94
50	26.89	19.82	18.99	15.80	10.11	19.85	29.42	11.77

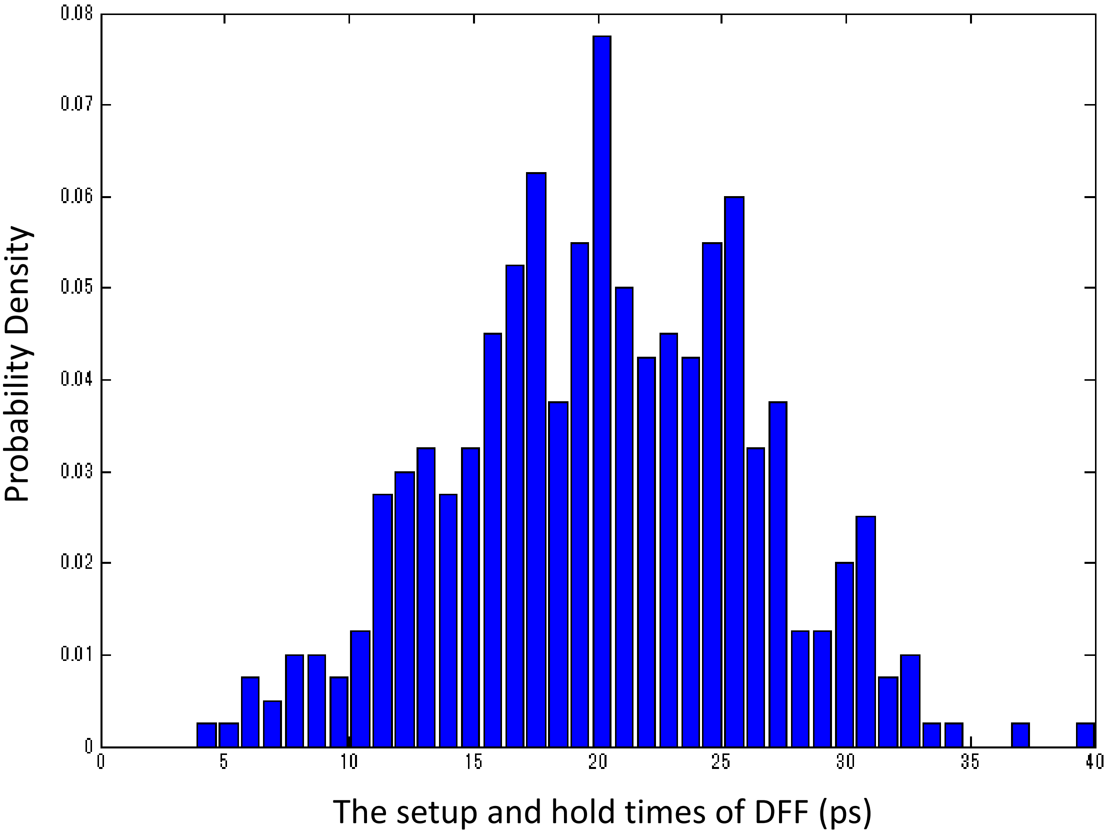


Fig. I The probability density function of DFFs' setup and hold times (case #2)

Table II The setup and hold times of each DFF (case #3)

(Normal distribution, mean = 20ps, standard deviation = 6, unit: ps)

Column Row	1	2	3	4	5	6	7	8
1	10.25	18.99	20.19	3.94	12.33	30.01	19.62	22.08
2	29.60	16.89	12.21	24.82	20.48	21.63	17.57	13.10
3	15.40	18.66	33.78	18.15	14.50	15.33	10.98	21.02
4	11.86	22.87	21.84	22.31	21.14	17.85	25.88	21.66
5	21.27	19.50	7.99	26.04	17.53	19.54	15.85	33.98
6	20.93	21.34	18.55	26.89	12.01	24.36	20.53	31.54
7	24.03	26.85	21.91	21.75	17.15	19.15	22.31	16.45
8	22.87	23.17	19.55	27.96	31.20	19.40	2.87	16.63
9	23.05	20.61	34.18	20.86	12.56	20.28	11.93	23.58
10	16.88	17.14	17.72	17.56	29.45	14.89	32.87	12.86
11	18.68	29.08	21.54	18.93	10.01	26.58	19.71	22.80
12	29.88	22.77	26.75	19.58	12.68	27.14	15.65	31.60
13	15.58	24.16	21.07	20.83	13.35	24.81	18.51	27.62
14	19.23	24.66	21.12	24.44	14.09	19.04	28.10	17.90
15	13.69	28.11	28.59	19.30	14.18	24.32	20.47	23.52
16	19.79	9.92	9.85	21.52	26.29	8.22	13.49	24.05
17	11.80	9.36	27.21	22.73	22.02	17.03	15.74	12.28
18	28.57	19.43	13.49	24.31	21.97	31.03	25.34	16.65
19	16.59	18.52	14.15	17.99	20.15	26.06	25.43	25.55
20	27.28	25.91	14.16	9.14	21.07	28.68	11.31	21.45
21	23.77	17.41	12.86	17.71	28.14	24.75	20.71	31.03
22	20.41	30.58	15.39	25.26	23.04	14.33	14.25	23.26
23	31.88	13.80	13.28	17.55	36.79	23.71	8.97	14.57
24	13.45	18.29	21.28	19.31	28.85	29.10	9.25	20.69
25	25.90	19.26	20.68	12.73	17.91	19.13	7.67	14.11
26	21.97	20.71	23.52	19.30	31.19	12.70	19.26	27.44

27	17.25	23.82	32.83	24.32	19.18	22.28	9.79	19.61
28	22.51	22.39	21.55	13.33	18.97	21.95	22.35	19.34
29	13.92	8.33	14.66	8.80	7.00	24.10	26.99	24.13
30	19.81	27.78	16.86	22.63	11.87	17.27	26.66	20.64
31	19.87	20.77	21.96	18.88	7.27	18.10	12.90	23.11
32	25.31	20.09	25.29	17.54	28.38	19.91	30.69	25.77
33	18.14	25.38	17.65	21.26	11.59	19.55	18.55	22.22
34	14.51	14.28	25.49	16.18	23.88	27.85	21.84	8.91
35	23.08	15.42	30.19	11.32	19.27	18.51	14.72	29.96
36	23.47	18.54	19.42	19.63	23.82	17.31	12.75	18.44
37	23.21	25.72	18.83	29.42	22.43	23.30	26.97	28.83
38	16.09	21.00	17.38	18.66	16.71	9.58	21.15	29.50
39	25.38	11.92	16.70	15.65	18.64	24.28	26.67	25.14
40	29.62	31.92	14.28	18.69	20.74	24.85	12.97	17.76
41	13.77	18.82	24.38	12.74	26.78	17.03	31.03	18.21
42	24.26	19.63	17.37	32.84	22.75	20.59	16.34	22.28
43	9.46	25.42	16.84	18.99	22.57	24.27	14.01	16.05
44	15.39	13.38	22.01	18.72	32.63	23.15	10.11	5.10
45	28.91	20.76	21.34	19.35	14.25	28.59	28.53	22.26
46	17.13	27.18	18.76	22.35	35.64	24.61	17.75	28.54
47	14.04	22.69	22.43	14.26	11.07	23.96	8.31	13.20
48	20.68	22.08	20.68	13.80	24.51	12.68	24.30	18.21
49	19.59	6.06	18.57	8.72	22.55	27.42	23.46	19.39
50	20.05	24.53	18.87	23.76	26.46	18.40	20.47	23.09

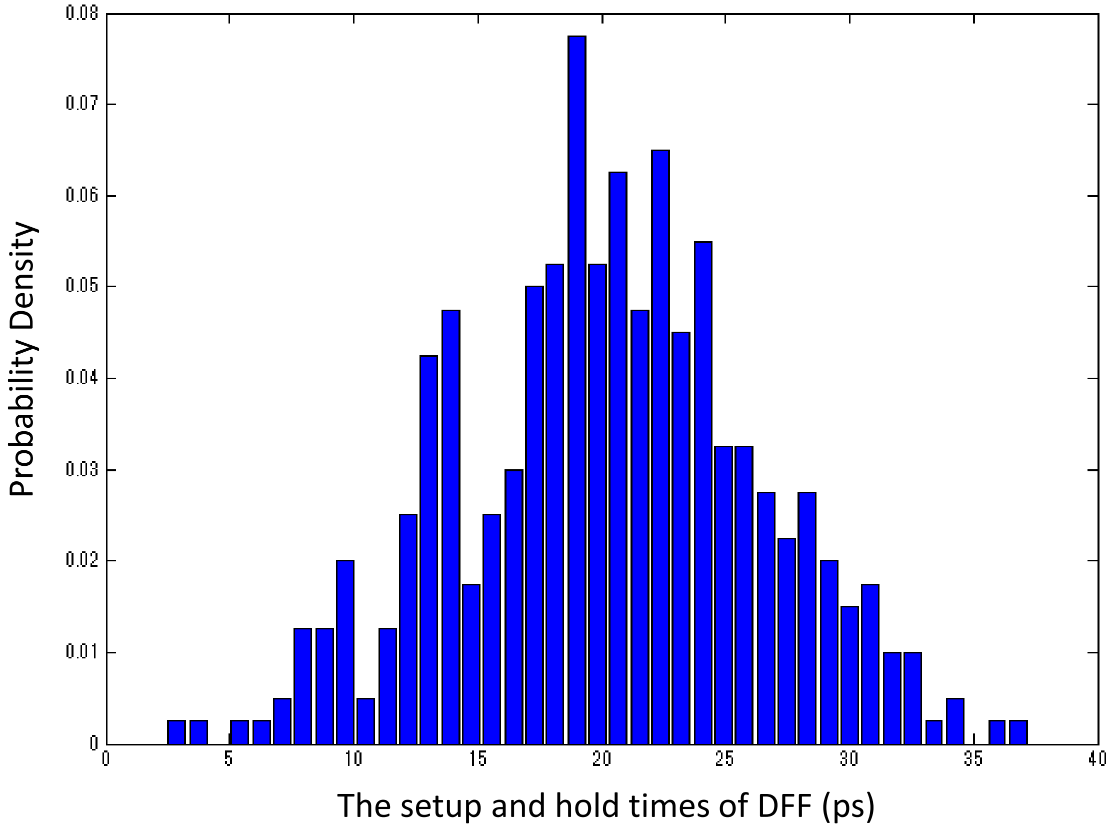


Fig. II The probability density function of DFFs' setup and hold times (case #3)

Table III The setup and hold times of each DFF (case #4)

(Normal distribution, mean = 20ps, standard deviation = 1, unit: ps)

Column Row	1	2	3	4	5	6	7	8
1	20.52	18.76	19.27	21.29	21.95	21.36	21.67	20.97
2	20.26	20.40	17.70	19.15	20.08	18.80	21.12	21.57
3	19.27	19.19	21.11	21.27	19.58	20.02	19.85	18.43
4	18.96	20.12	21.94	19.87	19.77	17.99	20.16	20.79
5	20.13	18.64	19.26	20.15	19.50	21.06	19.97	19.32
6	18.82	19.14	19.74	19.72	21.28	19.79	21.18	20.98
7	20.24	18.51	20.99	21.32	20.05	21.76	19.33	19.69
8	20.06	20.43	20.84	21.87	21.16	22.49	20.78	20.33
9	21.08	21.35	19.74	19.47	18.80	19.41	20.64	21.04
10	19.04	20.19	21.84	20.45	20.74	20.81	20.64	21.25
11	17.98	20.63	18.83	19.78	20.08	20.91	20.32	22.26
12	19.04	19.86	20.55	21.21	19.79	18.56	19.75	21.56
13	20.86	19.58	19.73	19.43	18.88	20.05	21.02	20.25
14	19.01	20.55	19.64	19.03	20.32	19.32	19.38	18.98
15	19.92	17.41	19.41	19.40	18.19	20.90	19.09	19.97
16	19.14	18.95	19.69	19.42	20.16	19.52	18.39	20.38
17	20.90	20.19	19.35	19.99	20.05	21.91	19.89	19.06
18	19.06	19.28	20.69	18.57	20.62	21.81	19.97	19.83
19	20.39	20.85	21.08	19.30	20.36	20.25	21.81	20.45
20	18.36	18.78	20.60	19.94	19.48	18.23	20.99	19.88
21	20.51	19.06	21.73	18.48	20.46	20.08	19.57	19.98
22	20.92	18.52	19.01	18.44	20.50	19.98	19.46	20.10
23	20.74	20.43	19.82	20.40	21.95	19.94	21.30	18.88
24	18.84	19.44	19.93	19.40	19.36	20.21	21.82	19.72
25	19.97	20.56	19.27	19.33	21.14	19.49	20.65	20.28
26	19.75	19.41	19.16	19.21	19.70	19.78	19.36	17.96

27	20.76	19.20	20.32	19.44	20.77	18.72	18.68	18.37
28	20.21	19.78	20.33	21.16	20.50	19.06	19.51	20.34
29	20.07	20.23	20.31	20.11	19.74	21.29	19.32	18.42
30	20.39	22.44	21.53	20.41	21.31	18.76	20.97	21.04
31	21.89	20.22	19.47	20.40	20.36	19.04	20.17	22.28
32	20.34	20.94	18.27	20.45	18.55	20.48	20.24	20.83
33	19.53	19.68	21.12	21.34	20.50	18.66	19.28	18.99
34	18.56	22.21	17.88	20.47	18.98	20.77	20.31	19.73
35	20.51	18.35	20.70	21.22	19.97	21.30	21.72	21.14
36	21.01	20.15	20.16	18.79	20.31	19.53	20.87	21.96
37	19.33	19.85	19.71	20.55	21.47	19.48	19.78	19.75
38	19.15	18.52	19.75	19.23	20.86	20.04	18.67	20.58
39	19.30	17.77	19.86	19.58	19.82	19.69	20.24	19.82
40	21.10	21.28	19.08	22.59	17.88	19.90	18.95	18.48
41	19.26	19.53	18.98	20.97	20.06	20.70	19.39	19.14
42	21.12	18.71	20.50	20.60	21.53	21.18	20.15	20.70
43	18.57	19.61	18.87	19.27	20.34	20.32	20.04	20.59
44	19.95	20.93	19.40	19.92	19.32	19.67	20.58	20.78
45	18.91	21.40	19.06	19.36	20.26	19.65	21.10	21.15
46	19.48	18.51	21.80	20.82	18.78	17.85	19.18	19.94
47	21.77	20.23	19.67	19.63	19.41	20.06	19.09	18.43
48	20.67	21.44	19.58	19.85	19.75	18.58	18.89	21.29
49	20.63	19.11	20.35	19.21	17.99	21.19	20.41	19.48
50	19.82	20.75	19.80	20.48	19.10	20.57	21.70	19.63

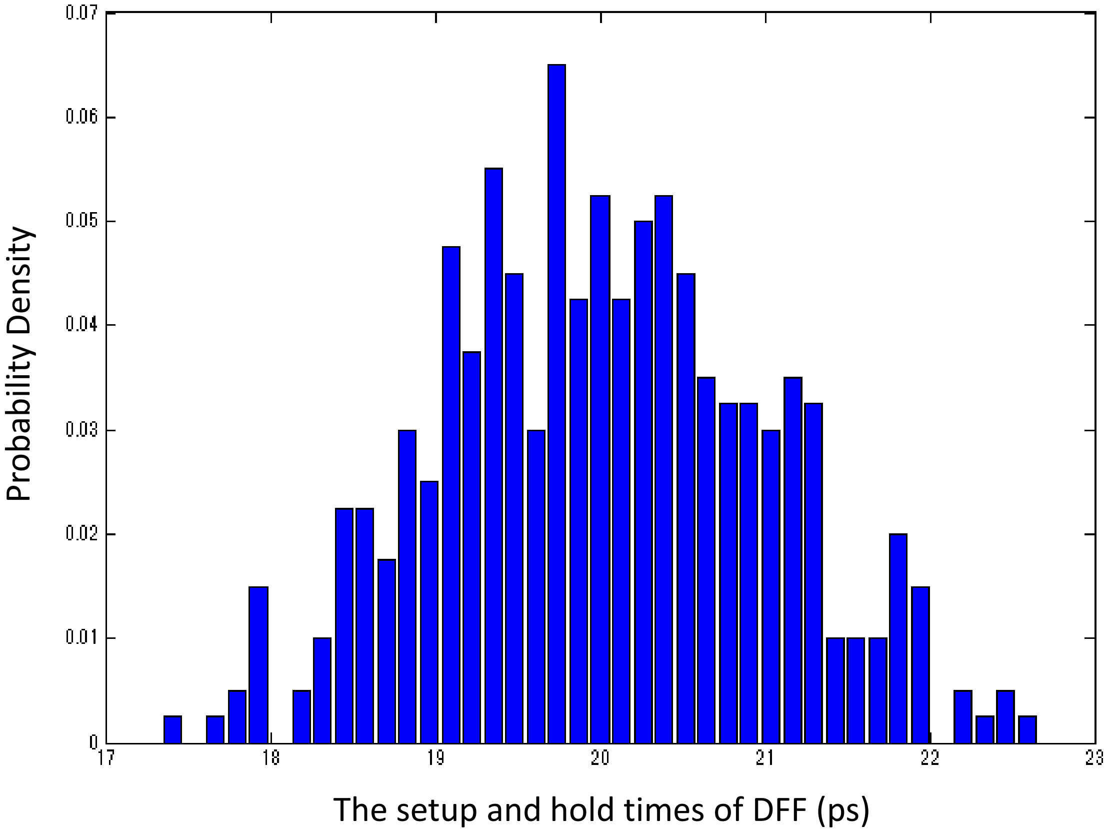


Fig. III The probability density function of DFFs' setup and hold times (case #4)

Table IV The setup and hold times of each DFF (case #5)

(Normal distribution, mean = 20ps, standard deviation = 2, unit: ps)

Column Row	1	2	3	4	5	6	7	8
1	17.97	17.35	19.92	20.11	17.30	21.55	15.36	15.88
2	20.44	16.42	20.64	23.28	19.19	20.89	20.11	16.00
3	18.10	17.59	16.52	18.87	22.42	15.80	24.62	23.68
4	14.56	23.43	20.06	23.91	19.02	22.29	19.78	19.04
5	21.93	21.94	17.20	19.50	21.98	21.60	19.98	18.72
6	19.77	18.13	17.90	18.30	22.86	19.62	18.80	21.26
7	17.92	21.17	20.30	19.28	17.38	23.21	20.99	18.38
8	19.08	17.04	18.08	16.01	17.24	18.74	19.85	21.03
9	22.23	24.59	23.28	17.38	18.42	17.47	17.47	17.53
10	21.16	17.85	22.15	20.87	17.26	21.92	16.04	23.12
11	21.18	21.16	19.40	21.08	20.96	21.04	17.95	19.41
12	21.91	19.39	21.52	22.03	18.19	19.15	24.05	18.61
13	19.75	19.38	19.39	20.08	21.55	17.91	21.07	15.83
14	19.48	21.74	19.22	19.82	20.19	17.65	19.97	22.36
15	22.40	19.17	18.07	17.00	16.44	15.96	23.91	21.39
16	19.04	19.45	23.75	18.71	20.14	19.25	21.34	19.97
17	23.31	16.37	18.06	23.51	20.07	18.18	22.37	21.02
18	19.26	20.62	18.41	20.94	20.47	19.37	21.19	19.09
19	19.28	20.19	19.32	20.16	25.93	20.51	20.36	18.07
20	20.33	19.00	21.46	20.83	18.85	19.86	19.56	21.28
21	20.14	18.03	19.24	19.69	20.45	21.81	21.69	19.49
22	19.86	17.92	16.84	20.67	18.83	20.09	21.72	20.74
23	16.81	20.02	17.74	21.04	20.77	20.70	20.19	16.71
24	22.29	21.77	19.90	19.91	18.34	18.38	23.78	18.82
25	22.50	21.05	18.85	16.37	21.93	21.17	22.30	17.49
26	20.06	18.38	17.90	20.07	22.07	18.47	22.30	23.92

27	20.52	21.25	18.87	21.24	18.72	20.92	23.16	21.40
28	19.83	17.77	20.50	20.73	16.56	19.01	21.65	20.74
29	16.43	19.67	19.27	19.52	17.19	22.08	19.77	22.91
30	17.55	21.01	21.75	17.37	18.82	20.33	18.04	20.01
31	19.27	15.71	17.33	20.98	18.83	21.50	18.71	19.20
32	17.97	16.89	20.77	18.67	23.00	18.22	21.09	22.14
33	21.16	22.69	17.92	20.54	17.64	21.65	20.69	16.17
34	19.30	20.95	21.89	20.86	24.01	22.70	19.57	23.11
35	19.28	20.57	17.37	18.16	23.99	21.39	20.62	17.84
36	20.08	20.45	20.00	19.60	19.67	23.34	20.43	24.06
37	22.21	22.38	17.85	17.14	20.57	16.32	17.69	23.61
38	22.04	15.03	20.02	19.91	19.48	17.46	19.08	19.20
39	19.84	19.75	23.47	21.00	19.41	18.28	18.10	17.64
40	17.72	19.65	18.75	20.16	20.49	19.74	17.13	18.86
41	16.02	20.41	17.96	19.04	16.03	19.66	18.56	22.19
42	22.40	24.14	22.22	20.10	17.94	20.40	19.42	23.01
43	18.33	17.52	13.86	21.88	18.95	14.48	17.42	19.42
44	22.37	16.57	20.69	21.17	17.35	21.92	22.27	18.29
45	20.16	22.32	20.23	22.66	21.70	20.01	20.17	21.78
46	21.94	22.65	18.36	17.92	18.55	18.86	22.57	21.72
47	21.01	18.22	23.43	20.34	18.65	22.55	16.30	18.28
48	19.28	16.86	17.84	20.25	18.68	24.90	20.39	17.09
49	21.99	19.69	15.94	15.08	18.44	19.91	17.75	22.03
50	14.92	20.21	16.19	18.19	21.90	19.92	22.65	16.71

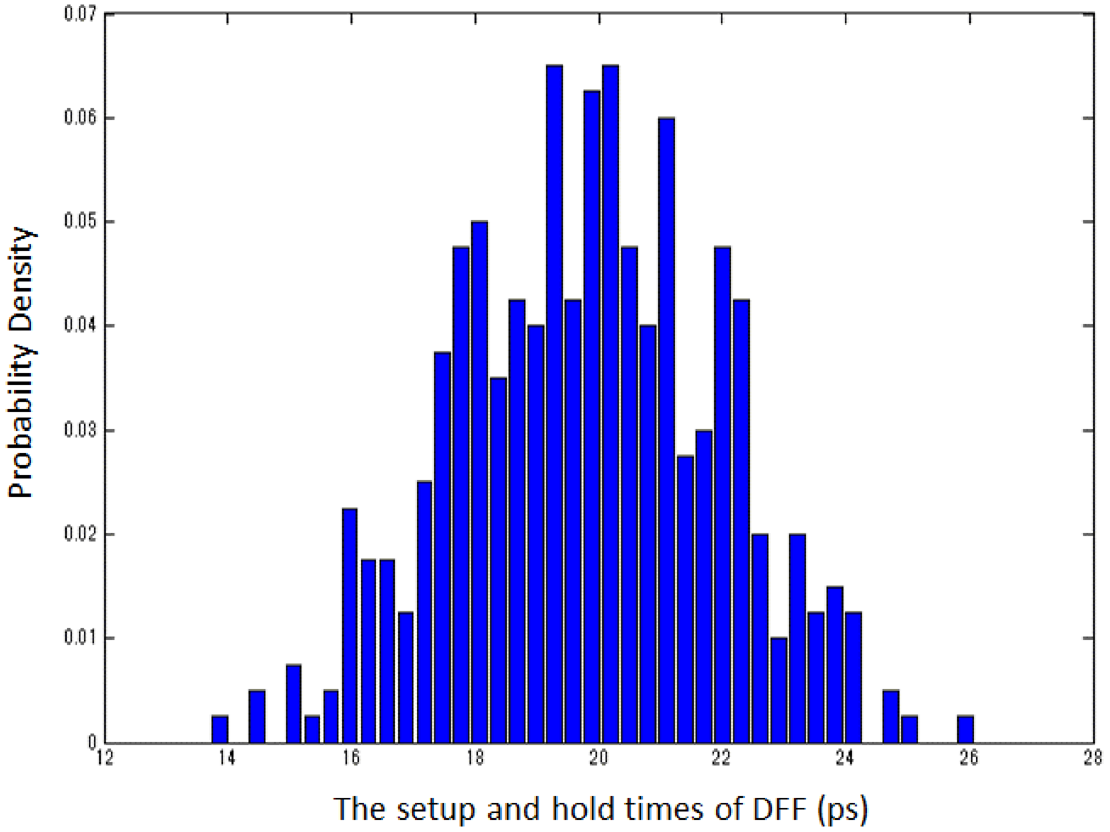


Fig. IV The probability density function of DFFs' setup and hold times (case #5)

Table V The setup and hold times of each DFF (case #6)

(Normal distribution, mean = 20ps, standard deviation = 3, unit: ps)

Column Row	1	2	3	4	5	6	7	8
1	20.46	21.50	17.80	16.80	18.15	22.46	21.43	22.26
2	18.45	19.65	19.69	13.82	22.34	21.74	14.56	22.19
3	14.93	18.84	15.07	24.12	17.55	16.75	16.29	21.36
4	18.41	20.05	19.77	18.34	23.22	21.82	22.24	26.16
5	20.63	22.44	19.64	15.30	18.43	24.62	17.48	18.77
6	22.38	15.48	20.88	21.11	14.90	21.08	21.58	21.14
7	21.06	17.68	20.36	21.24	21.08	15.07	23.20	19.15
8	22.09	16.71	22.83	15.93	15.49	23.51	23.39	18.16
9	19.33	20.19	18.23	16.84	19.75	20.04	21.23	19.04
10	20.16	23.00	16.74	19.20	26.96	18.86	22.29	24.92
11	17.32	22.24	24.09	22.36	19.55	16.58	15.11	24.72
12	19.77	14.25	25.23	18.29	20.98	19.14	21.45	19.63
13	19.59	23.12	21.59	16.81	26.71	19.33	19.40	20.85
14	25.95	23.84	14.52	17.13	23.02	17.23	24.65	17.18
15	22.51	20.41	26.41	17.43	24.64	13.53	18.11	16.86
16	22.07	20.75	15.01	23.15	18.07	18.20	19.97	23.58
17	20.09	25.07	21.21	27.07	11.96	18.35	13.07	20.92
18	19.41	23.21	20.50	21.93	21.29	15.17	19.52	20.37
19	22.62	21.17	18.21	11.12	22.65	18.00	18.40	16.55
20	20.24	22.56	16.01	21.98	23.67	20.08	26.23	23.35
21	19.20	26.89	15.13	28.39	18.49	19.23	23.60	20.64
22	20.55	15.58	22.79	24.15	24.85	22.10	18.95	19.45
23	19.95	17.13	18.29	21.38	18.14	19.85	14.56	24.51
24	20.25	18.84	18.80	20.06	18.82	17.06	17.16	19.83
25	20.27	19.41	18.75	22.84	17.54	19.04	19.17	19.57
26	18.94	16.90	22.80	16.83	19.45	17.06	21.16	24.50

27	18.76	18.58	20.09	20.81	21.38	19.92	19.53	17.12
28	18.48	20.71	14.40	20.95	21.49	22.31	17.59	21.09
29	18.28	22.11	22.17	20.62	23.00	17.86	20.22	18.10
30	20.74	13.38	18.33	20.94	18.92	19.21	21.19	15.03
31	21.40	21.39	19.29	17.12	23.77	20.19	21.49	19.72
32	17.74	23.53	18.91	17.09	19.84	15.92	18.88	15.89
33	18.92	18.78	15.05	17.56	15.73	19.44	19.91	24.97
34	16.85	21.98	21.26	22.27	19.11	19.46	22.75	25.15
35	20.72	24.52	20.98	21.51	28.45	17.00	15.55	19.01
36	25.33	24.95	22.19	19.85	23.94	18.52	19.12	19.95
37	21.57	19.28	24.60	15.69	24.14	21.45	22.08	21.67
38	20.60	16.57	19.17	19.28	22.75	19.60	20.31	23.80
39	15.68	20.66	20.01	16.15	22.89	20.41	19.58	20.78
40	19.38	21.79	17.83	18.56	19.75	20.39	18.47	21.78
41	22.65	22.61	20.67	19.64	13.43	14.17	19.06	22.07
42	17.80	16.81	19.31	22.02	19.35	23.62	19.18	23.51
43	15.67	21.20	20.91	16.49	27.30	22.91	20.14	24.70
44	16.06	18.45	20.51	18.35	19.45	20.29	16.62	23.02
45	24.81	16.56	21.43	21.50	22.43	22.59	17.52	19.69
46	17.53	14.69	20.58	17.72	20.11	18.50	20.15	16.80
47	19.19	18.39	24.38	21.59	20.74	23.69	21.53	16.77
48	21.22	19.66	24.91	18.86	21.60	21.08	19.61	11.38
49	22.74	21.72	19.51	24.99	20.12	19.50	20.75	23.95
50	18.67	16.28	20.17	21.64	22.07	18.51	16.17	14.72

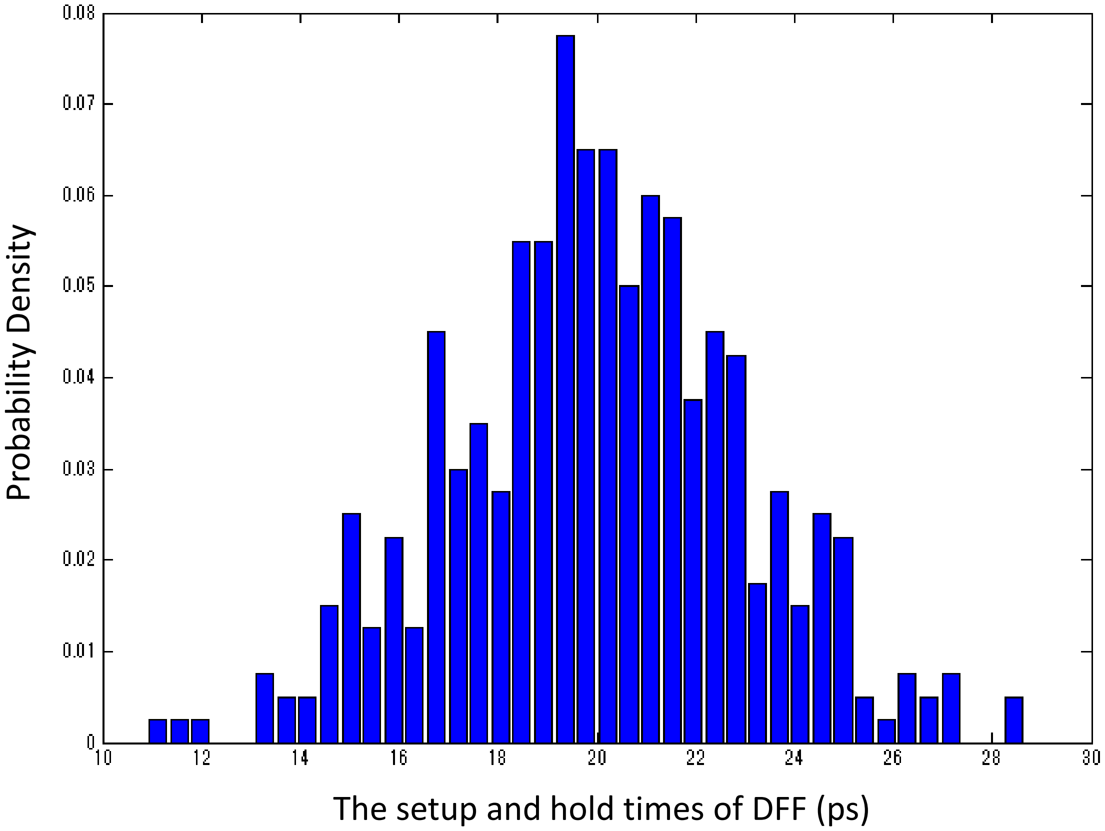


Fig. V The probability density function of DFFs’ setup and hold times (case #6)