October 2019

TIME-DIFFERENCE CIRCUITS: METHODOLOGY, DESIGN, AND DIGITAL REALIZATION

Shuo Li

TIME-DIFFERENCE CIRCUITS: METHODOLOGY, DESIGN, AND DIGITAL REALIZATION

A Dissertation Presented

by

SHUO LI

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

September 2019

Electrical and Computer Engineering

TIME-DIFFERENCE CIRCUITS: METHODOLOGY, DESIGN, AND DIGITAL REALIZATION

A Dissertation Presented

by

SHUO LI

Approved as to style and content by:

Wayne P. Burleson, Chair

Russell Tessier, Member

Emily Kumpel, Member

Christopher V. Hollot, Department Head
Electrical and Computer Engineering

DEDICATION

To Elaine and Xiaolin

ACKNOWLEDGMENTS

There are so many people I want to thanks to help me, encourage me, support me to finish my PHD dissertation. First of all, I mush express my gratitude to my two advisors during the whole PHD study: Professor Wayne Burleson, I can say without him, I cannot approach here. He pulled me out from dilemma and gave me the chance to continue pursuing my goal; he has always shown his kindness, knowledge, patience, and understanding to help me and support me, which is very important for me and my family, thus we can stay together after my babys born. Working with him has simply become a pleasure since he gave me freedom to work on the project, and provided lots of updated experience from industry to lead the advancing and practical topics in my thesis.

Also I am thankful to Dr. Christopher Salthouse, who led me into this area, getting in touch with integrated circuits design from the beginning. He gave me a very precious opportunity to finish a tape-out independently when I was still a fresh master student right after joining the lab. He was using his knowledge and experience to guide me whenever I needed. My first understanding and training of research are all from him. Working in BEL lays good foundation of IC design, which I think will benefits for long term career and future life.

I gratefully acknowledge the financial support provided by BEL and the department of Electronic and Computer Engineering. In particular, I want to thank Prof. Christopher V. Hollot, Prof. Robert Jackson, Prof. Paul Siqueira, and Prof. Russell Tessier, who generously gave me advices and help; encourage me to complete my thesis when I encountered difficulties. Also my friends and colleagues at UMASS provided great company, feedback, and wonderful friendship during the past years.

They made the beautiful town Amherst my first family in US. And all these memories have become invaluable reward of my PH. D degree.

It has been 9 years I left home to come aboard to pursuit my dreams. My parents provided all physical, mental and financial support through these years. Whenever I felt upset either from study or life, they can immediately comfort me and make me relaxed. They are the foundation and freedom to undertake many adventures. I hope I can have more time to company with my family in return.

Finally, I should express my special gratitude to my beloved husband, my best friend, Dr. Xiaolin Xu. His faithfulness to me and my family becomes the best support for me to continue finishing the work. I appreciated everything we experienced together over the past 10 years. The best thing we work together is raising our lovely daughter Elaine. Elaine has kept me focused on the important priorities; she reminds me each day of small joys that would otherwise go unnoticed. She taught us from her joy of learning and explorations.

There are many others to thank as well, too many to list here. So to my extended family and friends who have supported me financially, spiritually, and emotionally, I want to sincerely say thank you.

ABSTRACT

TIME-DIFFERENCE CIRCUITS: METHODOLOGY, DESIGN, AND DIGITAL REALIZATION

SEPTEMBER 2019

SHUO LI

B.E., UNIVERSITY OF ELECTRONIC SCIENCE & TECHNOLOGY OF CHINA

M.S., UNIVERSITY OF MASSACHUSETTS AMHERST

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Wayne P. Burleson

This thesis presents innovations for a special class of circuits called Time Difference (TD) circuits. We introduce a signal processing methodology with TD signals that alters the target signal from a magnitude perspective to time interval between two time events and systematically organizes the primary TD functions abstracted from existing TD circuits and systems. The TD circuits draw attention from a broad range of application fields. In addition, highly evolved complementary metal-oxide-semiconductor (CMOS) technology suffers from various problems related to voltage and current amplitude signal processing methods. Compared to traditional analog and digital circuits, TD circuits bring several compelling features: high-resolution, high-throughput, and low-design complexity with digital integration capability. Further, the fabrication technology is advancing into the nanometer regime; the reduction in

voltage headroom limits the performance of traditional analog/mixed-signal designs. All-digital design of time-difference circuit needs to be stressed to adapt to the low-cost, low-power, and high-portability applications.

We focus on Time-to-Digital Converters (TDC), one of the crucial building blocks in TD circuits. A novel algorithmic architecture is proposed based on a binary search algorithm and validated with both simulation and fabricated silicon. An all-digital structure Time-difference Amplifier (TDA) is designed and implemented to make FPGA and other all-digital implementations for TDC and related TD circuits feasible. Besides, we propose an all-digital timing measurement circuit based on the process variation from CMOS fabrication: PVTMC, which achieves a high measurement resolution: $< 0.5ps$. Moreover, our experimental results demonstrate that the PVTMC is fully compatible with the CMOS technology nodes from $180nm$ to $7nm$ with enhanced performance in more advance nodes.The design of PVTMC is realized on two FPGAs: Spartan-6 ($45nm$) and Artix-7 ($28nm$) and also obtained a high resolution of $< 1ps$.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER 1

BACKGROUND AND MOTIVATION

The characteristics that are commonly used to describe electronics circuits include voltages, currents, or digital bits. This thesis focuses on the investigation of a different variable: Time Difference (TD), which is essential for a wide variety of both on-chip and off-chip applications. Examples include measuring clock skew and jitter [3][4], building LIDAR remote sensing systems [5], synthesizing and optimizing digital systems. The main contributions of this thesis are 1) a design methodology that systematically organizes the TD circuits, 2) several novel and practical time-to-digital converter (TDC) designs, and 3) a new approach to an All-Digital design of a Time-Difference Amplifier, which forms the foundation of the TD circuits digital realization.

Voltage, current, and charge are three commonly used parameters in the description of analog circuits. These three parameters are closed related with each other: voltage is defined between each node and ground, current is defined through each element, and charge is stored on capacitors, as shown in Fig. 1.1.

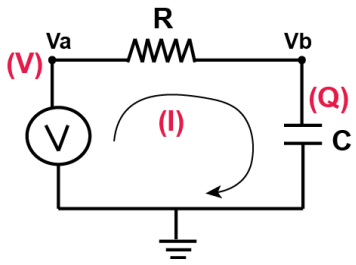


Figure 1.1: Traditional analog variables represent circuits.

Though these parameters are closely related, different parameter-based circuits have unique strengths and weaknesses. For example, voltage-mode circuits are the

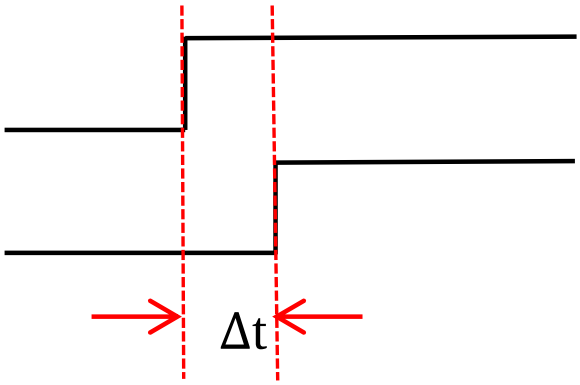


Figure 1.2: Definition of TD variable. The rising edges of two pulses determine the time difference.

most commonly used because voltage can be easily distributed and nondestructively measured with high input impedance instruments, such as voltmeters and oscilloscopes [6]. The weakness of voltage-signal is that its value is usually limited within the range of the supply voltage. As the CMOS feature size shrinks down below 100-nanometers (nm), the transistor gate oxide thickness forces the supply voltage to decrease below 1 volt. Correspondingly, the signal headroom becomes too small to design circuits with sufficient dynamic range [7]. Although the current-mode circuits take advantage of easy and accurate scaling and addition [8]. They have unique weaknesses that measuring current usually means high power consumption for large signal values and the need to make copies of currents with current mirrors to distribute a signal.

Time Difference (TD) circuits are becoming attractive for two types of applications: 1) where the physical signal being processed is inherently a time difference between two events; and 2) the processing of analog signals should be at very low supply voltage [9]. Specifically, the Time Difference (TD) in this thesis is defined as the time interval between two pulse edges, shown in Fig. 1.2, which is obtained by differentiating two pulsing edges. The advantage of TD variable that is independent with power supply makes it an ideal parameter for analog signal processing with requirement for higher resolution and lower power consumption than traditional analog signal processing [10].

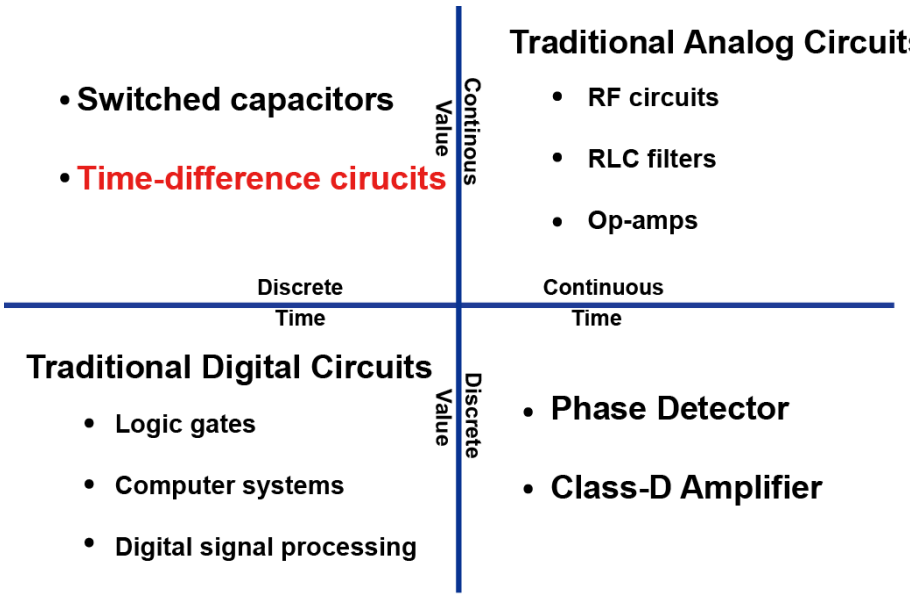


Figure 1.3: Classification of circuit types in time-value plane.

1.1 Classification of circuits

All types of circuits can be classified according to the time-value quadrants plane. The TD circuits presented in this thesis operate on continuous electronic parameters like voltage and current, which is different from circuits that work in discrete time points. The circuits belong to the class are known as the discrete-time continuous-value circuit, as shown in Fig. 1.3, which shows that the sampled value at discrete time point can be any real number within a range, but it can only change at discrete times. TD circuits can be compared to analog circuits in the time-value plane where signals vary continuously in a range. In comparison, traditional digital circuits quantize both time and voltage, so the signal only switches at the clock frequency between two stable logic voltages "0" and "1".

The less common quadrants of the time-value plane are Continuous-Time Discrete-Value (CTDV) and Continuous-Value Discrete-Time (CVDT). The most common CTDV circuit is digital phase detector, while the most commonly used DTCV circuit is switched-capacitor circuit [11] [12]. Switched-capacitor circuits hold an analog signal

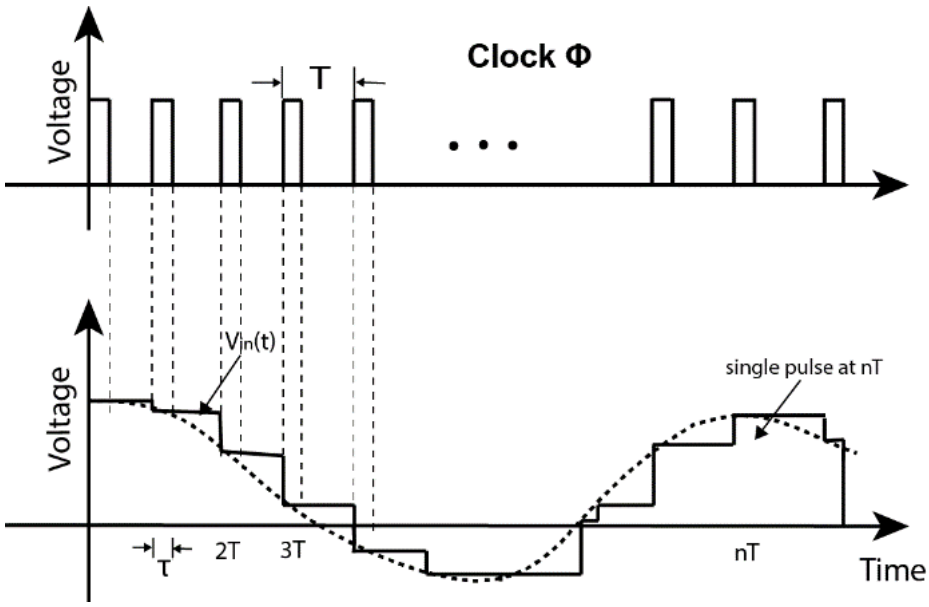


Figure 1.4: Discrete-time continuous-value signal sampled with clock cycle T.

as the charge on a capacitor, which is only get charged during clock edges, as shown in Fig. 1.4. Switched-capacitor circuits are good for discrete-time signal processing because the transfer function is governed by capacitance matching, not transistor parameters [13]. The TD circuits studied in this thesis are also DTCV circuits, but unlike switched-capacitor circuits whose value is invalid between changes, it is only valid during a single event.

1.2 Applications of time-difference circuits

Time-difference circuits can be employed in various fields [14] [15] [16] [17] [18] [19], a few of them are summarized in Fig. 1.5. This chapter highlights three applications: LiDAR [5], biomedical imaging, timing characterization for integrated circuit.

1.2.1 Application of time difference circuits in LiDAR

Light Detection and Ranging (LiDAR), like RADAR (Radio Wave Detection and Ranging), obtains the distance information by measuring the time difference between sending an excitation pulse and receiving a reflected pulse from a target object [20].

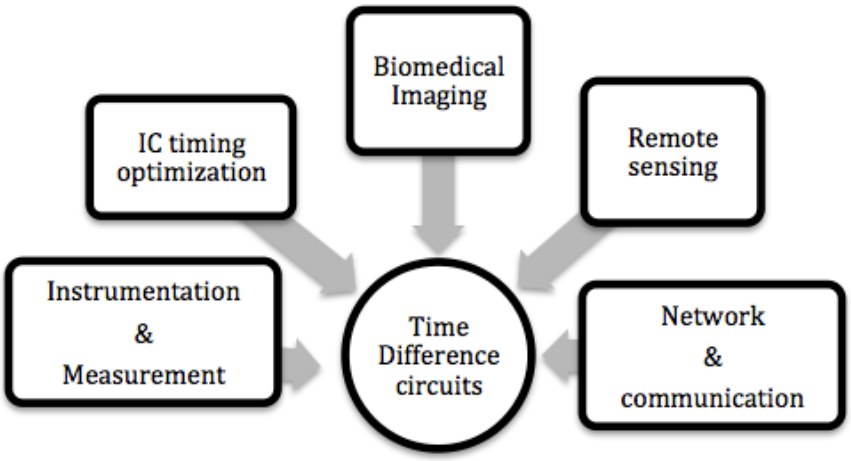


Figure 1.5: Applications of time-difference circuits.

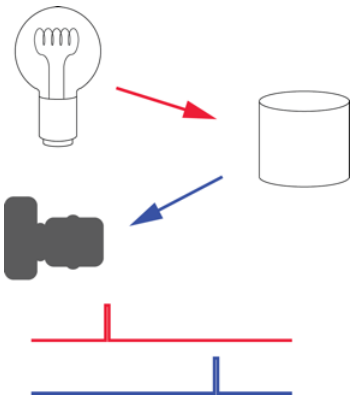


Figure 1.6: An example schematic of LiDAR system. The distance between the object and sensor is measured by time interval between two pulses.

$$|D| = \frac{1}{2}(v_{light} * t_{bounce}) \tag{1.1}$$

LiDAR and RADAR differ in the wavelength of the electromagnetic pulse. The shorter wavelength of light used in the LiDAR system compared to radio waves used in RADAR creates a more focused beam [20]. The narrower beam means that LiDAR can image smaller objects [15] [5]. LiDAR systems consist of three major components: a light source (laser), a scanner, and a sensor (photo-detector), as shown in Fig. 1.6.

A light source emits a pulse that bounces off the objects. The receiver contains high-speed photo-detectors that can sense the reflected laser pulse. Time-difference circuits are then used to measure the time interval between these two pulses. Since the speed of light is $299,792,458$ meters per second, the measurement resolution of picosecond to nanosecond timescale are required to achieve excellent accuracy for measuring small objects such as forestry layer measurements [21], atmospheric aerosol measurements [22], and chemical particle detections.

Because LiDAR uses light wave, it applies a similar basis with the time-of-flight (TOF) camera [23], which can be used to capture 3D images. The TOF camera measures the distance from objects to its every pixel, requiring high-resolution detector and TD circuits to obtain the small range accurately. Real-time motion imaging systems require a short time-difference measurement circuit with a high-speed operation that can be integrated within each pixel in the imaging sensor.

1.2.2 Application of time difference circuits in fluorescence lifetime imaging

The time difference between excitation of a light pulse and recording of a photon is also of interest for fluorescence lifetime imaging [24]. Fluorescence imaging is widely used in biology because of the ease of labeling biological molecules, the availability of multiple colors and high signal-to-background ratio. Fluorescent labeling of proteins is an important technique for understanding protein structure, protein folding, and chromosome formation [25] [26] [27]. RNA and DNA labeling are important techniques for disease diagnosis [28].

Fluorescence imaging is performed by illuminating a sample with filtered light by an excitation filter and imaging the sample through an emission filter [24]. The fluorescence lifetime is the time constant of exponential energy decay, as shown in

Eq. 1.2 [24] [29]. Fluorescent lifetime is used to measure the chemical environment in solutions of pH or calcium ion concentration [30] [31].

$$p(t) = \alpha e^{-\frac{t}{\tau}} \tag{1.2}$$

TD circuits are significantly used in two techniques for CMOS fluorescence lifetime sensors: direct integration/digitization and time-correlated single photon counting (TCSPC), both techniques can be used in high-throughput biomedical imaging [24]. The usage of the pixel topology determines the choice of the sensors and the corresponding timing control circuits. Direct integration and digitization use a high-speed detector to record the exponential decay in a single excitation pulse [32]. This is a conventional technique with which the fluorescence lifetime can be measured directly by exciting the fluorophore with an impulse of light and measuring the fluorescence with a high-speed detector [24]. Rapid lifetime determination (RLD) calculates the fluorescence lifetime using images from two time windows [33], as shown in Fig. 1.7. The ratio of the signals from the two windows is then used to calculate the fluorescence lifetime using Eq. 1.3.

$$\tau_{RLD} = \frac{-\tau_{win}}{ln(\frac{w_2}{w_1})} = \tau \tag{1.3}$$

Optimum width of the time windows, τ_{win}, for creating a low noise measurement, is equal to the expected fluorescence lifetime [24]. Since fluorescent lifetime floats between a hundred picoseconds and hundreds of nanoseconds, TD circuit is required to precisely set the adjustable integration time windows with wide time range for optimum signal-to-noise sensing of fluorescence lifetime [34].

Unlike integrating pixels that convert the number of photons hitting the photodiode into a voltage, TCSPC applies single-photon avalanche diodes (SPADs) to generate the signal as soon as the first individual photon is detected [35][32]. The sample is excited periodically with a pulse source, each time a photon is detected,

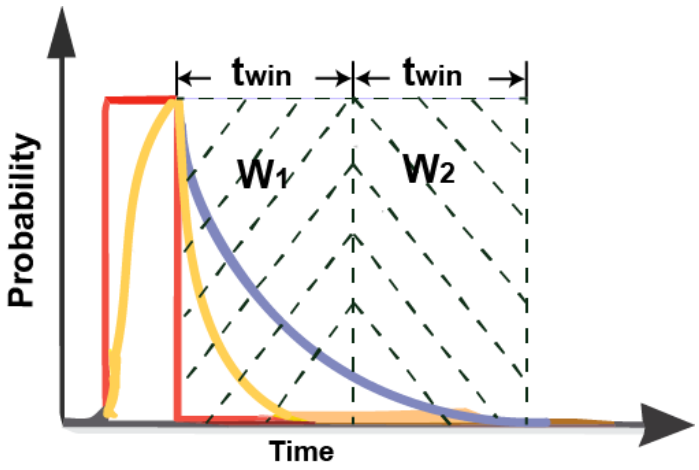


Figure 1.7: RLD method for fluorescence lifetime measurements. When a pulse excites the sample, the intensity falls off exponentially after the pulse. The fluorescence lifetime can be calculated using RLD method with the fluorescence intensity in two time windows W1 and W2.

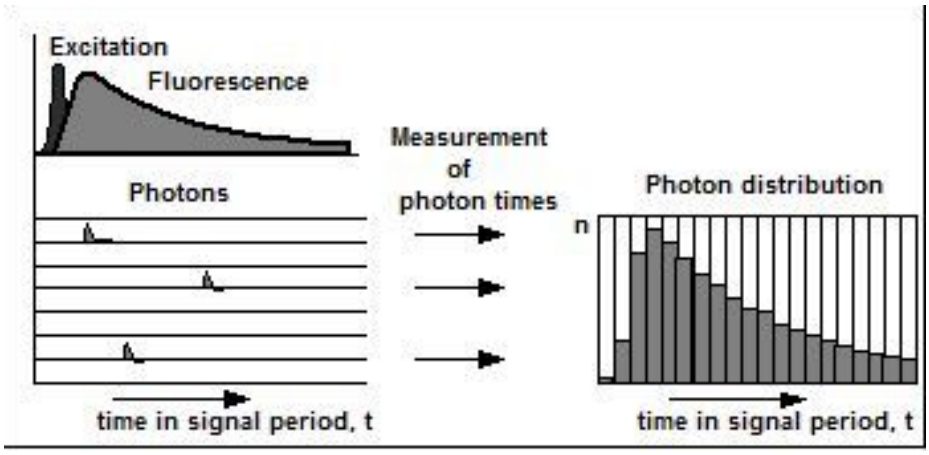


Figure 1.8: Operation principle of time-correlated single photon counting (TCSPC) measurements. The sample is excited by a pulsed laser source with a high repetition rate. By counting many events a histogram of the photon distribution over time is built up.

time difference circuits measure the time interval between corresponding detection and excitation pulses. The photon is counted as "1" at the location that is proportional to the detection time in the memory. After collecting a group of photons, a histogram of detection time reconstructs the optical decay waveform, as presented in Fig. 1.8.

Because at very low light levels every photon that hits a photo-diode is collected and its arrival time is precisely measured, the sensor arrays built with SPAD pixels are more sensitive to low fluorescence signal levels than sensor arrays built with integrating pixels. In this case, a time-to-digital converter (TDC) is an essential component in SPAD pixel-based fluorescence sensor and requires compact structure, high resolution and throughput [35].

1.2.3 Application of time difference circuits in timing characterizing and IC optimization

As transistor characteristics are shrinking down to sub-ten nanometers in CMOS fabrication process, minimum gate delay decreases to dozens of picoseconds or even sub-ten picoseconds. In such a fine time scale, the supply voltage is reduced below one volt, which makes the manipulation of analog parameters more challenging. To overcome these issues, TD circuits are proposed to offer an alternative signal processing approach for conventional A/D converter and Phase-Locked Loop (PLL) that can avoid the limitations on voltage headroom, but without degrading the voltage resolution. Meanwhile, digital-friendly mixed-signal circuits such as all-digital phase-lock loop (PLL)/delay-lock loop (DLL) are highly developed as the interface that can help better leverage digital circuits to improve analog processing capability, as shown in Fig. 1.9. In these circuits, the digital representation of the phase difference and time mismatch is an essential function for the digital-controlled synchronization and calibration [36][10][37]. In both scenarios, the TD circuits plan an essential role for the mixed-signal circuits to maintain high-quality performance.

The time jitters and clock skew in critical paths, or high-speed interconnections can directly impact the timing characteristics of the electronic systems, which may even cause function failure [3][4]. Jitter is the timing variations of a set of signal edges from their ideal values, which in clock signals are typically caused by noise or other

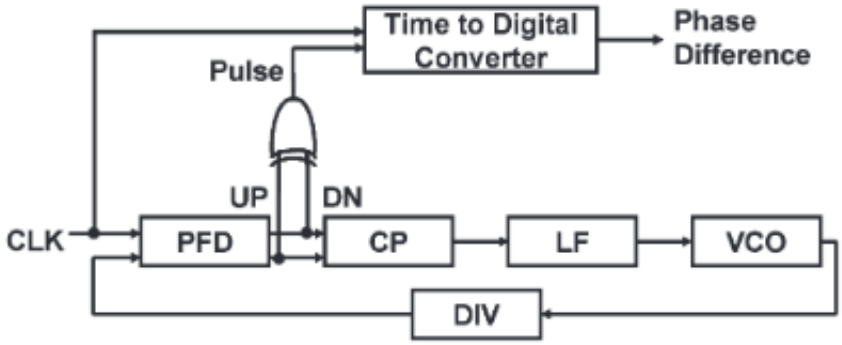


Figure 1.9: Build-in timing measurement circuits using TD circuits (Time-to-Digital Converter) for real-time digitizing and characterizing the phase error for PLL/DLL.

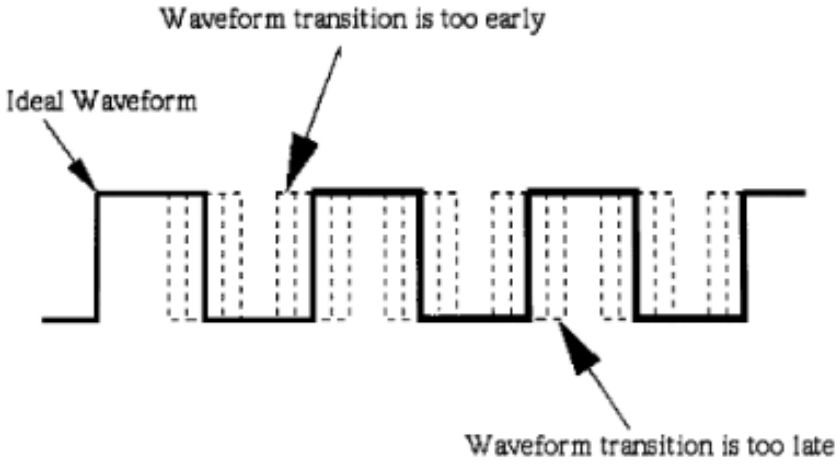


Figure 1.10: Jitter description in a clock signal.

disturbances in the system, as demonstrated in Fig. 1.10. The composition of jitter includes thermal noise, power supply variations, loading conditions, device noise, and interference coupled from nearby circuits [4]. Consider a microprocessor-based system in Fig. 1.11, in which the processor requires data setup time before clock rise, if the clock encounters negative period jitter, then the rising edge of the clock could occur before the data is valid [38][39]. Similarly, the hold time may be violated if positive jitter happens. Hence the microprocessor will be presented with incorrect data.

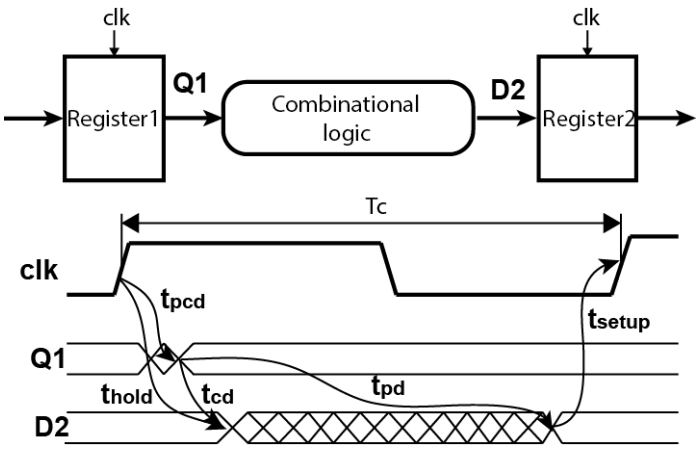


Figure 1.11: Time constraints of data setup/hold time for sequential circuits.

In summary, the demanding for a robust and efficient testing block of the jitter, phase, and distributed clock skew measurements have rapidly risen in recent years. In both scenarios, TD circuits, especially Time-to-Digital Converters (TDCs) are playing an essential role in timing measurement. TDCs realize the non-invasive measurement of timing/phase characteristics by precisely digitizing the time interval between two events, usually two rising edges. On-chip integration with the DUT blocks fully reduces the effects of the external loads and noise in the off-chip testing system [20]. The state-of-art performance of integrated systems requires that TDC must provide high resolution within a very small area in order to be distributed around the die.

1.3 Thesis Outline

This chapter mainly describes the background of time difference circuits, a class of circuits that measure the time interval between two events. Chapter 2 presents the methodology of signal processing based on the time-difference circuits and provides a new organization of TD function blocks that are used to build comprehensive time-difference circuits. Chapters 3 introduces the different types of TDCs by comparing their topologies, performance, and cost. And a novel compact TDC architecture is proposed and fabricated. Chapter 4 solves practical and industrial problems in

proposed algorithmic TDC; the design is improved to increase the digitizing throughput and save more power. In Chapter 5, we come up with an all-digital time-difference amplifier with only standard digital blocks, the implementation of which can be successfully transferred to difference advanced CMOS technologies, which lays the foundation of the digital realization for the time difference circuits. Chapter 6 presents a novel timing measurement circuit: PVTMC that constructively leverages the process variations from circuit fabrication to measure timing signals, which achieve high resolution as of $< 0.5ps$. Chapter 7 concludes this thesis.

CHAPTER 2

METHODOLOGY

The concept and classification of time difference (TD) circuits are introduced in Chapter 1. In this chapter, we discuss the TD signal processing methodology in details and provide a systematic organization of the TD function modules. In TD circuits, the most interesting information is the time difference between between two pulses, rather than the nodal voltages or branch currents of electric networks. Therefore, TD circuits offer a new and promising way for mixed-mode systems to deal with the challenges of conventional voltage/current-mode designs.

TD signal, which two time instants occur at specified timing points, will disappear at any timing point in the future. Since the original events are gone, so does the TD information stored in the signals. Therefore, the first difficulty of measuring TD signal is to storage and re-usage it with considering the discrete-time and continuous-value feature. In this chapter, we systemically study the fundamental of TD circuits, new methodology for TD signal processing definition and circuits expression are also included.

2.1 Signal processing in time-mode

Time-mode circuits depict an analog signal using the difference between the timing points at which two timing events take place. Ideally, the amount of time difference is linearly proportional to the amplitude of the analog signal. A time variable is a pulse-width-modulated signal with its pulse width directly proportional to the magnitude of the signal that it represents. A time variable possesses a unique duality characteristic;

specifically, it is an analog signal as the continuous amplitude is represented by the duration of the pulse. However, we can also deem it as a digital signal since it only has two largely distinct values. The duality of time variables enables them to conduct analog signal processing in a digital environment. This unique characteristic is possessed by neither analog nor digital variables.

Time-mode signal processing deals with addition, multiplication, amplification, integration, and comparison, etc., of time variables. Information to be processed by time-mode circuits is represented by the time difference between digital signals, for example, pulses. These circuits are necessarily digital systems that perform analog and mixed analog-digital signal processing without using power-greedy and speed-limited digital signal processors.

2.1.1 Time-mode vs. voltage-mode signal processing

Nowadays, the typical requirements of advancing CMOS technology is the capability of integration with high-performance digital systems. As a result, CMOS analog circuits are losing the flexibility to adapt specific and process-controlled components critical to the performance of these circuits [40]. Also, the voltage headroom, that is, the difference between the given supply voltage of a circuit and the minimum supply voltage of the circuitry required for MOS transistors to operate in saturation is continuously reducing and approaching to the performance constraints [7][40] [41]. The shrinking voltage headroom not only limits the maximum achievable signal-to-noise ratio (SNR), it also enlarges the effect of the non-linear characteristics of MOS devices which subsequently reduces the dynamic range of voltage-mode circuits [42]. As a result, the accuracy of voltage-mode circuits, or the minimum detectable voltage, typically degrades with technology development [40].

2.1.2 Time-mode vs. current-mode signal processing

Current-mode circuits apply current branches of circuit networks to represent the signal processing, compensating for the dropping of the voltage headroom. These circuits achieve a low voltage swing by lowering the impedance of nodes. The existence of low-impedance nodes throughout current-mode circuits, however, gives rise to large branch currents, typically consuming more power as compared with their voltage-mode counterparts. Lowering the power consumption of current-mode circuits while meeting other design constraints at the same time is rather difficult. The characteristics of the low-impedance nodes of current-mode circuits, on the other hand, offer an inherent advantage of a low time constant at every node of the circuits. As a result, current-mode circuits are suitable for applications where speed is more critical than power consumption. Since voltage and current are inherently related to each other via impedance or conductance, the characteristics of voltage-mode circuits and current-mode circuits do not fundamentally differ with each other. As a result, the performance of both circuits does not scale well with CMOS technology development.

2.1.3 Time-mode signal processing strengths and challenges

As the intrinsic gate delay of digital circuits accommodates most from technology scaling, time-mode circuits become a promising applicant of rapid signal processing. For example, the oscillation frequency of ring oscillators implemented in state-of-the-art CMOS technologies has reached tens of GHz, providing a large oversampling ratio while consuming a small amount of power [40]. It is showing that time-based signal processing has many desirable characteristics such as excellent scalability with the advancement of CMOS technology, such as reconfigurability, ease of portability, low-power consumption, and high-speed operation, which are competitive to mixed analog-to-digital signal processing outperform voltage and current driven circuits.

Although time-based signal processing possesses many critical advantages, challenges also need to be considered. The intrinsic gate delay of digital circuits benefits the most from technology scaling and device mismatch. To minimize the effect of device mismatch, minimum-sized delay cells are usually used, which can cause the impact on the speed and subsequently on other specifications of time-mode circuits. For time-mode circuits, to against of process, supply voltage, and temperature (PVT) variation is the other inevitable challenges [40]. Usually, corresponding calibration techniques and assisted circuits are applied to minimize the PVT effects on the performance of TD circuits [36].

Withholding or storing a time variable is rather difficult due to the irretrievable nature of time, which rises unique challenges to operate circuits using conventional function blocks [40]. To overcome the obstacles, we introduce a new methodology to organize and describe the TD circuits function in the next section, with which we can solve different problems with specified targets. The practical examples are demonstrated in the following chapters.

2.1.4 TD circuits operation functions

TD circuits are a new class in the circuits family branches where the particular circuits feature drives the unique understanding of the circuits expression. However, due to the relatively less study on it, the systematic organization of the function blocks is rare to see. In this section, we study all types of TD circuits from widespread application fields and give a summary of TD circuits function analogous to the conventional voltage-mode circuits, four primary functions are standardized as a sample and hold, the arithmetic unit, comparison, and amplification, to build comprehensive TD systems.

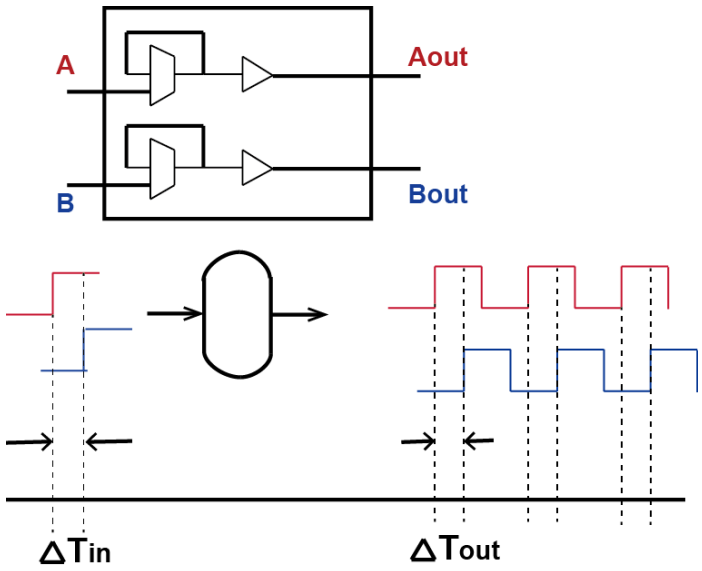


Figure 2.1: Function block and timing diagram of time-difference memory. The time-difference information is stored in the repeated events.

2.1.5 TD Sample and Hold (S/H)

As TD signal is transient, discrete time signal, the conventional sample and hold function is not applicable here. In order to realize multiple operations on the same TD at different time, the signal must be stored or rerouted around the circuit. A design is shown in Fig 2.1(top), in which the multiplexing paths let TD signal propagate through with delay buffers, this operation is more likely sampling the TD signal and holding it for the later use. Instead of recording the time difference between two events directly, TD S/H can be realized by repeating the timing events as well, as shown in Fig 2.1(bottom). With the repeated events, the TD value can be reused in each repeating period. Time-difference storage can be achieved with a feedback-loop multiplexer that is used to select feedback signal after the original input TD event finishes, this feedback-loop can also block interference signal from input. Delay buffers can be used to adjust the repeating period.

2.1.6 TD arithmetic operation

TD arithmetic operation such as TD adding/subtracting and TD integration is critically needed in time-based signal processing. The accumulation of a variable in the voltage or current mode can be realized by representing the variable as current and integrating the current onto a capacitor or inductor. However, the arithmetic operation in TD circuits is different.

2.1.6.1 TD adding/subtracting

The mechanism to add or subtract a certain amount of time from the target TD between two signals is making either of the signals delayed by a fixed delay element. If the 'start' signal is delayed, the subtraction is archived; while opposite operation realizes adding. A fixed amount of delay time generated by a delay buffer or delay-line plays the role of an adder or a subtractor. The sign of the operand can be easily switched by moving the delay time between two input nodes.

The delay element shown in Fig 2.2 can be implemented with a simple CMOS inverter. The propagation delay t_{pd} of an inverter is a fixed value for a given CMOS process. The CMOS inverter-based delay element and its transient waveform are shown in Fig 2.3, in which the inverter delay element produces an inverting propagation delay. The transition time of the basic inverter is determined by the charging time of a load capacitor, C_{load}, with drain current I_D, as shown in equation 2.1.

$$t_{pd} = C_L \int \frac{dV_{out}}{|I_D|} \tag{2.1}$$

There are alternative ways of varying the propagation delay of delay-elements by adjusting either load capacitance or drive resistance. In Fig 2.4, we illustrate all these options using a control voltage, a control current, or a digital control value. The driven-resistance adjustable delay elements are also called a current-starved inverter. However, the tunable range of these methods are relatively smaller.

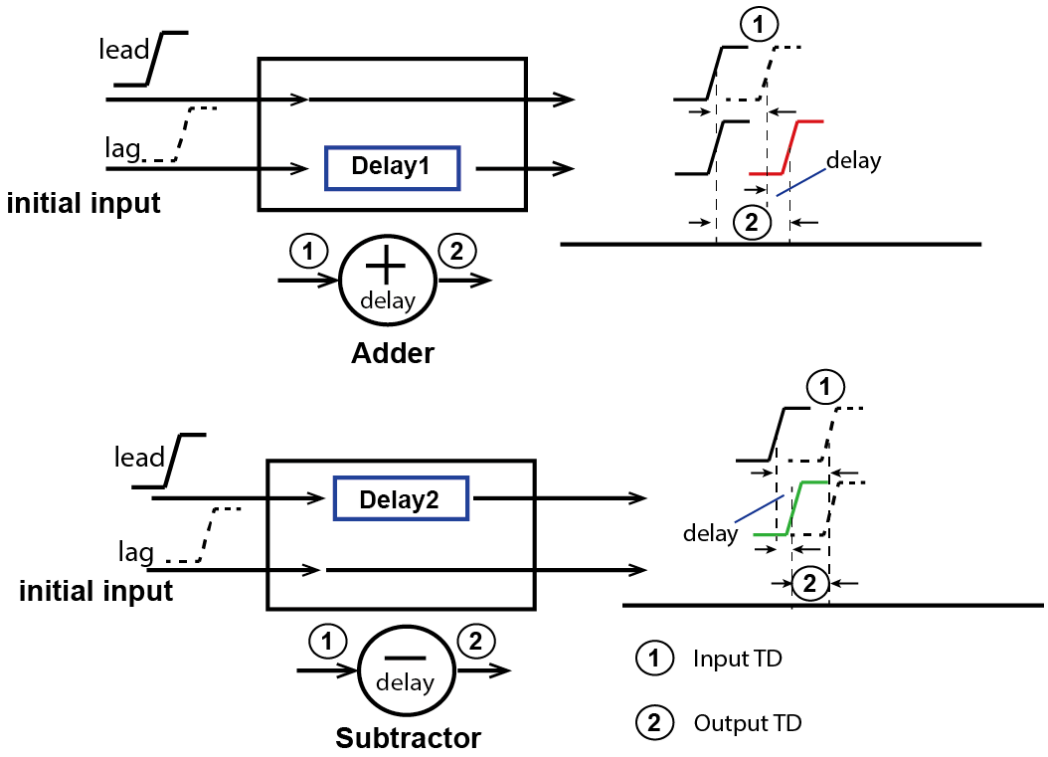


Figure 2.2: The symbol and basic function of a time-difference adder. TD ① labels the input TD value and ② labels the output TD value. Delay1 and Delay2 are two constant delay values

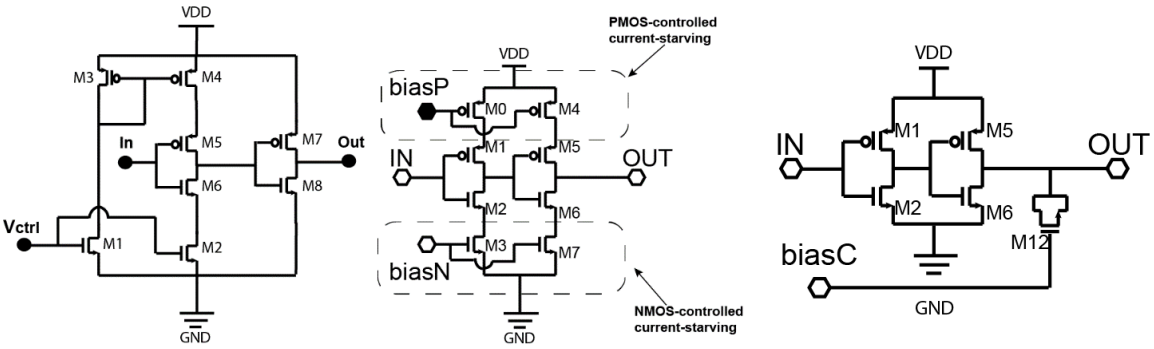


Figure 2.3: Alternative Designs of adjustable delay elements by controlling current, voltage and load capacitance.

The digital-controlled delay-element can also achieve similar performance of adjusting the delay time with small step. A useful feature of these designs stands out for directly interfacing with the digital system and microprocessors.

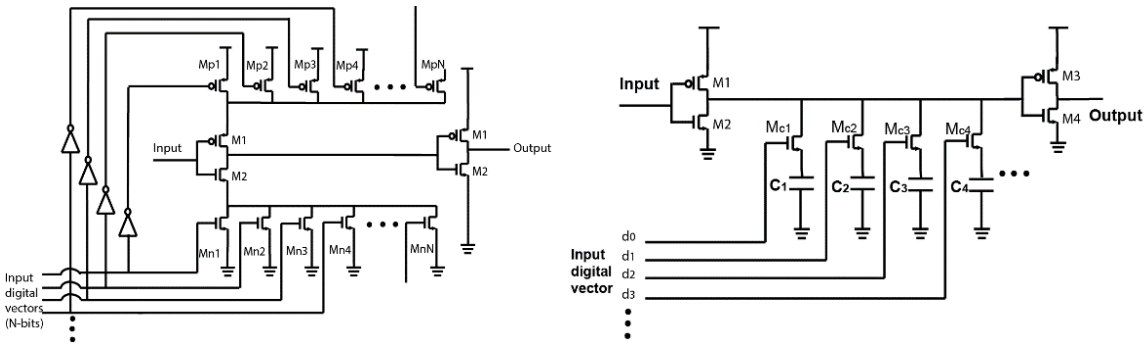


Figure 2.4: Alternative Designs of digital-controlled delay elements.

A digital-to-time converter (DTC) is the compound version of the digital-controlled delay element [43][44]. DTCs convert the digital input strings into precise TD value and are functioned as a programmable TD generator [34], which can configure a wide range of desired TD value as a reference of a clock or time windows for TD systems.

2.1.6.2 TD integration

It is incredibly challenging to hold a TD signal due to the instantaneous nature of time, not even to integrating the TD value continuously along the time axis. To fix this problem, time latch is designed to store the information of signal propagation and mimic the accumulation process of the signal integration. In the transparent mode, the signal propagates through the time latch; while in the opaque mode, the latch holds the signals. The two modes of the time latch are controlled by the enable signal ($EN/\overline{(EN)}$). When EN is high, the signal D propagates along the time-latch; in contrast, the signal propagation stops and is retained when the EN becomes active again. The time-latch function diagram is presented in Fig 2.5. When an input pulse becomes high, the signal **D** passes through **m** delay cells until the input pulse transition back to low, labeled as the gray square. When a trigger signal is applied to the time latch, the EN becomes active again and signal D resumes to propagate from the x^{th}th stage to the end of the delay chain to obtain the **FULL** signal. In this design, the input pulse is stored in its negative version -t_d.

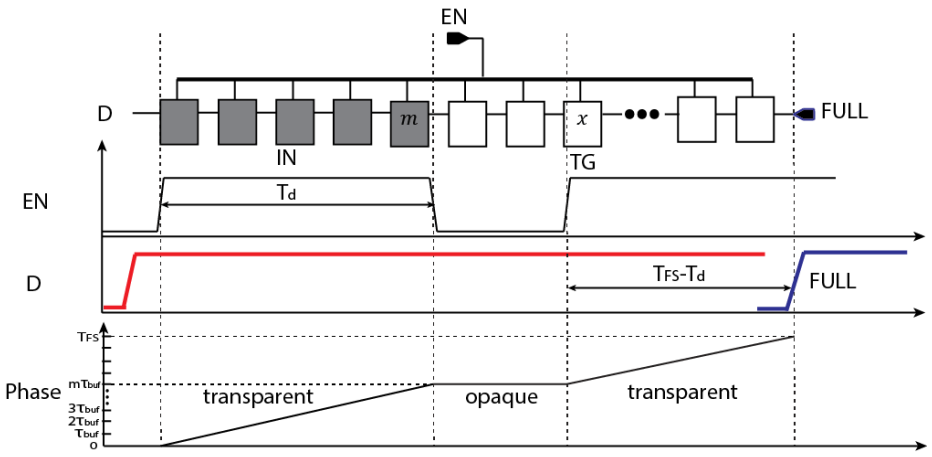


Figure 2.5: Time-latch function diagram, signal D propagates in transparent mode, and is held during the opaque mode.

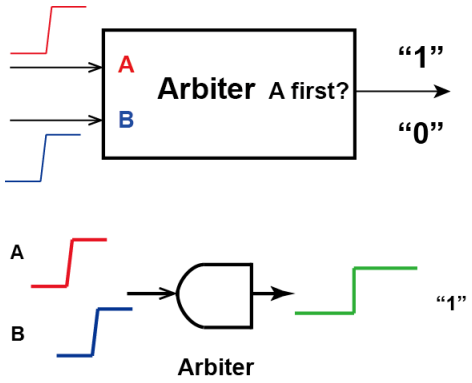


Figure 2.6: Symbol and function block of two-input TD arbiter.

2.1.7 TD comparison or quantization

A TD signal can be compared with zero or any non-zero reference T_{ref}: $TD - T_{ref} = TD' <$ **comp** > 0. An arbiter built with SR latch that generates mutual-exclusively output is used to perform the comparison. We define if the comparison result is greater than 0, meaning the original 'start' event leads 'stop' to occur at the arbiter and the comparison result gives '1'; otherwise the result outputs '0', as illustrated in Fig 2.6.

An arbiter can be built using an SR latch to determine the polarity of the time difference, as shown in Fig 2.7. Both input signals start at zero, and both outputs start at V_{dd}. After input A rises, OutA begins to fall. Since input B rises before OutA

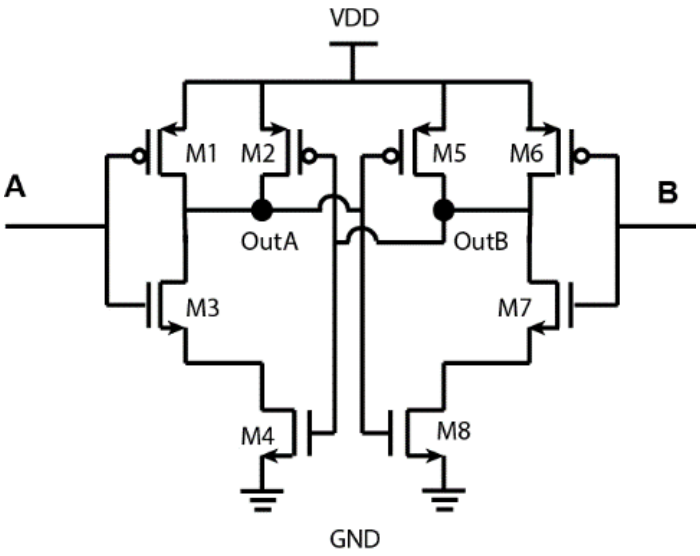


Figure 2.7: Schematic of SR-latch-based arbiter with two input events.

has finished falling, OutB starts to fall as well. OutA disables the other side first, so OutB eventually returns to V_{dd}. If the input time difference is small enough, the circuit can spend a significant amount of time with both outputs between V_{dd} and ground, illustrated in Fig 2.9. The decay relationship between input TD and ΔT_{out} is as shown in Eq. 2.2.

$$\Delta T_{out} = \tau * (logV_{th} - log(\alpha * \Delta T_{in})) \tag{2.2}$$

Besides the clear trend charging to 1 and discharging to 0, there is another state known as the metastable state for A and B. Metastability can cause problems in conventional electronic systems because the output is not a valid digital value during this time. To fix this problem, a metastable filter called MUTEX is applied to improve the SR-latch based TD arbiter. MUTEX improves on the SR latch-based arbiter by ensuring that the outputs are always valid digital values. Fig 2.10 presents the schematic of a MUTEX circuit, including two inverters. Note that the supply voltage nodes of the two inverters are attached to the output of the opposite inverter. These cross-coupled inverters function as a metastability filter: the PMOS transistors remain

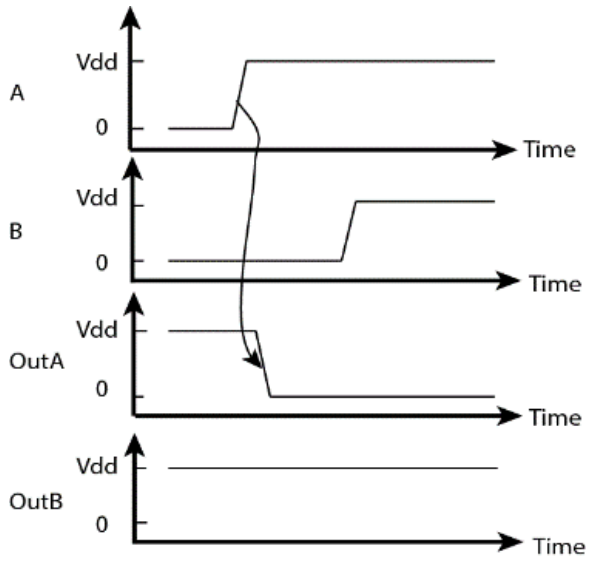


Figure 2.8: Arbiter function diagram. Two signals asynchronously arrive at the arbiter. The lead signal In1 is selected to be output by the arbiter, Out1 respond to transmit from Vdd to 0.

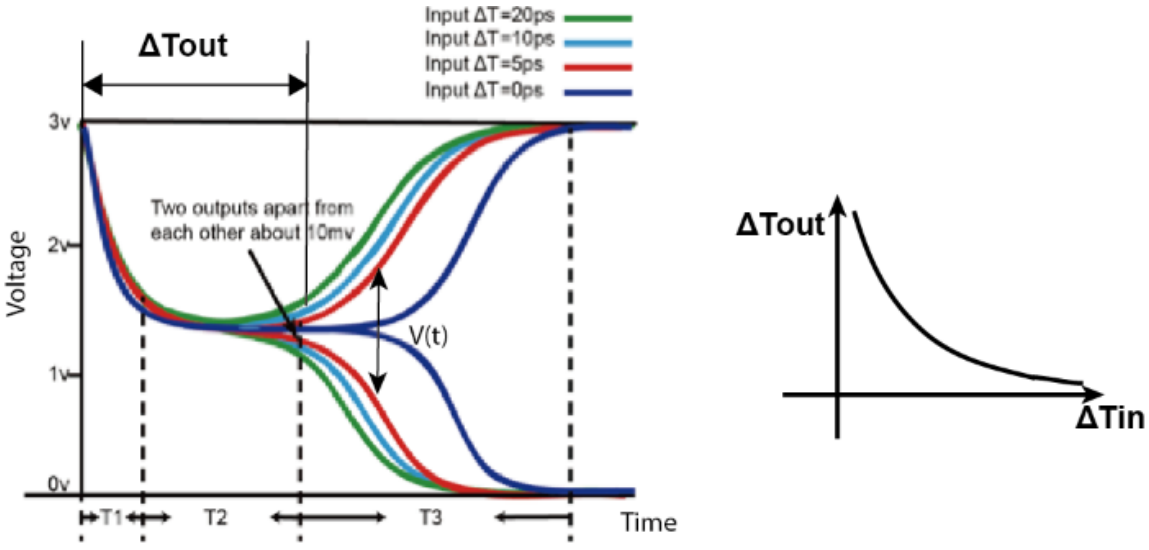


Figure 2.9: Output transient waveforms for different input time difference. ΔT_{out} exponentially increases with reduced ΔT_{in}.

in cut-off until the voltage difference between X and Y reaches the PMOS threshold voltage.

Once the two outputs of SR latch are distinguished by the value that at least reaches the threshold voltage of PMOS transistor, the pull-up transistor of the lead

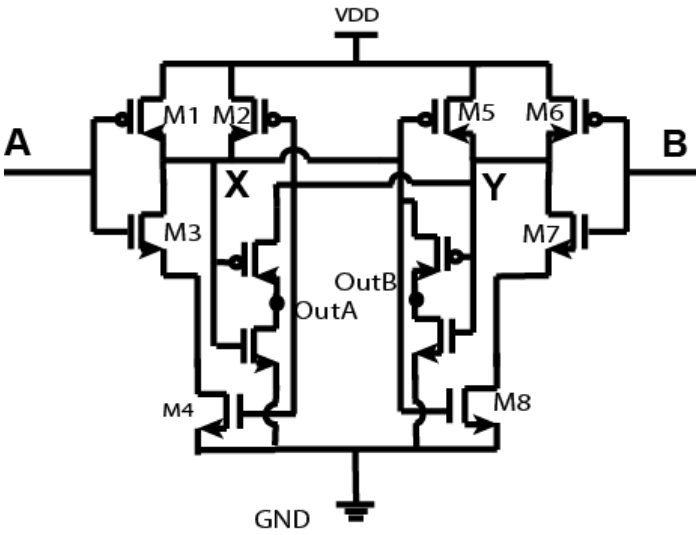


Figure 2.10: Schematic of MUTEX-based arbiter.

signal inverter will turn on. The inverter then toggles the output to V_{dd} to complete arbitration. The transient diagrams to illustrate the working function of MUTEX-based arbiter are shown in Fig 2.11.

$$V_{gs\_inverter} = |V_Y - V_X| \geq V_{th} \tag{2.3}$$

2.1.8 TD amplification

An effective method to measure a timing signal of very small scale is to amplify it and then measure the amplified version. Time-difference amplifier (TDA) is a such circuit, which takes a time difference signal ΔT_{in} and output a time-difference signal $T_{out} = A \times \Delta T_{in}$, as shown in Fig 2.12. As presented in Fig. 2.13, the quality of an amplifier can be characterized by several parameters: gain, linearity, operation range, and noise. The definition of amplification gain is described in Eq. 2.4. The gain error of a TDA circuit represents its linearity in timing amplification. A small error means that the amplifier is highly linear. Operation range determines the limits of input TD value can be amplified. Usually, the operation range is referred to as the linear amplification range.

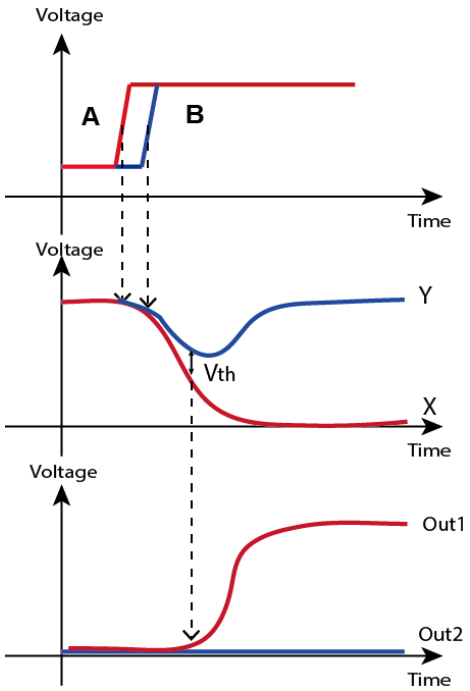


Figure 2.11: Timing diagram of MUTEX-based arbiter. SR latch enters metastable state due to the small time interval between two rising edges of In1 and In2. X leads Y during metal stable state, and the filter helps Out1 win the arbitration when the voltage difference between X and Y is larger than V_{th}.

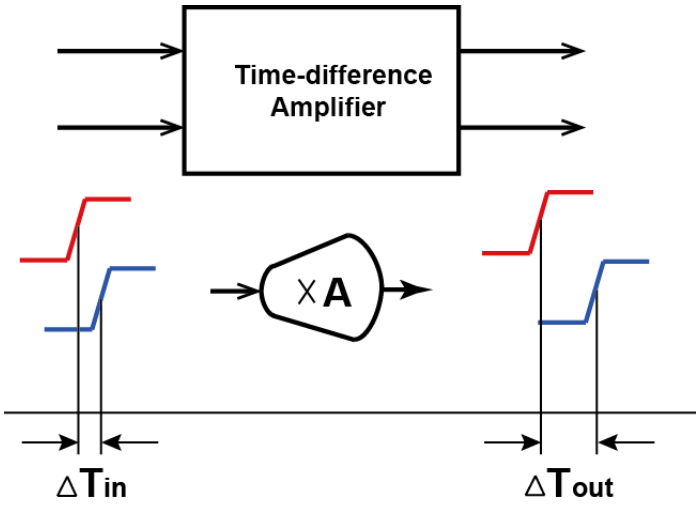


Figure 2.12: Function diagram of time-difference amplifier. The output of TD is multiplied by a constant value A of input TD.

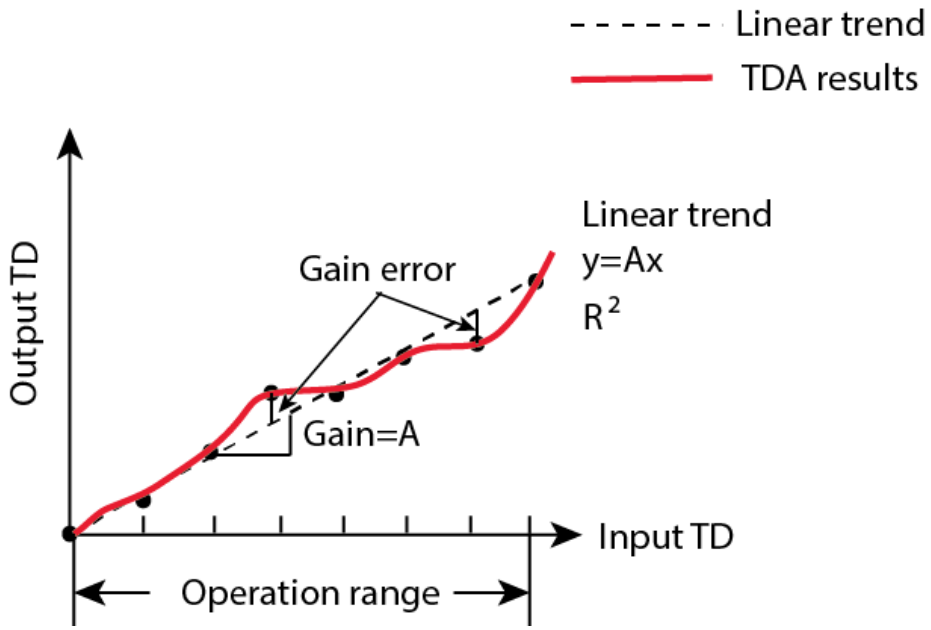


Figure 2.13: TDA works with specifications. The amplification gain is A. The operation range is determined by the maximum input TD. The variation between real TDA data and linear trend is the gain error.

$$A = \frac{\Delta T_{out}}{\Delta T_{in}} \tag{2.4}$$

TD amplifier is the key component in our proposed algorithmic TDC, which is also the main design challenge. Since it performs operation in analog-domain that amplifies the actual TD value, therefore, the characteristics of TDA determines the linearity and linear range, as well as the TDC conversion range and accuracy. A.M. Abas first came up the concept of TD amplifier [45], which is based on the exponential relationship between metastable time and input TD $T_{mst} = c * ln(\Delta T_{in})$, if two events are not far apart. A linear amplification range is obtained by taking the difference between two opposite offset MUTEX arbiter delays.

$$\Delta T_{out} = T_{right} - T_{left} = \tau * (log \frac{\Delta T + T_{offset}}{T_{offset} - \Delta T_{in}}) \tag{2.5}$$

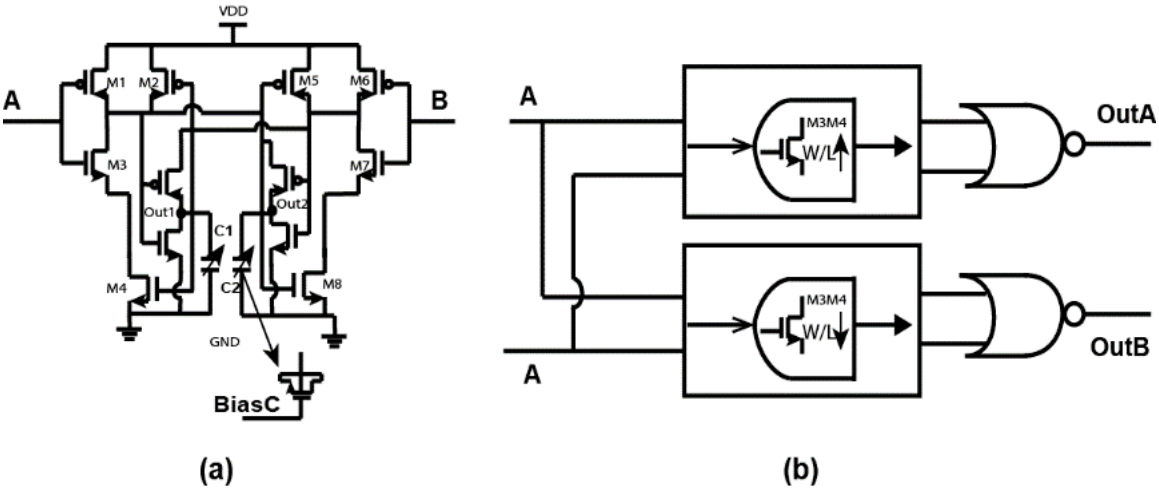


Figure 2.14: Schematic of developed arbiter-based TDA presents that the gain can be adjusted by BiasC. (a) schematic of arbiter with voltage-controlled MOScap as load capacitors of the NAND gate. (b)TDA schematic based on the arbiter design in (a) with different scaled size of transistor in NAND gates.

$$\Delta T_{out} = \tau * (log \frac{1 + \Delta T/T_{offset}}{1 - \Delta T/T_{offset}}) \tag{2.6}$$

The arbiter-based TDA performs the inverse hyperbolic tan function of input TD. The final TDA gain in the linear range is obtained with Eq. 2.7, where $\tau = \frac{C_{NAND}}{g_{m\_NAND}}$.

$$\begin{aligned} A &= \frac{\Delta(\Delta T_{out})}{\Delta(\Delta T_{in})} = \frac{\tau log(\frac{T_{off}+x1}{T_{off}+x1})}{x1} \\ &= \lim_{x1 \to 0} \frac{\tau log(\frac{T_{off}+x1}{T_{off}+x1})}{x1} = \frac{\tau log(\frac{T_{off}+x1}{T_{off}+x1})'}{x1'} \\ &= \frac{2\tau}{T_{off}} \end{aligned} \tag{2.7}$$

The schematic of MUTEX-arbiter based TDA is presented in Fig 2.14. The shifted plots in Fig 2.15 are realized by skewing the transistor size in the circuit. Increasing the W/L ratio of the NMOS transistors (positive skew) speeds up the transmission from high to low, making the pull-down network faster while decreasing NMOS W/L ratio (negative skew) slows down the gate transmitting speed.

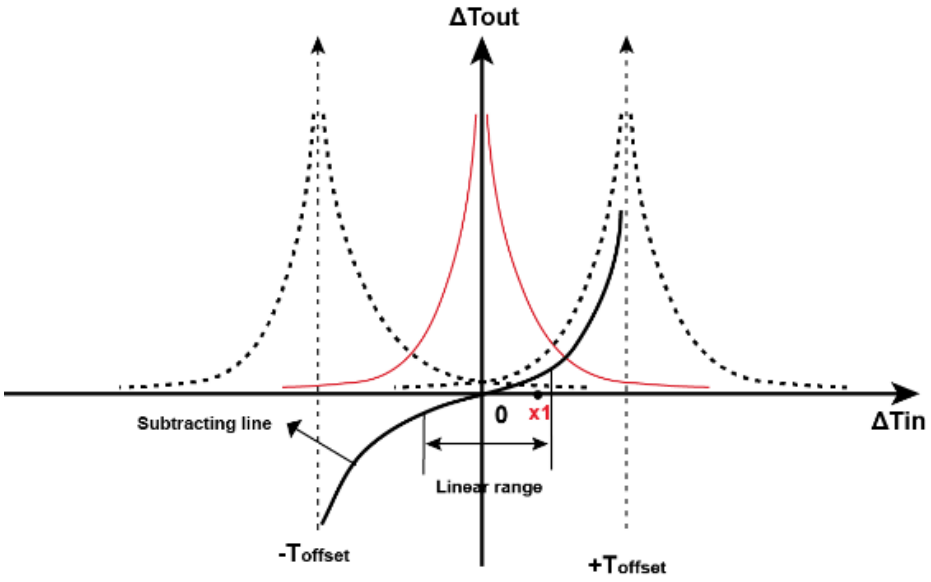


Figure 2.15: Two shifted version of exponential relationship of input and output time difference.

This TD amplifier design is small and of high-speed, which has been used in many TD circuit design. The weaknesses are that its linear range is very limited, moreover, in advanced CMOS technology, the resolution of metastability is reduced. Thus, the arbiter-based TD amplifier is not scalable with different CMOS technology nodes. Alternative TDA designs include switched delay line amplifier and pulse-stretched TDA. A switched delay-line amplifier is formed by creating two lines of switching delay elements that can shift the delay time between fast or slow, as shown in Fig 2.16. Input A enters the top line of switched delay elements on the left and propagates to the right. Input B enters the bottom line on the right and propagates to the left. These switches on the delay elements are attached to the other delay line.

In the pulse-stretched TDA topology, the circuit in Fig 2.17 performs amplification by first converting the time difference to a voltage. Fig 2.18 presents the detail amplification process of the pulse-stretches TDA, in which capacitor 1 is charged from the rising edge of input A until a time T_{offset} after the rising edge of input B to a voltage V_1; capacitor 2 is charged from the rising edge of input B until a time T_{offset}

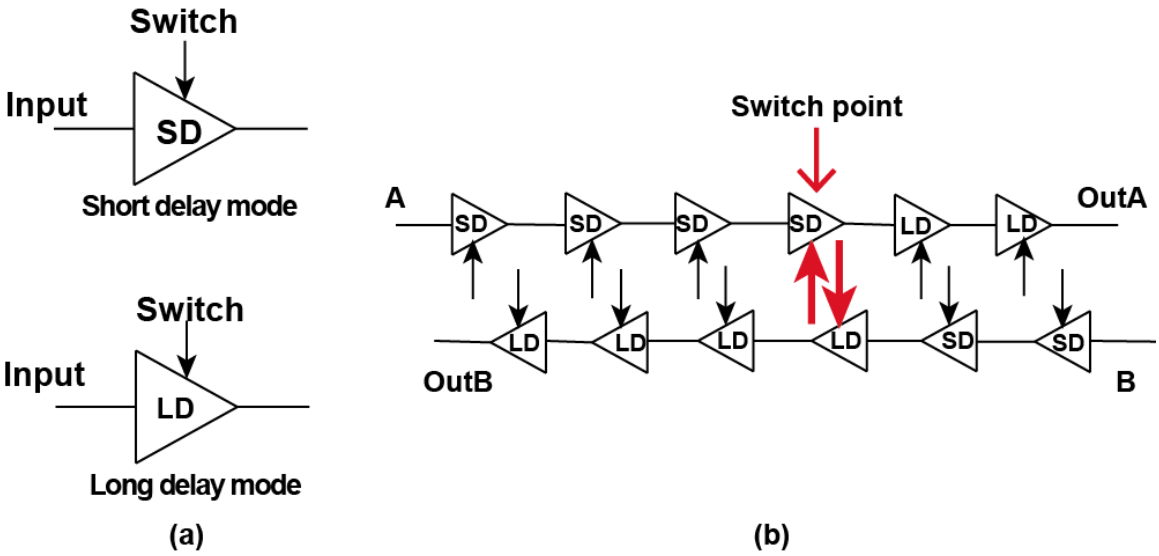


Figure 2.16: Implementation of TDA based on switched-delay element. (a) The delay element can be switched with two delay modes. (b) Delay line configuration.

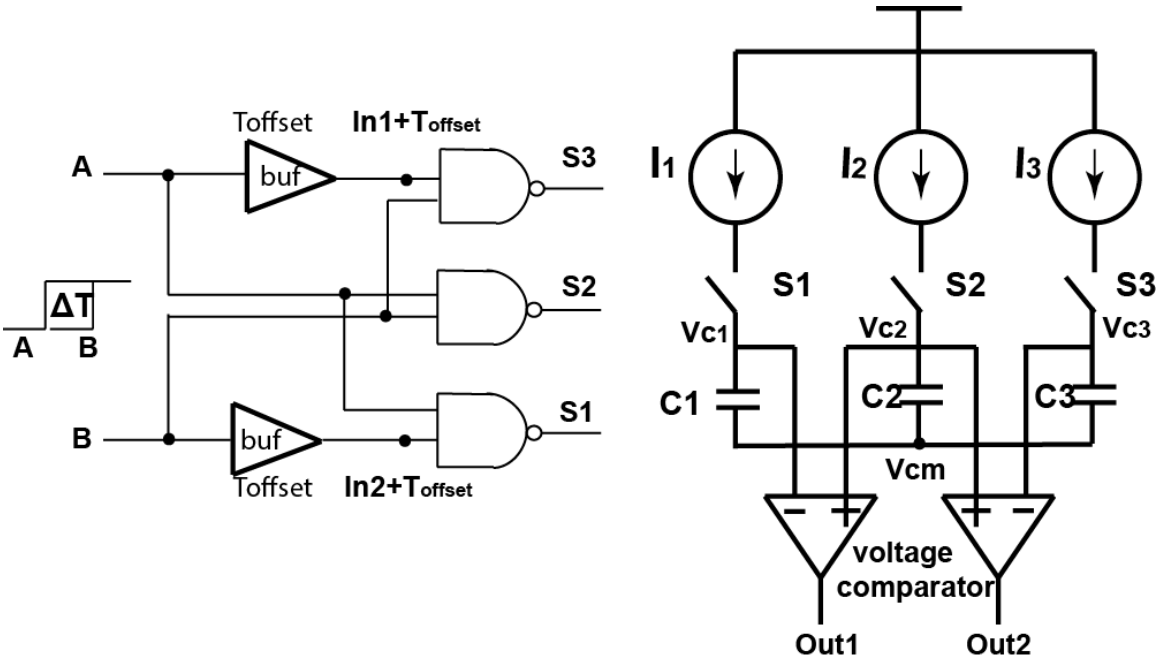


Figure 2.17: Implementation of pulse-stretched TDA.

after the rising edge on input A to a voltage V_2. A slower voltage ramp then converts the difference between these two voltages into a time difference.

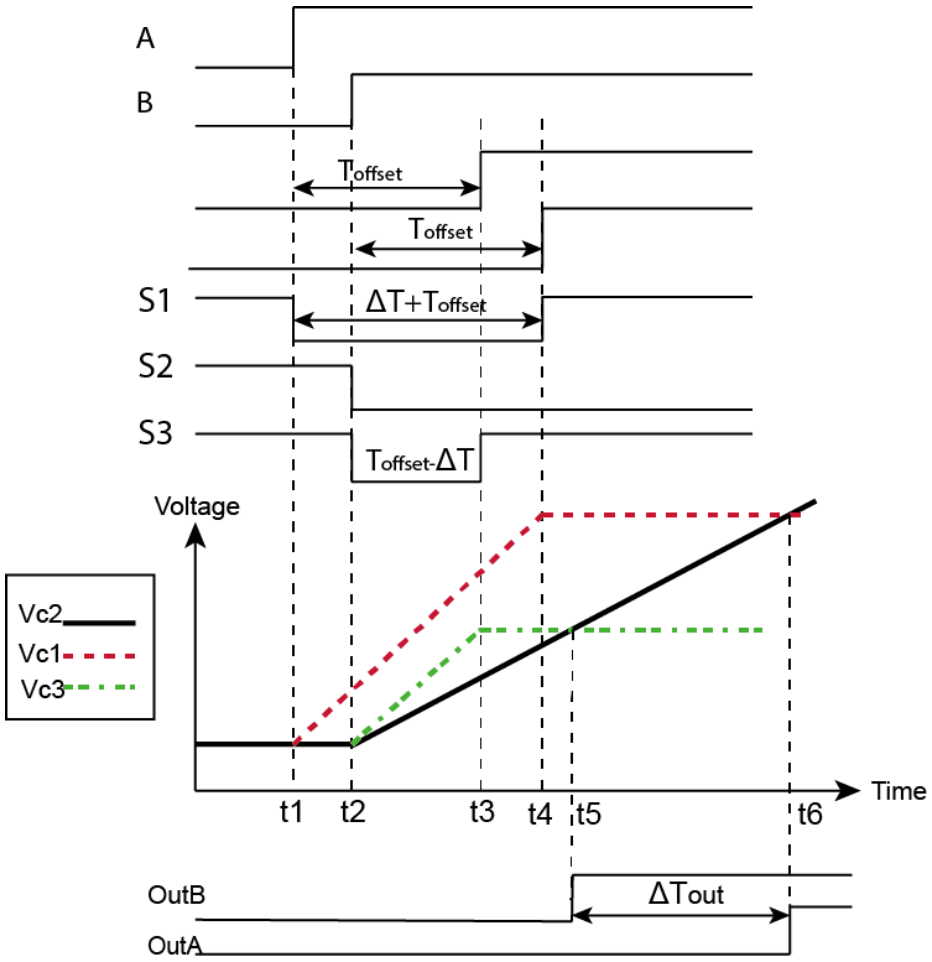


Figure 2.18: Timing diagram of pulse-stretched TDA. The input TD ΔT is amplified to ΔT_{out}.

The gain of the pulse-stretched TDA is controlled by the current sources and capacitances. Given the fixed capacitance charged with constant current sources, the pulse-stretched TDA can obtain a stable amplification gain. This design saves area by getting rid of the delay line structure, but at the cost of lower operation speed caused by the use of a voltage-to-time converter.

A novel TDA design is proposed in Chapter 5, which amplifies the TD by converting it to a single pulse and duplicating this pulse for N times, finally, the width summation of N pulses is used to obtain the gain of N. The pulse-based TD amplifier archives high linearity, and the linear range is no longer the bottleneck to limit the TDC

performance when scaling down to advanced CMOS technology. The specification of the application requirements drives the selection of different TD amplifiers between their trade-off.

CHAPTER 3

TIME-TO-DIGITAL CONVERTER ARCHITECTURES

With the discussion of fundamental TD signal processing and TD function modules, in this chapter, we focus on a comprehensive time difference circuit: time-to-digital converter (TDC), which has seen a rapid development in recent decades [40]. TDCs precisely digitize the time difference between the edges of two timing events, the function block of a TDC example is shown in Fig 3.1. A digital counter driven by a clock is an example of a coarse TDC, with its resolution limited by the clock cycle [38][39]. Many applications require higher-resolution time measurements, such as measuring the time-of-flight of particles in a nuclear science experiment [46][23], fluorescence imaging [47], LiDAR systems [48][5], or measuring the propagation delay in digital circuits [49]. These applications require mixed-signal TDCs that can offer sub-clock cycle resolution.

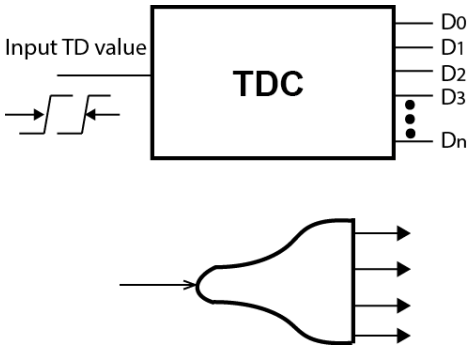


Figure 3.1: TDC is presented in the traditional schematic symbol, in which each wire is drawn as a line or as a time difference symbols. Digital symbols are drawn with filled arrow heads and a TD signal is drawn as a an arrow with narrow arrowhead.

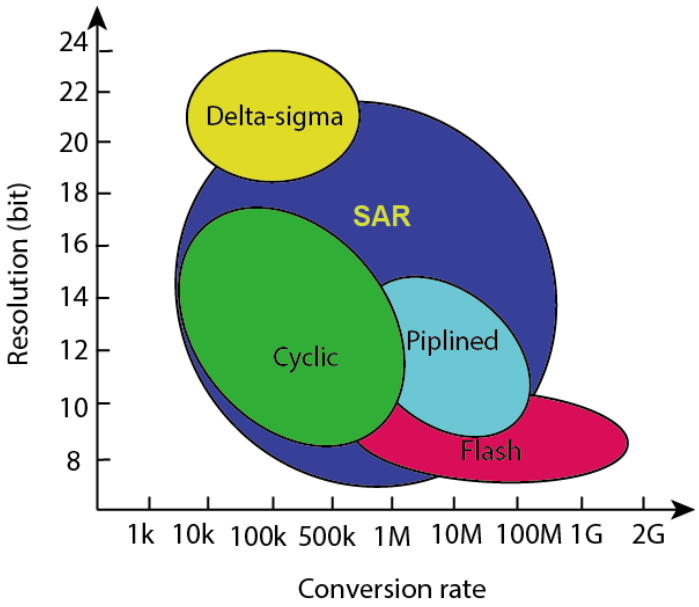


Figure 3.2: Tradeoffs of different ADC architectures in resolution and conversion rate. (ADC-survey from [1]).

This chapter first reviews several conventional TDC architectures, and then proposes a TDC based on the algorithmic design with performance simulation results. To study the architectures of TDCs, a similar guideline to the standard ADC topologies can be developed. For voltage-mode ADCs, the resolution and conversion rate determines the optimum structure among flash, successive approximation register (SAR), pipelined, algorithmic, also called cyclic, and delta-sigma, as shown in Fig 3.2.

3.1 Flash TDC

Flash ADC is arguably the most straightforward design for analog-to-digital conversion. A flash ADC is also called parallel converter because the input signal in a flash converter is fed to $(2^N - 1)$ comparators in parallel to perform an N-bit conversion. As shown in Fig. 3-3 (a), each comparator uses a different reference voltage, hence, the comparators create a thermometer output that is encoded into an N-bit digital output:

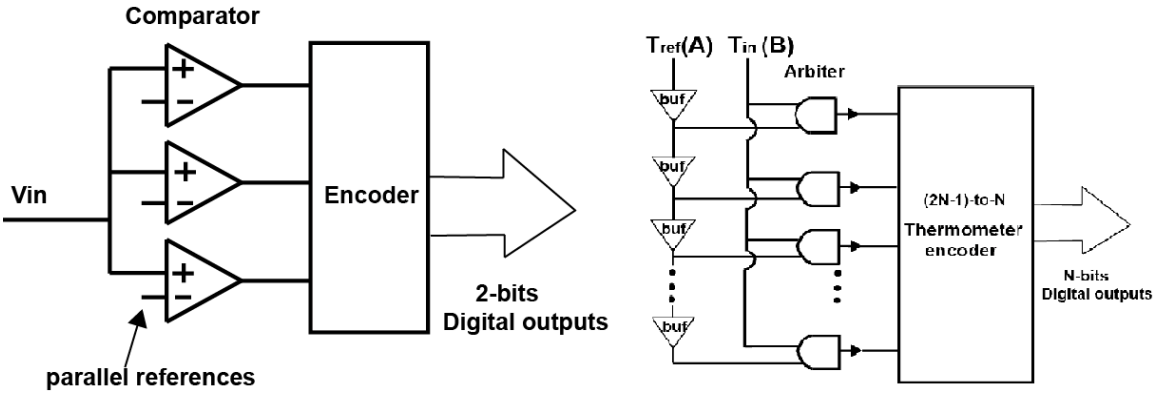


Figure 3.3: Diagram of flash converter. Left: A general 2-bits flash analog to digital converter is consisted of 3 comparators and voltage references, encoding the compared results by a thermometer encoder to 4 digital outputs. Right: N-bits flash TDC function similar with flash ADC replacing the voltage comparator to TD arbiter.

$$(x-1)\cdot T_{buf} \leq T_m \leq x\cdot T_{buf} \tag{3.1}$$

$$x = \sum_{m=0}^{N-1}(2^{D[m]\cdot D[m]}) \tag{3.2}$$

$$x-1 = \sum_{m=0}^{N-1}((2^{D[m]-1)\cdot D[m]}) \tag{3.3}$$

Flash TDCs use delay buffers to generate time differences. An arbiter compares the arrival time of each delayed output with the input signal. When the input signal overpasses the delayed signal due to the delay of multiple-stage buffer, the output of arbiter switches. As in a flash ADC, the arbiter outputs create a thermometer code. An encoder converts this code into an N-bit digital output, as shown in Fig 3.3(right figure). The measured time difference between T_{ref} and T_{in} is represented with N-bit digital output.

For flash TDCs, the resolution is determined by reference buffer delay time T_{buf} and the number of stages N, these two parameters jointly determines the operational range TDCs. The vernier delay-line (VDL) technique, shown in Fig. 3.4 is used to

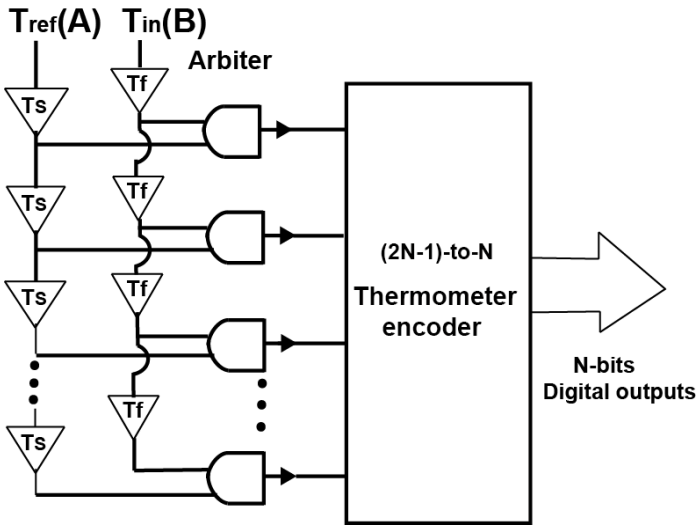


Figure 3.4: N-bit Vernier flash TDC is implemented with two delay lines with short single delay time T_f and long delay time T_s.

improve the resolution of flash TDC beyond the minimum gate delay [50] [51] [52]. Two delay lines are built from delay elements with different delay values to form the vernier structure. At each stage, the input signal catches up to the reference by $(T_s - T_f)$.

Since flash TDCs complete multi-bit conversion in parallel with simple delay elements, they have been used as the standard approach for realizing high-speed converters. However, it should be noted that flash TDCs suffer from a large area and high power consumptions due to a large number of comparators and delay buffers chain.

3.2 Coarse-fine interpolation TDC

Interpolation TDC is composed of two different design schemes: counter-based TDC and flash TDC. Using a simple counter that is clocked with the reference clock gives a coarse measurement for the time difference: T_m.

$$T_m = C \cdot \tau_{ref} \tag{3.4}$$

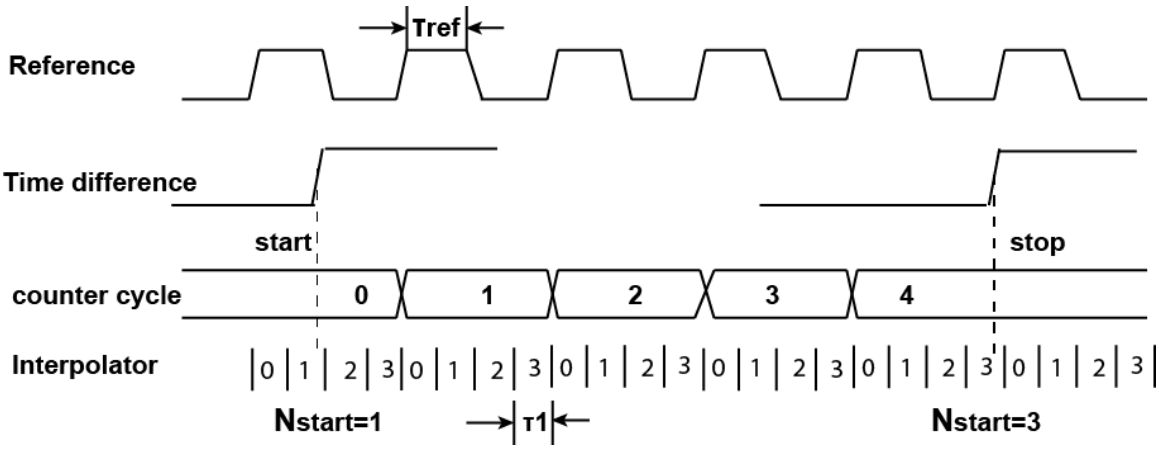


Figure 3.5: TDC function diagram with a counter and interpolation.

The reference clock period τ_{ref} determines the resolution of counter-based TDCs. Delay-line interpolation technique provides a way to locate the timing signal within the reference clock cycle. By splitting one reference clock period into multiple shorter cycle τ_1, as shown in Fig 3.5, the final result is composed of both counter and interpolating results:

$$T_m = C \cdot \tau_{ref} + (N_{stop} - N_{start}) \cdot \tau_1 \tag{3.5}$$

Multi-level interpolation further improves the resolution by performing higher-level interpolation within the results of lower-level interpolation [53]. A two-stage TDC is similar to a two-stage ADC, which realizes the first coarse conversion and amplifies the residue for the fine conversion state [54]. The two-stage TDC uses the counter-based coarse measurement for each conversion stage, and amplification stage is based on the switched-delay line TDA without feedback-loop applied for cyclic conversion.

3.3 SAR TDC

SAR converters use less power and area than flash converters because they perform a "binary search" algorithm in conversion. In SAR ADC, the input value is first compared to $V_{ref}/2$, if the input is greater than $V_{ref}/2$, it is next compared with

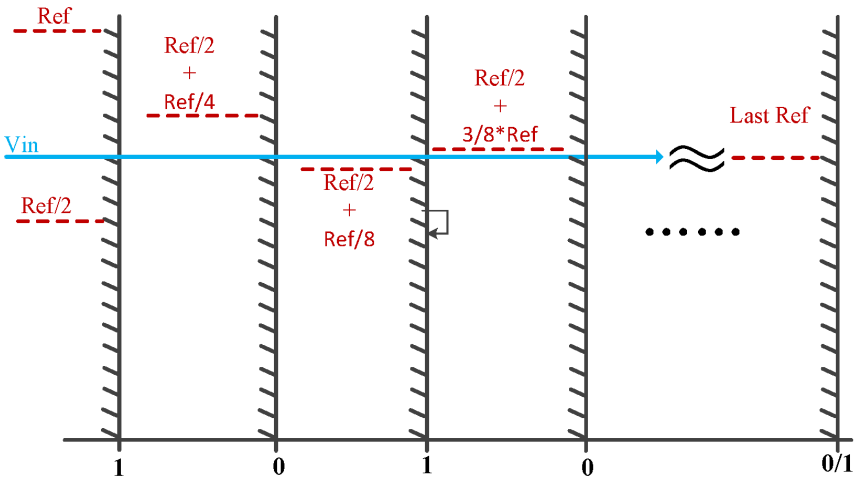


Figure 3.6: 3-bit successive approximation conversion diagrams. The input signal value is compared to the reference dependent on the previous conversion results.

$V_{ref}/2 + V_{ref}/4$; otherwise, the input value is compared with $V_{ref}/4$. This process is repeated until all the required bits are converted. The schematic of a three-bit SAR conversion is shown in Fig 3.6.

The SAR TDC is implemented based on the same algorithm, in which a time difference $\Delta T = T_2 - T_1$ is successively approximated with TD reference values that are produced by a DTC [43][44]. The digital code is adjusted until the generated TD reference is equal to the input time difference. To improve the resolution and conversion rate, a high-speed, high-resolution DTC is required. In a recent work presenting a SAR TDC implementation, a structure of adjustable delay elements with load capacitors composed of a 128 programmable MOS-capacitor matrix was used in [55]. However, the large area makes this TDC implementation infeasible and it is also power hungry in operation.

SAR TDC is theoretically more efficient than flash TDC because only N comparisons are required to perform an N-bit conversion. However, it is difficult to reuse circuits for multiple cycles because each time event only happens once. Therefore, N TD comparators (arbiters) and N DTCs are needed to perform an N-bit conversion [43][44].

3.4 Proposed Compact Algorithmic TDC

Although flash TDCs are easy to implement and can generate highly linear digitized results, they are large, slow, and power hungry [49]. Flash TDCs are large because an N-bit conversion requires 2^N delay cells. The conversion is slow for high-resolution because the signal has to pass through all of the 2^N delay cells. Similarly, high power consumption is incurred by the large number of switching delay elements.

Attempts to avoid the limitations of flash TDCs have been made by transforming other ADC topologies into TDC, for example, successive approximation [55], sub-ranging [49] and algorithmic [56] TDCs have all been presented. However, these design schemes either a require a digital-to-time Converter (DTC) or a ring-oscillator (RO) to generate unfixed time references for digitizing the unknown time differences, which consumes extra power and overhead [43][44]. The delay-line structure used in building DTC or RO significantly limits the performance of these architectures. The compact algorithmic TDC (CATDC) presented int his section eliminates these components by using a single delay and a time-domain amplifier (TDA) with a gain of 2.

3.4.1 CATDC: architecture and implementation

The CATDC performs a one-bit conversion each time, as shown in Fig 3.7. For each bit, the "start" signal is delayed by a fixed reference T_{ref}, the arbiter then compares this delayed signal to the "reference". The arbitration result "high" (1) or "low" (0) is one bit of the conversion. This CATDC structure also enables selection between the delayed signal and the original "start" signal for next operation: amplified by 2 in a time-difference amplifier. Then the amplified signal is processed in the same loop to calculate the next bit. The n-bit conversion of the measured time difference is: $T_{ref}(\sum \frac{1}{A^{n-1}})$, where T_{ref} is the time reference, and A is the amplification gain (ideal value of which is 2). This design makes it feasible that time differences are signals like voltages that can be moved around in a circuit from one node to another node.

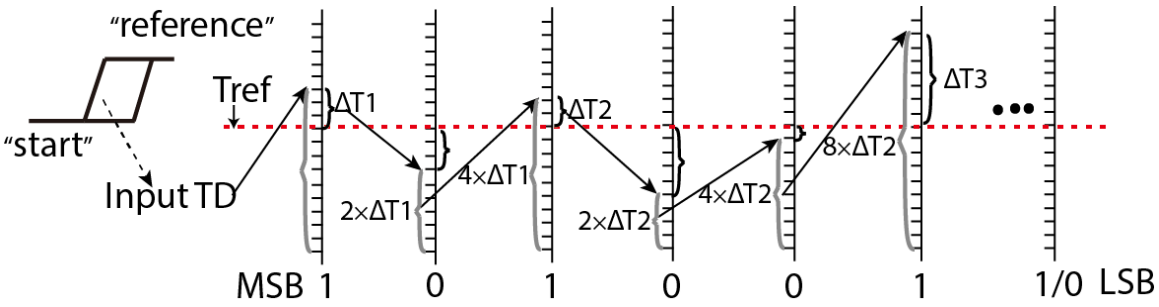


Figure 3.7: Multi-bit function diagram of proposed CATDC with a fixed reference value; for each comparison the time residue is further processed for the next loop.

This binary search algorithm is not possible for a TDC since the time difference is not available for two events in such a sequence [40]. The challenge of designing TDCs is that the time differences are continuously propagating signals that must be routed around the circuit. Therefore, the algorithm has to be modified by switching the operating sequence.

In the revised algorithm, two residues are pre-calculated based on the possible arbiter outputs, the flow diagram is as shown in Fig 3.8. The implementation of the algorithm shown in Fig 3.9 demonstrates the necessary changes that are required. First, both signals with and without delay are amplified to introduce a sufficient delay to perform the arbitration. Second, the arbitration result is re-initialized when the current input pulse ends. However, the select signals for the MUXs must keep effective until the amplification outputs reach the MUXs. To fulfill this job, a simple set-reset (SR) latch built with cross-coupled NAND gates is applied here after the comparator to conserve the useful outputs.

The proposed TDC is composed of a delay element, two amplifiers, an arbiter, four variable delay buffers, and four multiplexers. The transistor level schematics of these main blocks are shown in Fig 3.10. The delay element is built with a pair of current-starved buffers, shown in Fig 3.10a. The value of this delay sets the most

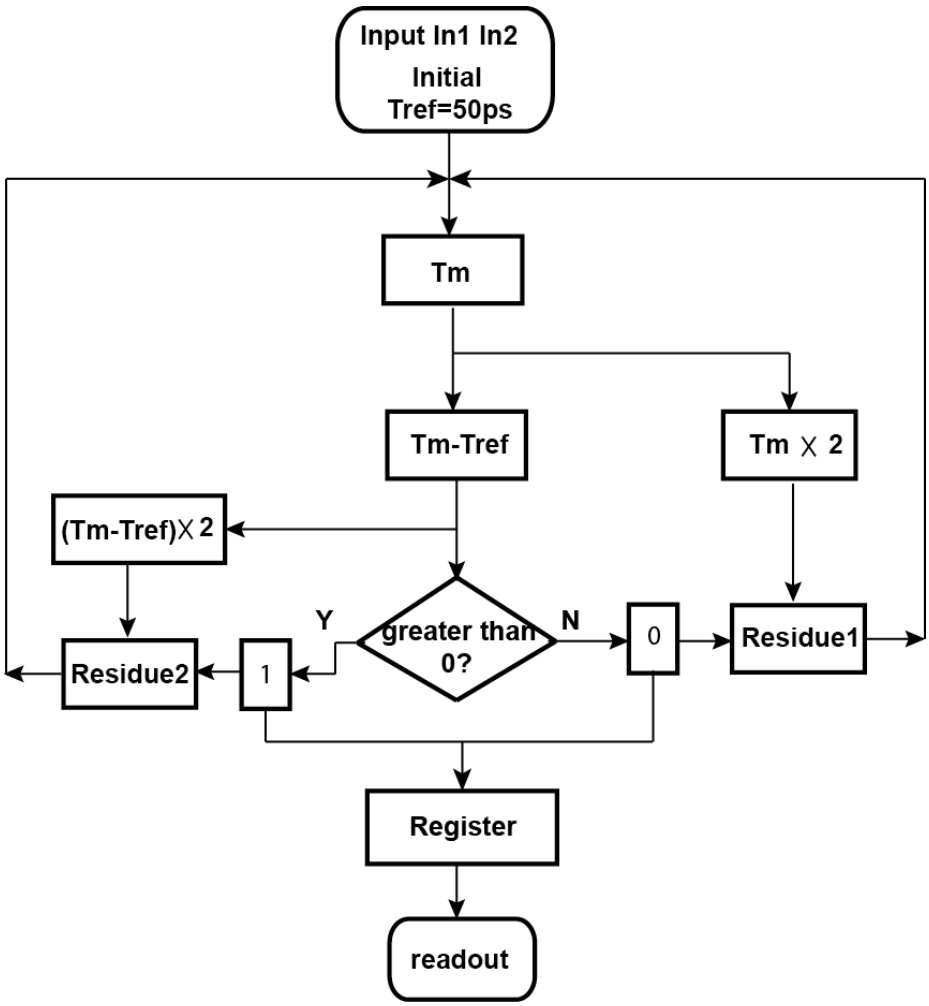


Figure 3.8: The algorithm needs to be revised for time-mode conversion by splitting the amplification into two parallel paths. The algorithmic TDC conversion flow diagram is shown here.

significant bit of the conversion. The multiplexers (MUXs) are route one of two input signals to the output, as shown in Fig 3.10.

The arbiter showed in Fig. 3.10c is built on two cross-coupled NAND gates. Initially, inputs A and B are both low and driving the output low. The first rising input between these two will cause the output of its NAND gate to low and lock the other NAND gate high. For example, if the rising edge of pulse A comes earlier than B, then X=1 and Y=0; otherwise, X=0 and Y=1.

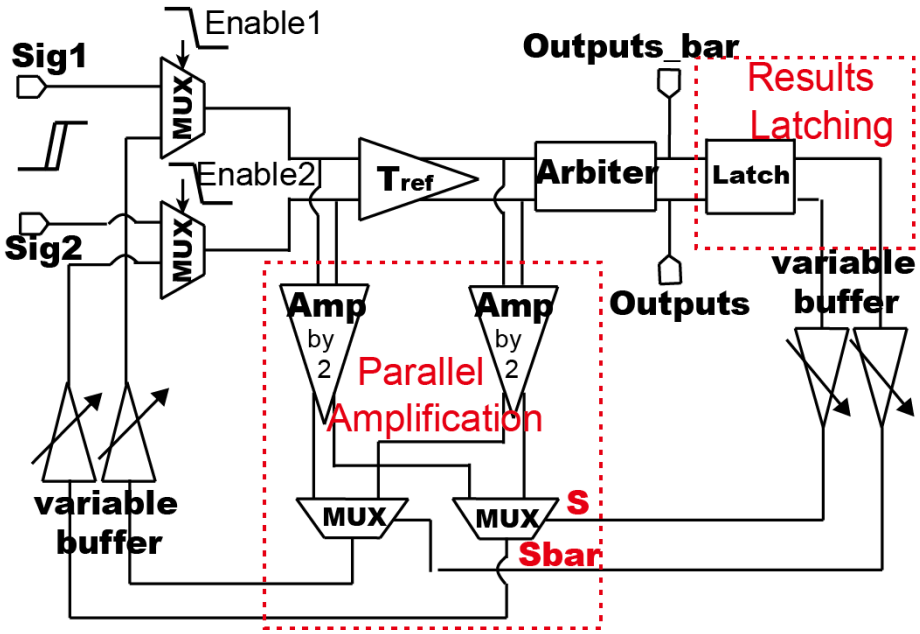


Figure 3.9: The full implementation of proposed CATDC. The variable buffers are applied for compensating the variation along the delay paths due to PVT.

The time-difference amplifier shown in Fig. 3.10d uses two arbiters to perform the amplification [45]. This amplifier works because the propagation time of these arbiters is dependent on the time difference between the arrival edges of the two input signals. The required decision time will exponentially increase as the input time difference decreases, as plotted by the solid black line in Fig. 3.11. By adjusting the widths of the pull-down transistors in the two NAND gates that make up an arbiter, this exponential curve can be shifted left or right along the time-difference axis, as shown with the dashed and gray curves in Fig. 3.11. An amplifier is formed by taking the time difference between two arbiters with opposite shifts. The transistor sizes presented in Fig. 3.10d are used to set the gain to be 2 for the CATDC, thus one bit is computed in each conversion cycle. A number variable delay buffers are introduced along the path to match the timing condition and fine calibration of the variations on the propagation delay.

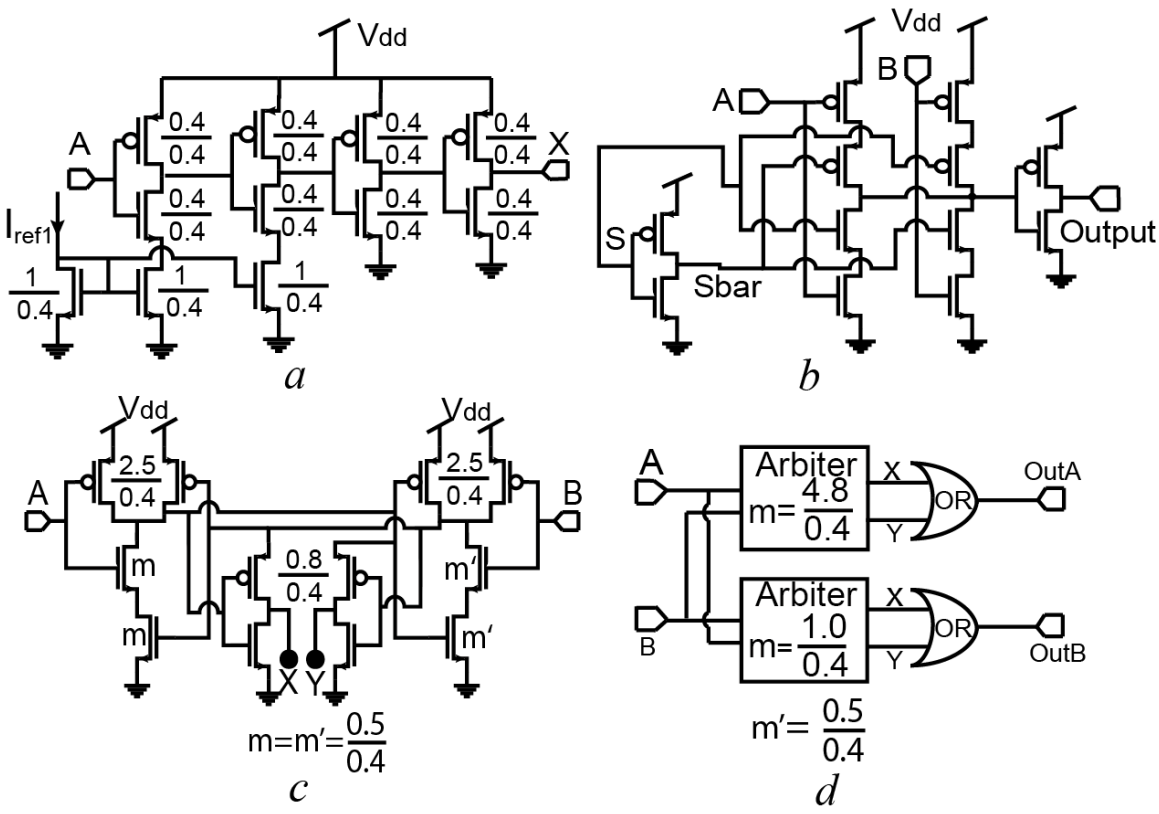


Figure 3.10: Transistor level schematics of main building blocks of CATDC. a. Current-starved variable delay buffers. b. Multiplexer (MUX). c. Arbiter-based Time-Difference Amplifier. d. Balanced MUTEX-based Arbiter.

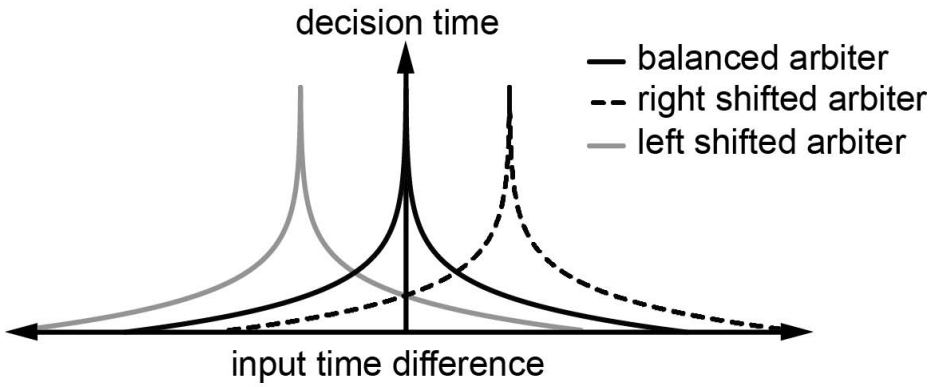


Figure 3.11: TDA function diagram.

3.4.2 Simulations results and analysis

The proposed CATDC was designed and fabricated with the TSMC 0.35μm high-voltage process with a total area of 200μm200μm. Before fabrication, the full TDC

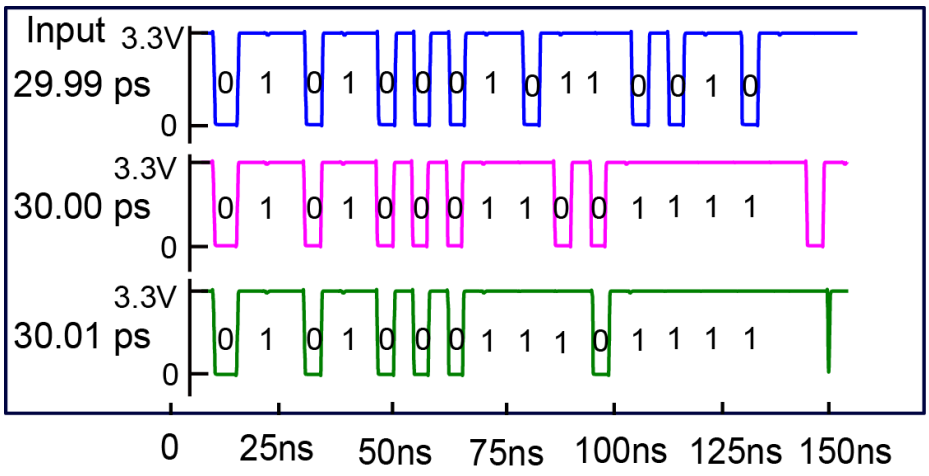


Figure 3.12: CATDC transient simulation results with 0.01ps resolution.

design was simulated with Cadence Spectre using a netlist extraction with parasitic components from the physical layout. The transient simulation results for different input time difference demonstrate the conversion function of CATDC, as shown in Fig 3.12. The corresponding 15-bits digital codes generated within 150ns are labeled on the plots, respectively. The timing resolution keeps increasing by adding the conversion cycle to get more bits. Within the simulation, a very high resolution 0.01-picosecond can be obtained.

To calculate the linearity and accuracy of the proposed TDC, we swept the input time difference from 0ps to 100ps ($2T_{ref}$), with a 0.2ps step, we observed that the digital codes linearly increasing with the input time difference. The sweep simulation results are presented in Fig. 3.13(a). The differential nonlinearity was calculated by converting the digital code back to time using a line of best fit and taking the difference between each adjacent value. The DNL showed in Fig. 3.13(b) demonstrates a maximum nonlinearity of -0.37ps over a range from 0 to 100ps corresponding to slightly less than 1 LSB at 8 bits. The integral nonlinearity was calculated by subtracting the line of best fit from the input-output relation. The INL showed in Fig. 3.13(c) shows a maximum error of 2.5ps or 6.4 LSB for 8 bits, due to the mismatch

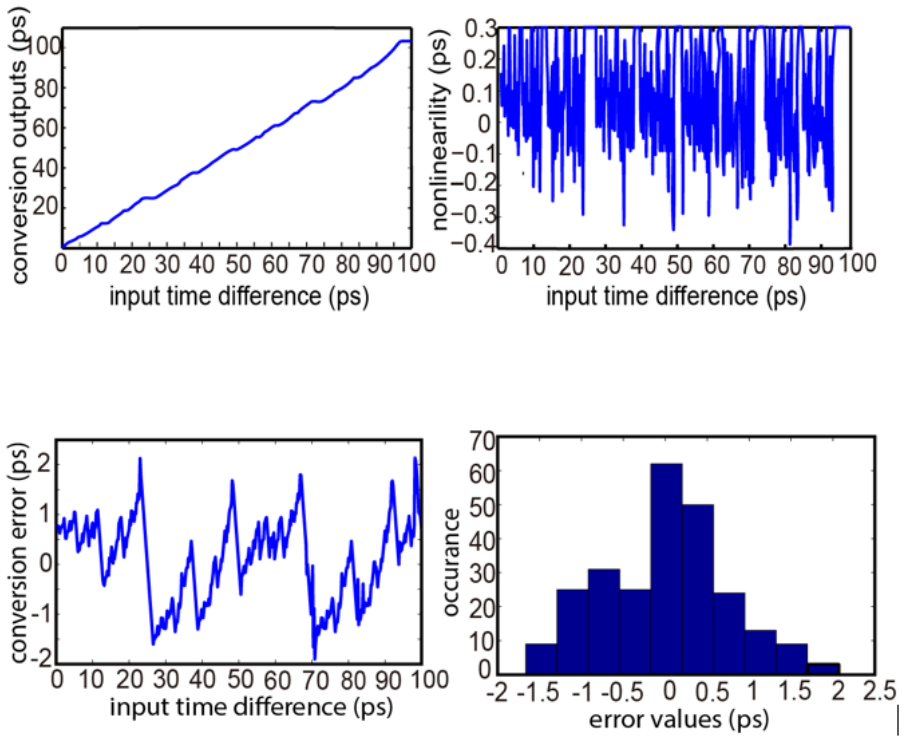


Figure 3.13: CATDC simulation results and analysis. a. Input time difference sweep from 0 to 100ps. b. Differential non-linearity presented in picosecond. c. Integral non-linearity, conversion error presented in picosecond. d. Error distribution.

between the practical gain of TDA and the ideal gain=2. The errors approximately follow the normal distribution, as shown in Fig. 3.13(d).

3.4.3 Fabricated chip experimental results

The die photo of the fabricated chip is shown in Fig. 3.14. The entire chip size is 2mm2mm containing the proposed CATDC structure labeled with a red rectangular box. The fabricated chip was tested using a custom printed circuit board that is connected to precisely delayed input signals using the Micrel, Inc, SY89297 digital delay lines, which have a resolution of 5ps. The digital output strings of the TDC were collected with Tektronix MSO4104 oscilloscope. The operation of this loop has been demonstrated with the four waveforms shown in Fig. 3.15. The measured power is 0.7mW for 100ns/bit. The TDC achieved the closed-loop and function

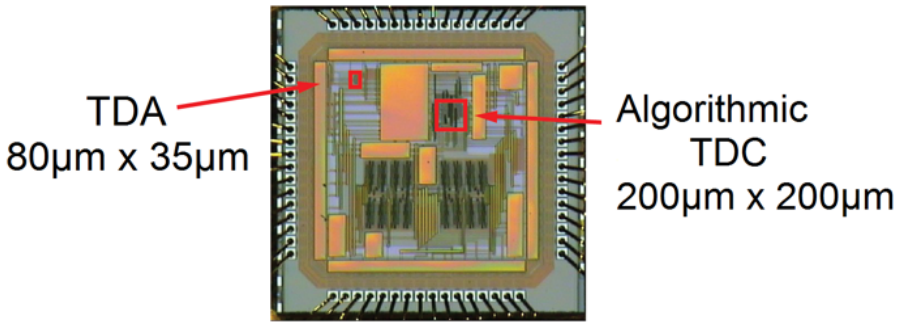


Figure 3.14: The die photo of test chip, in which the overhead of CATDC and TDA are labeled.

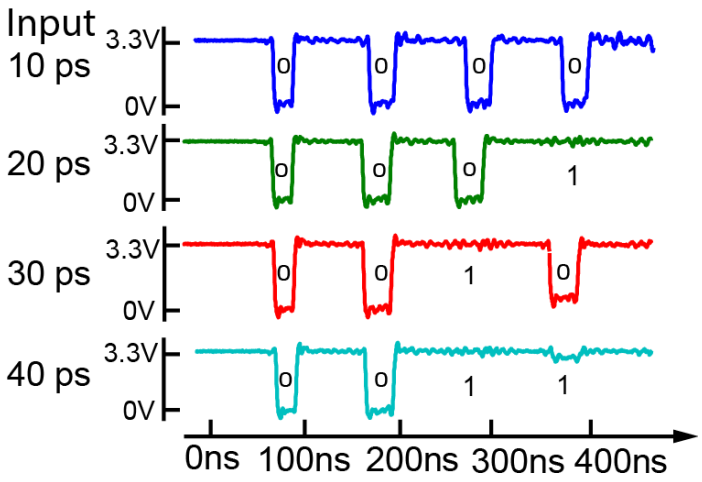


Figure 3.15: Experimental test results of CATDC for 10ps, 20ps, 30ps, and 40ps input time differences.

appropriately, and these test results demonstrate the practical feasibility of the CATDC implementation.

In this chapter, a new algorithmic TDC achieving sub-picosecond resolution is proposed. This new design significantly reduces the overhead and power consumption of of conventional TDCs, thus greatly broadening the applicability of them. To further validate the practical feasibility of this method, a test chip was fabricated and tested. The test chip demonstrated the essential operation of this novel design.

CHAPTER 4

CCATDC: A CONFIGURABLE COMPACT ALGORITHMIC TIME-TO-DIGITAL CONVERTER

High-resolution time-to-digital converters (TDCs) are important timing measurement circuits discussed in the previous chapters. The proposed Algorithmic TDC achieved promising performance with high conversion rate, low power, and compact structure. However, there are still problems need to be solved due to industrial fabrication issues, and practical applications requirements. The development of semiconductor technology not only favors the resolution improvement of TDCs but also introduces negative impact like process variations. Since modern TDCs are designed to achieve resolution of picosecond magnitude, process variations become a big concern. In this chapter, we present a configurable compact algorithmic TDC (CCATDC) to improve the functionality and performance of the CATDC work. To mimic the real-time circuit calibration and measurement in an industrial scenario, a Backpropagation-based Machine Learning framework is proposed. To overcome the mismatch between two signal channels and reduce overhead, we propose a bidirectional digitization method that is flexible to any input order. This method also increases the throughput by 50%. Experimental results demonstrate that our proposed CCATDC can achieve high resolution (<1ps). Comparison with other TDC structures shows that our proposed CCATDC consumes 75.4% less energy and 60% overhead than other design schemes.

4.1 Introduction

High-resolution time measurement is a common need in today's science and engineering applications, such as time-of-flight measurement in remote sensing [57][23]

, nuclear science [58] [46], biomedical imaging [59], and frequency synthesizer and time jitter measurement for RF transceiver in wireless communication system [3][4][38], and time jitter measurement for the all-digital phase-locked loop (PLL)/delay-locked loop (DLL) [18]. To favor the post-processing of different time signals, time-to-digital converter (TDC) is proposed, which bridges various applications and electronic devices. As a high-resolution time measurement system, TDC is able to digitize the time interval between two events. Different schemes have been proposed to build TDCs, such as the cyclic successive-approximation (SAR) based TDC, pulse interpolation based TDC, and delta-sigma based TDCs [55][60][61].

Delay line is a commonly used element to build TDCs, in which the delay length of each delay element is designed to be the same. While two timing signals "start" and "reference" are applied onto the TDC, the "start" signal will be postponed by each delay element and compared with the "reference" signal to derive a binary output. The delay-line based TDCs were firstly proposed to measure the on-chip time jitter [3][4][62][63][20]. The highest resolution of a delay-line based TDC is the delay of every single element, which is limited by the fabrication technology node. To improve the resolution and enhance the accuracy of such TDCs, several techniques have been proposed to leverage the sub-gate delay length. The well-known technologies include stretching pulse [64], using tapped delay line [65], and utilizing differential delay line [51]. However, though many variants of delay line based TDCs have been proposed in these years, it is still a challenge to achieve better than sub-10 ps resolution [40].

More recently, a newly compact algorithmic TDC (CATDC) is proposed in [2]. The CATDC is implemented in a compact structure in which a self-clocked loop methodology is utilized, and this scheme greatly reduces the overhead and power consumption of TDCs. However, one problem with CATDC (or maybe the common problem with other TDC schemes) is that the user has to set up the connection of the "start" and "reference" signals. In other words, the user has to know which signal

comes earlier than the other and then fed them to corresponding channels; otherwise, the converted output will be wrong. However, in practical applications, it is hard to differentiate the "start" signal from the "reference" one, due to the small time interval between them.

With the development of semiconductor technology, it is more difficult to control the fabrication process at small-feature technologies. In this scenario, the deviation of process, voltage, and temperature values from nominal specifications becomes a big concern [66] [67]. This phenomenon directly impacts the performance of TDCs; for example, the delay length of each delay elements is not equal with each other anymore, thus decreasing the overall accuracy [48]. Several techniques have been proposed to tackle the negative impact from process variation on some specific application, for example, Yousif *et al.* proposed to use the full characterization method to erase the process and temperature variations in positron emission tomography scanner [59].

To address the above-stated problems, we adhere to the design philosophy of CATDC in this work. More specifically, we propose a configurable compact algorithmic TDC (CCATDC) design. Our methodology is based on the truth that once a chip is fabricated, it becomes difficult (or even impossible) to remove process variations, especially for the mix-signal processing circuit. In this scenario, we adopt a reconfigurable design scheme in TDC realization and use Machine Learning technique to manipulate/configure the microscopic process variations of the delay chain. The rest of this chapter is organized as follows: section 4.2 reviews some related work about TDC design. Section4.3 presents a statistical analysis of the conversion error from different components in CATDC scheme. Section4.4 proposes the configurable TDC design methodology and presents each block with details. Section4.5 shows experimental results. Section 4.6 concludes this chapter and presents some future work.

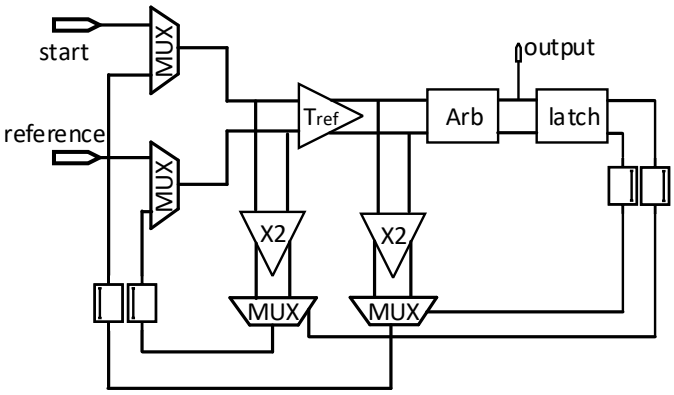
4.2 Related Work

Most conventional TDCs are realized with delay chains, such TDCs are classified as flash TDC, since the scheme of these TDCs is similar to that of flash analog-to-digital converters (ADCs), which are built with a series of comparators. In flash TDCs, the target signals: "start" is delayed by delay-element. Theoretically, each delay element will postpone the "start" signal by a constant time scale, and the output of each component is compared with the "reference" signal to generate a binary output.

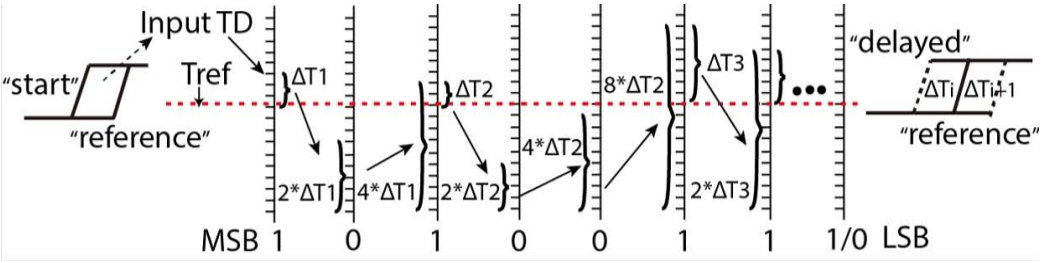
Though the usage of delay chain helps with digitizing the time difference, its resolution is limited by the minimum delay that can be achieved by each single delay element. To achieve better resolution, vernier technique is proposed, in which the "start" and "reference" signal are both delayed at each stage, but by different time length [51]. Though the implementation of flash TDCs is straightforward, and the outputs are highly linear, a big concern with this technique is the high overhead and power consumption since higher accuracy requires more delay elements [68]. And moreover, the frequency switching also limits the operation speed and power efficiency of flash TDCs.

Several works have been proposed to address the problems of flash TDCs; the most well-known technique is leveraging the ADC topology to build TDCs. Commonly used technologies include successive approximation [55], sub-ranging [49]. However, all of these techniques are either realized with digital-to-time converters (DTCs) or ring-oscillators (ROs) to generate adjustable time reference while digitizing the input time difference [43][44]. Hence, the power consumption and area overhead is still a big problem in these TDC implementations.

To address resolution problem in flash TDCs as well as reducing overhead and power consumption, Li *el al.* proposed a compact algorithmic TDC (CATDC) in [2], the schematic of which is shown in Fig. 4.1. Different from conventional TDCs, the delay constant elements T_{ref} is reused in each time-to-digital conversion round. More



(a) Schematic of CATDC



(b) An example time to digital conversion flow

Figure 4.1: Schematic of a time-to-digital converter design in [2]. T_{ref} stands for the reference signal, the time difference TD between "start" and "reference" signals are iteratively compared with T_{ref} to get each conversion bit. If the TD is larger than T_{ref}, then the output is 1, otherwise the output is 0 and TD will be amplified by 2 for next comparison. The time can be rebuilt with the binary output as $\sum_0^{n-1} o_i \times \frac{T_{ref}}{2^i}$.

specifically, a reference T_{ref} is employed to postpone the "start" signal, a constant gain=2 is utilized to amplify (double) the time difference (TD) between "start" and "reference" if it is smaller than T_{ref}, as shown in Fig. 4.1b. An arbiter element is used to compare this delayed "start" signal with the "reference" and output a binary bit (1 if larger than T_{ref} or 0 if smaller than T_{ref}). The time under test can be rebuilt with the binary output as $\sum_0^{n-1} o_i \times \frac{T_{ref}}{2^i}$, where o_i is the binary bit from the i^{th} conversion round, as shown in Fig. 4.1b.

4.3 Conversion Error Analysis

In this section, we analyze the main causes of the time digitization error in CATDC. As stated in section 4.1 and 4.2, the development of semiconductor technology not only advance the performance of electronic design but also brings a challenge [40]. This rule also applies to the realization of TDC development, smaller delay elements can abe fabricated with an advanced technology node, but the process variation also exacerbates the inequality between different delay elements. Before analyzing the primary source of conversion errors in CATDC, lets first go through the operation mechanism.

4.3.1 Scheme of CATDC

The main difference between CATDC and conventional TDCs lies in their operation scheme, most popular TDCs leverage chains of delay elements to compensate the difference between the "start" and "reference" signals, thus usually a long chain is needed to realize high resolution. In CATDC, the two signals "start" and "reference" are fed as the inputs, as shown in Fig. 4.1, once the two signals are inserted into the CATDC, the oscillation channel of MUXes will be enabled, and their connection with inputs will be turned off. In the first round, the "start" signal will be directly compared with the "reference" signal, the difference (residue) between them will be amplified by the *amp* block and the ideal gain is 2. In each of the subsequent conversion rounds, the *amp* block is reused to double the time residue if it is smaller than T_{ref}, and the newly derived "start" signal will be compared with the "reference" again and again until all the required conversion bits have been generated.

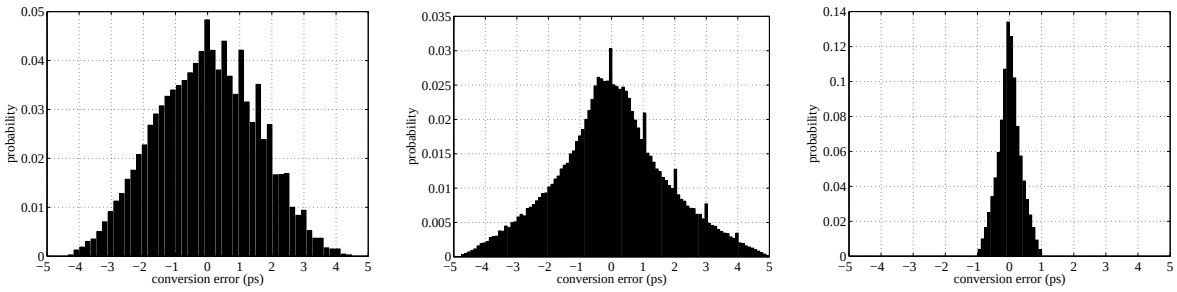
4.3.2 Error Analysis

As shown in Fig. 4.1, the main blocks of CATDC include a constant time reference T_{ref}, a time amplifier (*amp*), various MUXs and buffers. Any of these components in this structure can introduce conversion errors. Our objective is to find the dominant

factor and propose corresponding solutions. To analyze the effect of each block, the three main factors are simulated separately as follows; the results are obtained from circuit simulation using Synopsys HSPICE, version K-2015.06-SP1-3. The transistor and interconnect models used are from the open-source Predictive Technology Model (PTM). More specifically, the transistor models are PTM models for a 45nm process [69].

4.3.2.1 Gain Error

In CATDC design, amplification component is the fundamental block that accomplishes the digitization job; thus if the gain differs from the ideal value 2, there will be conversion errors. To analyze the conversion errors caused by inaccurate gain values, we assign a 5% variation on the gain, that is, from 1.9 to 2.1. The simulation results can be found as in Fig. 4.2a, the maximum conversion deviation is around ±4.5 ps. Note that for the sake of generality, various input values from 10 ps to 100ps are tried in all the simulations.



(a) Conversion error caused by amplification inaccuracy. (b) Conversion error caused by T_{ref} inaccuracy. (c) Conversion error caused by process variation and transient noise.

Figure 4.2: The simulated conversion error caused by different factors; it can be seen that the inaccurate amplification and time reference block introduce more conversion error than process variations. This is because, in CATDC, the primary conversion procedure is accomplished by T_{ref} and amplification module. While the effect of process variations between different transistors is canceled out along the delay chain, thus incurring less error. Due to the self-oscillation mechanism of CATDC, the impact of transient noise is automatically averaged.

4.3.2.2 Time Reference

The time reference is another block that fulfills the conversion procedure, similar to the amplification block; the reference components impact the conversion accuracy in each conversion round. In simulating the impact of different time difference values, we applied a 5% deviation range on the ideal $T_{ref} = 50ps$, from 47.5 to 52.5 ps. The conversion errors brought by time difference deviation can be found in Fig. 4.2b, where the maximum conversion error is around $\pm 5ps$.

4.3.2.3 Process Variations and Transient Noise

To simulate the impact of process variations, we utilize random threshold voltage values for different transistors. According to [70], the standard deviation (σ) of threshold voltage depends on transistor size, which can be calculated with Equation 4.1 using a $A_{VT} = 1.8mV\mu m$. Note that in our simulation, the threshold voltage deviation of all of the transistors in CATDC circuit follows Gaussian distribution.

$$\sigma_{VT} = \frac{A_{VT}}{\sqrt{WL}} \tag{4.1}$$

Besides process variations, transient noise is another factor that would introduce time conversion error. In our simulation, to evaluate the impact of noise, two noise types are considered: flicker noise and channel thermal noise. Flicker noise is usually considered in higher frequency simulation since the CATDC works on self-oscillation. Thus we consider this feature. The flicker noise is usually caused by charge fluctuation in oxide traps, and such noise will result in fluctuations of both mobile carrier numbers and mobilities in the channel. Channel thermal noise is related to the voltage variations, which is caused by the random motion of electrons. To add these two noise features,

the BSIM4[1] simulation card is used. In which parameters $fnoimod$ and $tnoimod$ are manipulated to implement flicker and thermal noise respectively [2].

The simulation result of process variations and transient noise is shown in Fig. 4.2c. By comparing with the conversion errors caused by amplification and time reference blocks, it can be concluded that the general process variations and transient noise have less impact on the version error. Note that the deviation of T_{ref} and gain is also caused by process variation to a certain extent, but for simplicity, we discuss them separately. This is because the process variation from a single transistor introduce either negative or positive bias, the overall effect of them will be averaged. And moreover, since the time digitation of CATDC is from several oscillation rounds; thus, the impact from transient noise will also be decreased.

4.4 Proposed Methodologies

From section 4.3, it can be concluded that the time conversion error of CATDC mainly comes from the amplification and time reference block. To eliminate the impact of these two blocks, we propose a configurable algorithmic compact TDC (CCATDC), in which the delay time reference block can be manipulated to compensate the bias from the amplification block.

4.4.1 Gain Compensation

In CATDC, the binary bit O_i of the $(i)^{th}$ conversion round can be expressed with Equation 4.2, in which $T^i_{reference}$ and T^i_{start} denote the "reference" and "start" signals in the i^{th} conversion round.

---

[1]BSIM4 model is built on industry standard and the BSIM modeling group at UC Berkeley.

[2]the parameters in our simulation of $fnoimod$ are set as noia = 6.25e41, noib = 3.125e26, noic = 8.75; of $tnoimod$ are set as ntnoi = 1, tnoia = 1.5e6 and tnoib = 3.5e6

$$O_i = \begin{cases} 0, & if\ (T^i_{reference} - T^i_{start}) * 2 < T_{ref} \\ 1, & if\ (T^i_{reference} - T^i_{start}) * 2 > T_{ref} \end{cases} \tag{4.2}$$

While in practical scenario, usually the gain of amplification is not equal to 2, thus by replacing the constant gain and time reference in Equation 4.2 with actual gain $\widehat{gain}$ and time reference $\widehat{T_{ref}}$, we get:

$$O_i = \begin{cases} 0, & if\ (T^i_{reference} - T^i_{start}) < \frac{\widehat{T_{ref}}}{\widehat{gain}} \\ 1, & if\ (T^i_{reference} - T^i_{start}) > \frac{\widehat{T_{ref}}}{\widehat{gain}} \end{cases} \tag{4.3}$$

Equation 4.3 implies that, to guarantee the correctness of each output bit, we can utilize a pair of proportional $\widehat{gain}$ and $\widehat{T_{ref}}$. That is, even if the value of $\widehat{gain}$ or $\widehat{T_{ref}}$ are different from the ideal value due to process variations, if we can configure/adjust the $\frac{\widehat{T_{ref}}}{\widehat{gain}}$, we can still get a correct conversion result. For example, if the ideal T_{ref} is 50 ps and ideal gain is 2, then if we can keep the ratio between them as: $\frac{\widehat{T_{ref}}}{\widehat{gain}} = \frac{50}{2} = 25ps$, we can get the correct conversion bit. Note that a pair of $\widehat{gain}$ or $\widehat{T_{ref}}$ satisfying Equation 4.3 only guarantees the correction of MSB, but will introduce conversion errors to other converted bits. To fix this problem while realizing Equation 4.3, we propose to use adjustable delay-line technique as follows.

4.4.2 Adjustable Delay-line

To address time conversion errors caused by the gain and reference inaccuracy, we propose to use an adjustable delay line. Our proposal is based on the truth that: once a CCATDC is fabricated, it is not easy to manipulate the gain [3]. According to Equation 4.3, a proportional reference/gain setup will help with improving the

[3]Note that the designer can still use some technique to change the chip in semiconductor level, for example, using Focused ion beam. However, using these techniques will significantly increase the cost.

conversion precision. To facilitate the robustness of CCATDC, we propose to use an adjustable delay-line, in which the actual in-path delay can be adjusted (note that there are many possible realizations of the adjustable delay element, such as voltage-, current- and digital-controlled.

The schematic of an adjustable delay element is shown in Fig. 4.3, in which the delay-line is composed by two parts: constant delay-line D_{const} and adjustable delay-line D_{adj}. Note that there still exists process variations in constant delay-line, that deviates it from the designed delay length. The purpose of adding it is roughly calibrating the reference time T_{ref}, just like the coarse tuning. While the high-resolution adjustment is realized with the adjustable delay-line, that fulfills the job of fine-tuning. To achieve higher adjustable resolution, two parallel delay-lines are utilized in two channels to form a vernier adjustable delay-chain. An external calibration circuit can supply the control signals to the adjustable delay line, due to the page limitation, we do not discuss this part in this work.

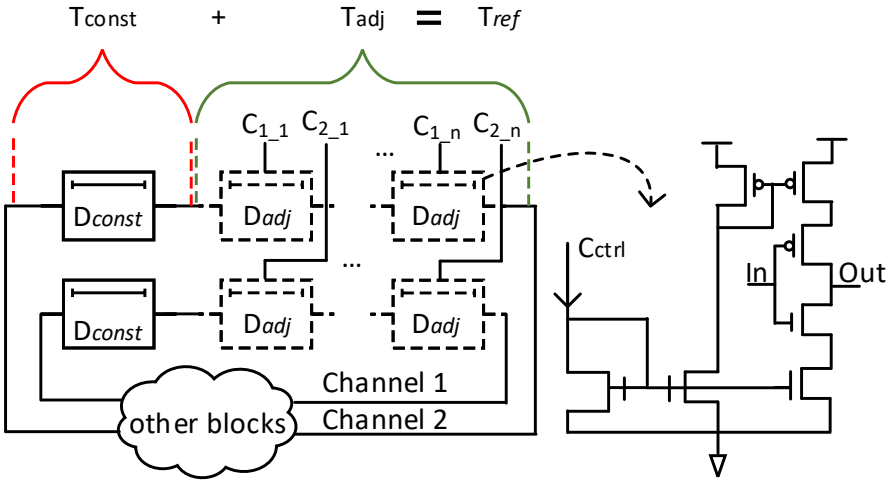


Figure 4.3: Schematic of the proposed configurable delay element, which is composed of two blocks, D_{const} and D_{adj}. There are many possible implementations of the adjustable delay element, in this figure a current-controlled delay element is shown as an example.

4.4.3 Delay Chain Configuration with Machine Learning

An important purpose of conducting the configuration is set up the delay of each element to realize Equation 4.3. However, since all of the components in the CCATDC design is of nanometer magnitude, it is impractical to measure such features with the external instrument. In this section, we propose to use Machine Learning techniques to help with characterizing and configuring the delay length of CCATDC. More specifically, the backward propagation of errors (Backpropagation) algorithm is used in this work. Backpropagation is a well-known technique in training artificial neural networks and is often used with other optimization methods such as gradient descent. Usual Backpropagation training is composed of two phases: propagation and weight updating. Once an input vector is applied to the neural network model, it will be propagated from the input layer to the output layer. The output per each input vector will be obtained and compared with the desired (golden) one, and a loss function will be used to calculate the error for each neuron in the trained model and change the weight parameters.

In this work, the controlling signals (for example, current) of the configurable delay-elements (as shown in Fig. 4.3) are fed into the Backpropagation model as input vectors: $\mathbf{C} = (C_0, C_1, \ldots, C_{n-1})$ in Algorithm 2. To comprehensively configure the delay chain, m different time inputs $\mathbf{T} = (t_0, t_1, \ldots, t_{m-1})$ and corresponding ideal conversion outputs $\boldsymbol{O} = (O_0, O_1, \ldots, O_{m-1})$ are also utilized to train the model. The accuracy of CCATDC is optimized by setting Δ_{err} as the threshold of conversion error. Before the model is being trained, weight parameters are initialized as 0. The controlling signals are fed into the HSPICE simulation and corresponding conversion outputs are obtained (line5), the simulated results are then compared with the golden values (line7). The error is backpropagated to the network (line8) and controlling vector is optimized to achieve high conversion accuracy (line9). The training procedure will end while the pre-set error tolerance has been met (line10).

Procedure 1 Given a CCATDC chip, configure the delay element and improve the time conversion accuracy.

Input: a designed configurable compact algorithmic TDC

Input: n input control vector $\mathbf{C} = (C_0, C_1, \ldots, C_{n-1})$

Input: m time input $\mathbf{T} = (t_0, t_1, \ldots, t_{m-1})$ and corresponding ideal conversion output $\boldsymbol{O} = (O_0, O_1, \ldots, O_{m-1})$ for the designed T_{ref} and amplification gain

Input: Δ_{err}, the threshold of conversion error

Initialize weight parameters $\theta^i \leftarrow 0$
Initialize error parameters $\Delta^i \leftarrow \Delta_{err}$
while $\max(\Delta^i) >= \Delta_{err}$ **do**
  for $i := 0$ **to** $m - 1$ **do**
    $\hat{O}_i = HSPICE(t_i, \mathbf{C})$
  end for
  compute conversion error $\boldsymbol{\Delta}= (\hat{\boldsymbol{O}}$ - $\boldsymbol{O})$
  BackwardPropagateError($\boldsymbol{\Delta}$)
  UpdateWeights($\boldsymbol{\Theta}$, $\mathbf{C}$)
end while
return $\mathbf{C}$

4.4.4 Bidirectional Flexibility

As mentioned in section 4.1 and 4.2, a common problem with current TDC design is that users have to know "start" and "reference" signals in advance, and correctly connect them with the corresponding pin, since the delay length is different between two channels. An alternative method is that users have to employ two TDCs and switches the connection to make sure one of them generates the correct output, but this will increase the power and overhead consumption since two circuits will be used. Another drawback of using two channels in CATDC lies in the mismatch auxiliary blocks (such as latches), such mismatch is constant and will introduce error in every conversion round.

To solve the two problems above, we propose to use a "signed" delay-chain in the CCATDC design, the schematic of which is as shown in Fig. 4.4. In this new structure, the "start" does not necessarily come earlier than "reference" signal. An Arbiter can detect the first arrival signal, and then a switch signal will be generated to distribute

the adjustable delay elements into either channel1 or channel2, correspondingly. For example, if the "start" signal is still ahead of "reference" more than T_{ref}, the top Arbiter will switch the sign of T_{ref} to +. The sign of both $T_{ref}/2$ will be +; thus it works the same as that in Fig. 4.1. If "reference" is ahead of "start" for less than T_{ref}, then the top Arbiter will change the sign of T_{ref} and the right $T_{ref}/2$ to −. The middle Arbiter will change the sign of left $T_{ref}/2$ to +. Thus the $+T_{ref}/2$ and $-T_{ref}/2$ will cancel out each other, and the time difference will be amplified by 2.

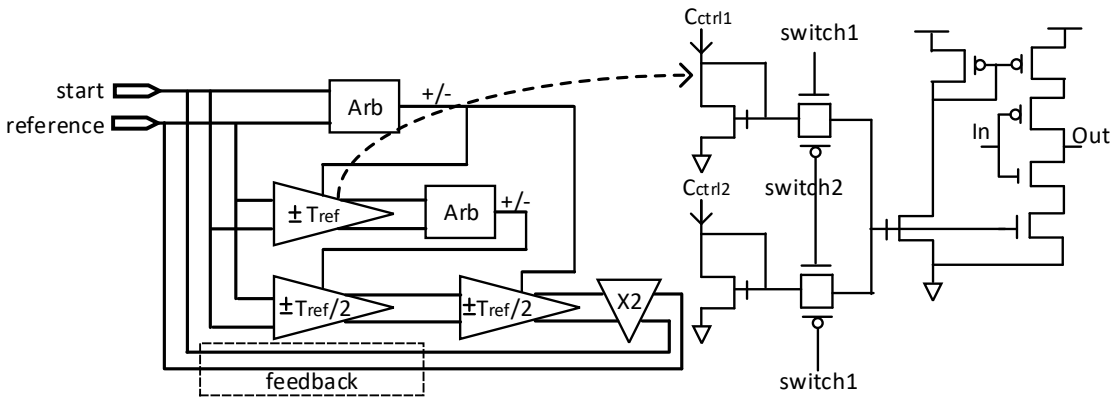


Figure 4.4: Schematic of bidirectional CCATDC. Two Arbiter circuit is employed, the outputs of them are used to switch the effect of delay-line. The switch signal is used to determine which channel is delayed more than the other. Note that for brevity, some auxiliary blocks like latch circuit are not shown in this figure.

4.5 Implementation and Performance

In this section, we validate the performance of the proposed CCATDC in an interactive mode, a controller is written in Python is used to control the HSPICE simulation, and update the voltage inputs for the adjustable delay line. This controller is also responsible for updating the weight values and Backpropagation training. The reason we use this mode is to mimic that in the practical scenario, real-time testing and calibration of a chip can be fulfilled with a software.

	Energy/bit	overhead: # of gates
Flash TDC	0.353pJ	406
CATDC	0.156pJ	196
CCATDC	0.087pJ	162

Table 4.1: Overhead and energy comparison between different TDC structures.

In our experiment, the proposed CCATDC is implemented with PTM 45nm standard cell libraries [69]. To test the conversion accuracy, we applied random process variations on all transistors used to build the CCATDC circuit, and transient noise is also added in all simulations. The experimental result is as shown in Fig. 4.5, in which input time difference from 10ps to 100ps is applied, while an ideal $T_{ref} = 50ps$ is utilized to rebuild the time difference. Comparing the result with that in Fig. 4.2, it can be seen that CCATDC is more robust against process variation and random noise, and more so if more conversion bits are utilized. We also compare the performance of different TDC implementations, and it can be seen that our proposed CCATDC outperforms the other two structures in both overheads (60% and 17.3%) and power consumption(75.4% and 44.2%).

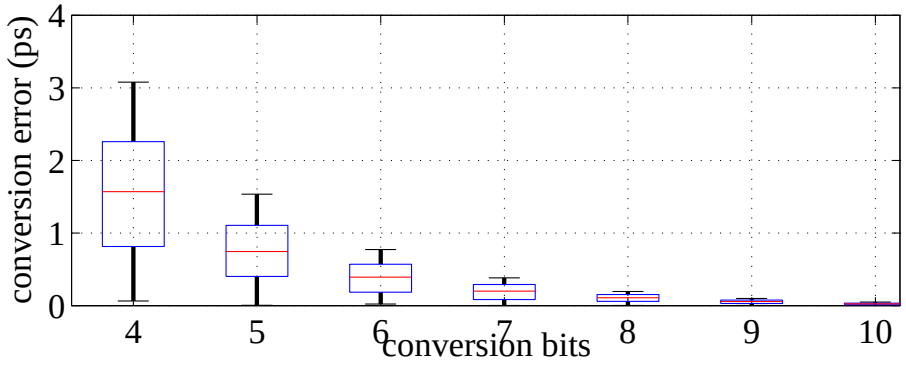


Figure 4.5: Time conversion results of different input time differences (10-100ps), the designed CCATDC has an ideal reference time $T_{ref} = 50ps$, and an ideal gain=2. It can be found that while more conversion rounds are utilized, the error become smaller.

4.6 Conclusion and Future Work

The development of semiconductor technology enables the realization of high-resolution delay-chain but also introduces negative impact from process variations. In this work, we present CCATDC, a configurable compact algorithmic time-to-digital-converter design that can mitigate the conversion errors caused by process variations. CCATDC consists of adjustable logic that can be tuned per the performance of conversion accuracy. A Machine Learning based framework is proposed to find the best configuration inputs for a wide input time range. The experimental results demonstrate that our design is lightweight compared to the other two TDC structures. Interesting future work is to explore the implementation of calibration circuit, while also minimizing the overall power consumption. The next two chapters will explore the possibility of FPGA-based TDC implementations.

CHAPTER 5

DESIGN OF PVT-RESISTANT ALL-DIGITAL TIME-DOMAIN AMPLIFIER WITH VARIABLE GAIN AND WIDE OPERATION RANGE

Time-domain amplifier (TDA) has found significant use in high-resolution time measurement. However, conventional TDA designs are mainly built with analog characteristics of the circuit, which limits the integration with digital designs. Besides, these designs have other problems like narrow operation range, high sensitivity to process variations, supply voltage, and temperature fluctuations. This chapter proposes an all-digital TDA architecture that facilitates the compatibility with modern digital electronic systems. The proposed TDA design can provide adjustable gain values for timing amplification. Experimental results demonstrate that the proposed TDA achieves high linearity ($< \pm 1\%$ gain error) under a variety of temperature and supply voltage conditions. Moreover, we show that the proposed TDA architecture can be implemented in different CMOS technology nodes: 45nm, 32nm, 22nm, and 16nm without performance loss.

5.1 Introduction

Time-of-flight measurement has found great use in modern scientific and engineering applications [23], such as remote sensing [57] [34], nuclear science [46] [58], biomedical imaging [59], frequency synthesizer and time jitter measurement of RF transceiver [38], and time jitter measurement for the all-digital phase-locked loop (PLL)/delay-locked loop (DLL) [18]. To facilitate the processing of analog timing signal with digital systems, time-to-digital converter (TDC) has been proposed as the interface. Some

well-known TDCs include flash time-to-digital converter (TDC), cyclic TDC, and pipelined TDC. Most of these TDCs are implemented with digital blocks such as D flip-flop, inverter, etc. However, while time signals scale to the magnitude of the picosecond, these TDC can not measure them with high precision. This is because the timing signal of picosecond is smaller than the resolution of these TDCs, and therefore can not be directly quantified. This problem becomes more severe with the advancement of CMOS fabrication technology since the impact of process variation cannot be ignored [67][40].

Time-domain amplifier (TDA) is a promising method to address the problem of limited resolution in timing measurement [71] [72]. Instead of directly improving the resolution of measurement systems, TDA amplifies the timing signal under test, the magnification of this amplification is called gain of TDA. While the amplified timing signal can be precisely measured within the resolution of existing TDCs, the original timing signal can be calculated by dividing the measured value with gain. It is desirable that a TDA has high linearity (consistent gain) and resolution, and low sensitivity to the fluctuation of environmental conditions. Considering that TDA can be used as a component of many other digital systems such as TDC, it is also desirable to implement TDA with all-digital blocks.

In this work, a novel TDA architecture is proposed, that amplifies timing signal under test by duplicating it into a *pulse train*[1] that is composed of several pulses. The number of pulses in the pulse train stands for the gain, for example, if the described gain is 2, two copies of the timing signal are fed into the TDA in sequence. The proposed TDA structure is composed of three components: pulse extraction, pulse duplication, and pulse summation, the schematic of each element is described with details. The novelties of proposed TDA technique are as follows:

[1]The length of a timing signal under test can be converted into the width of a pulse or the time difference between the rising edge of two adjacent pulses.

- The proposed TDA architecture consists of all standard digital blocks, and its implementation can reduce the circuit design complexity and the restriction of the physical layout.
- The implementation of proposed TDA structure is flexible, that facilitates the integration with post-processing systems.
- The proposed TDA has high linearity across a range of temperature and supply voltage conditions.
- The proposed TDA can be implemented with various CMOS technology nodes, and show high performance even with the existence of process variations.

The rest of this chapter is organized as follows: Section 5.2 reviews several related works on TDA design. Section 5.3 presents the main idea of the proposed all-digital TDA. The components of TDA are also presented in this section. Section 5.4 presents the experimental evaluation of the proposed TDA architecture, its performance under a range of temperature points, supply voltage values, and even process variations is demonstrated. Section 5.5 concludes this chapter.

5.2 Related Work

A conventional way to measure timing signal is using the analog feature of a circuit, such as the charging time of the RC circuit. For example, if a constant current source charges a capacitor, its voltage will reflect the charging time. Therefore, by measuring the voltage value of a capacitor, the charging time can be calculated. In [72], the rising edge of the pulse under measurement turns on the charging, while the falling edge turns it off. The timing under testing (i.e., the pulse width) can be calculated with the voltage value of the capacitor. This TDA design achieves excellent linearity and operational range. However, it requires three individual current sources, and two comparators, which are hard to be integrated with the digital circuit. The speed

of this design is also limited by the low-speed of the comparators, which hinders its applicability.

The response time of SR-latch is utilized to amplify the timing signal in [71]. When the time difference between the two inputs: the set and reset are close to each other, SR-latch will produce two outputs with an amplified time difference. This is because the SR-latch will first go into a metastable state. The duration of the metastable state has a reverse log relation to the time difference between these two inputs. To realize linear amplification, two SR-latches are implemented with the opposite time offset. This implementation is compact and achieves high-speed amplification. However, the applicability of the TDC design is limited by the small metastable time range of SR-latch. Since the gain is reversely related to the offset, the amplification requires plenty of time offset. Though techniques are developed to improve the operation range [73], it is still challenging to measure the timing signals with a length above 100ps. Moreover, the metastability of SR-latch is sensitive to the device mismatch and process variation, that poses strict requirements in symmetric layout design and accurate device sizing.

Prior art of TDA also utilizes a chain of cross-coupled delay elements to amplify timing signal [74] [75]. By applying two input signals into two paths separately, that is composed of many variable delay cells. These delay cells are specialized that they can output two different delay lengths, depending on the delay switch is "High" or "Low". These two delay lengths are proportional to each other. By switching the delay time of the delay elements, this TDA can generate amplified time difference with a linear relationship to the input. However, it is complicated to implement this TDA because of the design complexity of two fixed delay elements that are proportional to each other. For large input range and gain, the accumulation error caused by the long delay lines introduces nonlinearity.

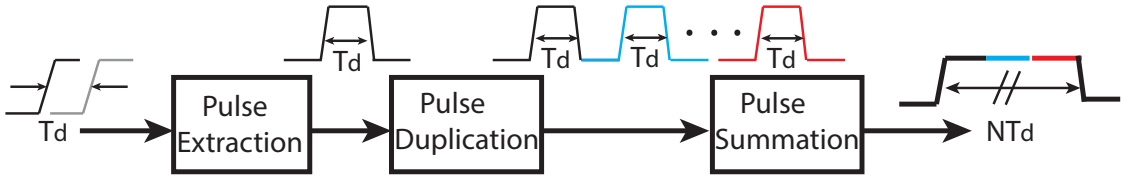


Figure 5.1: Proposed Digital-Time-Difference-Amplifier block diagram.

5.3 All-Digital Time-Domain Amplifier

The mechanism of all-digital TDA is converting the time signal under test into the width of one pulse. The amplification is realized by replicating this pulse for $N-1$ times and taking the summation of the width from these N pulses. Schematic of the proposed TDA is as shown in Fig.5.1. The TDA consists of three blocks: pulse extraction, pulse duplication, and pulse width summation. The pulse extraction block generates a single pulse with a pulse width of T_d equals the time difference between the two inputs. The pulse duplication block generates $N-1$ copies of the extracted pulse in sequence, like a pulse train. The last step is to sum the width of N pulses to get the amplified output with a width of $N * T_d$. If needed, a pulse generator can be used to separate the amplified signal back to two signals with a time difference $N * T_d$ in between. The schematic of each block is described in this section.

The pulse duplicator produces multiple non-overlapped pulses from the original pulse, and make them into one pulse train, as shown in Fig. 5.1. To sum the pulse width of the pulse train, two time-latches are applied to exclude the non-active time between two pulses. The circuit implementations for each block are presented in this section.

5.3.1 Pulse extraction

A phase detector for single-edge detection is used as the pulse extractor. Assuming that the timing signal under test is the difference between inputs A and B, the single-edge phase detector converts this time difference (T_d) into the width of a single

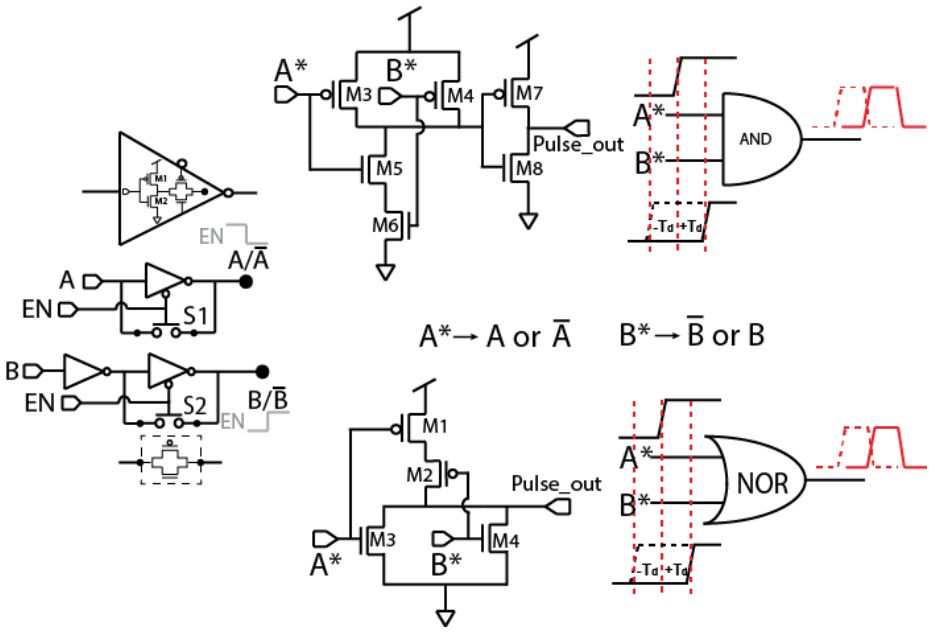


Figure 5.2: Phase extraction based on two alternative designs of positive-edge phase detector. Inputs A and B are always opposite polarity.

pulse, as shown in Fig. 5.2. Following de Morgan's law: $A \cdot \overline{B} = \overline{\overline{A} + B}$, the phase detector is implemented with AND gate or NOR gate, as shown in Fig. 5.2. Note that only A input ahead of B input is shown in this schematic, this design is applicable to extract the pulse if B input is ahead of A input with the switches **S1** and **S2**. The resolution of the phase detector determines the lower bound of the TDA operation range. To reduce this limitation, an extra time constant T_C can be added to enlarge the time difference between A and B. For example, an additional delay element with constant delay T_C can be added in one of the two input channels. As T_C is set higher than the minimum phase (t_{PD_min}) that can be detected, the effective time difference guarantees that TDA can operate below the phase detector resolution.

5.3.2 Pulse duplication

The pulse duplicator produces $N-1$ copies of the initial pulse to build a pulse train. One solution is to pass the original pulse to a group of buffers; each buffer can delay the pulse signal with a fixed delay length of τ_{buf}. An OR gate is connected

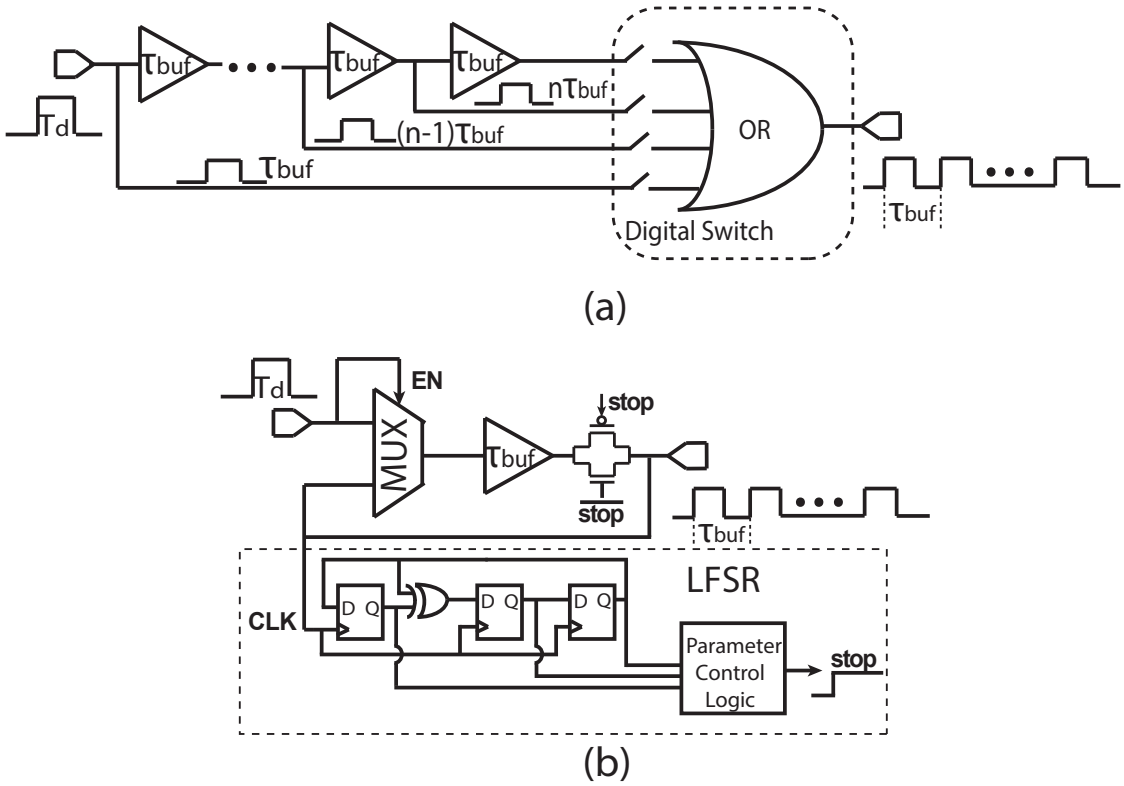


Figure 5.3: Replicating pulses in parallel (a) and series (b).

with the outputs of all delay buffers to generate the pulse train, as shown in Fig. 5.3(a). Since every two adjacent pulses should be separated from each other, therefore τ_{buf} determines the upper bound of the TDA operation range. The OR gate based pulse duplicator is straightforward to implement. However, the implementation of this method brings a relatively larger overhead, because the number of buffers and the size of OR gate increase linearly with the number of pulse copies. An overhead-efficient way to duplicate the pulses is using a feedback loop to generate multiple pulses in series, as shown in Fig.5.3(b). A linear feedback shift register (LFSR) can be used as a counter to count the number of pulses. The control logic of LFSR determines the size of the counter. Therefore different gain values can be realized without modifying the hardware implementation.

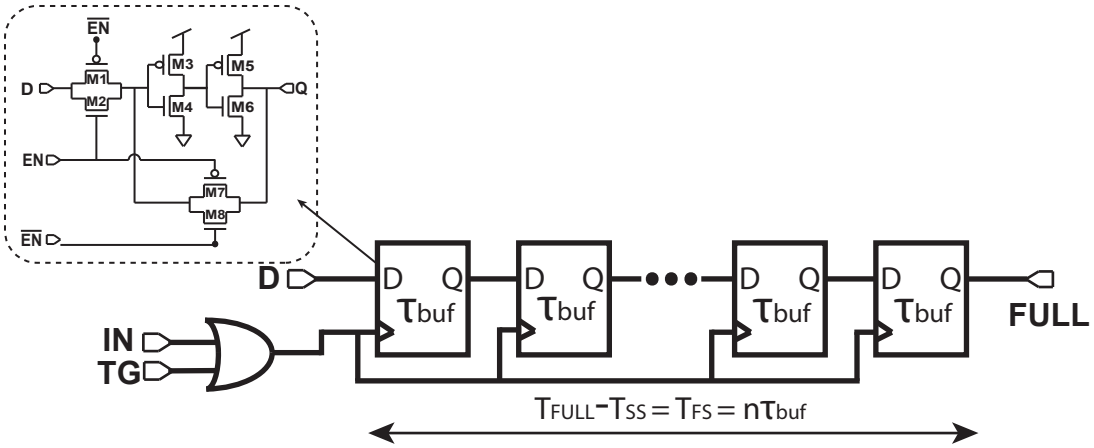


Figure 5.4: Time-Latch implementation diagram using a chain of D-latch.

5.3.3 Pulse summation

In the proposed TDA, pulse summation is the critical step to amplify the time signal with high linearity. The mechanism of pulse summation is adding all duplicated pulses into a single one. The pulse summation is realized with a specialized time-latch, as shown in Fig. 5.4, this time-latch is composed of a series of D latches. Besides the regular input D and output Q signals, each D latch has a **EN** signal. Depending on the state of **EN**, the D latch has two states: transparent state and opaque state. In the transparent mode, the input signal D can propagate to Q; while in the opaque mode, the transmission is held. The latch modes are controlled by the enable signal **(EN/$\overline{\text{EN}}$)** of two transmission gate switches(M1, M2 and M7, M8), which can be activated by input signal **IN** or trigger signal **TG**. When **EN** is high, the signal **D** propagates along the time-latch; in contrast, the signal propagation stops and is retained when the **EN** becomes active again. Once the signal **D** is passing through the whole delay chain, the signal **FULL** is generated. The time-latch function diagram is presented in Fig.5.5.

When an input pulse becomes high, the signal **D** passes through **m** delay cells (labeled as gray in Fig. 5.6) until the input pulse becomes low. Then the delay cells are disabled, and the signal propagation stops after a duration of input pulse T_d.

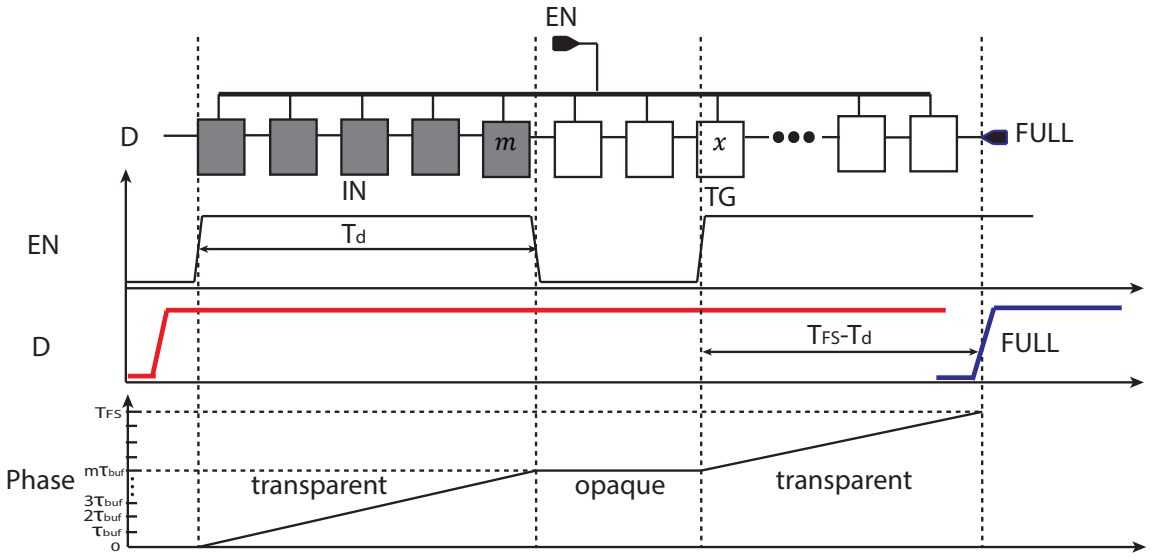


Figure 5.5: Time-Latch function diagram with two modes: propagating and holding.

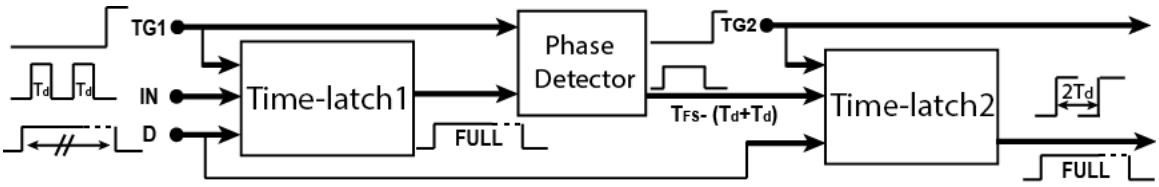


Figure 5.6: Pulse summation by cascading two time-latches.

When a trigger signal is applied to the time latch, **EN** becomes active again, and the signal **D** resumes to propagates from the xth stage to the end to obtain the **FULL** signal. Therefore, the output of the time-latch is the time difference between the **TG** and **FULL**: T_{FS}-T_d. The width of the input pulse is stored in its minus version: -T_d. The summation of multiple pulse width is realized by cascading two latches. The example of amplification gain=2 is demonstrated in Fig.5.6. Two identical pulses from pulse duplicator are applied to the first time-latch, and the output pulse of the first time-latch T_{FS}-(T_d+T_d) imports to the second time-latch. With the same operation as a single time-latch, the final output: T_{FS}-(T_{FS}-(T_d+T_d))=2T_d is achieved.

5.3.4 The multi-path high resolution time-latch

Due to the switching between two modes, every time the time-latch is reactive after a period of holding time, it should pick up where it stops. However, the quantization level is introduced by the propagation delay of the single delay cell, which limits the resolution of the analog TD amplification, and introduces quantization error that degrades the linearity of the TDA. To solve this problem, multi-path topology is designed to achieve a higher clock frequency or better time resolution [42]. Each delay cell has multiple input paths to drive its output, which senses the state from different outputs of previous stages. Earliest arrival transition firstly triggers the output to reduce the average propagation delay on each stage.

The schematic of multi-path time-latch in TDA design is shown in Fig.5.7. The delay element is built with D-latch. It is developed to add a parallel input path to the latch, but disconnect the initial net between the two inverters. The modified latch cell has three inputs, **D1**, **D2**, **midin**; and two outputs, **midout**, **Q**. The nth stage inputs **D1**$_{(n)}$ and **D2**$_{(n)}$ are connected to the two previous stages outputs: Q$_{(n-1)}$ and Q$_{(n-2)}$ respectively. Input midin$_{(n)}$ is connected to previous stage midout$_{(n-1)}$. Therefore, connecting to the earlier stage pushes to get faster response on the output and reduce the propagation delay from initial input to the output. The two parallel feedback paths are connected from outputs to inputs D1 and D2. In this implementation, **D1** is connected with the earlier transition; the feedback transmission gate(M7 and M8) to D2 can be omitted since it will not cause the change of the output.

5.4 Function and Performance Evaluation

The pulse train summation of this design offers tunable integer gain without modifying hardware implementation. To evaluate this feature, pulses trains composed of two, three, and four pulses are applied to the TDA, respectively. The amplified

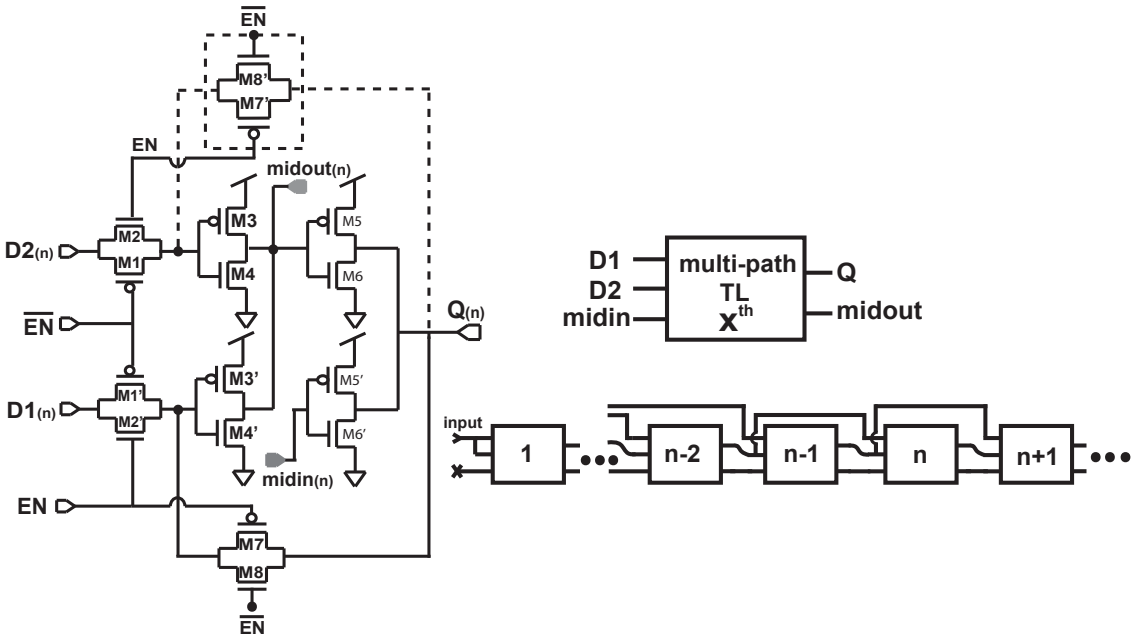


Figure 5.7: Implementation of multi-path time-latch to improve the time resolution of ADTDA.

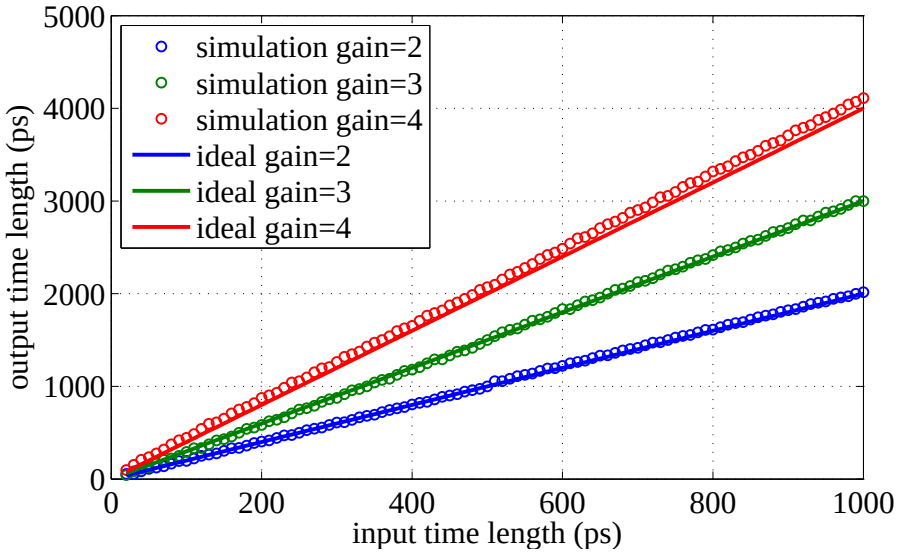


Figure 5.8: The gain is tunable with integer multiplier.

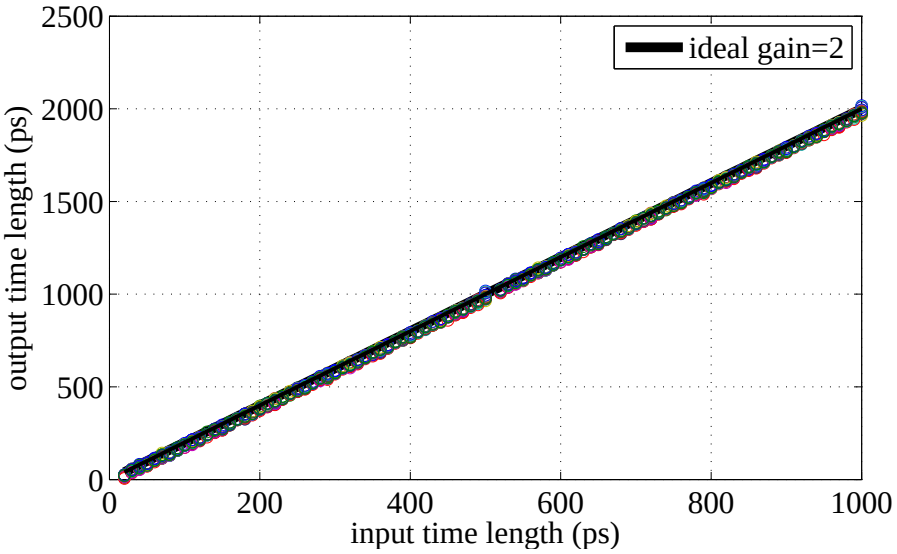
outputs of gain (=2, 3, and 4) are shown in Fig.5.8. The amplification demonstrates the good fit between the observed amplification gain and theoretical gain.

Feature size	*Resolution*	*Amplification error*
$45nm$	$12ps$	4.47%
$32nm$	$8ps$	3.27%
$22nm$	$6ps$	2.61%
$16nm$	$1ps$	1.63%

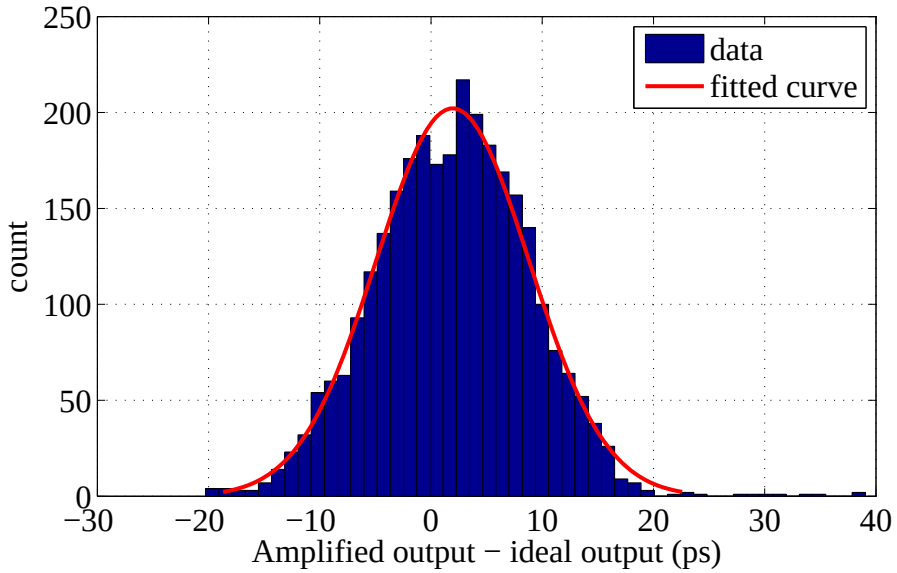
Table 5.1: ADTDA performance using different CMOS technologies (with gain=2)

As fabrication technology scales, process variations become more uncontrollable. High-precision electronic systems are sensitive to both process variations and fluctuations of environmental conditions, such as temperature and supply voltage. Process variation means that the fabricated physical parameters deviate from the designed ones, thus greatly impacts the accuracy of designs that have strict requirements on the physical dimension. The robustness of the proposed TDA with process variation is evaluated with Monte-Carlo simulations, as well as under different environmental conditions. In Fig.5.9, the experimental result shows high reliability and robustness of this implementation. Random process variations ($\pm$15% V_{supply}) are added to the threshold voltage of each transistor in the Monte-Carlo simulations, which are conducted with the temperature changing from 0 to 100 °C and supply voltage varying within $\pm$20% V_{supply}, the simulation results under various temperatures and supply voltages are shown in Fig.5.10a and Fig.5.10b. It can be observed that the amplification gain error is under $\pm$1%.

The performance of proposed TDA is also evaluated across a variety of CMOS technology nodes: 45nm, 32nm, 22nm, and 16nm. The amplification gain=2 is evaluated with Hspice simulation for each technology node. The experimental results demonstrate that the TDA design achieves constant gain across all technology nodes, as shown in Fig.5.11. As the feature size scales down, the resolution becomes higher, as concluded in Tab 1. This is because the minimum propagation delay that can be achieved is smaller with more advanced CMOS technology nodes.



(a) Monte Carlo simulation of 30 TDA instances.

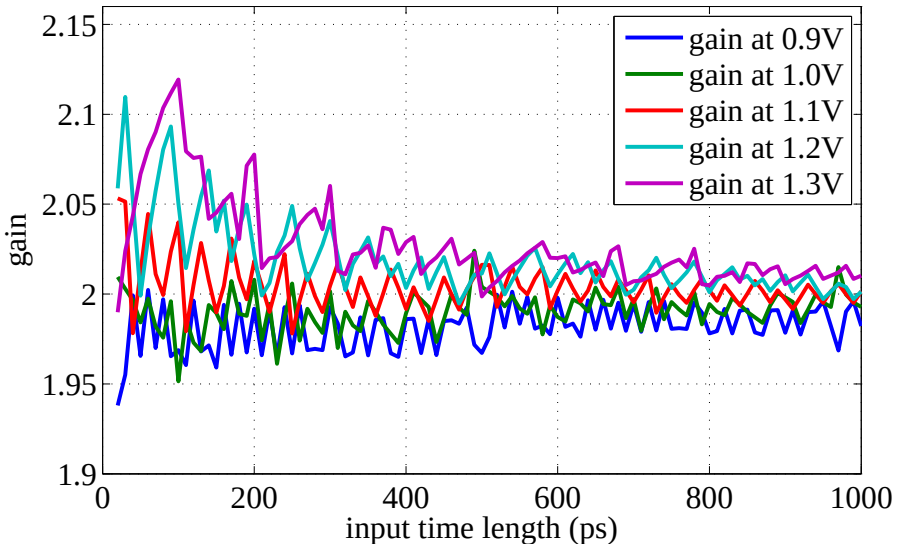


(b) Gain error distribution from Monte Carlo simulation of 30 TDA instances.

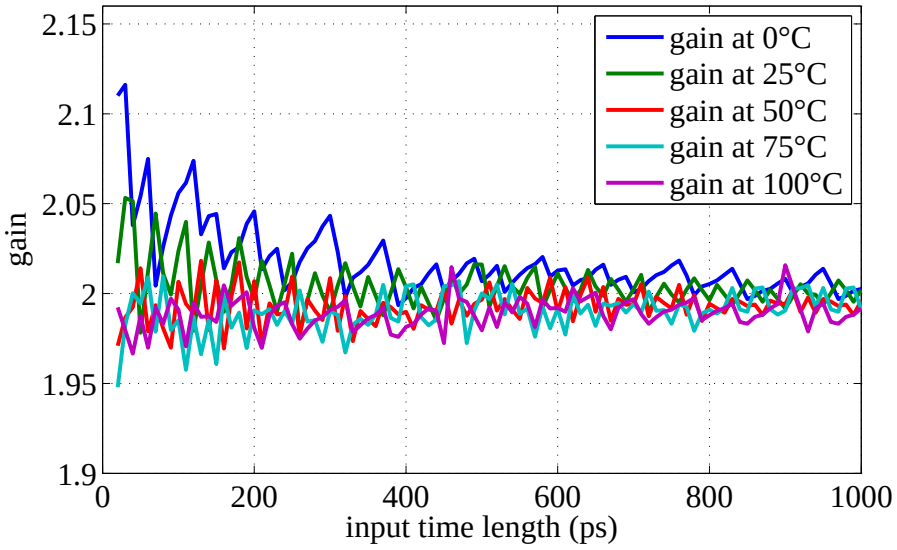
Figure 5.9: Monte Carlo simulation results for 30 TDA instances. It can be observed that the gain of the proposed TDA architecture is reliable against process variations.

5.5 Conclusion

In this work, an all-digital TDA design is proposed, its performance is evaluated across a variety of environmental conditions and CMOS technology nodes. The experimental results demonstrate that the proposed TDA can operate in a wide linear range, that can be expanded from picosecond to nanosecond. The multi-path time-



(a) The gain of proposed TDA architecture under different supply voltages.



(b) The gain of proposed TDA architecture under different supply temperatures.

Figure 5.10: The gain of proposed TDA architecture under different environmental conditions.

latch realizes measurement resolution that is below the minimum gate propagation delay, and high accuracy with no more than 12*ps* error. Monte Carlo simulation conducted under different process corners shows the robustness of the design. The proposed TDA is composed of standard all-digital blocks, and thuscan be integrated

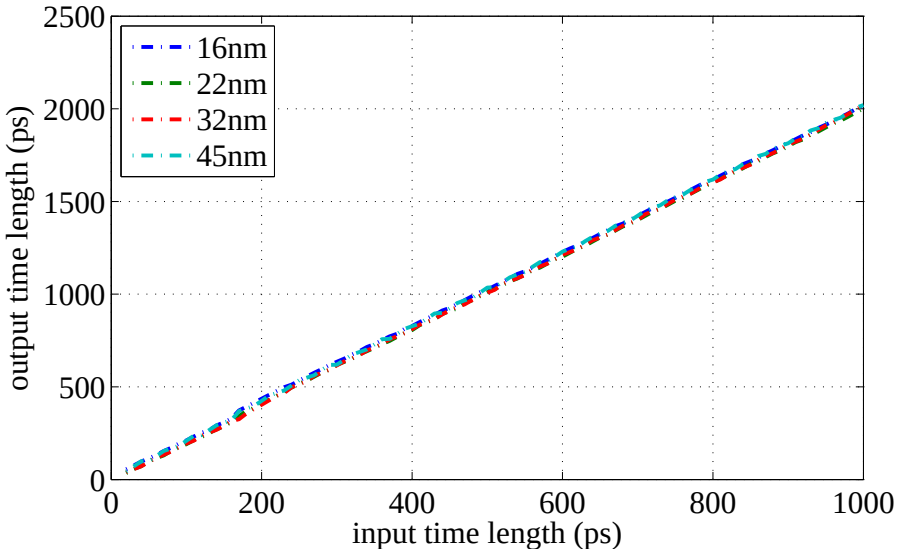


Figure 5.11: The implementation of all-digital TDA is portable in other advanced CMOS technologies.

with other digital systems. Future work will explore the feasibility of FPGA realization of the proposed design scheme.

CHAPTER 6

AN ALL-DIGITAL PVT-RESISTANT TIMING MEASUREMENT CIRCUIT WITH RESOLUTION OF SUB-PICOSECOND

Measuring timing of sub-picosecond has become a common but urgent need in today's high-speed electronics design. However, conventional timing measurement circuits are either built with analog circuits that are hard to integrate with digital systems, or differential delay-lines whose resolution is significantly impacted by process variations or environmental conditions. To overcome these issues, in this chapter, we present a novel all-digital timing measurement circuit: PVTMC, which for the first time, constructively leverages process variations to measure timing signals of sub-picosecond. We present the design scheme of PVTMC with great details. In our experiment, we use both HSPICE simulation and FPGA implementations to validate the performance of this new timing measurement circuit. Our experimental results demonstrate that PVTMC achieves high resolution ($< 0.5ps$), and its performance is stable against the fluctuations of environmental conditions, such as supply voltage and temperature. Moreover, two algorithms are proposed to improve the performance of PVTMC. They are: 1) A random search-based method that leverages the statistical characteristics of PVTMC to speed up the measurements and improve its accuracy. 2) A hybrid method based on machine learning modeling and binary search, which can fully characterize the internal delay variations of PVTMC circuit and exponentially reduce the number of measurement iterations. Beside the performance, we also show that the proposed circuit structure is compatible with the most prevalent CMOS technology nodes and FPGA chips due to the widely existence of process variations.

6.1 Introduction

The rapid development of the semiconductor industry makes it possible to design high-speed electronics for versatile applications. Correspondingly, accurate quantification of such high-speed timing signals (i.e., of sub-picosecond) becomes an urgent need in many application scenarios, such as time-of-flight measurement in remote sensing [23] [76], quantum computing [77], phase-locked loop frequency synthesizer [78], and the lifetime imaging of fluorescence [24] [79]. In order to measure time intervals of sub-picosecond magnitude, several methodologies have been proposed over the past few years. Roughly speaking, these methods can be classified into two classes: 1) Analog circuit-based methods, such as the employment of capacitor and constant current source [72] and the usage of analog memory and current switching [80]; and 2) digital circuit-based methods including faster counters [38][39], vernier delay-line [81], time-to-voltage converter [80], time-to-digital amplifier [82], and time-to-digital converter (TDC) [34].

Although these above-mentioned methods have achieved measuring timing intervals with resolution of picoseconds, they also have their weaknesses. For example, most high-frequency circuits have been using digital systems for signal post-processing, however, it is difficult to integrate these analog circuit-based timing measurement designs with digital systems. On the contrary, for digital circuit-based methods, the measurement resolution is always limited by the minimum propagation delay of gate-level components. For example, the resolution of TDC is determined by the smallest propagation delay provided by the gate-level delay components, like a buffer. However, in practice, the delay length of such delay components is significantly impacted by several issues, such as the process variations from CMOS fabrication.Correspondingly, the effective delay length of such components is usually different from the designed value, which significantly degrades the measurement accuracy [83]. Moreover, the sensitivity of electronic characteristics to environmental conditions like temperature

also brings negative impact on the precision of the timing measurement circuit. For example, higher temperature would incur longer propagation delay on these gate-level components. These weaknesses make most existing methods unusable while the interval of time of interest goes below the magnitude of picoseconds of even sub-picosecond. Interestingly, several works have already considered the negative impact of process variations and proposed compensation solutions for the timing measurement [60]. However, these methods used compensation circuit that needs large overhead, and the compensation circuit itself also faces these problems.

To solve the above-mentioned issues in designing timing measurement circuit with high-resolution, in this work, we propose an all-digital timing measurement circuit: PVTMC[1], which constructively leverages the process variations of CMOS circuit fabrication to measure timing signals. The structure of the proposed circuit design has the same design philosophy with the well-known hardware security primitive: physical unclonable functions (PUFs) [84], which also uses the process variations of circuit in a constructive way, i.e., to build fingerprinting of a chip. Unlike the conventional time-to-digital converter that relies on the propagation delay of gate-level elements, PVTMC uses a statistical feature of its binary outputs to quantify the duration of timing signals, which significantly improves the fault tolerance to PVT issues. The usage of such statistical features guarantees that the performance of PVTMC is stable against the fluctuations of environmental conditions, which do not change the statistical characteristics used by PVTMC. Another advantage of the proposed design is that it provides adjustable measurement range and precision, which can be realized by reconfiguring the number of elements in the circuit. The proposed design is specially good at measuring high-speed periodic time signals which cover

[1]PVTMC is short for *Process Variation-based Timing Measurement Circuit.*

most majority timing signals of interest, such as the jitters of high-frequency clock signals. The contributions of this chapter are summarized as follows:

- We propose a timing measurement circuit that constructively leverages the naturally existing process variations of nanometer CMOS technologies. To the best of our knowledge, this is the first work that constructively leverages the process variations of circuit to measure timing signals with sub-picosecond resolution.

- We propose and prove a math model that can be jointly used by a set of PVTMC circuits fabricated with the same structure and technology node; this model significantly reduces the workload of calibration and measurements with PVTMC.

- We propose a hybrid method based on machine learning and binary search to model and characterize the behavior of PVTMC circuit. This method exponentially reduces the number of measurement iterations of using PVTMC.

- We evaluate the performance of the proposed circuit architecture across a variety of CMOS technology nodes and environmental conditions; the experimental results demonstrate low measurement error ($< 0.5\ ps$) and high reliability across different temperature and supply voltage conditions.

- We implement the proposed PVTMC circuit on FPGAs and evaluate its performance. The results demonstrate that PVTMC is fully compatible with commodity reconfigurable platforms with good performance.

The rest of this chapter is organized as follows: Section 6.2 reviews the background of timing measurement circuit design and some related works; Section 6.3 elaborates the main idea and principle of the proposed timing measurement circuit; Section 6.4 presents the experimental evaluations of the proposed timing measurement circuit

with HSPICE simulations; Section 6.6 presents the experimental results from FPGA implementation; Section 6.7 concludes this chapter and gives the directions for future work.

6.2 Background and Related Works

One of the most commonly known timing measurement circuits is the so-called time-to-digital converter (TDC). As indicated by its name, a TDC can convert the length of timing signals under measurement into digital outputs, such as `1110`, with each bit denotes a corresponding weight like binary numbers. Delay-line is an important component used by most conventional TDCs. A delay-line is usually composed of a series of delay elements. The propagation delay of every single delay-element determines the best resolution that can be achieved by the delay-line based TDCs. The resolution of a single delay-line based TDC is the propagation delay of its delay-element, such as a buffer [].

In order to achieve higher resolution, vernier delay-line-based TDC was proposed [51], the schematic of which is shown in Fig. 6.1. In this schematic, the timing signal under measurement is denoted with the interval between "start" and "reference". In a vernier delay-line-based TDC, two channels are built with components of different propagation delay lengths: td_1 and td_2. While passing through the delay-lines, these two input signals "start" and "reference" are *postponed* by two different magnitudes: td_1 and td_2. Correspondingly, the length of the interval between "start" and "reference" is gradually reduced by these delay elements. Meanwhile, the interval between the two input signals is converted into the outputs (o_i) of the D Flip-flops (DFFs), and therefore, one of the them will first output a `1` instead of `0`, which indicates that the "start" signal is postponed enough and its rising edge overpasses that of "reference". Then the length of the interval between "start" and "reference" can be calculated with the number of DFFs output `1`. Unlike the normal delay-line based TDC whose

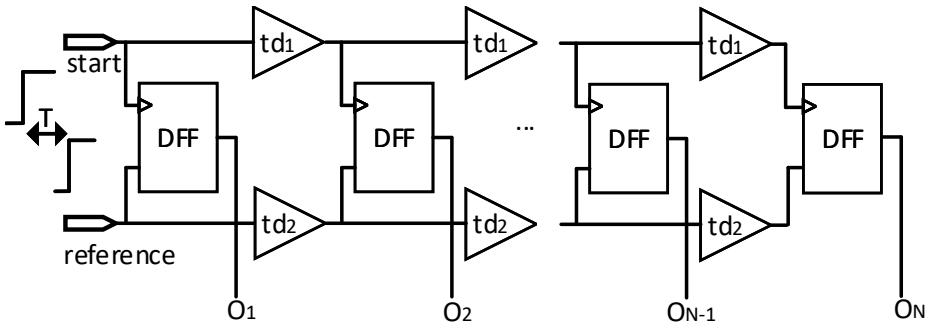


Figure 6.1: Schematic of a vernier TDC.

resolution is limited by the minimum propagation delay of a single element, the vernier TDC has a higher resolution $td_1 - td_2$. However, the practical performance and applicability of such delay-line-based TDCs are still limited by several issues:

6.2.0.1 The process variations from CMOS fabrication procedure

With the advancement of semiconductor technology, it has become more difficult to precisely control the fabrication procedure of CMOS circuitry, which makes process variation a critical issue in high-performance and high-accuracy electronic designs. As a result for TDCs, the delay elements will have deviated delay length from the designed value (i.e., td_1 and td_2), which degrades the measurement accuracy [48].

6.2.0.2 The order of two input signals must be pre-known

In a vernier delay-line based TDC, the delay length of td_1 is always designed to be larger than that of td_2, which ensures that the leading signal (i.e., "start" in Fig. 6.1) can be delayed more than the "reference" signal. However, in practical scenario, it is difficult to predict the order of two timing signals like which one comes earlier than the other due to the small time interval between them.

6.3 Introduction of PVTMC

In this section, we elaborate on the design scheme, statistical characteristics, and working principles of the proposed timing measurement circuit: PVTMC.

6.3.1 Schematic of PVTMC

The schematic of a N-bit length PVTMC is shown in Fig. 6.2, which is composed of two delay-channels named as *top (t)* and *bottom (b)*. In each delay-channel, N 2–1 MUXs are connected with each other by their inputs and outputs. The *select* signals of the MUXs are provided by a binary vector $\mathbf{S} = \{\mathbf{S}_t, \mathbf{S}_b\}$, where $\mathbf{S} = \mathbb{B}^{2N}$ and $\mathbb{B} = \{0, 1\}$. Similar with that shown in Fig.6.1, the timing signal of interest is the interval between the two pulse inputs: in_t and in_b. These two pulses propagate through two delay-channels t and b via the paths determined by the select signals: $\mathbf{S}_t = \{s_t^0, s_t^1 \dots s_t^{N-1}\}$ and $\mathbf{S}_b = \{s_b^0, s_b^1 \dots s_b^{N-1}\}$. For example, if the select signal s_t^i is '1', in_t signal will propagate through the '1' channel of i^{th} MUX in the *top* delay-channel. The outputs of these two channels are denoted as o_t and o_b, which are used as the inputs of a D Flip-flop (DFF). The DFF can generate a binary output d depending on which signal between o_t and o_b arrives earlier than the other one. If denoting the timing interval between o_t and o_b with t_{out} and that between in_t and in_b with t_{in}, then for a given *select* vector S, d can be defined with Eq.6.1, where $t_{out} = f(t_{in}, S)$ means that t_{out} is jointly determined by t_{in} and selector vector S.

$$d = \begin{cases} 1 & t_{out} = f(S, t_{in}) > 0 \\ 0 & t_{out} = f(S, t_{in}) < 0 \end{cases} \tag{6.1}$$

6.3.2 Statistical characteristics of t_{out}

In a PVTMC circuit, the signal that can be directly measured by user is only the output d, which is derived from the sign of t_{out} in Eq. 6.1. In this section, we firstly study the statistical characteristics of t_{out}.

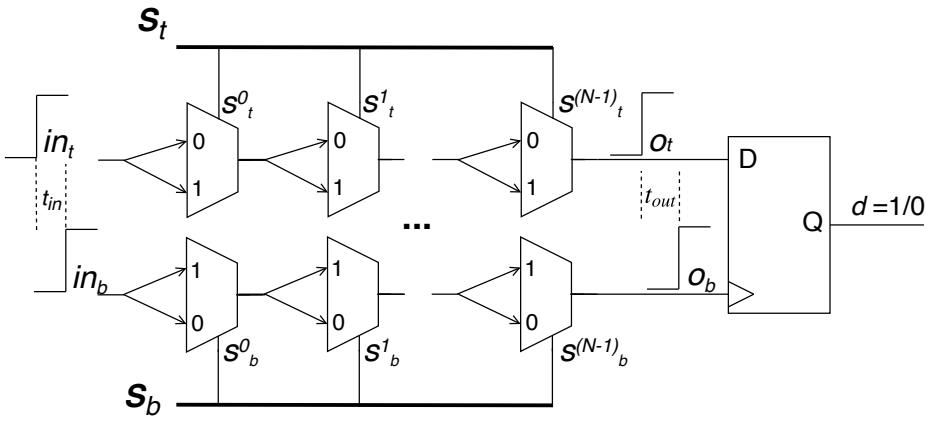


Figure 6.2: Schematic of PVTMC.

6.3.2.1 Simulation setup:

It has been reported that the physical parameters of CMOS transistors like threshold voltage (V_{th}) and effective channel length/width (L_{eff}/W_{eff}) follow an approximately Gaussian distribution [85] [86], adhering to this rule, we firstly used HSPICE simulation to study the statistical characteristics of t_{out}. Our simulation adopted the publicly-available Predictive Technology Model (PTM) for a 45 nm process [69]. The process variations of transistors were set as follows: the mean value of each process variation-impacted parameter was set as its nominal value in the transistor model library; the standard deviation $\sigma_{V_{th}}$ of threshold voltage was calculated with Eq. 6.2, in which $A_{V_{th}}$ was set to be 1.8 $mV\ \mu m$ according to [87] [88]; the standard deviation of L_{eff} and W_{eff} were set to be 10% of their nominal values following [88].

$$\sigma_{V_{th}} = \frac{A_{V_{th}}}{\sqrt{W_{eff} L_{eff}}} \tag{6.2}$$

We used Monte Carlo simulation to extract the t_{out} of a PVTMC of 64 stages ($N = 64$) for $50,000$ random *select* vectors. In this simulation, the input t_{in} was set to be 0, i.e., $t_{out} = f(S, t_{in} = 0)$. The probability density function (PDF) of the simulation results are shown in Fig. 6.3, which indicates that the distribution of t_{out} is approximately Gaussian in the range $[-80\ ps, +80\ ps]$ with a mean value $\mu_{t_{out}}$

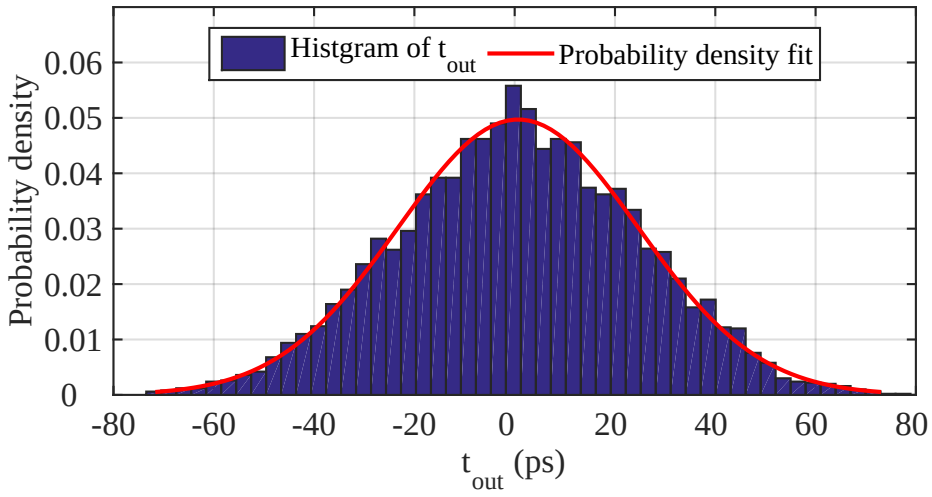


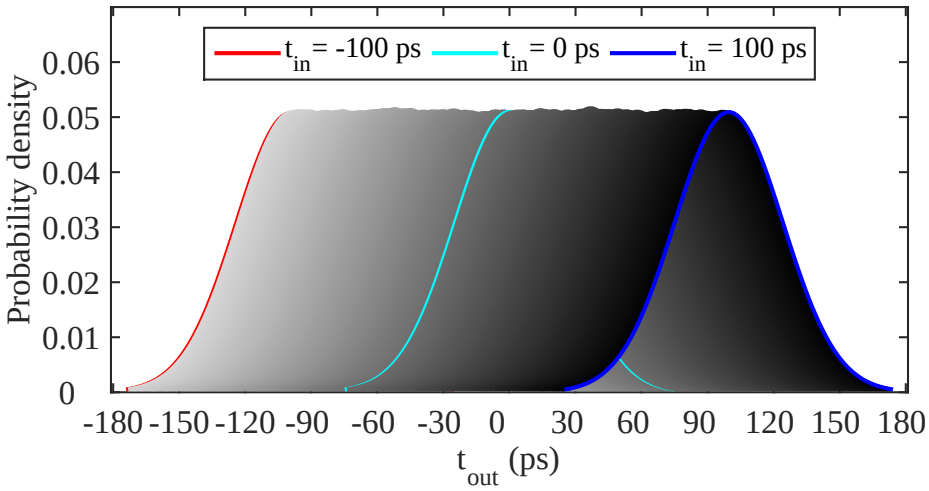
Figure 6.3: The t_{out} values follow approximately Gaussian distribution.

near 0 ps, these results are in agreement with previous studies [89]. According to Eq. 6.1, these simulation results demonstrate that the probability of $d = 1$ ($P(d = 1)$) is approximately equal with that of $d = 0$ ($P(d = 1)$), yet both are $\sim 50\%$.

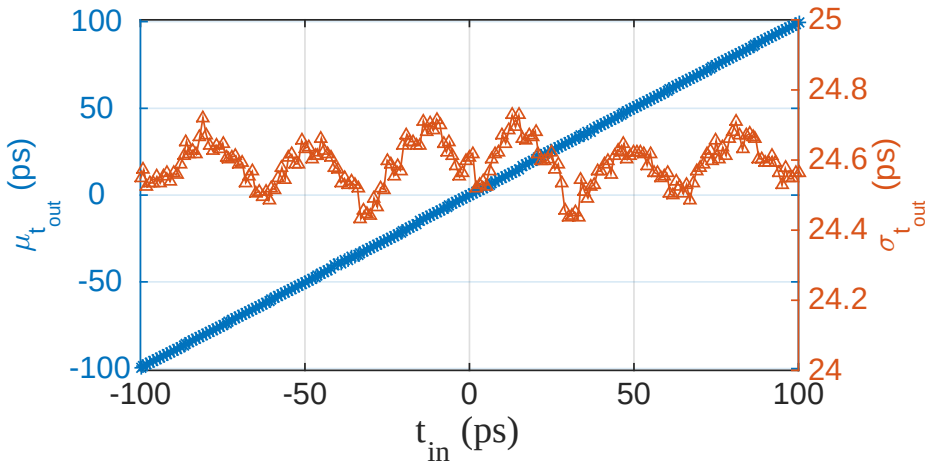
We then swept t_{in} from -100 ps to 100 ps with a 1 ps step for the same *select* vectors and extracted 201 sets of t_{out} values. The fitted PDFs of each set of t_{out} values[2] are presented in Fig. 6.4a, which shows that the deviation of t_{in} from 0 ps changes the distribution of t_{out}, most notably the mean value $\mu_{t_{out}}$. The mean value $\mu_{t_{out}}$ and standard deviation $\sigma_{t_{out}}$ of each set of t_{out} values are plotted in Fig. 6.4b. It can be found that in each fitted curve, $\mu_{t_{out}}$ approximately equals with t_{in} and the standard deviation $\sigma_{t_{in}}$ is relatively stable within the range $[24.4\ ps, 24.8\ ps]$. Therefore, for simplicity, we use Eq. 6.3 to define the PDF of t_{out}.

$$PDF(t_{out}) = \frac{exp^{-\frac{(t_{out}-\mu_{t_{out}})^2}{2\sigma^2_{t_{out}}}}}{\sqrt{2\pi}\sigma_{t_{out}}} \tag{6.3}$$

[2]Here, each set of t_{out} values are from the same t_{in}.



(a) The approximately Gaussian distribution of t_{out} for different t_{in} inputs from -100 ps to 100 ps with a 1 ps step.



(b) The mean value $\mu_{t_{out}}$ and $\sigma_{t_{out}}$ of t_{out}.

Figure 6.4: Simulation results showing the distribution of t_{out}. (a) plots the fitted distribution of t_{out} values for different t_{in} from -100 ps to 100 ps. (b) plots the mean value $\mu_{t_{out}}$ and the standard variance for each fitted curve in (a).

6.4 Algorithms Used in PVTMC Measurements

6.4.1 Principles of PVTMC

From Fig. 6.4a it can be learned that the deviation of t_{in} from 0 ps is propagated to the statistical distribution of t_{out}. More specially, the $\mu_{t_{out}}$ is approximately equal with t_{in}, as shown in Fig. 6.4b. However, it is infeasible to directly use $\mu_{t_{out}}$ to measure t_{in}. As defined in Eq. 6.1, the output value d is in accordance with the sign of t_{out}, therefore,

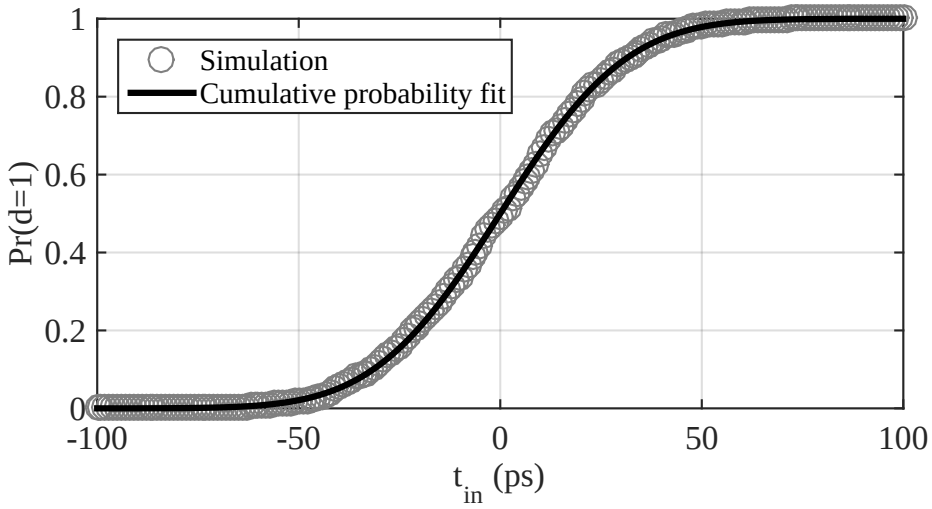


Figure 6.5: The probability of generating $d = 1$ by PVTMC circuit for different t_{in} input values is in agreement with the CDF of Gaussian distribution $t_{in} \sim \mathcal{N}(0, \sigma_{t_{out}})$. Thus, for a given t_{in}, its value can be measured with the probability $P(d = 1)$.

if there exists a relationship between d values and t_{in}, we can use this relationship to measure t_{in} since we can directly collect d as digital outputs. From Fig. 6.4a, we can see that for different t_{in} values, the probability of generating $d = 1$ ($P(d = 1)$) also changes. In order to analyze this, we plotted $P(d = 1)$ and corresponding t_{in} in Fig. 6.5 with gray circle. The fitted curve of cumulative distribution function (CDF) of the normal distribution $t_{in} \sim \mathcal{N}(0, \sigma^2_{t_{out}})$ is also plotted with black line. It can be found that these two data sets have good agreement with each other. In Appendix 6.8, we provide a formal proof that the $P(d = 1)$ for a given input timing signal t_{in} is equal with the CDF of the normal distribution $t_{in} \sim \mathcal{N}(0, \sigma^2_{t_{out}})$.

Based on the observed relationship between $P(d = 1)$ and t_{in}, the following two steps are proposed to measure t_{in}:

1. *CDF model formulation*: the detailed algorithm is shown in Proc. 12. For a PVTMC circuit, a set of golden inputs $\mathbf{T} = (t^0_{in}, t^1_{in}, \ldots, t^{l-1}_{in})$ is applied, the

corresponding probability values $\mathbf{P}=\left(P^0, P^1, \ldots, P^{l-1}\right)$[3] of generating $d = 1$ are extracted across a set of random *select* vectors $\mathbf{S}$. The $\mathbf{P}$ and $\mathbf{T}$ are used to formulate the CDF model of $CDF(t_{in})$ (line 12 in Proc. 2). Note that this step is common in most measurement instruments, which serves for the calibration purpose.

2. *Timing measurement*: the detailed algorithm is shown in Proc. 3. When the PVTMC circuit is used to measure a unknown timing signal t_{in}, another set of random *select* vectors are applied, the probability of observed $P(d = 1) =$ is used together with $CDF(t_{in})$ to extract the measurement result $\hat{t}_{in}$[4].

Based on the observations from Fig. 6.5, PVTMC can provide two types of timing measurements:

1. *Coarse-grained measurement:* The timing signals with magnitude near or beyond the two "tail bounds" the CDF curve can be coarsely measured. For example, when t_{in} is beyond or on the very left side of the CDF curve in Fig. 6.5, it is very possible that we will always get $P(d = 1) = 0$. With the coarse-grained measurement, we can figure out the general range of the input timing signal t_{in}.

2. *Fine-grained measurement:* This is an important feature of the proposed "model formulation" approach, a model can describe the relationship between $P(d = 1)$ and t_{in} as continuous variables, which means that once a model is obtained, user can avoid the laborious characterization and enrollment for discrete t_{in} values. Moreover, in Sec. 6.4.2, we will show that a generic CDF model can be used by a group of PVTMC instances of the same size and fabricated together.

---

[3]For simplicity, we only use $d = 1$ in this chapter, however, $d = 0$ can be used for measurement as well.

[4]Note that there exists error in any measurement, here we use t_{in} and $\hat{t}_{in}$ to denote the difference.

Procedure 2 CDF model formulation: build the cumulative distribution function of $CDF(t_{in})$ for future use.

Input: A PVTMC circuit
Input: A set of l golden inputs $\mathbf{T} = \left(t_{in}^0, t_{in}^1, \ldots, t_{in}^{l-1}\right)$
Input: A set of m random *select* vectors $\mathbf{S} = \left(S^0, S^1, \ldots S^{m-1}\right)$
Output: $CDF(t_{in})$

Initiate a probability value collector $\mathbf{P} = \left(P^0, P^1, \ldots, P^{l-1}\right)$
for $i := 0$ **to** $l - 1$ **do**
  for $j := 0$ **to** $m - 1$ **do**
    apply t_{in}^i on PVTMC
    apply S^j and extract d from the PVTMC circuit
    if $d == 1$ **then**
      $P^i = P^i + 1$
    end if
  end for
  $P^i = P^i/m$
end for
$CDF(t_{in}) = normcdf\_fit(\mathbf{P}, \mathbf{T})$
return $CDF(t_{in})$

Procedure 3 Random sample-based timing measurement with the formulated function $CDF(t_{in})$.

Input: A PVTMC circuit and its CDF model
Input: A input time signal t_{in} A set of k random *select* vectors $\mathbf{S} = \left(S^0, S^1, \ldots S^{k-1}\right)$

Output: Measured result $\hat{t}_{in}$

Initiate a counter $c = 0$
for $i := 0$ **to** $k - 1$ **do**
  apply S^j and extract d from the PVTMC circuit
  apply t_{in}^i on PVTMC
  if $d == 1$ **then**
    $c = c + 1$
  end if
end for
$P(d = 1) = c/k$
$\hat{t}_{in} = norminv(P(d = 1), CDF(t_{in}))$
return $\hat{t}_{in}$

6.4.2 Random search-based timing measurement

To make better use of the model, we propose a random search-based method to measure timing signals, in which the $P(d = 1)$ is formulated by applying random

select vectors. In order to validate the performance of the random search-based timing measurement, we simulated 100 PVTMC instances with 45 nm technology node [69]. The simulation setup of process variations was the same as Sec. 6.3.2. The t_{in} of each instance was swept from -100 ps to 100 ps with 1 ps as the step and used as golden data for validation purpose. For each t_{in} value, $50,000$ random *select* vectors were applied, with which $P(d = 1)$ were formulated. Note that this setup accomplishes the job of *CDF model formulation* (line 12 of Proc. 2).

In order to improve the efficiency of the timing measurement, one solution is to find a generic CDF model for different PVTMC instances, i.e., if a set of PVTMC chips are fabricated with the same technology node, a generic CDF model can be used for them. In our validation, we found that the CDF models of these 100 PVTMC instances had high similarity with each other: the μ and σ values of them fluctuated within a very small range $[-0.2ps,\ +0.2ps]$, which is in accordance with Fig. 6.4b. Therefore, we adopted the CDF model of one specific PVTMC instance as a generic one in our validation[5].

6.4.3 Binary search-based timing measurement

Although the random search-based method is easy and straightforward to implement, it requires users to repeatedly sample enough d outputs to formulate $P(d = 1)$, which consumes more resources like power and latency. The main reason is because that if fewer randomly applied *select* vectors are used, then the collected d values might not yield enough discriminating information about $P(d = 1)$ for a given t_{in}. To mitigate this issue, we introduce a more efficient method based on machine learning modeling and binary search.

---

[5]Note that using a generic CDF model might decrease the measurement accuracy of PVTMC, but this impact is trivial since the small statistical difference between different CDF models, as confirmed by our experimental results.

6.4.3.1 Using machine learning to model PVTMC

To model a PVTMC circuit, we first introduce four parameters: t^i_{t0} and t^i_{d0}, which denote the propagation delay of channel 0 in the i^{th} *top* and *bottom* MUXs, and t^i_{t1} and t^i_{d1} that are used to denote the propagation delay of channel 1 in the i^{th} *top* and *bottom* MUXs. With these definitions, we formulate two parameters α^i and β^i in Eq. 6.4:

$$\begin{aligned} \alpha^i &= \frac{(t^i_{t0} + t^i_{t1}) + \hat{s}^i_t(t^i_{t0} - t^i_{t1})}{2} \\ \beta^i &= \frac{(t^i_{b0} + t^i_{b1}) + \hat{s}^i_b(t^i_{b0} - t^i_{b1})}{2} \end{aligned} \tag{6.4}$$

where α^i and β^i represent the propagation delay of the i^{th} *top* and *bottom* MUXs respectively, and $\hat{s}^i$ is a linear conversion of s^i: $\hat{s}^i_{t/b} = 1 - 2 * s^i_{t/b}$, where t/b stands for *top* or *bottom*. Based on Eq. 6.4, we can rewrite the relationship between t_{out}, t_{in}, and S with Eq. 6.5:

$$t_{out} = f(S, t_{in}) = t_{in} + \sum_{0}^{N-1} \alpha_i - \sum_{0}^{N-1} \beta_i \tag{6.5}$$

Eq. 6.5 indicates that there exists a linear additive model between the *select* vectors S and t_{out}. Since it is infeasible to directly measure t_{out} and formulate these parameters like α_i and β_i, we chose to use a modeling method. More specifically, a Support Vector Machine (SVM) model is trained with a set of known *select* vectors $\mathbf{S}$ and corresponding output value set $\mathbf{d}$, as shown in line 1 of Proc. 4. Note that the objective of training the data with SVM model is to get a model $\mathbf{P}_{model}$ that can mimic the behavior of PVTMC circuit, i.e., predicting the d values for unknown *select* vectors. Although the SVM model is not used to directly characterize the exact value of t_{out}, however, if it can predict d for any given *select* vector with a high success rate, this means that $\mathbf{P}_{model}$ can formulate a numeric value mp_t_{out} (mp_t_{out}: model predicted t_{out}) that has high linearity with the actual t_{out}, whose sign is correlated with d [90][91].

In order to validate the performance of using SVM classifier for t_{out} prediction, we used a training set with $1,000$ random *select* vectors and corresponding d values when $t_{in} = 0$[6] from HSPICE simulation. The prediction accuracy of the trained model is shown in Fig. 6.6a, it can be found that while more samples are used for training, the model prediction rate grows higher. When $1,000$ training samples are used, the prediction accuracy gets close to 100%. To further study the characteristics of the SVM model, we also extracted the corresponding t_{out} for $50,000$ *select* vectors from HSPICE simulation[7]. The $\mathbf{P}_{model}$ predicted and simulated t_{out} are depicted in Fig. 6.6b, which shows that even the two sets of data are of different scales, the correlation coefficient between them is very high, i.e., there exists a linear relationship between them: $mp\_t_{out} = coe \times$(simulated t_{out}), where *coe* denotes the correlation coefficient. Proc. 4 shows how to calculate *coe* with the results from Proc. 2. This inspires that once we get the $\mathbf{P}_{model}$ for a PVTMC circuit, we can use it to study the statistical characteristics of the circuit without repeatedly measuring it. More specifically, for a given *select* vector, we can formulate its d based on the sign of mp_t_{out}.

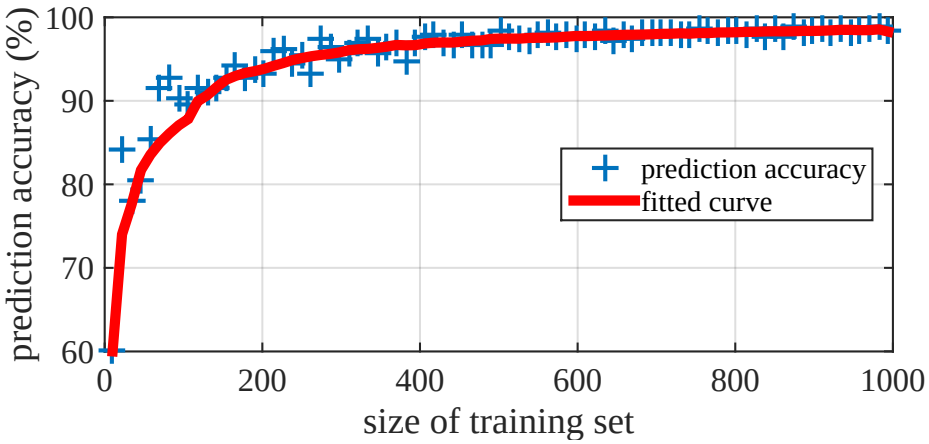
6.4.3.2 Binary search-based measurement

Though the randomly sampled *select* vectors can help with formulating $P(d = 1)$, most of them are redundant with each other in providing useful information to characterize t_{in}. In this section, we will show how to use the trained Machine Learning model $\mathbf{P}_{model}$ to measure t_{in}, which avoids repetitively applying *select* vectors. As shown in Fig. 6.4a, the change of t_{in} from 0 to other values shifts the probability distribution of t_{out}, i.e., $\mu_{t_{out}}$. To visualize how to use this feature in measuring t_{in}, we

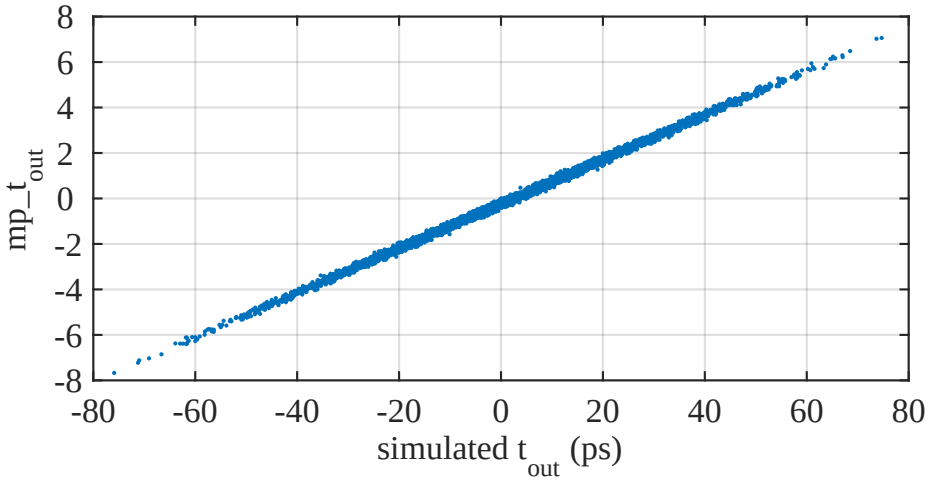
---

[6]Following the phenomenon shown in Fig. 6.4a, the probability distribution model for a specific PVTMC circuit is same for any t_{in}. Therefore, without loss of generality, we chose $t_{in} = 0$ in our model training.

[7]Note that it is difficult to extract the practical t_{out} from a real circuit, here it is only used for visualize the relationship between mp_t_{out} and simulated t_{out} values in Fig. 6.6b.



(a) Prediction rate increases while more training samples used.



(b) Linear relationship between mp_t_{out} and HSPICE simulated t_{out} for $50,000$ *select* vectors.

Figure 6.6: (a) shows that while more training samples are used, the prediction rate of $\mathbf{P}_{model}$ becomes higher, and 1000 training samples achieves around 100% prediction rate. (b) presents simulated t_{out} by HSPICE and model predicted t_{out} by $\mathbf{P}_{model}$. It can be found that there exist a linear relationship between these two parameters.

depict two curves in Fig. 6.7 as an example: when $t_{in} = 0$, suppose that there exists two *select* vectors: S_i and S_j which correspond to $t_{out} = 0$ and $t_{out} = -\tau$, respectively. S_i is the boundary between $d = 1$ and $d = 1$. When the t_{in} becomes τ, the PDF curve will be shifted to right by τ (similar to Fig. 6.4a), and the *select* vector S_j will become the new boundary between $d = 1$ and $d = 0$.

Procedure 4 Train a Machine Learning model for a PVTMC circuit and formulate the coefficient (*coe*) between actual t_{out} and model predicted t_{out}: mp_t_{out}.

Input: A set of random *select* vectors **S** and corresponding output values **d** from a PVTMC circuit with $t_{in} = 0$.

Output: The $\mathbf{P}_{model}$ for PVTMC circuit.

Output: The coefficient *coe* between mp_t_{out} and actual t_{out}.

1: $\mathbf{P}_{model} \leftarrow \mathbf{SVM}(\mathbf{S}, \mathbf{d})$ {train a SVM model for the PVTMC circuit}

2: $\mu_{mp\_t_{out}} = \text{avg}\langle \mathbf{P}_{model}, \mathbf{S} \rangle$ {mean mp_t_{out} calculated by $\mathbf{P}_{model}$}

3: $\sigma_{mp\_t_{out}} = \text{var}\langle \mathbf{P}_{model}, \mathbf{S} \rangle$ {variance of mp_t_{out} calculated by $\mathbf{P}_{model}$}

4: The distribution of mp_t_{out} follows Gaussian distribution and expressed with $mp\_t_{out} \sim \mathcal{N}(\mu_{mp\_t_{out}}, \sigma^2_{mp\_t_{out}})$

5: Since the cumulative distribution function $CDF(t_{in})$ expresses the relationship between t_{in} and $\sigma_{t_{out}}$, therefore we can use $CDF(t_{in})$ from Proc. 2 to calculate $\sigma_{t_{out}}$

6: **return** $coe = \frac{\sigma_{t_{out}}}{\sigma_{mp\_t_{out}}}$

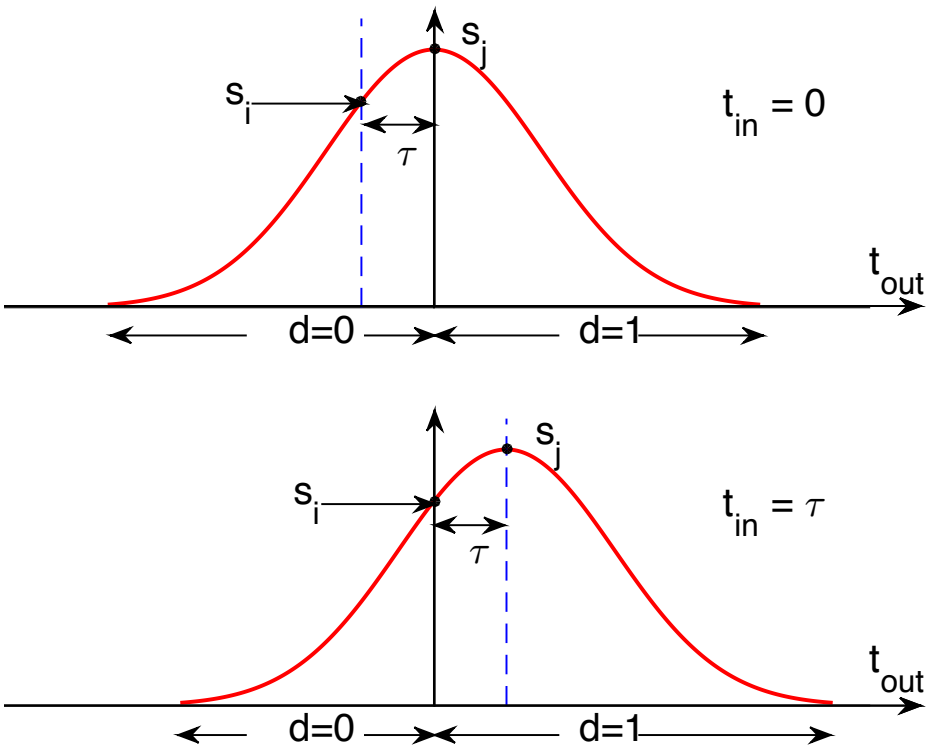


Figure 6.7: Visualization of the PDF curve shift caused by t_{in}. It can be found that, while t_{in} changes, the boundary of $d = 1$ and $d = 0$ will shift correspondingly.

From the results shown in Fig. 6.3 and Fig. 6.4a, we know that the t_{in} is just the boundary that divides the two zones: $d = 1$ and $d = 0$. Therefore, if we can use a

new search method to quickly determine the *select* vector that corresponds to this boundary, then we can use its corresponding mp_t_{out} and *coe* to formulate the actual t_{in} that we need to measure. As we mentioned before, the distribution of t_{out} has boundary, i.e., within a range $[-80ps, 80ps]$. Therefore, we can apply binary search to quickly find the location of corresponding *select* vector S_j. Suppose the resolution we want to achieve with binary search is ρ, the distribution range of t_{out} is Δt, and the minimum number of searches need to achieve this k, then we will have:

$$\frac{\Delta t}{2^k} <= \rho \tag{6.6}$$

Therefore, the number of binary searches we will need is: $log_2(\frac{\Delta t}{\rho})$. For example, with the 45nm technology nodes, the simulated time differences of outputs is within the range from -80ns to +80ns, therefore, to achieve a resolution of 0.5ps, theoretically, we just need to do 9 searches. In next section, we will show how to use the model $\mathbf{P}_{model}$ and binary search together to measure t_{in}.

6.4.3.3 Combining Machine Learning model and binary search

From the discussion above, it inspires us that the measurement of t_{in} can be converted into the search for S_j, the boundary of $d = 1$ and $d = 0$. With S_j, we can formulate its correspondingly t_{out} from $\mathbf{P}_{model}$. Proc. 4 and 5 present the formulation of $\mathbf{P}_{model}$ and how to use it in binary search. The details can be divided into the following steps:

1. Given a set of random *select* vectors $\mathbf{S}$ and corresponding output values $\mathbf{d}$ from a PVTMC circuit while $t_{in} = 0$, a Machine Learning model $\mathbf{P}_{model}$ is trained, as shown in line 1 of Proc. 4.

Procedure 5 Apply binary search with $\mathbf{P}_{model}$ and *coe* for a PVTMC circuit from Proc. 4 to measure t_{in}.

Input: $\mathbf{P}_{model}$ and *coe* for a PVTMC circuit from Proc. 4.
Input: A set of random *select* vectors $\mathbf{S}$.
Input: A timing signal t_{in} to be measured.
Output: The measurement result $\hat{t}_{in}$.

1: $\mathbf{mp\_t}_{out} \leftarrow \mathbf{SVM}(\mathbf{S}, \mathbf{P}_{model})$ {extract a set of mp_t_{out} with $\mathbf{P}_{model}$ and $\mathbf{S}$}
2: $\hat{\mathbf{S}} \leftarrow \text{sort}(\mathbf{mp\_t}_{out}, \mathbf{S})$ {sort $\mathbf{mp\_t}_{out}$ from low to high and record the corresponding *select* vectors with order in $\hat{\mathbf{S}}$= $\{\hat{S}_0, \hat{S}_1, \hat{S}_1, \ldots \hat{S}_{n-1}\}$}
3: $low = 0$
4: $high = n - 1$
5: **while** $(low < (high - 1))$ **do**
6: d_{low} =PVTMC($\hat{S}_{low}, t_{in}$)
7: d_{high} =PVTMC($\hat{S}_{high}, t_{in}$)
8: $mid = \frac{low+high}{2}$
9: d_{mid} =PVTMC($\hat{S}_{high}, t_{in}$)
10: **if** $(d_{mid} = d_{low})$ **then**
11: $low = mid$
12: **else**
13: $high = mid$
14: **end if**
15: **end while**
16: $\widehat{mp\_t_{out}} = \mathbf{SVM}(\hat{S}_{mid}, \mathbf{P}_{model})$
17: Return $\hat{t}_{in} = -\widehat{mp\_t_{out}} \times coe$

2. Two parameters $\mu_{mp\_t_{out}}$ and $\sigma_{mp\_t_{out}}$ are calculated with $\mathbf{P}_{model}$ and $\mathbf{S}$. Note that with these two parameters, the distribution of mp_t_{out} can be expressed as $mp\_t_{out} \sim \mathcal{N}(\mu_{mp\_t_{out}}, \sigma^2_{mp\_t_{out}})$.

3. With the $CDF(t_{in})$ from Proc. 2, the actual $\sigma_{t_{out}}$ can be calculated following the math relationship presented in Appendix 6.8. Then, the coefficient *coe* between mp_t_{out} and actual t_{out} can be calculated, as shown in line 6 of Proc. 4.

4. A set of random *select* vectors $\mathbf{S}$ is used with $\mathbf{P}_{model}$ to formulate $\mathbf{mp\_t}_{out}$, with which the order of $\mathbf{S}$ is sorted and recored in $\hat{\mathbf{S}}$ based on their mp_t_{out} values from low to high.

5. Binary search is used to find a *select* vector $\hat{S}_{mid}$ (as shown from line 5 to 15 in Proc. 5), and its $\widehat{mp\_t_{out}}$ is calculated with $\mathbf{P}_{model}$.

6. The measurement result for t_{in} is calculated with $\hat{t}_{in} = -\widehat{mp\_t_{out}} \times coe$.

6.5 Performance Evaluation with HSPICE Simulation

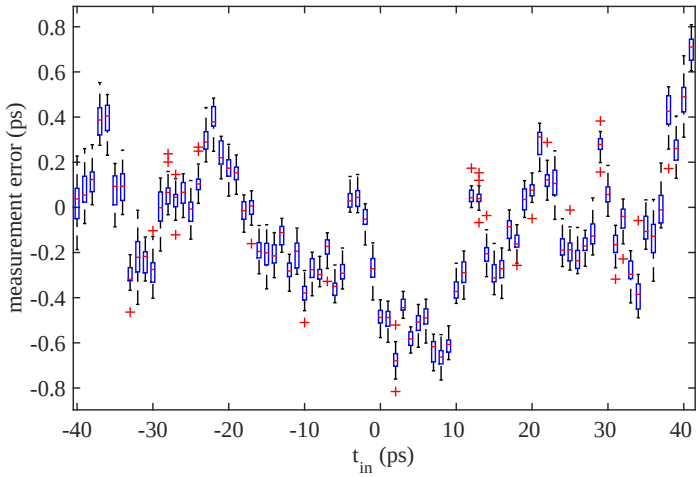
6.5.1 Timing Measurements

Following the random sampling-based algorithms proposed in Proc. 2 and Proc. 3, we firstly validated the relationship between the number of samples and the measurement precision. For each timing input, we did $1,000$ independent measurements and in each measurement, we used 100 and 500 d samples out of the $50,000$ values to formulate $P(d = 1)$, respectively. The experimental results are shown in Fig. 6.8, which demonstrates that: 1) Using more samples can improve measurement accuracy, as doing this makes the formulated $P(d = 1)$ closer to the fitted golden model. 2) The proposed method achieves higher measurement accuracy of $< 0.5\ ps$.

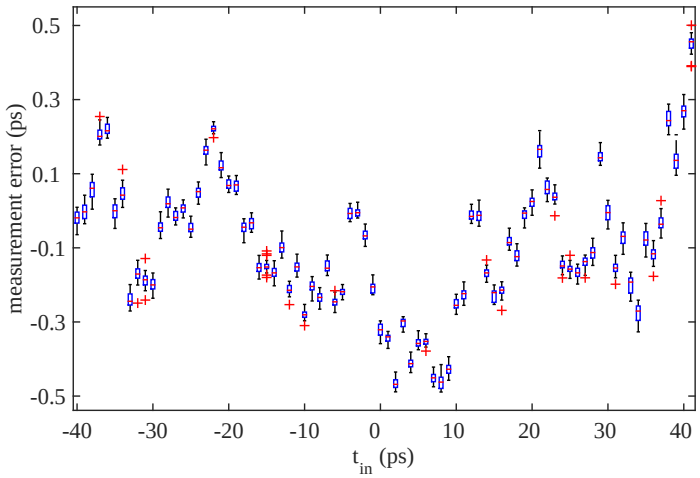
6.5.2 Performance evaluation under different environmental conditions

Besides process variations, the performance of the CMOS circuit is also impacted by environmental conditions like temperature variations and supply voltage changes. Since the measurement method is based on random search to formulate the probability of $d = 1$ which is determined by t_{out}, therefore, considering that the proposed PVTMC scheme leverages the microscopic process variations to measure timing signals, the changes of environmental conditions may flip the sign of t_{out} from $+$ to $-$, or vice versa. As a result, the probability of generating $d = 1$ by PVTMC might be changed by the fluctuations of environmental conditions and the measurement accuracy will be impacted.

To thoroughly explore this potential issue, we simulated the distribution of t_{out} across a variety of environmental conditions. The distribution of $50,000$ t_{out} values



(a) 100 d samples were used to formulate $P(d = 1)$.



(b) 500 d samples were used to formulate $P(d = 1)$.

Figure 6.8: Plot shows the timing measurement errors of random search. $1,000$ iterations were analyzed for each t_{in}. (a) In each iteration, 100 d values were collected to formulate $P(d = 1)$, (b) In each iteration, 500 d values were collected to formulate $P(d = 1)$.

simulated under golden condition $(25°C, 1.1\ V)$[8] are plotted with gray color in Fig. 6.9. We then changed the supply voltage from 1.1 V to 0.9 V, 1.0 V, 1.2 V and 1.3 V, the temperature from $25°C$ to $50°C$, $75°C$ and $100°C$ respectively in our simulation. The simulated t_{out} values under these changed conditions are plotted in black color in Fig. 6.9. As the environmental condition changes, the sign of t_{out} flips between

[8]For clarity, we only showed $t_{in} = 0ps$ here.

− and + as we predicted. However, these t_{out} values whose signs were flipped by environmental conditions are 1) only t_{out} values that are close to 0; 2) also following an approximately Gaussian distribution. These results indicate that the probability of generating $d = 1$ and $d = 0$ by the PVTMC circuit are not impacted even under the changed environmental conditions. We then repeated the timing measurement with these flipped t_{out} values and found there is no performance degradation compared to Fig. 6.8a (For brevity, these results are omitted).

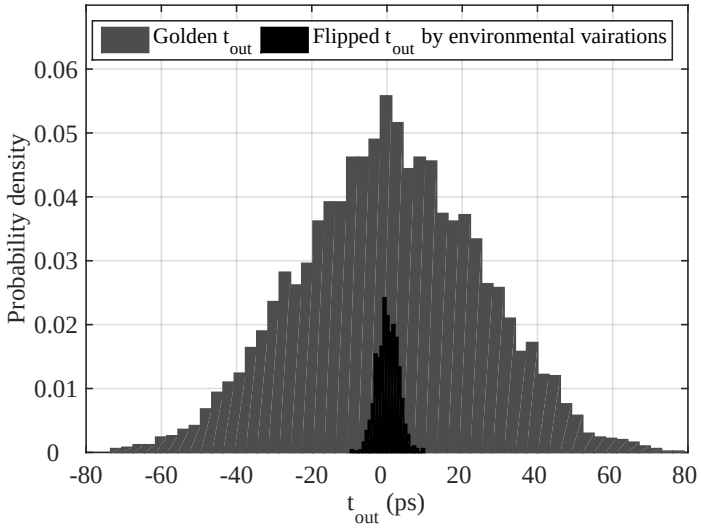


Figure 6.9: Simulation results of t_{out} at standard environmental condition $(25°C, 1.1\ V)$ and a set of changed conditions.

6.5.3 Impact of circuit length

As shown in Fig. 6.4a and Fig. 6.5, the measurable range of PVTMC circuit is determined by the distribution of t_{out}. In other words, the capability of timing measurement range is determined by the additive delay difference from MUXs. Therefore, a longer delay chain can introduce more process variations yet a larger measurable range. To validate this feature, we simulated the distribution of t_{out} for a set of PVTMC instances of different length: $N = 128$, 64, 32 and 16 respectively, the fitted PDF curves of simulation results are shown in Fig. 6.10. The results demonstrate

that as the PVTMC circuit becomes longer, it can cover a larger measurable time range. However, this also increases the power consumption and area overhead of the circuit, which should be taken into consideration by the designer.

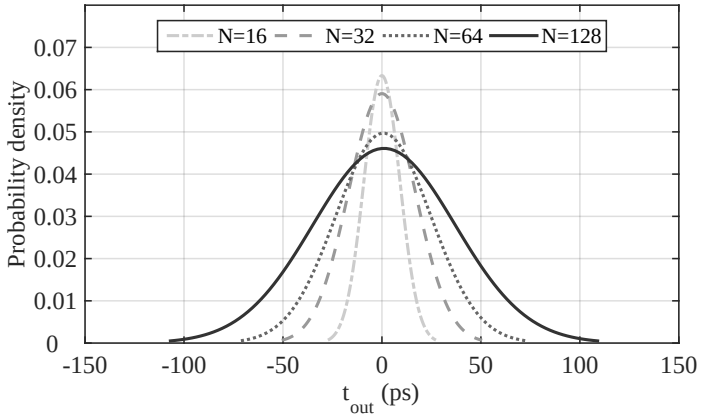


Figure 6.10: Plot shows that a PVTMC circuit with more MUXs can coverage a larger measurable time range.

6.5.4 Compatibility with different CMOS technology models

In order to evaluate the compatibility of the proposed PVTMC scheme, we implemented the circuit with all 12 CMOS technology models on the PTM website [69], from 7 nm to 180 nm. The process variations of the designed circuit were introduced following the same setup in Sec. 6.3. We found the proposed methodology is compatible with all these CMOS technology models due to the existence of process variations. However, since the propagation delay of more advanced technology models like 7 nm becomes relatively smaller, the measurable time range is also smaller. For example, in our simulation, we found that a 64 stage PVTMC built with 7 nm process can only measure timing signals within $[-10\ ps, 10\ ps]$.

6.6 Validation with FPGA Implementations

In order to validate the practical performance of the proposed PVTMC method, we implemented the PVTMC circuit of 16-bit (N=16) on an Atlys FPGA trainer

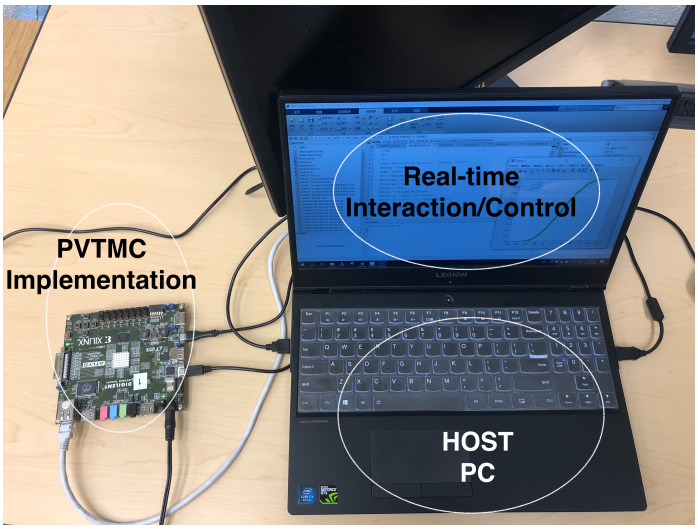


Figure 6.11: The experimental setup for PVTMC circuit implemented on FPGA. The HOST PC is used for programming and controlling the FPGA chip, i.e., sending and receiving data. In the HOST PC, a Matlab-based framework is built, which can visualize and analyze the measurement data in real-time. Note that the real-time control is an important component to realize the binary search-based measurement.

board from Digilent [92], which has a Spartan-6 FPGA chip from Xilinx. We used Xilinx ISE for the design and programming and UART port on the board for sending and collecting data. The experimental setup is as shown in Fig. 6.11, in which we built a framework that can collect and visualize the measurement data, this setup facilitates real-time interaction/control and the binary search-based measurement.

6.6.1 Implementation of PVTMC circuit on FPGA

The first step of our experiment is to implement the PVTMC circuit on FPGA. Unlike ASIC implementation or HSPICE simulation, which can built the circuit with transistors. In FPGA, the basic component that we can use is look-up table (LUT). In the implementation, each MUX in Fig. 6.2 was instantiated with a LUT. Note that the bias introduced by placement and routing will significantly decrease the performance of PVTMC, for example, a biased PVTMC implementation on FPGA may only generate $d = 1$ (or $d = 0$). To alleviate the possible bias, we chose to use

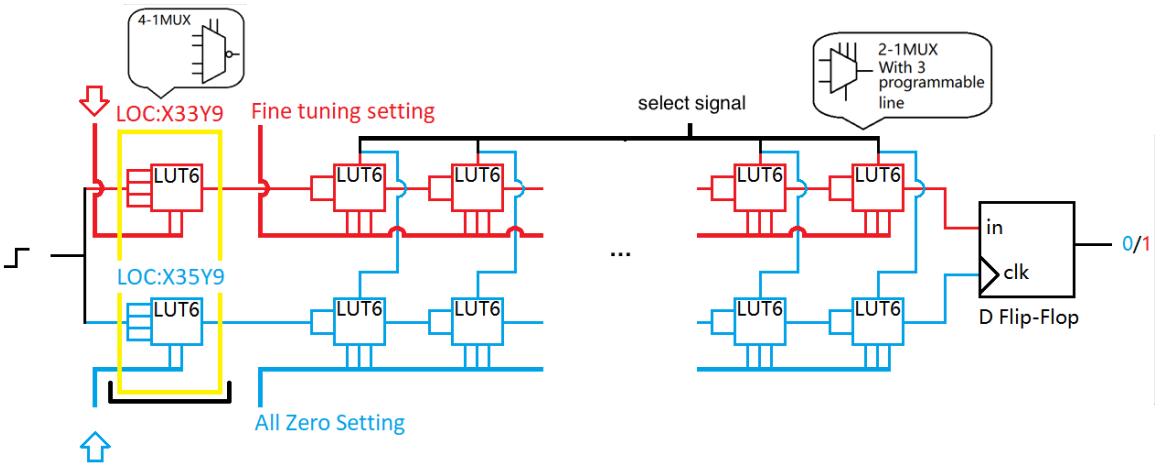


Figure 6.12: Schematic of PVTMC implementation on FPGA.

one out of the 4 available LUTs in each slice[9], which helps with the replication of placement and routing for each MUX. Each LUT on the Spartan-6 FPGA chip has six input pins and two output pins. To avoid the asymmetric circuit layout caused by automatic placement and routing, we took the following steps in our design:

1. The MUX components of PVTMC were instantiated as hard macro to facilitate replication;

2. The two delay-lines were placed over two adjacent rows of slices on the FPGA to guarantee symmetric layout;

3. In each LUT, three input pins were used for two inputs and one *select* signal of the MUX, other three pins are used for tuning the bias between two delay-lines, as shown in Fig. 6.12. The usage of the tuning setup will be introduced in next section.

[9]There are 4 LUTs in each slice of Spartan-6 FPGA named from A to D, and we used LUTA in our implementation

6.6.2 t_{in} generation within FPGA

The second problem that needs to be solved is generating t_{in} for testing. In our experiment to evaluate the performance of PVTMC implementations on FPGA, we built a on-chip t_{in} generator instead of using other external instruments. There are several reasons that we tested PVTMC in this way:

- Unlike HSPICE that can provide t_{in} of any wanted resolution in the simulation environment, it is impractical to provide extremely small t_{in} with the precision of sub-picoseconds from an external function generator,
- Even if a pair of pulses with a small gap in between can be generated by an external function generator, the connection and transmission of these signals will inevitably introduce extra bias. For example, if an external function generator is used, the two cables connecting it and the IOs pins of FPGA will introduce bias to the timing signals for test.

To solve this problem, we chose to generate the timing signals within the FPGA chips to test our design. More specifically, the timing difference between in_t and in_b are derived from the the process variations between hardware implementations of the same digital topology. The schematic of the signal generator is shown on the left side of Fig. 6.12, which is composed of two 4-1 MUXs. To make the design more compact, each 4-1 MUX is implemented within a LUT. These two MUXs share the same input signal, which is a pulse and their output are connected with the PVTMC circuit to provide in_t and in_b.

Since each LUT in Spartan-6 FPGA has 6 input inputs, 4 of them were used as input pins in our design and other 2 pins were used for *channel selection* purpose. Therefore, there are in total $2^4 = 16$ channel configurations and the propagation delay difference between them are used to generate t_{in}. Due to the placement and routing scheme of FPGA, there will exist bias between the two delay-lines after connecting

the signal generator with PVTMC. Thus, we first calibrated the PVTMC circuit to erase the bias to guarantee the measurement accuracy.

The tuning was realized through real-time interaction between FPGA and HOST PC. Note that according to our analysis, a symmetric PVTMC will show a $P(d = 1) = 50\%$ while $t_{in} = 0$. To realize this, the procedure of tuning PVTMC is as follows: We first applied `00` as the select signals for both 4-1 MUXs, which share the same input pulse (i.e., the input timing signal $t_{in} = 0$). We then applied 1024 random *select* vectors, and the corresponding output d values were collected to formulate $P(d = 1)$. The HOST PC kept changing the tuning bits until the observed probability of $d = 1$ get close to 50%[10]. Moreover, in order to make the tuning easier, the tuning bits of the bottom delay-line was set to be all-zeros, and that of the top delay-line were used to do the fine tuning.

To validate the performance of the PVTMC circuit, we quantified the t_{in} generated by the two MUXs. More specifically, each MUX was included as a stage of a ring oscillator (RO) and the overall propagation delay of the RO was measured. Note that in this way we cannot directly get the propagation delay of each channel of the MUX, but it provides the delay difference between them. For example, the propagation delay of channel `00` is X*ps* shorter than that of channel `01` for MUX1, while the propagation delay of channel `00` is Y*ps* shorter than that of channel `01` for MUX2. Since we have fine tuned the delay-line of PVTMC to be balanced while connecting with `00` channels of both MUXs, and then they can provide a t_{in} of $(X - y)ps$ when configured as `0101`.

6.6.3 Experimental results of FPGA validation

All the 16 t_{in} from MUXs were used in our experiment to formulate the CDF model (line 12 of Proc. 2), the fitted curve and data points are shown in Fig. 6.13.

[10]Note that in practice, it might be difficult (if not impossible) to get an exact $P(d = 1) = 50\%$ with a random set of *select* vectors, while tuning the PVTMC implementation. Thus in our experiment, we chose to set [49%, 51%] as an acceptable range.

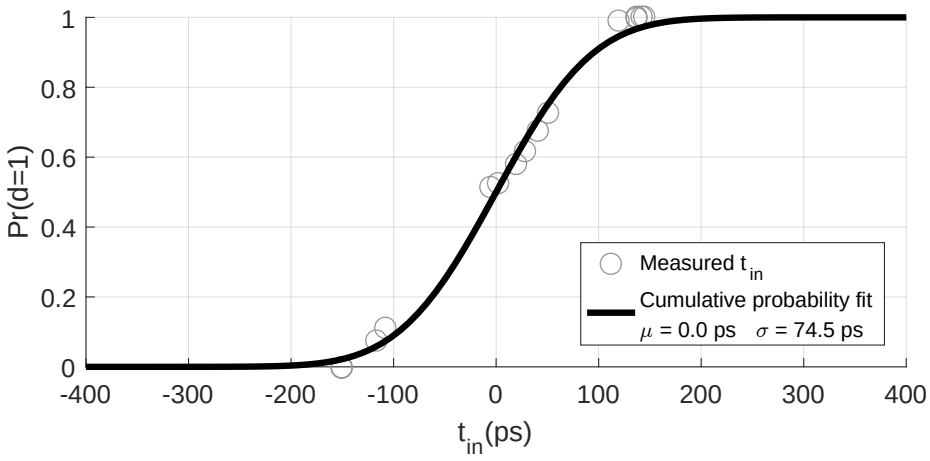


Figure 6.13: 16 measured t_{in} values and the corresponding fitted CDF curve. It can be found that the values of $\mu_{t_{out}}$ and $\sigma_{t_{out}}$ formulated from this curve are $0ps$ and $74.5ps$, respectively.

6.6.3.1 Binary search-based measurement

To implement binary search, we first trained a SVM model for the PVTMC circuit following following Proc. 5, with which we generated the CDF model. More specifically, we used different number of d samples in training the model, the results are as shown in Fig. 6.14. It can be found that, even though the number of training sample differs, the CDF models and fitted curves generated by them only have negligible difference, such as the statistical features like μ and σ. This means that we can always use fewer samples to do the model training and measurement, which significantly reduces the workload while using the binary search-based method. We then used the setup shown in Fig.6.11 to conduct the real-time measurement. Since only 1024 samples as used, therefore, we only need at most $log_2(1024) = 10$ searches to finish the measurement. While the errors are shown in Fig. 6.15, with which we can draw three conclusions:

- The measurement error is as expected: i.e., larger measurement error occurs on the two tails with larger t_{in} values.
- Using fewer number of samples (i.e., 1024) to train a model does not degrade the performance of binary search-based measurement.

- While the input timing signals are within the range [-50ps, 50ps], the measurement error is no more than 1.5ps.

Note that the measurement resolution was characterized with the smallest delay difference that can be provided by our designed signal generator on FPGAs. In practice, we believe that a timing signal of shorter duration can also be measured with PVTMC. Moreover, the resolution and measurable scope of FPGA implementation are both larger than that of HSPICE simulation, this is because the lowest-level logic components that can be manipulated within FPGA are LUT and slice, which have much larger propagation delays than MUX implemented with ASIC.

6.7 Conclusion

In this chapter, we proposed a timing measurement circuit that can measure timing signals of sub-picosecond. Unlike previously proposed timing measurement circuit such as TDC, whose performance is impacted by the process variations of CMOS transistors. Instead, this new design constructively leverages the process variations to measure timing signals. The performance of the proposed circuit structure has been validated with a variety of CMOS technology nodes and across different environmental conditions, as well as with FPGA implementations. The experimental results demonstrate that PVTMC achieves good performance on these platforms and is fully compatible with these reconfigurable devices with good performance.

6.8 Cumulative probability distribution of t_{in}

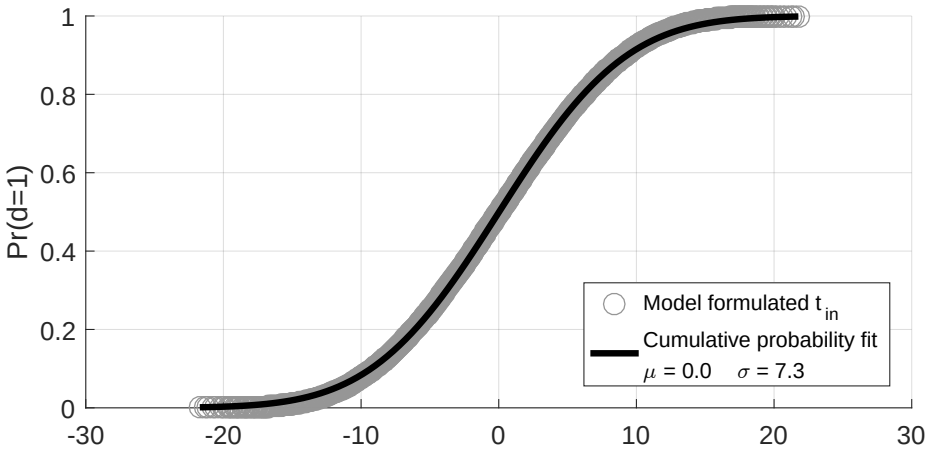
Lemma 1. *Given a timing input t_{in}, the probability of generating $d = 1$, i.e., $P(d = 1)$ from a PVTMC circuit is equal with the cumulative distribution function (CDF) of the normal distribution $t_{in} \sim \mathcal{N}(0, \sigma_{t_{out}})$.*

Proof.

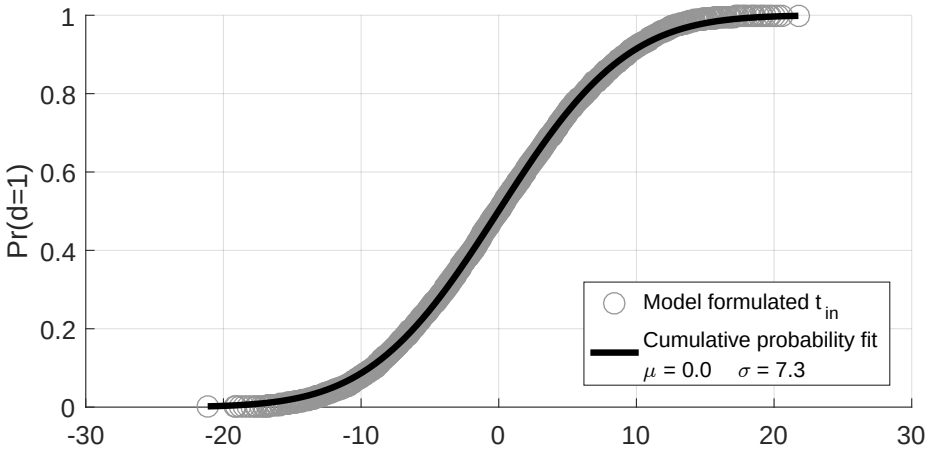
$$
\begin{aligned}
P(d=1) &= P(t_{out} > 0) \\
&= 1 - P(t_{out} \leq 0) \\
&= 1 - P\Big(\frac{t_{out} - t_{in}}{\sigma_{t_{out}}} \leq -\frac{t_{in}}{\sigma_{out}}\Big) \\
&= 1 - F\Big(-\frac{t_{in}}{\sigma_{t_{out}}}\Big) \\
&= 1 - \frac{1}{2}\Big[1 + erf\Big(\frac{t_{in}}{\sigma_{out}\sqrt{2}}\Big)\Big]
\end{aligned}
\tag{6.7}
$$

where $erf()$ denotes the commonly used Gaussian error function [93] and F is the CDF of the standard normal distribution $\mathcal{N}(0, 1)$.

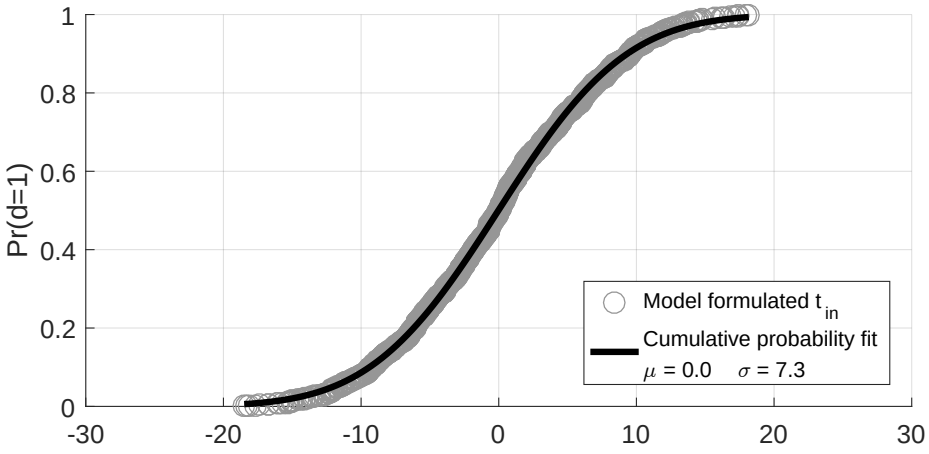
Hence proved. □



(a) Training SVM model with all 65536 d samples.



(b) Training SVM model with 8192 random d samples.



(c) Training SVM model with all 1024 random d samples.

Figure 6.14: The CDF curve fit results from SVM models trained with different number of training samples. It can be found that is the training samples are randomly selected, using fewer number of training samples does not degrade the performance.

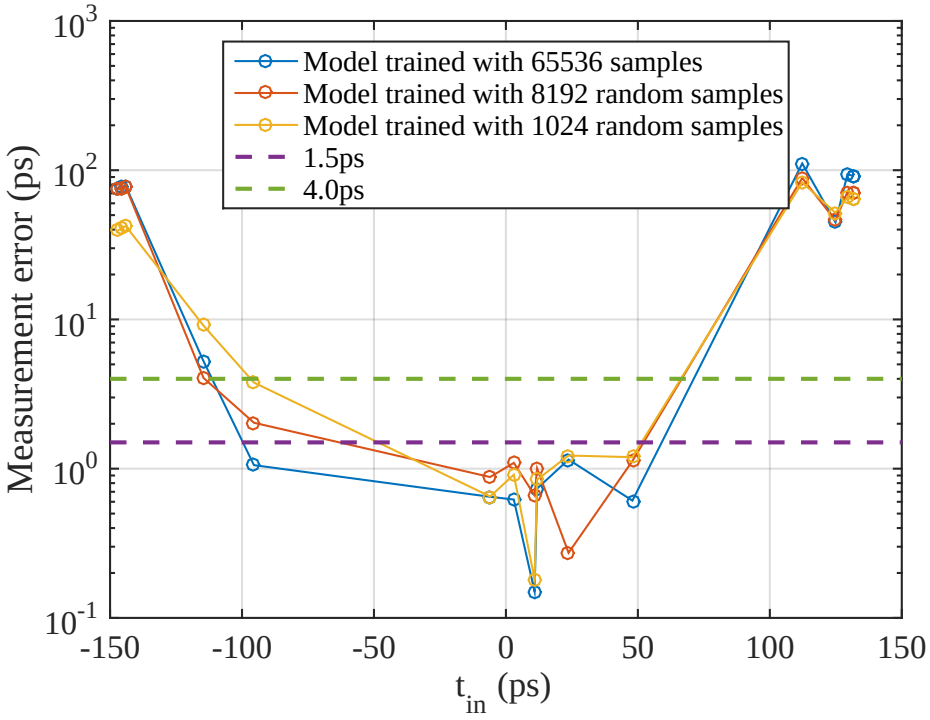


Figure 6.15: The timing measurement results with PVTMC model trained with different amount of samples. It can be found that using fewer number of training samples does not degrade the measurement accuracy.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

Driven by Moore's law, the geometrics of semiconductor device have experienced continuous physical scaling in the past few decades, while such advancement has significantly facilitated the development of electronic devices, but also proposed new challenges for hardware designers. This dissertation presents some of our recent work in advancing the measurement of timing signals with very high resolution, i.e., $< 0.5ps$. Specifically, we focus our investigation on the Time Difference Circuits in three perspectives: methodology, design, and digital realization. The contribution of each chapter in this dissertation is briefly summarized as follows:

Chapter 1 mainly introduces the background and motivation of developing time difference (TD) circuits. More importantly, the main design challenges like overhead, power consumption, digital realization, and throughput are also investigated.Many important applications of TD circuit are also introduced in this chapter.

In Chapter 2 first discusses various methodologies for TD signal processing with details. Beyond that, this chapter provides a systematic perspective of the TD function modeling, we also predicts that TD circuits can offer a new and promising way for mixed-mode systems to deal with the challenges of conventional voltage/current-mode designs.

Chapter 3 reviews and compares several conventional TDC architectures. Based the strengths and weaknesses of these existing techniques, we propose a novel compact algorithmic TDC design: CATDC that realizes algorithmic time-to-digital conversion with small overhead [2]. Both simulation and fabricated silicon are used for the

performance validation of this design, which outperforms the state-of-the-art TDC designs.

A common issue that significantly limits the resolution of existing TDC circuits is the process variations from circuit fabrication. To solve this problem, we propose another novel TDC design in Chapter 4: configurable compact algorithmic TDC (CCATDC), which is an extension of the CATDC work in Chapter 3 but with advancement in reconfigurability [83]. Specifically, a Backpropagation-based Machine Learning framework is proposed and used to mimic the real-time circuit calibration and measurement in an in-field use scenario.

Besides the resolution issues, most conventional TDC designs use analog device, which suffers from the integration with digital systems for data post-processing. Chapter 5 focuses on solving this problem by proposing an all-digital TDA architecture that is synthesizable with modern digital design library. Moreover, this TDA design provides reconfigurable gain values for timing amplification and is scalable with the most prevalent CMOS technology nodes [94].

Following the research directions in Chapter 4 and 5, Chapter 6 jointly solves the two problems with a novel design: PVTMC, which for the first time, constructively leverages the process variations from CMOS fabrication to measure timing signals [95][96]. As our theory contribution, two algorithms: random- and binary-search are proposed to improve the performance of PVTMC. We also show that the structure of PVTMC is scalable with the most prevalent CMOS technology nodes and FPGA chips due to the widely existence of process variations. Tab. 7 summarizes the comparison between our work and the state-of-the-art TDC implementations with ASIC and FPGA.

Future work of this dissertation mainly has three directions: 1) Fabricating the proposed timing measurement architectures with ASICs and validate their practical performance with both on-chip and off-chip measurements. 2) Both the ASIC and

Resolution	All-digital Algorithmic TDC with TDA [94] [2]	PVTMC [95]	Reference[4]	Reference[97]
Design	ASIC	ASIC (7-180nm) FPGA(45/28nm)	ASIC(16nm)	FPGA(40nm)
Application	Single-shot Wide range	Periodic pulse	ADPLL phase	TOF HEP
Resolution	Limited by the TDA(2ps 16nm)	Sub-picosecond	8.4ps	10ps
Accuracy	10bit<1ps	<1ps	1-2ps	12.8ps
PVT	< 5% TDA gain error	PVT resistant	2.3MHz/$^{\circ}C$	2ps varied nominal value
Calibration Technique	Post processing modeling	One-time pre-measurement enrollment	Multi-core; Coarse/fine tuning adjust	Code density test and pipelined OTFC for bin alignment

Table 7.1: Comparison between our work and the state-of-the-art ASIC and FPGA TDC implementations.

FPGA implementations for these timing measurement circuits can be applied for different applications on jitter mornitering, ToF measurment. 3) Looking for collaboration opportunities with industry to commercialize these proposed architectures and integrate them with real-world applications. Therefore, there might be two more publications about topics on VLSI, and instrument and measurement respectively based on the future work.

BIBLIOGRAPHY

[1] Boris Murmann. Adc performance survey 1997-2012. *Online, October*, 2012.

[2] Shuo Li and Christopher D Salthouse. Compact algorithmic time-to-digital converter. *Electronics Letters*, 51(3):213–215, 2015.

[3] Lei Qiu, Yuanjin Zheng, and Liter Siek. Multichannel time skew calibration for time-interleaved adcs using clock signal. *Circuits, Systems, and Signal Processing*, 35(8):2669–2682, 2016.

[4] T. Tsai, M. Yuan, C. Chang, C. Liao, C. Li, and R. B. Staszewski. 14.5 a 1.22ps integrated-jitter 0.25-to-4ghz fractional-n adpll in 16nm finfet cm0s. In *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, pages 1–3, Feb 2015.

[5] Man Keun Kang and Tae Wook Kim. Cmos ir-uwb receiver for 9.7-mm range finding in a multipath environment. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 59(9):538–542, 2012.

[6] Clyde F Coombs. *Electronic instrument handbook*. McGraw-Hill Professional, 1999.

[7] A-J Annema, Bram Nauta, Ronald Van Langevelde, and Hans Tuinhout. Analog circuits in ultra-deep-submicron cmos. *IEEE Journal of Solid-State Circuits*, 40(1):132–143, 2005.

[8] John Monteith and Mike Unsworth. *Principles of environmental physics*. Academic Press, 2007.

[9] Salim Alahdab, Antti Mäntyniemi, and Juha Kostamovaara. A 12-bit digital-to-time converter (dtc) with sub-ps-level resolution using current dac and differential switch for time-to-digital converter (tdc). In *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, pages 2668–2671. IEEE, 2012.

[10] Jun-Seok Kim, Young-Hun Seo, Yunjae Suh, Hong-June Park, and Jae-Yoon Sim. A 300-ms/s, 1.76-ps-resolution, 10-b asynchronous pipelined time-to-digital converter with on-chip digital background calibration in 0.13-μm cmos. *IEEE Journal of Solid-State Circuits*, 48(2):516–526, 2012.

[11] Roubik Gregorian, Kenneth W Martin, and Gabor C Temes. Switched-capacitor circuit design. *Proceedings of the IEEE*, 71(8):941–966, 1983.

[12] Chris Toumazou and John B Hughes. *Switched-currents: an analogue technique for digital technology*. Number 5. Iet, 1993.

[13] David A Johns and Ken Martin. *Analog integrated circuit design*. John Wiley & Sons, 2008.

[14] Dan Elson, Jose Requejo-Isidro, Ian Munro, Fred Reavell, Jan Siegel, Klaus Suhling, Paul Tadrous, Richard Benninger, Peter Lanigan, James McGinty, et al. Time-domain fluorescence lifetime imaging applied to biological tissue. *Photochemical & Photobiological Sciences*, 3(8):795–801, 2004.

[15] Takashi Fujii and Tetsuo Fukuchi. *Laser remote sensing*. CRC press, 2005.

[16] David J Kinniment and JV Woods. Synchronisation and arbitration circuits in digital systems. In *Proceedings of the Institution of Electrical Engineers*, volume 123, pages 961–966. IET, 1976.

[17] Dan I Porat. Review of sub-nanosecond time-interval measurements. *IEEE Transactions on Nuclear Science*, 20(5):36–51, 1973.

[18] Gordon W Roberts and Mohammad Ali-Bakhshian. A brief introduction to time-to-digital and digital-to-time converters. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 57(3):153–157, 2010.

[19] Robert Bogdan Staszewski, Dirk Leipold, Chih-Ming Hung, and Poras T Balsara. Tdc-based frequency synthesizer for wireless applications. In *Radio Frequency Integrated Circuits (RFIC) Symposium, 2004. Digest of Papers. 2004 IEEE*, pages 215–218. IEEE, 2004.

[20] Wanghua Wu, Robert Bogdan Staszewski, and John R Long. A 56.4-to-63.4 ghz multi-rate all-digital fractional-n pll for fmcw radar applications in 65 nm cmos. *IEEE Journal of solid-state circuits*, 49(5):1081–1096, 2014.

[21] Ralph O Dubayah and Jason B Drake. Lidar remote sensing for forestry. *Journal of Forestry*, 98(6):44–46, 2000.

[22] H Sang Lee, IH Hwang, James D Spinhirne, and V Stanley Scott. Micro pulse lidar for aerosol and cloud measurement. In *Advances in Atmospheric Remote Sensing with Lidar*, pages 7–10. Springer, 1997.

[23] Augusto Ronchini Ximenes, Preethi Padmanabhan, and Edoardo Charbon. Mutually coupled time-to-digital converters (tdcs) for direct time-of-flight (dtof) image sensors. *Sensors*, 18(10):3413, 2018.

[24] David Tyndall, Bruce R Rae, David Day-Uei Li, Jochen Arlt, Abigail Johnston, Justin A Richardson, and Robert K Henderson. A high-throughput time-resolved mini-silicon photomultiplier with embedded fluorescence lifetime estimation in 0.13mum cmos. *IEEE transactions on biomedical circuits and systems*, 6(6):562–570, 2012.

[25] B Albert Griffin, Stephen R Adams, Jay Jones, and Roger Y Tsien. Fluorescent labeling of recombinant proteins in living cells with flash. *Methods in enzymology*, 327:565–578, 2000.

[26] Satoshi Karasawa, Toshio Araki, Miki Yamamoto-Hino, and Atsushi Miyawaki. A green-emitting fluorescent protein from galaxeidae coral and its monomeric version for use in fluorescent labeling. *Journal of Biological Chemistry*, 278(36):34167–34171, 2003.

[27] Chang-Cheng You, Oscar R Miranda, Basar Gider, Partha S Ghosh, Ik-Bum Kim, Belma Erdogan, Sai Archana Krovi, Uwe HF Bunz, and Vincent M Rotello. Detection and identification of proteins using nanoparticle–fluorescent polymer chemical nosesensors. *Nature Nanotechnology*, 2(5):318–323, 2007.

[28] Derick G Wansink, Wouter Schul, Ineke Van Der Kraan, Bas Van Steensel, Roel Van Driel, and Luitzen De Jong. Fluorescent labeling of nascent rna reveals transcription by rna polymerase ii in domains scattered throughout the nucleus. *Journal of Cell Biology*, 122:283–283, 1993.

[29] JR Lakowitz. Principles of fluorescence spectroscopy, springer. *New York, NY*, 1999.

[30] Philippe IH Bastiaens and Anthony Squire. Fluorescence lifetime imaging microscopy: spatial resolution of biochemical processes in the cell. *Trends in cell biology*, 9(2):48–52, 1999.

[31] Joseph R Lakowicz. Fluorescence anisotropy. In *Principles of fluorescence spectroscopy*, pages 291–319. Springer, 1999.

[32] W Becker, H Hickl, C Zander, KH Drexhage, M Sauer, S Siebert, and J Wolfrum. Time-resolved detection and identification of single analyte molecules in micro-

capillaries by time-correlated single-photon counting (tcspc). *Review of scientific instruments*, 70(3):1835–1841, 1999.

[33] Richard M Ballew and JN Demas. Error analysis of the rapid lifetime determination method for the evaluation of single exponential decays. *Anal. Chem.;(United States)*, 61(1), 1989.

[34] Shuo Li and Christopher Salthouse. Digital-to-time converter for fluorescence lifetime imaging. In *Instrumentation and Measurement Technology Conference (I2MTC), 2013 IEEE International*, pages 894–897. IEEE, 2013.

[35] Ahmet T Erdogan, Richard Walker, Neil Finlayson, Nikola Krstajić, Gareth OS Williams, John M Girkin, and Robert K Henderson. A cmos spad line sensor with per-pixel histogramming tdc for time-resolved multispectral imaging. *IEEE Journal of Solid-State Circuits*, 2019.

[36] Dongyi Liao, Hechen Wang, Fa Foster Dai, Yang Xu, Roc Berenguer, and Sara Munoz Hermoso. An 802.11 a/b/g/n digital fractional-n pll with automatic tdc linearity calibration for spur cancellation. *IEEE Journal of Solid-State Circuits*, 52(5):1210–1220, 2017.

[37] Jen-Chien Hsu and Chauchin Su. Bist for measuring clock jitter of charge-pump phase-locked loops. *IEEE Transactions on Instrumentation and Measurement*, 57(2):276–285, 2008.

[38] J-P Jansson, Annti Mantyniemi, and Juha Kostamovaara. A cmos time-to-digital converter with better than 10 ps single-shot precision. *IEEE Journal of Solid-State Circuits*, 41(6):1286–1296, 2006.

[39] Jozef Kalisz. Review of methods for time interval measurements with picosecond resolution. *Metrologia*, 41(1):17, 2003.

[40] Zeng Cheng, Xiaoqing Zheng, M Jamal Deen, and Hao Peng. Recent developments and design challenges of high-performance ring oscillator cmos time-to-digital converters. *IEEE Transactions on Electron Devices*, 63(1):235–251, 2015.

[41] Christopher D Salthouse, Fred Reynolds, Jenny M Tam, Lee Josephson, and Umar Mahmood. Quantitative measurement of protease activity with correction of probe delivery and tissue absorption effects. *Sensors and Actuators B: Chemical*, 138(2):591–597, 2009.

[42] Min Zhang, Hai Wang, and Yan Liu. A 7.4 ps fpga-based tdc with a 1024-unit measurement matrix. *Sensors*, 17(4):865, 2017.

[43] Hai Wang, Min Zhang, and Yan Liu. High-resolution digital-to-time converter implemented in an fpga chip. *Applied Sciences*, 7(1):52, 2017.

[44] Ion Vornicu, Ricardo Carmona-Galán, and Ángel Rodríguez-Vázquez. Time interval generator with 8 ps resolution and wide range for large tdc array characterization. *Analog Integrated Circuits and Signal Processing*, 87(2):181–189, 2016.

[45] AM Abas, Alex Bystrov, DJ Kinniment, OV Maevsky, Gordon Russell, and AV Yakovlev. Time difference amplifier. *Electronics Letters*, 38(23):1437–1438, 2002.

[46] Stefano Russo, Nicola Petra, Davide De Caro, Giancarlo Barbarino, and Antonio GM Strollo. A 41 ps asic time-to-digital converter for physics experiments. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 659(1):422–427, 2011.

[47] Justin Richardson, Richard Walker, Lindsay Grant, David Stoppa, Fausto Borghetti, Edoardo Charbon, Marek Gersbach, and Robert K Henderson. A 32×

32 50ps resolution 10 bit time to digital converter array in 130nm cmos for time correlated imaging. In *Custom Integrated Circuits Conference, 2009. CICC'09. IEEE*, pages 77–80. IEEE, 2009.

[48] Jussi-Pekka Jansson, Vesa Koskinen, Antti Mantyniemi, and Juha Kostamovaara. A multichannel high-precision cmos time-to-digital converter for laser-scanner-based perception systems. *IEEE Transactions on Instrumentation and Measurement*, 61(9):2581–2590, 2012.

[49] Minjae Lee and Asad A Abidi. A 9 b, 1.25 ps resolution coarse–fine time-to-digital converter in 90 nm cmos that amplifies a time residue. *IEEE Journal of solid-state circuits*, 43(4):769–777, 2008.

[50] Antonio H Chan and Gordon W Roberts. A jitter characterization system using a component-invariant vernier delay line. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 12(1):79–95, 2004.

[51] Piotr Dudek, Stanislaw Szczepanski, and John V Hatfield. A high-resolution cmos time-to-digital converter utilizing a vernier delay line. *IEEE Journal of Solid-State Circuits*, 35(2):240–247, 2000.

[52] Tetsutaro Hashimoto, Hirotaka Yamazaki, Atsushi Muramatsu, Tomio Sato, and Atsuki Inoue. Time-to-digital converter with vernier delay mismatch compensation for high resolution on-die clock jitter measurement. In *VLSI Circuits, 2008 IEEE Symposium on*, pages 166–167. IEEE, 2008.

[53] Jussi-Pekka Jansson, Antti Mantyniemi, and Juha Kostamovaara. Synchronization in a multilevel cmos time-to-digital converter. *IEEE Transactions on Circuits and Systems I: regular papers*, 56(8):1622–1634, 2009.

[54] Shingo Mandai and Edoardo Charbon. A 128-channel, 9ps column-parallel two-stage tdc based on time difference amplification for time-resolved imaging. In *ESSCIRC (ESSCIRC), 2011 Proceedings of the*, pages 119–122. IEEE, 2011.

[55] Antti Mantyniemi, Timo Rahkonen, and Juha Kostamovaara. A cmos time-to-digital converter (tdc) based on a cyclic time domain successive approximation interpolation method. *IEEE Journal of Solid-State Circuits*, 44(11):3067–3078, 2009.

[56] Pekka Keranen and Juha Kostamovaara. Algorithmic time-to-digital converter. In *NORCHIP, 2013*, pages 1–4. IEEE, 2013.

[57] Daniele Marioli, Claudio Narduzzi, Carlo Offelli, Dario Petri, Emilio Sardini, and Andrea Taroni. Digital time-of-flight measurement for ultrasonic sensors. *IEEE Transactions on Instrumentation and Measurement*, pages 93–97, 1992.

[58] Nir Bar-Gill, Linh M Pham, Andrejs Jarmola, Dmitry Budker, and Ronald L Walsworth. Solid-state electronic spin coherence time approaching one second. *Nature communications*, 4:1743, 2013.

[59] Abdel S Yousif and James W Haslett. A fine resolution tdc architecture for next generation pet imaging. *IEEE Transactions on Nuclear Science*, pages 1574–1582, 2007.

[60] Stephan Henzler, Siegmar Koeppe, Winfried Kamp, Hans Mulatz, and Doris Schmitt-Landsiedel. 90nm 4.7 ps-resolution 0.7-lsb single-shot precision and 19pj-per-shot local passive interpolation time-to-digital converter with on-chip characterization. In *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pages 548–635. IEEE, 2008.

[61] Wei-Zen Chen and Po-I Kuo. A $\delta\sigma$ tdc with sub-ps resolution for pll built-in phase noise measurement. In *European Solid-State Circuits Conference, ESSCIRC Conference 2016: 42nd*, pages 347–350. IEEE, 2016.

[62] Vadim Gutnik and Anantha Chandrakasan. On-chip picosecond time measurement. In *VLSI Circuits, 2000. Digest of Technical Papers. 2000 Symposium on*, pages 52–53. IEEE, 2000.

[63] Chin-Cheng Tsai and Chung-Len Lee. An on-chip jitter measurement circuit for the pll. In *Test Symposium, 2003. ATS 2003. 12th Asian*. IEEE, 2003.

[64] S Tisa, A Lotito, A Giudice, and F Zappa. Monolithic time-to-digital converter with 20ps resolution. In *Solid-State Circuits Conference, 2003. ESSCIRC'03. Proceedings of the 29th European*. IEEE.

[65] Robert Bogdan Staszewski, Sudheer Vemulapalli, Prasant Vallur, John Wallberg, and Poras T Balsara. 1.3 v 20 ps time-to-digital converter for frequency synthesis in 90-nm cmos. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 53(3):220–224, 2006.

[66] Shekhar Borkar, Tanay Karnik, Siva Narendra, Jim Tschanz, Ali Keshavarzi, and Vivek De. Parameter variations and impact on circuits and microarchitecture. In *Proceedings of the 40th annual Design Automation Conference*, pages 338–342. ACM, 2003.

[67] Smruti R Sarangi, Brian Greskamp, Radu Teodorescu, Jun Nakano, Abhishek Tiwari, and Josep Torrellas. Varius: A model of process variation and resulting timing errors for microarchitects. *IEEE Transactions on Semiconductor Manufacturing*, 21(1):3–13, 2008.

[68] Matthew Z Straayer and Michael H Perrott. A multi-path gated ring oscillator tdc with first-order noise shaping. *IEEE Journal of Solid-State Circuits*, 2009.

[69] Wei Zhao and Yu Cao. New generation of predictive technology model for sub-45 nm early design exploration. *IEEE Transactions on Electron Devices*, 53(11):2816–2823, 2006.

[70] Masood Qazi, Mehul Tikekar, Lara Dolecek, Devavrat Shah, and Anantha Chandrakasan. Loop flattening & spherical sampling: Highly efficient model reduction techniques for sram yield analysis. In *Proceedings of the Conference on Design, Automation and Test in Europe*, pages 801–806. European Design and Automation Association, 2010.

[71] AM Abas, A Bystrov, DJ Kinniment, OV Maevsky, G Russell, and AV Yakovlev. Time difference amplifier. *Electronics Letters*, 38(23):1, 2002.

[72] B Dehlaghi, S Magierowski, and L Belostotski. Highly-linear time-difference amplifier with low sensitivity to process variations. *Electronics Letters*, 47(13):743–745, 2011.

[73] ANM Alahmadi, G Russell, and A Yakovlev. Time difference amplifier design with improved performance parameters. *Electronics letters*, 48(10):1, 2012.

[74] Toru Nakura, Shingo Mandai, Makoto Ikeda, and Kunihiro Asada. Time difference amplifier using closed-loop gain control. In *2009 Symposium on VLSI Circuits*, pages 208–209. IEEE, 2009.

[75] Shingo Mandai, Toru Nakura, Makoto Ikeda, and Kunihiro Asada. Cascaded time difference amplifier using differential logic delay cell. In *SoC Design Conference (ISOCC), 2009 International*, pages 194–197. IEEE, 2009.

[76] Marvin Lindner, Ingo Schiller, Andreas Kolb, and Reinhard Koch. Time-of-flight sensor calibration for accurate range sensing. *Computer Vision and Image Understanding*, 114(12):1318–1328, 2010.

[77] Masayoshi Terabe, Akito Sekiya, Takahiro Yamada, and Akira Fujimaki. Timing jitter measurement in single-flux-quantum circuits based on time-to-digital converters with high time-resolution. *IEEE Transactions on Applied Superconductivity*, 17(2):552–555, 2007.

[78] Song-Yu Yang, Wei-Zen Chen, and Tai-You Lu. A 7.1 mw, 10 ghz all digital frequency synthesizer with dynamically reconfigured digital loop filter in 90 nm cmos technology. *IEEE Journal of Solid-State Circuits*, 45(3):578–586, 2010.

[79] Michelle A Digman, Valeria R Caiolfa, Moreno Zamai, and Enrico Gratton. The phasor approach to fluorescence lifetime imaging analysis. *Biophysical journal*, 94(2):L14–L16, 2008.

[80] Andrew E Stevens, Richard P Van Berg, Jan Van der Spiegel, and Hugh H Williams. A time-to-voltage converter and analog memory for colliding beam detectors. *IEEE Journal of solid-state circuits*, 24(6):1748–1752, 1989.

[81] KwangSeok Kim, Wonsik Yu, and SeongHwan Cho. A 9 bit, 1.12 ps resolution 2.5 b/stage pipelined time-to-digital converter in 65 nm cmos using time-register. *IEEE Journal of Solid-State Circuits*, 49(4):1007–1016, 2014.

[82] Ahmed Elkholy, Tejasvi Anand, Woo-Seok Choi, Amr Elshazly, and Pavan Kumar Hanumolu. A 3.7 mw low-noise wide-bandwidth 4.5 ghz digital fractional-n pll using time amplifier-based tdc. *IEEE Journal of Solid-State Circuits*, 50(4):867–881, 2015.

[83] Shuo Li, Xiaolin Xu, and Wayne Burleson. Ccatdc: A configurable compact algorithmic time-to-digital converter. In *2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pages 501–506. IEEE, 2017.

[84] G. Edward Suh and Srinivas Devadas. Physical unclonable functions for device authentication and secret key generation. In *Proceedings of the 44th annual Design Automation Conference*, DAC '07, pages 9–14. ACM, 2007.

[85] Aseem Agarwal, David Blaauw, and Vladimir Zolotov. Statistical timing analysis for intra-die process variations with spatial correlations. In *Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design*, page 900. IEEE Computer Society, 2003.

[86] Wei Zhao, Frank Liu, Kanak Agarwal, Dhruva Acharyya, Sani R Nassif, Kevin J Nowka, and Yu Cao. Rigorous extraction of process variations for 65-nm cmos design. *IEEE Transactions on Semiconductor Manufacturing*, 22(1):196–203, 2009.

[87] Kelin J Kuhn. Reducing variation in advanced logic technologies: Approaches to process and design for manufacturability of nanoscale cmos. In *Electron Devices Meeting, 2007. IEDM 2007. IEEE International*, pages 471–474. IEEE, 2007.

[88] M Anis and M H Aburahma. Leakage current variability in nanometer technologies. In *System-on-Chip for Real-Time Applications, 2005. Proceedings. Fifth International Workshop on*, pages 60–63.

[89] Georg T Becker. On the pitfalls of using arbiter-pufs as building blocks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 34(8):1295–1307, 2015.

[90] Xiaolin Xu, Wayne Burleson, and Daniel E Holcomb. Using statistical models to improve the reliability of delay-based pufs. In *VLSI (ISVLSI), 2016 IEEE Computer Society Annual Symposium on*. IEEE, 2016.

[91] U Rührmair, J Sölter, Frank Sehnke, Xiaolin Xu, Ahmed Mahmoud, Vera Stoyanova, Gideon Dror, Jürgen Schmidhuber, Wayne Burleson, and Srinivas

Devadas. Puf modeling attacks on simulated and silicon data. *Information Forensics and Security, IEEE Transactions on*, 2013.

[92] Atlys Spartan. 6 fpga trainer board (limited time). *URL: http://store. digilentinc. com/atlys-spartan-6-fpga-trainer-board-limitedtime-see-nexys-video*.

[93] Larry C Andrews and Larry C Andrews. *Special functions of mathematics for engineers*. McGraw-Hill New York, 1992.

[94] Shuo Li and Wayne Burleson. Design of pvt-resistant all-digital time-domain amplifier with variable gain and wide operation range. *IEEE Transactions on Circuits and Systems II: Express Briefs*, under review.

[95] Shuo Li, Xiaolin Xu, and Wayne Burleson. Pvtmc: An all-digital sub-picosecond timing measurement circuit based on process variations. In *2019 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2019.

[96] Shuo Li, Xiaolin Xu, and Wayne Burleson. Design of an all-digital sub-picosecond timing measurement circuit based on process variations. In *2019 ACM/ESDA/IEEE Design Automation Conference (DAC),WIP*, 2019.

[97] Jun Yeon Won, Sun Il Kwon, Hyun Suk Yoon, Guen Bae Ko, Jeong-Whan Son, and Jae Sung Lee. Dual-phase tapped-delay-line time-to-digital converter with on-the-fly calibration implemented in 40 nm fpga. *IEEE transactions on biomedical circuits and systems*, 10(1):231–242, 2015.