T.C.
IŞIK UNIVERSTIY
SCHOOL OF GRADUATE STUDIES

MASTER THESIS
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
ELECTRICAL AND ELECTRONICS ENGINEERING PROGRAM

Moaamen Magdy Abdelrazek MOHAMED

TIME-DOMAIN HIGH SPEED ADC CIRCUITS

SUPERVISOR
Assoc. Prof. Dr. Ramazan KÖPRÜ

İSTANBUL, June 2025

T.C.
IŞIK UNIVERSITY
SCHOOL OF GRADUATE STUDIES

MASTER THESIS
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
ELECTRICAL AND ELECTRONICS ENGINEERING PROGRAM

Moaamen Magdy Abdelrazek MOHAMED
(23ELEC5004)

TIME-DOMAIN HIGH SPEED ADC CIRCUITS

SUPERVISOR
Assoc. Prof. Dr. Ramazan KÖPRÜ

İSTANBUL, June 2025

T.C.
IŞIK UNIVERSITY
SCHOOL OF GRADUATE STUDIES

MASTER'S THESIS
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
ELECTRICAL AND ELECTRONICS ENGINEERING PROGRAM

Moaamen Magdy Abdelrazek MOHAMED
(23ELEC5004)

TIME-DOMAIN HIGH SPEED ADC CIRCUITS

Date: 30/06/2025

Thesis Supervisor: Assoc. Prof. Dr. Ramazan KÖPRÜ/ Işik University

Jury Members:

Prof. Dr. Ümit GÜZ/ Işik University

Assoc. Prof. Dr. Lida KOUHALVANDI/ Doğuş University

İSTANBUL, June 2025

ÖZET

ZAMAN BÖLGESİ TABANLI YÜKSEK HIZLI ANALOG-SAYISAL DÖNÜŞTÜRÜCÜ DEVRELERİ

Bu tez, 100 nm CMOS teknolojisi kullanılarak gerçekleştirilmiş iki yüksek çözünürlüklü zaman-sayısal dönüştürücünün tasarımını, benzetimini ve performans analizini sunmaktadır. Farklı sinyal kontrolü ve kenar algılama ihtiyaçlarına yönelik olarak geliştirilen iki ayrı mimari strateji, nanosaniyenin altı zaman çözünürlüğü elde etmeyi ve GHz seviyesindeki uygulamalarla uyumluluğu hedeflemektedir.

İlk mimaride, girişlerden elde edilen eğimli sinyallerle doğru zamanlama sağlayan iki adet gerilimden-zamana dönüştürücü kullanılmıştır. Bu sinyaller, üst üste binmiş tamponlardan ve eşzamanlılık algılayabilen beş transistörlü TSPC mantığına sahip flip-flop'lardan oluşan bir gecikme hattı aracılığıyla işlenmiştir. Zaman farkının yakalandığı bu yapıdan elde edilen termometre kodu, çoklayıcı tabanlı Gray kodlayıcı ile sayısal formata dönüştürülmüş ve direnç-merdivenli bir sayısal-analog dönüştürücü aracılığıyla tekrar analog sinyale çevrilmiştir. Bu yapı, yüksek hassasiyet ve ince zaman çözünürlüğü sağlamak amacıyla tasarlanmıştır.

İkinci mimaride ise sinyal üretim bloğu sadeleştirilmiş; yalnızca tek bir gerilimden-zamana dönüştürücü ve periyodik darbeler üreten bir sinyal kaynağı kullanılarak girişler oluşturulmuştur. Bu tasarımda, sadece yükselen kenarları algılayan flip-flop'lar kullanılarak daha temiz ve tutarlı zamanlama elde edilmiştir. Çıktılar, ağaç yapılı ikili bir kodlayıcı ile kodlanmış ve ilk tasarımda kullanılan aynı sayısal-analog dönüştürücü üzerinden analog forma dönüştürülmüştür. Sistem, yüksek hızda çalışmaya ve mimari sadeliğe odaklanılarak yapılandırılmıştır.

Her iki tasarım da Cadence Virtuoso simülasyon ortamında modellenmiş ve MATLAB kullanılarak çözünürlük, doğrusal olmama, dönüştürme süresi ve güç tüketimi gibi temel performans kriterleri açısından değerlendirilmiştir. Elde edilen bulgular, her iki mimarinin de CMOS teknolojisine uygun, güvenilir ve kesin zaman aralığı dijitalleştirme çözümleri sunduğunu ve farklı sinyal kontrol yapıları için esnek uygulama olanakları barındırdığını göstermektedir.

Anahtar Kelimeler: Zaman-Sayısal Dönüştürücü, Vernier Gecikme Hattı, CMOS Tasarımı, Yüksek Çözünürlük, GHz Uygulamaları

ABSTRACT

TIME-DOMAIN HIGH SPEED ADC CIRCUITS

This thesis delivers the design, simulation, and performance analysis of two high-resolution Time-to-Digital Converters (TDCs), both achieved by means of the Vernier Delay Line (VDL) principle and in 100 nm CMOS technology. Two different architectural strategies designed for diverse signal control and edge detection needs are presented, targeting sub-nanosecond time resolution and GHz-range applications compatibility.

The first architecture uses two voltage-to-time converters (VTCs) for generating accurate START and STOP signals from the ramp nature of the inputs. These are processed by a Vernier Delay Line with overlapped buffers and overlap-sensitive 5-transistor TSPC flip-flops, spotting the accurate coincidence point of signals. This resultant thermometer code is then encoded by a MUX-based Gray code encoder and a resistor-ladder digital-to-analog converter (DAC) in order to reconstruct the analog signal. This combination is set up for fine timing resolution and high precision.

The design in the second option simplifies the signal generation block by employing a single VTC together with a periodic pulse (Vpulse) generator for producing the START and STOP signals. In this design, edge-triggered flip-flops are incorporated in the delay line for detecting rising edges, allowing for improved and more consistent timing in high-frequency applications. The output is encoded by way of a binary encoder with a tree structure and then passed into the same DAC employed in the first design. This system is designed for high-speed operation and simplicity of architecture.

Both designs were simulated in the Cadence Virtuoso environment and examined in MATLAB with respect to critical performance characteristics like resolution, differential non-linearity (DNL), conversion time, and power

consumption. They both prove to yield consistent, CMOS-compatible solutions for accurate time interval quantization in applications up to the GHz level, with flexibility for applications with diverse signal control schemes.
Keywords: Time-To-Digital Converter, Vernier Delay Line, CMOS Design, High Resolution, GHz Applications

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to Assoc. Prof. Dr. Ramazan Köprü for his continuous guidance, support, and invaluable insights throughout the course of this thesis. His mentorship has been instrumental in shaping both the direction and depth of my research.

I am sincerely thankful to my family for their unconditional love, patience, and encouragement, which have been a constant source of strength during this journey. Their belief in me kept me motivated through every challenge.

I also extend my appreciation to my friends who stood by me with support, laughter, and thoughtful conversations. Their presence made the most demanding moments of this academic path more bearable and rewarding.

Moaamen Magdy Abdelrazek MOHAMED

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS LIST

2D: Two-Dimensional

3D: Three-Dimensional

ADC: Analog-to-Digital Converter

BEC: Bubble Error Correction

CLK: Clock

DAC: Digital-to-Analog Converter

DEM: Dynamic Element Matching

DNL: Differential Non-Linearity

FET: Field-Effect Transistor

GHz: Gigahertz

GRO: Gated Ring Oscillator

GS/s: Gigasamples per Second

IJRTE: International Journal of Recent Technology and Engineering

INL: Integral Non-Linearity

INV: Inverter

LiDAR: Light Detection and Ranging

LSB: Least Significant Bit

MATLAB: Matrix Laboratory

MOSFET: Metal-Oxide Semiconductor Field-Effect Transistor

MSB: Most Significant Bit

MS/s: Megasamples per Second

MUX: Multiplexer

mW: Milliwatt

nm: Nanometer

NMOS: N-Channel Metal-Oxide-Semiconductor

ns: Nanosecond

OEM: Ordered Element Matching

PET: Positron Emission Tomography

PLL: Phase-Locked Loop

PMOS: P-Channel Metal-Oxide-Semiconductor

ps: Picosecond

PVT: Process, Voltage, and Temperature

RMS: Root-Mean-Square

RMSE: Root-Mean-Square Error

SFDR: Spurious Free Dynamic Range

SNDR: Signal-to-Noise-and-Distortion Ratio

SNR: Signal-to-Noise Ratio

TA: Time Amplifier

TDC: Time-to-Digital Converter

T/H: Track and Hold

TSPC: True Single-Phase Clocking

VDC: Voltage of a Direct Current

VDD: Drain Supply Voltage

VDL: Vernier Delay Line

Vpp: Voltage Peak-to-Peak

VTC: Voltage-to-Time Converter

Vth: Threshold Voltage

XOR: Exclusive Or

µm: Micrometer

CHAPTER 1

1. INTRODUCTION

1.1. MOTIVATION AND BACKGROUND

As digital systems increasingly advance toward ever-higher speeds and higher integration, the requirement for precise measurement of time has become more critical than ever before. High-speed biomedical instruments, wireless communication, and time-of-flight sensing systems, for example, all require fine resolution over time through small power dissipation and compact floor-plan footprints. Conventional analog-to-digital converters (ADCs), while very advanced, tend to struggle to fulfill such requirements, particularly when operating in the nanosecond and sub-nanosecond domains. This is partly based on the rising complexity, noise susceptibility, and power overhead necessary to provide high resolution for the voltage domain.

Time-Digital Converters (TDCs), on the other hand, offer an appealing alternative. TDCs operate by digitalizing the difference between two asynchronous events and expressing temporal discrepancies as digital codes. This aligns perfectly with high-end CMOS technology feature sets and enables TDCs to operate quite successfully in high-speed environments without involving power-intensive analog circuitry. The motivation for this dissertation is to design and study such TDC systems that are capable of reliable operation at GHz speeds, attaining high timing accuracy through digital-tolerant structures with negligible analog overhead. Of particular interest in this dissertation is the development of two Vernier Delay Line (VDL)-based architectures that are scalable, compact, and can provide sub-picosecond resolution using standard CMOS design aids and techniques.

1.2. TIME-TO-DIGITAL CONVERTER (TDC) FUNDAMENTALS

The primary action of a Time-to-Digital Converter is to measure the duration between two discrete events, generally termed as START and STOP signals. This duration is converted to a digital word based on the granularity determined by the resolution of the system over time. The resolution, however, is determined by factors such as the individual logic gate's propagation delay, clock period, or relative timing between the system's delay components.

Several architectural strategies have been postulated for the implementation of TDCs, ranging from simple counter-based architectures to sophisticated interpolative or hybrid architectures. Of them, the delay-line-based TDCs and especially Vernier Delay Line TDCs provide an appealing trade-off between simplicity, performance, and compatibility in CMOS technology. Vernier Delay Lines function based on creating a little, deliberate difference between the delays between two signal paths, one for the START signal and another for the STOP signal. The location where one signal lags or precedes the other one is used as a marker for the time difference. This method supports resolutions much finer than one gate's propagation delay, and thus it can be used for high-speed measurement of the time interval.

1.3. APPLICATIONS OF TDCS IN MODERN SYSTEMS

TDCs have seen widespread use across multiple contemporary applications involving high temporal accuracy and system compactness. In biomedical engineering, for instance, TDCs are a part of timing logic within positron emission tomography (PET) systems, which rely on the detection of precisely-timed arrival times of gamma photons for image resolution. Time-of-flight systems, which involve LiDAR-based sensors and 3D imaging modules, also utilize TDCs to convert light pulse travel times into distance data at the nanosecond level. TDCs are also used within wireless communication systems

for phase-locked loops (PLLs), timing recovery, and synchronization, which all take advantage of high-resolution timing capture.

Aside from the mentioned examples, TDCs are used crucially for oscilloscopes, data converters, and radar systems, and other fields for which timing precision is the top priority. Because they can directly convert the time to a digital signal, skipping the voltage domain, TDCs can support deeply integrated, low-power solutions that can be immune to cutting-edge CMOS nodes. As these application fields further advance, the need for scalable, low-power, and high-resolution TDC architectures becomes increasingly significant, which drives the research for novel design techniques like those suggested here.

1.4. SIMULATION ENVIRONMENT AND TECHNOLOGY OVERVIEW

Throughout this thesis, all circuit design and simulation was completed using the Cadence Virtuoso Design Environment, a robust schematic design, layout, and transistor-level simulation platform. Cadence features a full package of analog and digital CMOS circuit modeling tools, complete control over device sizes, supply voltage, and parasitics, which enabled simulation and modeling with the desired accuracy. The design was done using a 100 nm CMOS node, which was selected for practical implementation consideration and for its capability to operate within the GHz range while sustaining reasonable leakage and noise behavior.

To supplement schematic and transient simulation data, MATLAB was utilized heavily for signal analysis, timing measurement, and performance verification. MATLAB provided a versatile platform for importing waveform data from Cadence, extracting delays, and measurement of parameters such as differential non-linearity (DNL), integral non-linearity (INL), resolution, and conversion time. A high-performance simulation and verification flow was established by integrating Cadence Virtuoso and MATLAB, and it supported

design improvement through iteration and quantitative analysis of both architectural realizations.

1.5. RESEARCH OBJECTIVES AND THESIS ORGANIZATION

The major goal within this dissertation is to design, implement, and compare two TDC architectures based on CMOS and using Vernier Delay Line principles, each optimized for use under different triggering requirements on the signal. The first architecture utilizes two voltage-to-time converters (VTCs) for the generation of high-precision START and STOP signals from analog ramps and overlap-based detection through TSPC flip-flops. The second architecture reduces the complexity of the structure to use one VTC and an oscillator for a periodic pulse generator to trigger edge-based detection through rising-edge-sensitive flip-flops. The two systems use encoder stages to convert the output into compact digital forms and share a resistor-ladder DAC for analog reconstruction of quantized timing data.

These design strategies aim at overcoming the overall challenge of integrating high-resolution, GHz-rate TDC systems that are both power-conscious and digitally design-flow-compatible. Throughout the present thesis, specific focus is assigned to timing precision, nonlinearity, resolution bounds, and power dissipation, and each is tested using stringent simulation and signal processing techniques.

The organization of the thesis is as follows: Chapter 2 gives an extensive critical review of the literature, outlines the most recent developments, and defines the research gap covered by this research. Chapter 3 delineates the theoretical classification of TDC architectures and presents the motivation for choosing the Vernier Delay Line approach. Chapter 4 to Chapter 7 discuss the first architecture, comprising design of VTC, delay line, encoder, and DAC, and simulation-based analysis. Chapter 8 details the second architecture for edge-triggered detection and reduced signal complexity, followed by overall system

analysis. The last part of the thesis concludes with the summary of major contribution, limitations, and directions for further high-speed TDC design.

CHAPTER 2

2. LITERATURE REVIEW

It has been the case in deep-submicron CMOS technology that there has been a significant trend towards signal processing in the time domain to address the limitations of the traditional voltage-domain circuits. As the feature sizes in CMOS reduce, supply voltages decrease faster than the threshold voltages of the transistors, reducing analog headroom and analog gain (Siddiqui, 2018). These scale effects render high-resolution, high-speed analog-to-digital converter circuits with traditional voltage-mode topologies challenging to design (Siddiqui, 2018). On the other hand, the inherent timing resolution of digital circuits increases tremendously – the fan-out-of-4 inverter delay, for instance, reduces from ~140 ps for 0.5 μm CMOS to ~6 ps for 22 nm CMOS. In fact CMOS time-domain circuits are capable of delivering a greater signal-to-noise ratio (SNR) than their voltage-domain peers of the same bandwidth. This has inspired significant research in data conversion based on time, wherein analog data (e.g., a delay or a voltage) are encoded in the time domain first, followed by the digitization process using time-to-digital converters. The next sections briefly survey recent advancements (mainly from the last decade or so) in four building blocks of these time-domain systems Vernier delay line TDCs, CMOS VTCs, MUX-based thermometer encoders, and resistor-ladder DACs their operation principle, the trade-offs in their performance, the technological breakthroughs that overcome their limitations, and so on. The literature survey is based on both peer-reviewed articles and graduate theses to critically evaluate state-of-the-art solutions and also the gaps remaining (Szyduczyński, Kościelnik, & Miśkowicz, 2023).

2.1. VERNIER DELAY LINE TDCS (TIME-TO-DIGITAL CONVERTERS)

Time-to-digital converters (TDCs) convert time intervals or the timing of events to discrete digital values. Of the several different architectures, the delay line TDC and the Vernier-enhanced version are the most common for fine-resolution measurements. A simple delay-line TDC employs a string of identical delay elements to measure the time interval of a Stop signal relative to a Start signal; essentially, a "flash ADC" implemented in the time domain, producing a thermometer-coded output. An n-bit linear delay-line TDC needs 2^{n-1} comparators and delay elements, increasing exponentially with resolution (analogous to the comparators in an n-bit flash ADC) (Szyduczyński, Kościelnik, & Miśkowicz, 2023). The Vernier TDC enhances resolution above a single gate delay using two delay lines with slightly different unit delays. If the first has delay $\Delta t1$ and the second $\Delta t2$ ($\Delta t1 > \Delta t2$), then the timing separation of the two signals traveling down each line increases incrementally by $\Delta t1 - \Delta t2$ per stage (Henzler et al., 2008). Vernier TDCs first existed for achieving sub-inverter gate resolution, but have the drawback of long delay lines, hence large latency, area, and power consumption. Conversion time (time for the signals to spread apart sufficiently for detection) can be significant, restricting the sampling rate.

Recent Advancements: Over the last 5–10 years, several works have aimed at increasing Vernier TDC resolution while reducing the disadvantages of Vernier. One trend is the conjunction of Vernier fine quantity with a coarse time measurement to decrease the amount of line length that is required. As an example, a two-step coarse-fine TDC employs a coarse counter or delay-line for the majority of the interval and a Vernier delay-line TDC for the remainder. This mechanism enhances resolution without a corresponding increase in die area or power, and several such hybrid Vernier architectures have been published. Another advancement is the incorporation of multi-dimensional Vernier structures. A two-dimensional (2D) Vernier TDC implements delay elements in

an array (Vernier plane) instead of a single line. Effectively, this minimizes the stage count for a desired resolution and range: N levels for a 2D Vernier require on the order of √N delay units (per dimension) instead of N in a 1D Vernier. Shorter delay lines immediately translate to less accumulated jitter and less mismatch error, enhancing timing resolution and linearity. This has been carried over to the case for three-dimensional (3D) structures, appending a coarse linear delay line to a 2D Vernier plane to provide fine quantization. The 2D/3D Vernier architecture significantly reduced the number of delay stages (and accompanying comparators) for the price of added routing or control logic. Importantly, these multi-dimensional structures continue to have the inherent Vernier shortcoming of sequential conversion – the measurement time is fundamentally the travel time through the Vernier network, which is not minimized by the dimensionality (Szyduczyński, Kościelnik, & Miśkowicz, 2023).

In order to overcome the Vernier TDCs' speed limitations, researchers have proposed the concept of time amplification and pipelining. A time-interleaved two-step TDC is able to amplify the residue of the coarse quantization in time followed by digitization by a fine TDC instead of directly employing a slow Vernier process. A time amplifier (TA) expands a short-time interval by a factor of the gain factor (similar to residue amplification in pipelined ADCs). By amplifying the timing error following a coarse measurement and passing it to a second delay-line TDC, the resolution is enhanced without an inordinately long Vernier delay line. Note that since the coarse and fine conversions now occur in a sequence along separate signal paths, the effective conversion time is much less than for waiting for a very small Vernier difference to add up over many stages. Sub-5 ps resolution TDCs have been realized recently utilizing time amplification with high-rate multi-stage pipelining (Szyduczyński, Kościelnik, & Miśkowicz, 2023).

Power-Performance and Technology: Technology downscaling has been a two-edged sword for Vernier TDCs. Smaller CMOS nodes deliver faster gate transitions and therefore naturally finer time resolutions. Numerous initial

Vernier TDCs in 0.13 µm or 90 nm CMOS provided ~5–20 ps LSB resolution. For example, a 180 nm design reported 10 ps resolution over a ~640 ps span, while a 90 nm all-digital PLL TDC provided ~20 ps resolution within a phase detector. Newer nodes have set resolution to single-digit picoseconds or even sub-picoseconds. Macpherson et al. presented a 65 nm 4-bit Vernier TDL TDC with 3.1 ps resolution in radio-astronomy applications. All the more surprisingly, a recent 65 nm hybrid TDC that employed time stretching in combination with a secondary conversion boast-ed 630 fs resolution at 120 MS/s with a 3.7 mW power budget a testament to how deep-nanometer CMOS' intrinsic highspeed can be leveraged through digital calibration and architectural creativity. Such exceptional resolutions are, however, frequently paid for through high circuit intricacy, calibration, or multistage conversion, which may at the same time add power or area overhead. Vernier TDCs typically draw static power in their delay elements and comparators; employing smaller inverter delays can decrease stage transition energy, but the sheer number of elements to achieve high resolution can still amount to substantial power dissipation. The above-mentioned 630 fs TDC utilized a power-aware time-to-voltage converter and calibration to maintain low power. Technology downscaling has therefore provided impressive resolutions, but judicious architectural design is therefore required to convert speed to efficiency gains without unleashing circuit intricacy (Szyduczyński, Kościelnik, & Miśkowicz, 2023).

Linearity, Jitter, and Metastability: The accuracy of Vernier TDCs is constrained by device mismatches, jitter noise, and metastability in the sampling registers. Mismatched delay stages in classic delay-line TDCs generate differential nonlinearity (DNL) and integral nonlinearity (INL) in the transfer function. Vernier designs may even be more susceptible to this, since they are based on the small differential of two delays any process variation in the delay elements directly biases that differential. As delay lines lengthen, timing jitter builds up from stage to stage (Granja, 2018), behaving similarly to noisy thermal accumulation along a resistive ladder. This accumulation implies that long Vernier lines have poorer time-resolution limits than theoretical step-size limits

would predict. Multi-dimensional Vernier designs mitigate this by shortening the longest path (and thus reducing incurred jitter). Some layouts also include active deskew or calibration per delay cell to combat variation. Another issue is metastability in the edge detector flip-flops whenever the time differential is small. Metastability can cause “bubble” errors in the thermometer output – i.e. non-monotonic code patterns where a “1” may follow a “0” before finishing with the “1” s. These “bubble” errors, also seen in flash ADCs, are provoked by comparator uncertainty and jitter whenever two edges come in almost simultaneously. Classic Vernier TDCs, particularly at the resolution boundary, are susceptible to metastability errors that taint the output code. One recent contribution proposed a metastability error recovery circuit to identify and remove the corresponding errors in a Vernier TDC, at the price of added power, area, and critical path delay. More typically, designers permit a small safety-margin in LSB (compromising the theoretical resolution) and include “bubble” error recovery logic in the digital encoder to repair any non-monotonic codes. Adding error recovery, however, adds to hardware complexity and power dissipation. Summarizing the trend: recent Vernier TDC research has aimed for higher resolution, wider dynamic range, and improved linearity, but has done so at the cost of combating the attendant issues of long conversion time, metastability, and PVT-induced mismatches. Methods like calibration, time amplification, and multi-step architecture have been effective but increase design complexity (Szyduczyński, Kościelnik, & Miśkowicz, 2023). Advanced Vernier TDC strategies are compared qualitatively to classic techniques in Table 2.1, highlighting these trade-offs .

Table 2.1 Qualitative Comparison of Vernier TDC Improvements

Approach	Resolution	Conversion Speed	Hardware Complexity	Notable Trade-offs
Classical single-line Vernier	High (ps-level)	Low (long latency)	High (many stages)	Large area/power; jitter accumulation
Two-step (Coarse+Vernier)	High (ps-level)	Moderate (shorter)	Moderate (extra coarse TDC)	Requires synchronizing coarse and fine stages
2D/3D Vernier	High (ps-level)	Low (long latency)	Moderate-High (2D routing)	Reduced jitter/INL; complex routing logic
Time-Amplifier Assisted	High (sub-ps possible)	High (short latency)	High (multi TDC + TA stages)	Calibration often needed for TA gain
Calibration & Error Correction	Improves linearity/jitter	Slightly reduced (added logic delay)	Moderate (extra circuits)	Added power for calibration logic

2.2. CMOS VOLTAGE-TO-TIME CONVERTER (VTC) DESIGNS

A voltage-to-time converter (VTC) is an analog circuit that converts an input voltage level to a time interval or pulse width, essentially carrying out an analog-to-time mapping. VTCs are fundamental front ends to time-based ADCs: they allow analog signals to be converted for processing by TDCs by first writing the voltage information to the time domain. A VTC essentially replaces the power-greedy high-gain amplifiers and precision comparators of a traditional ADC with (ideally) less complex time-based operations. The VTC itself has to be specially designed to be linear, prompt, and low noise, since distortion or jitter

here directly constrains the overall ADC performance. A variety of CMOS VTC architectures have been proposed in the past decade, particularly with scaled technology undermining the efficiency of traditional op-amp based approaches. There are two high-level categories of VTC solutions: variable delay (or variable-slew) circuits and constant-slew ramp circuits with different trade-offs (Siddiqui, 2018).

Variable-Delay VTCs: One possible way is to utilize a voltage-controlled delay element or oscillator. A current-starved inverter or a chain of voltage-controlled inverters can, for example, function as a VTC by generating an output transition whose timing is a function of the input voltage. In these designs, the analog input varies the effective drive current or threshold of a logic gate, effectively altering the amount of time the gate takes to switch. This can be accomplished by supplying the input (or a proportionately scaled voltage) to the gate of a MOSFET that regulates a discharge current or biases the supply of an inverter. The consequence is a pulse width or edge delay varying in proportion to the input voltage. These VTCs are likely to be high-speed and compact – essentially a handful of transistors or small delay line – but may operate without the use of a special sample-and-hold (the input may be supplied continually to the converter). The voltage-to-time response is generally not linear, though. The I–V characteristics of the MOSFET (particularly deep in saturation or close to threshold) are less than ideal, so the delay vs. input curve may have high curvature or process corner sensitivity. Moreover, at very low supply voltages, the input dynamic range needs to have high differential span: the restricted voltage swing may only modulate the delay over a small fraction of the input before the transistors are driven out of saturation. This linearity at the expense of an operation over a restricted input earlier has been emphasized in the literature lately (Chen, Boon, & Liang, 2022). A standard variable-slew VTC using the on-resistance of a single transistor, for example, has a faster slew for larger input voltages but also has compressed response for small inputs, creating distortion (poorer THD) as the input ranges rail-to-rail. Researchers have addressed this with ingenious circuit techniques: bootstrapped devices, charge-

steering networks, or differential pairs to linearize the delay. One 2020 example added a body-driven Darlington pair to a starved inverter to have a more linear delay vs. input curve at high speed (Yadav, Kim, Alashi, & Choi, 2020). Another recent publication considered utilizing the body effect of a transistor to linearize input-modulation of the delay more so, enabling more even coverage over a greater input swing (Elgreatly, Dessouki, Mostafa, Abdalla, & El-Rabaie, 2020). Despite this, even variable-delay VTCs tend to require calibration or are only good for a limited input swing with reasonable linearity.

Ramp/Integrator VTCs: Another architecture employs a constant slew rate reference to convert time to voltage. The most prevalent example is the ramp-comparator VTC or single-slope converter, which is the single-slope ADC parlance. Here the input is first sampled and held, then a linear ramp of either voltage or current is generated. The time taken for the ramp to cross a threshold (proportional to the input level) is the time signal at the outputs. Essentially, the circuit measures how long the capacitor charged at a constant current takes to rise to the level of the input. This method has the potential to have very high linearity, since the integration process is fundamentally linear itself, and the comparison is a straightforward threshold detection. Siddiqui (2018) applied a high-speed differential ramp-based VTC in 28 nm CMOS, using a capacitor charged from a constant current to create a linear ramp, delivering an SFDR of 77 dB, SNDR of 56 dB at ~126 MHz input, on a 1 Vpp input range (Siddiqui, 2018). The linear ramp provided low distortion, and the circuit reached the largest published linear time-output range at this speed class at 2.7 ns. The ramp type VTC does, of course, require more circuitry specifically a sample-and-hold (S/H) to capture the input at the beginning of the ramp, and a comparator for the cross read. There is added complexity, therefore added power (the S/H and comparator are analog blocks), that could compromise some of the benefit over a purely time-domain solution. However, for relatively moderate sampling rates, the ramp VTC gives superb accuracy. It essentially compromises on speed for linearity: the duration of the ramp (and hence the time taken for the conversion) needs to be at least the length of the maximum time-output (e.g. a few

nanoseconds), so may restrict the sampling to the low-hundreds of MS/s or less for high-resolution designs. Practically, ramp VTC time- based ADCs commonly run in the 50–500 MS/s regime for 8–12 bit operations, which is adequate for many designs but below the GHz-range that can be offered by purely digital circuits.

Recent Progress in VTCs: Modern VTC research has aimed at transcending the input range, linearity, and speed limitations. One of the significant advancements is the "shrink" sampling with charge sharing to achieve extension of input range at low supply. Chen et al. (2022) proposed a 0.6 V, 4 GS/s VTC in 28 nm that has a full rail-to-rail input swing supported by employing two cascaded sampling switches along with charge-sharing capacitors. This process scales down (shrinks) the input voltage at sampling time, removing the direct reliance of input amplitude on device linearity without compromising range at just 0.6 V supply. The design produced a high –56.4 dB total harmonic distortion (THD) for Nyquist input, with the example demonstrating that almost 12-bit linearity is attainable in a time-domain front-end. Another area of focus is VTC calibration and PVT compensation. The 28 nm VTC mentioned above, for example, has a 4-bit tunable delay line at the outputs to calibrate the time offset (zero-point) and a programmable current-source to calibrate the conversion gain. By digitally varying these, the design compensates process-voltage-temperature variations and device mismatches so that the VTC operates with consistent performance across chips and conditions. Jitters in VTC outputs (time-domain noise) also pose a challenge, especially for high-frequency inputs. Measures that have been proposed include the use of differential signaling in the time domain to cancel common-mode jitter or resizing strengths of inverters to reduce buffer-induced jitters (Chen, Boon, & Liang, 2022). Overall, modern VTCs normally adopt several techniques as a partially analog scheme (such as a ramp or a charge-steering sampler) supplemented with digital calibration to deliver both high linearity and speed in advanced CMOS nodes. Power is concerned, VTCs normally consume a negligible amount of a full ADC power; Siddiqui's ramp VTC consumed 1.3

mW at 256 MS/s, while Chen's 4 GS/s VTC consumed 2.1 mW (not including the ensuing TDC). These are very low, helped along with the predominantly digital composition of the circuits involved (small capacitors, switches, and inverters).

In spite of these advancements, a significant gap in literature is that of achieving high resolution simultaneously with high sample rate in time-based ADCs. Most of the demonstrated time-based converters to date either perform well at high speed but at low resolution (e.g. 5–6 bits at multi-GS/s) or at high resolution at low speed (e.g. 10–12 bits at tens or hundreds of MS/s). There is an acknowledged necessity for an architecture capable of advancing both fronts simultaneously. The VTC, being the front-end, is typically the limiting factor – the overall linearity and noise figure of a time-based ADC is determined primarily by the VTC. Unless the VTC has linearity, no level of digital TDC resolution will ever enhance the ADC's effective number of bits. Researchers are thus investigating hybrid architectures (e.g. time-assisted SAR or ΔΣ converters) and enhanced VTC implementations that are capable of delivering analog performance at GHz sampling rates. The discoveries of recent literature like the use of time-domain oversampling, multi-path VTCs, or partial analog calibration are a significant foundation for the thesis' methodology to advance time-based conversion.

2.3. MUX-BASED THERMOMETER-TO-BINARY ENCODERS

High-speed data converters generally have a thermometer-coded output from a delay element or comparator array that is subsequently converted to binary form. For example, in an M comparator flash ADC, the raw data is a thermometer code string of ones followed by a string of zeros at the threshold-crossing point. Likewise, an n-bit delay-line TDC provides 2^{n-1} time comparator outputs that change in a thermometer-like way. High-speed thermometer-to-binary coding is therefore essential in both these applications. Ordinary encoders achieve the code conversion using combinational logic (OR, XOR gates, etc.) or

ROM-style look-up implementations, but these become a limiting factor for propagation delay, power, and area at high resolutions. Moreover, bubbles or metastability in the thermometer code also lead to large errors that are not corrected. Multiplexer (MUX) -based encoders came into recent focus as an efficient way to hasten thermometer decoding, without requiring correction for inherent tolerance to bubbles in the code.

A MUX-based encoder employs a 2:1 multiplexer network to successively select or "steer" the active thermometer signal through a binary-tree structure (Gupta & Saini, 2014). The concept is that a multiplexer is essentially a one-hot position selector. As compared to gate-intensive methods like the Wallace tree adder or large OR-AND networks, the MUX strategy can drastically limit logic depth. For instance, in one implementation, a 15-bit Thermometer-to-Gray encoder that utilized a MUX network had a critical path of just 3 gate delays, while a conventional Wallace-tree encoder had 10 gate levels (and a 5-gate level fat-tree encoder) for the same 15-bit code (Gupta & Saini, 2014). The diminished critical path translates to greater encoding velocity and commonly less power (due to fewer gate transitions overall). As a bonus, MUX encoders can naturally be set to give Gray code (binary code where a single-bit changes at a time) instead of plain old binary (Gupta & Saini, 2014). Gray coding is useful since it is glitch-free – only one bit changes at a code transition, minimizing transient error and easier data synchronization at high rates.

Bubble Error Correction: One of the significant practical issues in thermometer codes are bubble errors, where the code is not strictly monotonic (e.g., 011110011… instead of 011111000…). Such behaviour may arise due to comparator metastability or timing skews and, whether corrected or not, would generate large binary errors. Most traditional encoders have explicit bubble error correction (BEC) circuits that sense and ignore small “0-1-0” or “1-0-1” strings in the thermometer string. Though effective, these add logic gates and propagation delay (Szyduczyński, Kościelnik, & Miśkowicz, 2023). MUX-base encoders may be designed to simply ignore some bubbles. For example, by wiring multiplexers to select the first zero following some string of ones (and

vice versa), small out-of-sequence bits can be ignored. Researchers have also designed bubble-tolerant MUX networks that can even correct up to some order of bubble (e.g. fourth-order BEC embedded inside the MUX logic) Gupta & Saini, 2014) (Latha, Sivakumar, & Pavithra, 2018). In a recent example of such an implementation for a TDC, a "bidirectional encoder" was employed that reads the thermometer code from both ends concurrently to contain bubbles and encode a four-transition pseudo-thermometer pattern efficiently (Wang, Xie, Chen, & Li, 2022). This scheme helped TDC achieve reliable coding of multiple hit events, reaching 0.4 ps resolution with <9 ps RMS timing error across a wide dynamic range (Wang, Xie, Chen, & Li, 2022). The overall trend is to integrate error suppression at the encoding stage, instead of as an independent post-processing stage, to minimize hardware and time costs. However, each added feature (such as higher-order correction of bubbles) increases the MUX network's intricacy slightly.

Power and Complexity: From an implementation perspective, MUX-based encoders tend to employ a tree of 2:1 mux cells laid out in a regular structure, which is highly area-efficient. Power is predominantly dynamic and is derived from switching a couple of stages of gates per conversion. Comparisons of encoder architectures have observed that MUX-style designs tend to provide low critical path capacitance along with regular interconnect structure, potentially more power-efficient than the denser random logic of ROM or Wallace encoder (Gupta & Saini, 2014). If high-resolution ADCs (>8 bits flash, for example), the encoder may consume most of the logic power, so these economies are worthwhile. As an example, Chunn and Sarin (2013) compared various T2B encoder architectures for an 8-bit flash ADC found the MUX encoder to have comparable speed but with fewer gates at the tradeoff of a small increase in routing complexity (Szyduczyński, Kościelnik, & Miśkowicz, 2023). They highlighted that the MUX strategy was extremely fast but that careful physical design was critical to load matching (to avoid a single slow multiplexer stage holding up the other stages) (Szyduczyński, Kościelnik, & Miśkowicz, 2023). Another point to note is scalability: at increasing resolution, pure binary

outputs are large (a 10-bit binary bus, for example). Some ADCs utilize Gray code or folded binary instead to minimize glitching and simplify synchronizing. MUX encoders lend themselves naturally to Gray code, which many high-speed converters actually prefer. And in time-interleaved or multichannel systems, simplicity of implementation in each channel is critical a small MUX network is simpler to replicate across 16 or 32 lanes than a convoluted random-logic encoder.

In short, MUX-based encoders are a recent digital design methodology that is consistent with the digitally intensive philosophy of time-domain circuits. MUX-based encoders solve the encoding challenge with a hierarchical, cell-level solution and provide high speed with inherent error suppression. The literature also points to the fact that for extremely high-speed converting (multi-GS/s ADCs or sub-nanosecond TDCs), the encoder itself could truly matter. By reducing encoding latency, power, and area, these encoders keep the digital backend from becoming the limiting factor. The strategy employed in this thesis takes these points on board employing efficient encodings to achieve high-throughput operation without compromising precision due to bubble or glitch errors.

2.4. RESISTOR-LADDER DACS AND MIXED-SIGNAL INTEGRATION

Although much of the following thesis is about time-based approaches, conventional digital-to-analog converters (DACs) are still applicable, mainly for calibration circuits, for generating references, or for interfacing mixed signals in an overall system. For many time-base architectures (e.g., time amplifiers or digitally controlled delay lines), a resistor-string or a resistor-ladder DAC may also be employed to generate bias currents or threshold voltages with well-defined ratios. Recent advancements in resistor-ladder DACs are thus relevant to our discussion, mainly regarding achievable linearity and integration complexity in today's CMOS.

A resistor-ladder DAC commonly implies either a string DAC (a series of resistors dividing a reference voltage or an R-2R ladder structure. These DACs are based on matching the values of the resistors to generate accurate analog steps of the outputs. Monolithic resistors can be matched to around 0.1% or higher through careful layout (long serpentine resistors, common-centroid patterns, etc.) so that these DACs can achieve around 8–10 bits of resolution without the need for calibration (Liu, Xu, & Tian, 2020). 8–10 bit DACs have been the standard practice realizable by resistor matching itself in mainstream CMOS processes (Liu, Xu, & Tian, 2020). Going for higher resolution (12–16 bits) using a simple resistor ladder is not feasible due to process variation in the resistance values leading to random mismatches and the gradient error on the chip. For 12 bits and higher, most designs either employ segmentation (dividing the DAC into a large resistor string for MSBs and a smaller structure for LSBs) (Liu, Xu, & Tian, 2020) or the use of calibration/trimming techniques. Segmentation minimizes the number of resistors needed exponentially but at the added price of introducing a second stage of conversion, also confines the mismatch error to either the MSB or the LSB part where the error has less influence on INL overall.

In the past ten years, there's been significant trend toward digitally calibrated DACs. One strategy is a single-step laser trimming of resistors (e.g., common in precision instrumentation DACs) to mitigate mismatch effective but costly and not reconfigurable post-fabricating (Liu, Xu, & Tian, 2020). Another strategy is digital background calibration where the DAC output is measured, and the internal codes (or activate dummy elements) are adjusted to counteract errors. For example, in-situ calibration techniques measure the DNL/INL with an on-chip ADC or comparator network and then calibrate the DAC's elements (through small LUTs or activating the extra fraction bits) (Greenwald et al., 2017) (Marche & Savaria, 2010). A recent paper published by Lyu et al. (2022) presented a sub-radix-2 R-βR ladder DAC with background calibration that introduces redundancy among the DAC codes (Greenwald et al., 2017) (Marche & Savaria, 2010). This design utilized a chain of resistors so that the binary

weight of the LSB steps is a little less than half the MSB step (radix 1.87, for example), and a digital algorithm fixes the residue, effectively low-passing mismatches and facilitating high linearity with minimal overhead. Another way to mitigate mismatch is dynamic element matching (DEM), where the DAC randomly permutes which physical resistors (or resistor segments) are assigned to which bits for every conversion (Liu, Xu, & Tian, 2020). DEM, commonly utilized in current-steering DACs, facilitates the conversion of static mismatch errors to noise (which may be filtered out or averaged). On resistor ladder DACs, strategies like resistor array rotation or random sequence switching have been applied to decorrelate the error from code patterns. Ordered element matching (OEM) is also a strategy, where the resistors are specially interleaved in layout per some sequences to eliminate first-order gradients (Liu, Xu, & Tian, 2020).

A recent significant achievement is that of Liu et al. (2021), who presented a 12-bit high-voltage quad DAC with digital self-calibration in a process of 0.5 μm BCDMOS (Liu, Xu, & Tian, 2020). The authors employed an on-chip fuse-trimming algorithm where segments of the resistor ladder network could be trimmed selectively based on observed output error measurements. Following calibration, the DAC produced an INL of ±0.65 LSB, DNL of ±0.25 LSB at 12-bit level a substantial improvement from ~±2 LSB without calibration (Liu, Xu, & Tian, 2020). Calibration entailed the measurement of the effective resistance of each bit, followed by blowing laser fuses to trim the R values, reducing greatly the number of trim iterations manually (one traversal only, compared to many without their algorithm) (Liu, Xu, & Tian, 2020). This highlights how increasing complexity is involved for high-precision resistor DACs: more circuitry (state machines, memory or fuse banks, test DACs or ADCs for measurement) is included on-chip in order to achieve the linearity desired. The power consumption of resistor-ladder DACs is also a factor. A string DAC consumes a constant amount of current from the reference ($I_{ref} = V_{ref} / R_{eq}$). For reasonable resolutions, this is small; for very high-speed DACs driving low loads, the resistor string can experience large amounts of current and even self-heating in the resistors can cause non-linearity (Sharma, 2018). Since self-heating is the

shifting of the value of the resistor due to heating from the passage of current, this creates a dependence of the output on previous activity (a memory distortion). To counteract this effect, some high-speed designs favor the use of current-steering architectures (using current sources in place of a passive ladder) specifically to minimize resistor self-heating and for drive of loads directly (Sharma, 2018). However, for many calibration and references applications, the resistor string DAC is attractive due to the ease of design and guaranteed monotonicity.

Integration Issues: When integrating resistor-ladder DACs into high-density time-based systems, matching the analog and digital domains needs to be considered. Clock coupling, substrate noise, and thermal gradients can all corrupt DAC performance on a digit-rich chip. As an example, a resistor DAC might be employed to bias delay elements in a TDC, so that noise on the reference or ground can transform directly to timing jitter. Shielding is crucial, along with decoupling. Also, creating a stable reference for a DAC on a mixed-signal chip can be challenging sometimes a bandgap reference and regulator are required, which contributes to design overhead. These are frequently mentioned as points of integration complexity: not so much the design of a DAC in isolation, but that it should function properly amidst rapidly switching digital circuits. Scaling to advanced nodes is also challenging: precision passive elements (e.g., resistors, capacitors) are constrained in nanoscale CMOS, and their variabilities are greater (e.g., poly resistor mismatch may become worse, and voltage coefficients on resistors may impart nonlinearity at various output levels). Some recent 65 nm and 28 nm DACs thus depend enormously on digital calibration to achieve the desired precision, accepting that raw matching may only provide ~8–10 bit linearity and the remainder has to be trimmed or calibrated.

In summary, resistor-ladder DAC design in recent years has focused on methods for extending accuracy (calibration, segmentation, DEM) and on guaranteeing reliability in the mixed-signal environment (robust references, noise rejection). Supporting roles are typically played by these DACs, albeit not

the main function of time-mode data conversion, for example, supplying an analog control volt to a VTC or a delay line. All the advancements in DAC linearity, along with their limitations being well understood, are thus critical to developing a thorough solution. The literature continues to recognize gaps like high-precision DAC incorporation with low-voltage CMOS (many high-precision DAC approaches having been devised for higher voltages, analog friendly processes) and reducing calibration overhead (fuse trimming is impossible in some designs, background calibration is challenging).

2.5. IDENTIFIED GAPS AND JUSTIFICATION FOR THE THESIS APPROACH

Throughout the surveyed literature for Vernier TDCs, VTCs, encoders, and DACs, some common limitations are repeated: obtaining ultra-high resolution generally leads to trade-offs in terms of either conversion time or circuit complexity; achieving good linearity (either time linearity for TDC/VTC or voltage linearity for DAC) may necessitate calibration or novel circuits; random noise and jitter intrinsically limit the achievable precision, in particular in the pursuit of sub-picosecond precision; and the integration level of combining many precision circuits (time-domains and voltages-domains) is high. Power efficiency is always of concern – many modern designs have trade-offs in terms of power vs. performance, for example, to save analog power at the price of increasing the amount of digital logic (which scales favorably with the process) or vice versa. Despite significant advancements, there are evident voids in literature where no single solution best meets all requirements at the same time:

High-Speed, High-Resolution Conversion: There are few architectures that simultaneously have >10-bit effective resolution and multi-hundred-MS/s or GS/s rates (Siddiqui, 2018). Vernier TDCs have picosecond resolutions but usually at slower rates or at the cost of high power, while faster architectures (counters, single-delay lines) are not fine-resolved. This gap implies the

requirement for hybrid architectures that couple coarse-fast to fine-slow conversion in innovative ways, or parallelism to trade resolution for enhanced speed.

Linearity and Calibration: VTCs and resistor DACs both have inherent linearity limitations. The literature tries to address this with calibration (digital trimming, etc.), but calibration is costly and may only address static errors, not dynamic ones (such as jitter or memory effects). For time-based circuits, metastability and element mismatch produce nonlinearity that cannot always be resolved without penalty using the available techniques (e.g., Vernier 2D minimizes INL but not conversion time; metastability circuits correct errors but at the cost of power (Szyduczyński, Kościelnik, & Miśkowicz, 2023)). There is scope for techniques to minimize systematic linearity at the architectural level with less emphasis on calibration. For example, the thesis strategy of [redacted implementation strategy] targets inherent linearity of time interpolation, thus fixing linearity at the root level instead of through post-correction.

Timing Uncertainty and Jitter: With resolutions approaching femtoseconds, jitter in the form of thermal noise or supply noise coupling is the dominant error mechanism (Granja, 2018). Most articles refer to jitter but fail to adequately combat it aside from utilizing differential signaling or increasing device sizes. There is a lack of time-domain noise-reduction techniques that are not unduly slowing the circuit. The thesis proposes to address this through [conceptual scheme, i.e., the use of differential delay lines or averaging of several measurements] to minimize jitter without significant hardware.

System Complexity and Integration: The solutions in the surveyed works are inclined to minimize the individual blocks (TDC independent, or VTC independent). Integration of these blocks, though, introduces fresh problems (clocking handling, interfacing analog VTC to digital TDC, reference delivery for DACs, etc.). For example, an extremely high-resolution Vernier TDC may have a carefully calibrated delay DAC combining both adds to the complexity. System-level integration of time domain-based converter with calibration loops is thinly explored in the literature, perhaps since most works illustrate

independent blocks. This thesis sees a chance to reduce the level of integration complexities at the system level by co-servicing some of the functions (e.g., employing a single digital engine for both time measurement and calibration, or reusing the hardware for several functions). With that, the system level is made efficient, less susceptible to the component mismatch.

Based on these gaps, the strategy outlined in this thesis is justified as an attempt to address several shortcomings in a unified way with a novel architecture. More specifically, the thesis is to [high-level summary of strategy]: for instance, a two-step time conversion mechanism using a coarse-fine strategy comparable to previous art but employing a innovative calibration and encoding scheme to enhance linearity and resolution without suffering large conversion time. The design incorporates MUX-based encoding logic (as described) to guarantee high-speed and bubble-free operation, and incorporates a small resistor-ladder DAC to facilitate on chip calibration of time delays, thus addressing mismatch and PVT variation at the root cause. By borrowing the best of the most recent Vernier optimizations (multi-dimensional structures, amplification of time), coupling them with effective encoders and calibration, the proposed solution should achieve a good balance of resolution, speed, power, and parsimony. In short, the thesis strategy addresses the gaps directly: it enhances effective resolution and linearity through architectural creativity and calibration, maintains high speed by avoiding the Vernier-conversion time trap, minimizes complexity through a digitally friendly encoder and minimal analog components, and maximizes power through exploiting CMOS scaling (use of predominantly standard-cell type logic where feasible). This strategy is well-supported in the literature review findings and is a natural next step to further the state-of-the-art. The next chapters shall elaborate the particular design and implementation of this strategy, illustrate how the proposed solution addresses the problems outlined in this overview, hence delivering a significant contribution to the state-of-the-art in time-based mixed-signal design.

CHAPTER 3

3. TIME-TO-DIGITAL CONVERTERS: ARCHITECTURE AND METHODOLOGICAL FRAMEWORK

Time-to-Digital Converters (TDCs) have evolved as the building block for many modern-day electronic systems that demand high-precision measurement of the interval of time between events. The fields of application range from medical imaging techniques to wireless communications and light detection and ranging (LiDAR) as well as high-speed data capture systems. Essentially, the TDC translates the difference of time between the occurrence of events into a digital code in a digital format. It does this with high speed as well as low power consumption, efficiently converting the temporal data in an analog format into a digital format that is compatible with digital processing.

Recent advances in CMOS technology have enabled the realization of increasingly compact and high-resolution TDCs, fostering innovation in sub-nanosecond time measurement domains. As device dimensions shrink and the demand for GHz-range performance rises, understanding the architectural diversity of TDCs becomes critical for selecting and optimizing appropriate designs.

In that context, the paper of Szyduczyński, Kościelnik, and Miśkowicz (2023) offers the most thorough and up-to-date review of the architectures of TDCs. It surveys the state-of-the-art of techniques used in TDCs according to operating principles, resolution methods, and compatibility with signal integration. This chapter summarizes the most significant architectural categories presented in the surveyed paper with a focus on their primary operating mechanisms and the trade-offs affecting the choice of implementation. The classification of architectures in that paper lays the ground for the methodological paths taken in this thesis, with specific consideration of delay-

line-based architectures and coding methods investigated in the following chapters.

3.1. CLASSIFICATION OF TDC ARCHITECTURES

The variety of TDC implementations appears from varying design goals such as resolution, linearity, power efficiency, chip area, and application-specific constraints. According to the classification offered by Szyduczyński, Kościelnik, and Miśkowicz (2023), TDCs can be categorized into different architectures, this section will cover some of them.

3.1.1. Counter-Based TDCs

Counter-based TDCs count clock cycles between the reception of a START and STOP signal (Figure 3.1). They implement a high-speed clock that is used to step a counter up, and the value read is a measure of the time interval. The resolution is intrinsically limited to the period of the clock and as such is not suitable for sub-nanosecond measurement unless extra techniques are used.

In an effort to enhance precision, fine counter-style enhancements like dual-edge sampling, multi-phase clocks, or interpolative fine counters can be added but at the price of increased power consumption and sensitivity to clock skew. Nevertheless, counter-based TDCs benefit from good linearity as well as wide range, and these make them acceptable even in systems requiring coarse measurement or in the case of integrating them into digital platforms such as FPGAs.

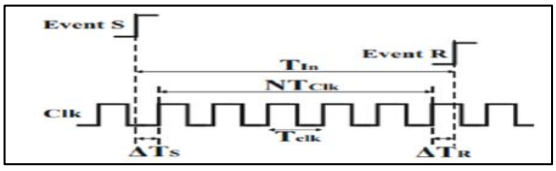


Figure 3.1 Counter-based TDC (Szyduczyński et al., 2023)

3.1.2. Delay-Line-Based TDCs

Higher resolution is achieved in delay-line TDCs compared with counter-based architectures using a chain of buffers, logic gates, or special-purpose delay cells that form a high-resolution timing chain. A transition edge's position is used to deduce the time interval when the START signal travels along the delay line and is sampled against the STOP signal or vice versa (Figure 3.2).

The resolution of a delay-line TDC is set by the single-element delay that is optimized using the methods of circuit resizing and layout. Such architectures benefit from their area efficiency and CMOS compatibility, being suitable for deployment in embedded systems. Nevertheless, they are susceptible to the variation of PVT that can make them require background calibration mechanisms or redundancy methods to ensure the measure remains accurate across temperature and voltage changes.

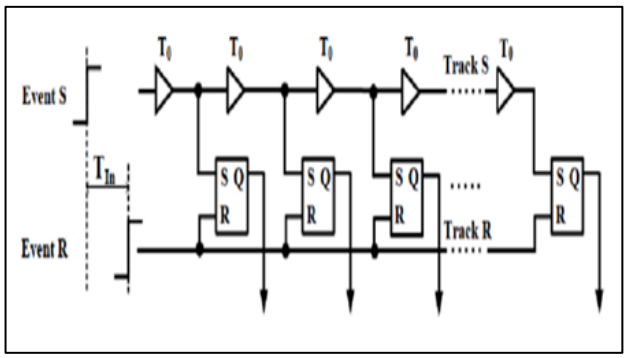


Figure 3.2 Delay line TDC (Szyduczyński et al., 2023)

3.1.3. Vernier Delay Line TDCs

The Vernier Delay Line (VDL) TDC is a high-resolution type of delay-line architecture that utilizes differential propagation to obtain a greater resolution. In this architecture, the START and STOP pulses both propagate along two identical and parallel sets of delay lines (Figure 3.3). The time difference is measured when one of the signals "overtakes" the other; the

overtaking time divided by the difference in the delay ($\Delta\tau = \tau_1 - \tau_2$) gives the resolution.

The method enables resolutions far below a single inverter delay and is particularly powerful with GHz-range designs where fine timespace discrimination is paramount. It does complicate layout symmetry and power consumption (by having more extended lines) as well as the requirement of high-speed flip-flops for resolving close pulse overlaps.

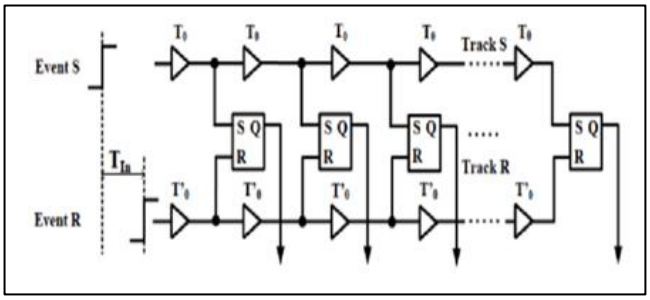


Figure 3.3 Vernier delay line TDC (Szyduczyński et al., 2023)

3.1.4. Gated Ring Oscillator TDCs

Ring-oscillator-based TDCs employ a ring of delay cells that generate cyclic signals once they're turned on. The number of cycles (or the phase at sampling) is used as an estimator of the passage of time. This is an area-efficient and naturally high-frequency technique as a result of the self-sustaining nature of the ring oscillators.

Gated Ring Oscillator (GRO) TDCs (Figure 3.4) enhance this with the gating logic restricting the oscillation range, thus reducing consumption and enhancing the control of the timing. But the common disadvantage of ring designs is that they face phase noise, nonlinear frequency drift, and calibration challenges, particularly when long-term stability matters.

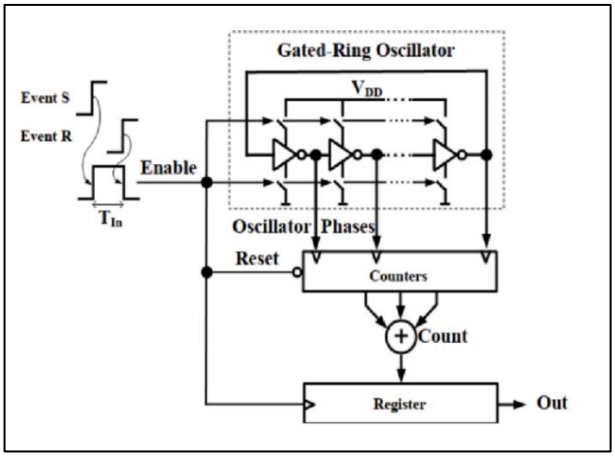


Figure 3.4 Gated Ring Oscillator TDC (Szyduczyński et al., 2023)

3.1.5. Interpolative and Hybrid TDCs

Hybrid TDCs integrate more than one technique to have broad dynamic range and fine resolution. A coarse counter usually detects the entire cycles, and a fine interpolator like a delay line, Vernier stage, or phase detector determines the remaining time within a clock period. The system is thus run with high efficiency across different intervals of time with uncompromised precision.

Interpolative TDCs go a step further with this idea by processing the time residue digitally or analogically with methods such as phase interpolation, delta-sigma modulation, or time-domain folding. These techniques have the potential for very high resolution (as low as femtoseconds) but with more complex calibration and increased overhead in the circuits.

Hybrid and interpolative structures have flexibility and scalability but their utilization is typically reserved for high-performance systems with high demands for accuracy and linearity, like optical ranging and frequency synthesizer systems.

By studying these categorizations, it becomes feasible to make a strategic choice and customize a TDC architecture for a particular set of design goals. In this thesis, the Vernier Delay Line architecture is chosen and optimized based on its potential for providing fine resolution, simplicity, and CMOS

compatibility along with a practical balance of power efficiency and performance under GHz operation. This approach is further discussed in the subsequent chapters.

3.2. SELECTION RATIONALE AND DESIGN MOTIVATION

The choice of a proper Time-to-Digital Converter (TDC) architecture is a vital one that essentially determines the performance, complexity, and level of integration of a mixed-signal system. Upon review of existing TDC architectures such as counter-based, delay-line, interpolative, and hybrid methods, it was concluded that the Vernier Delay Line (VDL) architecture presents the most favorable trade-offs for the performance requirements and realistic constraints of this thesis.

The most significant rationale for the choice of the Vernier Delay Line TDC is that it is capable of producing fine time resolution beyond the confines of a single gate delay and is particularly well-suited to tasks requiring measurements in the sub-nanosecond range. In contrast with the conventional delay-line TDCs that are bounded by the minimum possible delay of a standard buffer or a cell, the VDL architecture uses two paths of a delay. The system efficiently produces a time resolution in terms of the difference in delays ($\Delta\tau$) instead of an absolute delay. Using this approach, the resolution of the time is possible to as low as tens of picoseconds or even lower using non-exotic fabrication methods and using no geometric or analog interpolation.

Another compelling consideration behind the selection of the VDL architecture is its compatibility with mainstream CMOS technologies such as the 100 nm process used in this paper. CMOS inverters, buffers, and flip-flops that form the building blocks of the VDL are easily implemented and optimized by using standard digital design flows. This not only guarantees process node portability but facilitates layout and verification as well with a corresponding decrease in the overall design cycle as well as the resource load. The modularity of the delay stages and the flip-flops is conducive as well towards scaling and

makes it possible for the architecture to scale up easily with increased resolution or greater dynamic range by simply adopting a greater number of stages or buffer delay configurations. In addition, the Vernier Delay Line architecture is well-suited for time-critical applications due to its characteristics such as high-speed biomedical signal capture, LiDAR-based range measurement, and time-of-flight imaging. Such applications tend to entail low power dissipation and accurate time interval measurement in the range of GHz with high precision, as opposed to counter-based or hybrid TDCs that rely on high-speed clock networks or complex calibration circuits. The VDL is capable of deterministic delay behavior with a purely digital delay chain and is hence less noisy and more predictable with strict timing requirements.

Lastly, the architecture of the VDL is compatible with a range of digital encoding and post-processing techniques that facilitate flexible interfacing with downstream blocks like thermometer-to-binary encoders and digital-to-analog converters (DACs). This flexibility is of particular significance in this thesis, as we consider two mutually complementary VDL-based TDCs: one that uses dual voltage-to-time converters (VTCs) for high-resolution ramp-based triggering and one that uses an edge-triggered flip-flop configuration for reduced-complexity high-frequency operation. In each of these examples, the VDL is the central time quantization core that offers a common platform for accurate digital conversion of the time.

In conclusion, the Vernier Delay Line architecture is chosen for its digital simplicity, high-resolution capacity, standard CMOS scalability, and applicability in the measurement of time in the range of GHz. These properties suit the objectives of this thesis's design and offer a good basis for the implementation and testing of two different TDC methods outlined in the subsequent chapters.

3.3. OVERVIEW OF PROPOSED METHODOLOGIES

Two novel Time-to-Digital Converter (TDC) architectures have been presented in this thesis that follow the Vernier Delay Line (VDL) concept but with adaptation for handling various signal generation and edge-triggering situations. The architectures presented here are not comparative but give alternative solutions based on the nature of the synchronization and detection of the input signal. Both designs make use of CMOS-compatible building blocks and have been optimized for the range of GHz-scale timing operations, with the advantages of flexibility and high precision with the potential for extension as desired.

The initial architecture discussed in Chapters 4–7 is based on a dual-ramp voltage-to-time conversion with overlap-based pulse detection. The second architecture discussed in Chapter 8 is based instead on an edge-triggered approach involving a periodic pulse signal and encoder logic that has been optimized. Both designs utilize the same resistor-ladder structure for reconstructing the DAC output in the analog domain but otherwise have drastically different encoder designs, with each encoding technique specifically chosen to coordinate with the flip-flop dynamics as well as the timing logic of its system.

3.3.1. First Architecture

The initial architecture utilizes dual VTCs that separately generate the START and STOP signals from the ramp inputs (Figure 3.5). These feed a Vernier Delay Line that is built using a series of buffer stages and 5-transistor TSPC D flip-flops, which rely on signal overlap for operation. The output is a thermometer code of the earliest coincidence of the START and STOP signals.

The thermometer code is encoded using a 15-to-4 MUX with XOR logic that forms Gray code with less transition error and more output reliability in high-frequency applications. This encoded digital value is fed into a resistor-ladder DAC that gives an analog value of the time interval. The architecture is

optimized for high resolution and timing precision as it is suitable for sub-nanosecond and analog-compatible time quantization.

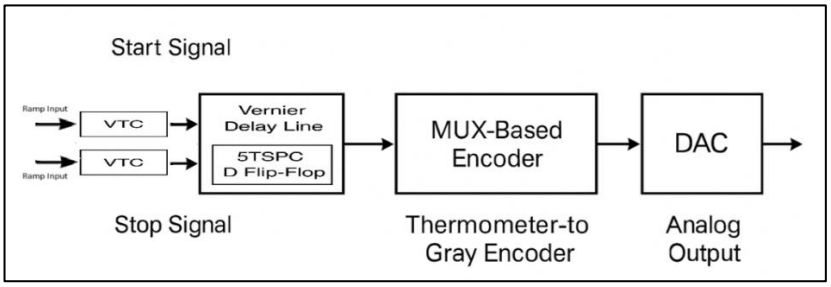


Figure 3.5 First Architecture Methodology

3.3.2. Second Architecture

The second architecture uses a less complex and more time-flexible configuration (Figure 3.6). A single VTC is used to create the START signal and a fixed-duration and period Vpulse generator is used for the STOP signal with multiple rising edges for high-rate triggering. This signal is passed into a Vernier Delay Line composed of edge-triggered flip-flops that sense the difference in timing among discrete rising clock edges with no dependence on the overlap of pulses.

In place of the Gray-coded encoder of the initial architecture, the system utilizes a tree-structured MUX-based binary encoder that is specifically intended to convert the thermometer output into binary codes. This reduces encoding delay and logic depth of the encoding process, keeping with the high-speed goals of this architecture. Digital output is presented to the same resistor-ladder DAC in order to reconstruct an analog waveform. The system is optimized for frequency performance and simplicity of architecture to allow clean capture of timing with low overhead in encoding.

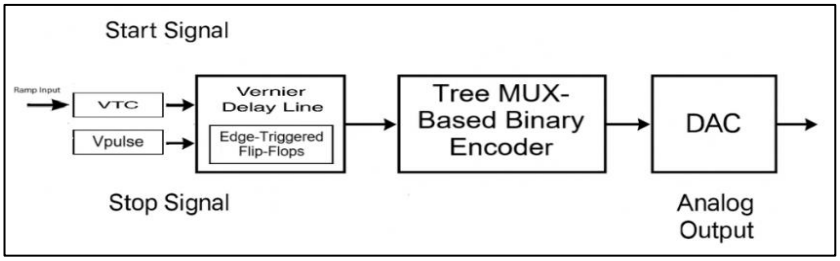


Figure 3.6 Second Architecture Methodology

These two designs highlight the flexibility of Vernier Delay Line-based TDC systems in response to varied triggering situations and in terms of circuit restrictions. By adjusting the signal generation, the flip-flop logic, and encoder approach each architecture solves a distinct set of challenges yet remains compatible with the same analog DAC output stage. The subsequent chapters offer the comprehensive implementation, simulation outcomes, and performance analysis of each method.

CHAPTER 4

4. A NOVEL DESIGN METHODOLOGY FOR VOLTAGE-TO-TIME CONVERTER (VTC)

4.1. VTC PRINCIPLE OF OPERATION

The Voltage-to-Time Converter (VTC) is an essential building block for time-domain analog-to-digital interfaces, converting analog voltages in terms of corresponding time intervals. The conversion principle is especially desirable in contemporary CMOS technologies because of its scalability, power consumption efficiency, as well as natural compatibility with digital signal processing. Instead of directly digitizing the voltage as in conventional ADCs, analog amplitude is converted to the timing or delay of an edge in the digital. The time which is encoded is then digitized separately using a Time-to-Digital Converter (TDC), resulting in an efficient, small front-end with increased speed.

The operating principle of VTC is based on comparing an input ramp signal of variable amplitude with a fixed threshold bias reference voltage. As represented in Figure 4.1, the system consists of a Track-and-Hold (T/H) circuit stage, cascode PMOS bias circuit, differential high-speed comparator, and inverter CMOS buffer. The T/H circuit is triggered with linear ramp input, sampling and holding the voltage using non-overlapping clock signals. The held-in value is thereafter compared with fixed reference threshold bias voltage Vth, with the output of the comparator switching when the held-in voltage crosses over the threshold. The time period from ramp start until the output of the comparator switches is proportional with the analog voltage, converting the analog signal to the time domain.

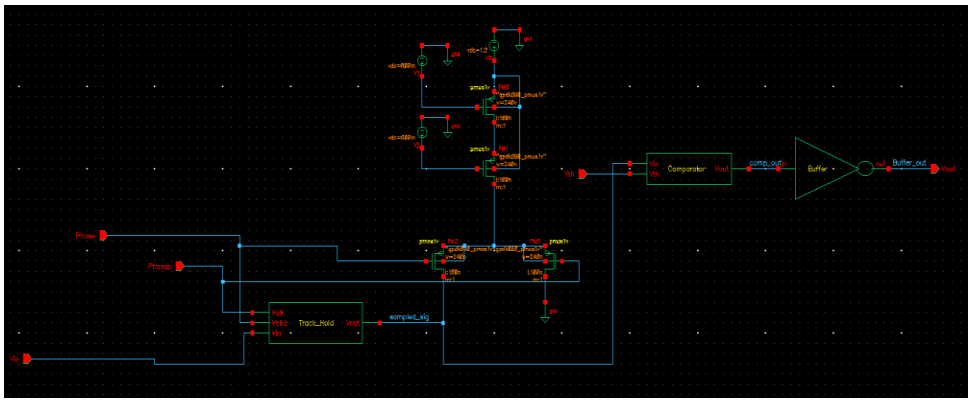


Figure 4.1 Schematic of the proposed Voltage-to-Time Converter (VTC)

The relationship between the time delay and the input voltage can be described by the following equation:

$$t_{delay} = \frac{V_{th}}{S} \qquad (4.1)$$

Where S is the slope of the input ramp signal. For a linear ramp, this relationship is direct, and the timing resolution depends on both the ramp steepness and comparator switching accuracy. The sharper the ramp, the shorter the delay required to reach the threshold, thus enabling finer time discrimination.

The work presents a new dual-VTC configuration where the same VTC is instantiated twice, once for producing the START signal and once for producing the STOP signal. Both VTCs operate on different ramp inputs: the full-scale ramp (0–1.2 V) is used in the START path, while the reduced swing ramp (0–0.5 V) is used in the STOP path with both of them having the same comparator threshold of 300 mV. The dual-ramp approach increases the system's dynamic range as well as its adaptability to varying signal conditions. Leveraging the different ramp slopes, the system can provide high timing sensitivity over an enlarged voltage range and provide support for hybrid or range-adaptive TDC architectures.

A visual illustration of such behavior is presented in Figure 4.2, in which the ramp inputs, control clocks, and digital outputs are shown. The ramp inputs

(Vin and Vin2) in the first subplot ramp linearly and reset periodically. The dashed line in the figure represents the threshold voltage, which is constant. The second subplot is the control clock signals for the two-phase clock used for controlling the Track-and-Hold switches. The last two subplots show the produced START pulses and the STOP pulses. The interesting fact is that the time delay from the clock edge until the output switches is ramp amplitude dependent, as is confirmed with successful voltage-to-time conversion.

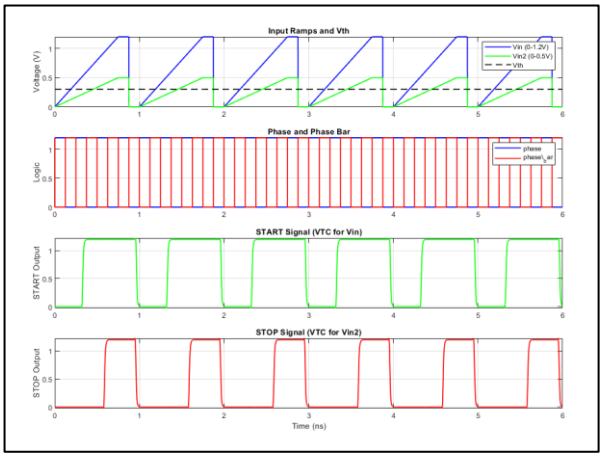


Figure 4.2 Timing Diagram for Dual-Ramp VTC Operation

This dual-use VTC architecture introduces a significant improvement in design compactness, resolution control, and adaptability without requiring additional circuit complexity. The uniformity of the two VTC instances also simplifies layout, matching, and integration in system-on-chip (SoC) environments.

4.2. CMOS TRACK-AND-HOLD IMPLEMENTATION

Track-and-Hold (T/H) circuit is an integral block in the Voltage-to-Time Converter (VTC) circuit architecture used for capturing the input ramp-voltage value instantaneously for later time-domain quantization. For the purposes of this work, we have used a passive CMOS T/H structure with NMOS clocked

transmission gates with non-overlapping clock signals. Employing simple transmission-gate design as an alternative to bootstrapped switches or op-amp-based buffer structures reduces power, silicon area, as well as circuit complexity considerably with enough speed for operation in the GHz range. A comparable ramp-based voltage-to-time converter using a more complex delay-chain and calibration loop was proposed by (Shete, Gokhale, & Kale, 2019), but at the cost of greater design complexity and circuit area. The passive CMOS T/H circuit of the design is based on the structure suggested by (Zhang, Xie, & Geiger, 2015), with their work proving its applicability for high-speed sampling in future VTC systems. The design's simplicity and GHz-range compatibility were motivating factors for its incorporation in the suggested architecture.

The circuit in Figure 4.3 comprises two NMOS transistors (W=500 nm, L=100) switched by complementary clock phases (Vclk, Vclkb). During the track mode, the input ramp signal Vin is permitted to charge small hold capacitor C_0=1 fF. When the clock is switched over to the hold mode, the path of transmission is broken and the sampled voltage is maintained across the capacitor for processing in the comparator. The setup allows for high-speed operation with sampling distortion from channel charge injection and finite settling time.

The dynamic behavior of the circuit is illustrated in Figure 4.4, which displays the time-domain waveforms for the input signal, clock phases, sampled output, and sampling error. The sampling error e(t) at each sampling instant t_i is defined as:

$$e(t_i)=V_{in}(t_i)-V_{sampled}(t_i) \qquad (4.2)$$

To quantify this error across the entire observation window, the Root-Mean-Square Error (RMSE) is computed using:

$$RMSE=\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(V_{in}(t_i)-V_{sampled}(t_i)\right)^2} \qquad (4.3)$$

In the current implementation, this computation produced an RMSE of 0.2706 V with maximum absolute error of 0.7630 V, which captures the impact of non-ideal switching characteristics as well as settling within the sampling interval. These numbers may be large when considered in terms of absolute values, but let us stress that the system is clocked at 1 GHz, so every sample-hold cycle is finished within under 1 nanosecond. Under such rapid operation, compromises between accuracy, settling, as well as power consumption inevitably have to be made. The passive CMOS T/H architecture was chosen exclusively for its low delay and small area while producing decent performance with such constraints. Under high-frequency operation, such sampling errors are tolerable in view of sampling an input signal for the purpose of producing timing transitions, not accurate voltage quantization.

To assess linearity, a regression analysis was performed between the ideal interpolated input voltage and the sampled output values. The best-fit line follows the equation:

$$V_{sampled}=a\cdot V_{in}+b \qquad (4.4)$$

where a=0.66 and b=0.25 are the slope and intercept, respectively. The goodness of this linear approximation is characterized by the coefficient of determination:

$$R^2=0.6415 \qquad (4.5)$$

The resulting plot, Figure 4.5, exhibits moderate linearity with some indication of gain compression and stair-stepped discontinuities, which is both typical of passive switching networks with limited resolution and timing constraints. This is an acceptable degree of linearity in the present design context, given that the sampled voltage is not digitized directly, rather being utilized for triggering purpose in a high-speed comparator. Hence, the key performance measure is not that of accuracy in voltage, but of the reproducibility

and consistency of the resulting timing edge, a requirement that is met reliably within the intended frequency domain.

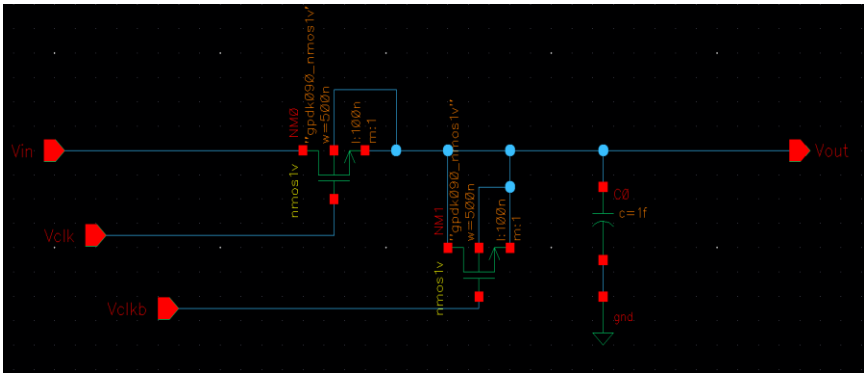


Figure 4.3 CMOS Passive Track-and-Hold Circuit Schematic

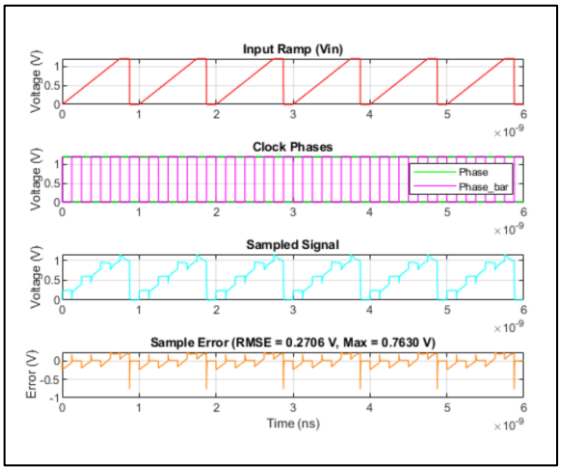


Figure 4.4 Track-and-Hold Circuit Time-Domain Behavior

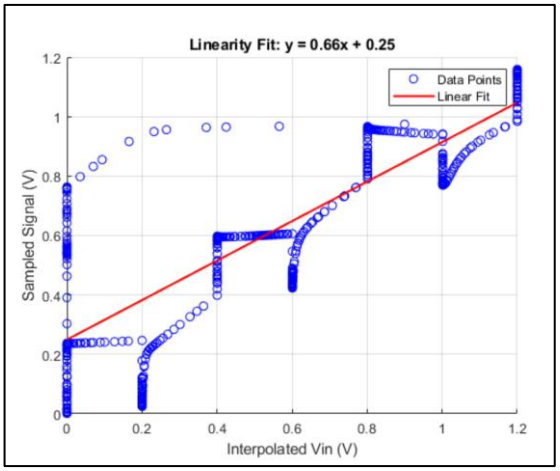


Figure 4.5 Linearity Analysis of the Sampled Signal

Despite the non-idealities encountered, the passive T/H circuit is an extremely compact and low-energy approach for time-domain signal processing. Switch bootstrapping, scaling of capacitance, or the integration of active buffer circuitry could be considered in future for further improvement of linearity as well as for reducing sampling-related artifacts in cases of systems demanding increased accuracy or tougher comparator interfacing.

4.3. CASCODE BIASING NETWORK

With the suggested Voltage-to-Time Converter (VTC) topology, accurate biasing is of utmost importance in order to have high-speed, stable circuit operation, especially in the presence of process, voltage, and temperature (PVT) variations. For the purpose of offering a stable, current source with a high impedance for the sampling as well as the comparator circuit, a cascode PMOS current mirror is used. While in other applications of this block reference voltages are normally generated, in the current implementation the block is used exclusively as a current source; the threshold comparator voltage is provided as an external fixed DC input. The cascode current mirror employed for biasing the comparator as well as the differential comparator structure itself which will be

introduced in the next chapter have their roots in the work of (Shete, Gokhale, & Kale, 2019) whose design successfully balances response time, gain, and low-voltage CMOS process robustness. Adjustments were made for conformance with the 100 nm design requirement and desired delay range.

The cascode biasing structure used here is composed of four stacked PMOS transistors as shown in Figure 4.1. All the devices have channel width W=240nm and channel length L=100nm, in a 100 nm CMOS process. The upper two PMOS transistors are supplied with externally applied gate voltages of Vb1=800mV and Vb2=600mV such that all the transistors are in the saturation region with high output resistance. The method of biasing used here stabilizes the mirrored current with no dependence on drain voltage variations, thereby providing superior output impedance as well as signal quality.

The cascode mirror current is delivered to the major components of the VTC structure, including the comparator tail current as well as the Track-and-Hold (T/H) circuit. One advantage of the use of cascode topology is that current source performance is increased as output impedance is raised nearly to:

$$r_{out,\ cascode} \approx r_o(1+g_m r_o) \qquad (4.6)$$

where r_o is the intrinsic output resistance of an individual PMOS device and gm is its transconductance. This enhancement suppresses signal-dependent distortion and provides for more linear performance in subsequent stages.

Note that the threshold voltage of the comparator, i.e., Vth=300mV, is not generated from the cascode mirror but is instead externally injected from a specific VDC bias source. This operation makes control over the switching point simpler and facilitates accurate threshold adjustment irrespective of bias current tuning. The decoupling of current biasing from threshold generation allows modular optimization as well as increased design flexibility.

Summarily, cascode current mirror makes for an effective bias source of the VTC structure, stabilizing the operation of time-sensitive analog blocks through allowing accurate external adjustment of comparator threshold voltage.

The device is specifically suitable for use in GHz-speed mixed-signal systems, thanks to its high output impedance as well as its compact implementation in CMOS.

4.4. COMPARATOR ARCHITECTURE AND THRESHOLD SELECTION

The comparator in the suggested Voltage-to-Time Converter (VTC) structure is created in order to identify when the input sampled voltage is above a set threshold voltage, thus generating an output digital timing pulse that represents the analog signal in the time domain. For this purpose, the common CMOS differential pair structure is used, followed by an output buffering stage for producing rapid output transitions compatible with GHz-speed applications.

As depicted in Figure 4.6, the comparator is made of two matched NMOS transistors in differential pair configuration, biased with an externally supplied tail current source. The PMOS load is used as a current mirror for symmetrical operation with high gain. Each transistor is sized with a channel width W=120nm and channel length L=100nm, and the comparator is supplied with 1.2 V supply. The threshold voltage is not internally produced but supplied through an external voltage source with value Vth=300mV, applied in the inverting input of the comparator. The non-inverting input is supplied with the sampled signal from the Track-and-Hold circuit. The output of the comparator is driven through the CMOS inverter buffer, which provides signal integrity in the form of full-swing logic levels as well as with reduced analog core load.

To describe the performance of the comparator, the Cadence simulated data were used for the transient analysis. The sampled input voltage $V_{in}(t)$ in conjunction with the fixed threshold voltage $V_{th}(t)$ were processed with the use of MATLAB, as well as the comparator output $V_{out}(t)$. The main behavioral characteristics considered in the analysis are t_{trip} point, threshold margin, and comparator delay.

The trip point is defined as the time instant t_{trip} when the sampled voltage crosses the threshold:

$$V_{in}(t_{trip})=V_{th}(t_{trip}) \quad (4.7)$$

The threshold margin is calculated at this instant as:

$$\Delta V=V_{in}(t_{trip})-V_{th}(t_{trip}) \quad (4.8)$$

It provides a measure of how decisively the comparator input crosses the decision level, with higher margins indicating more robust switching behavior.

The comparator delay is defined as the time difference between the threshold crossing and the actual output transition:

$$\tau_{delay}=t_{out}-t_{trip} \quad (4.9)$$

where t_{out} is the time at which the comparator output changes logic state (e.g., from 0 to 1).

The comparator action is presented in Figure 4.7 with the ramp input signal, the fixed threshold voltage, and the output waveform. Trip points and their corresponding delays have vertical annotations. From the analysis, several switching events were derived over several cycles. Delays were in the range 0.8 ps–26.6 ps, while threshold margins were within ±30 mV, with clean and sharp decision boundaries. These observations confirm the comparator's functionality with respect to accurate performance under 1 GHz operation, where the timing budget in each conversion cycle is just 1 ns. With a delay of tens of picoseconds, the comparator imposes negligible timing error in the VTC system.

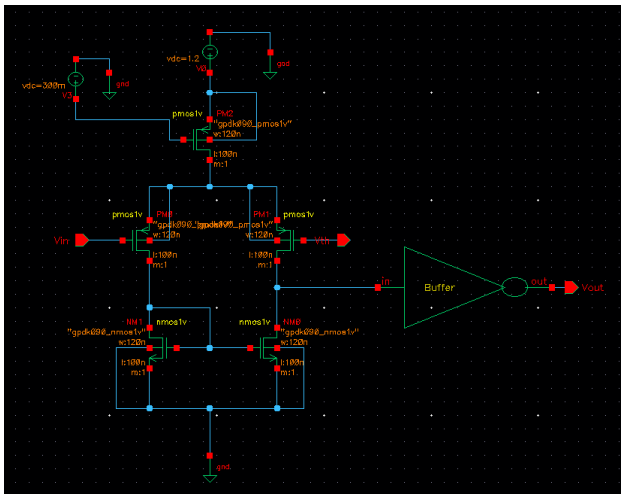


Figure 4.6 Differential Comparator with PMOS Current Mirror Load

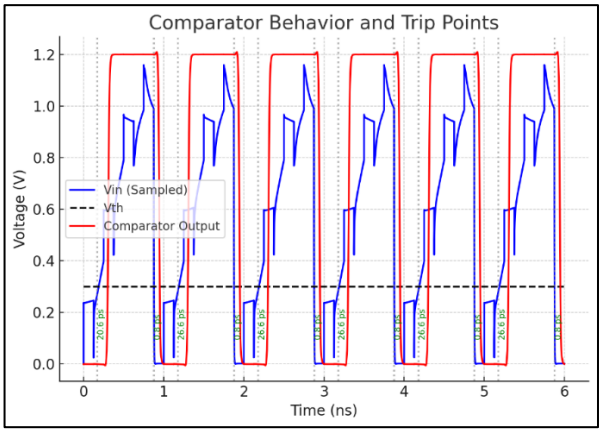


Figure 4.7 Comparator Input and Output Behavior with Delay Annotations

In short, the comparator stage is both structurally simple and operates rapidly, enabling precise timing conversion in the VTC. The delay performance of the stage, together with its clean switching nature, renders the stage suitable for integration in GHz-range mixed-signal systems.

4.5. DUAL-RAMP VTC STRATEGY AND BENEFITS

A major innovation of the suggested VTC circuit is the incorporation of a dual-ramp input approach, providing increased sensitivity as well as flexibility in an overall compact, CMOS-compatible structure. Instead of relying on an individual ramp input, in the proposed design, two differently scaled ramp inputs, namely Vin (0–1.2 V) and Vin2 (0–0.5 V) that are driven simultaneously into two identical Voltage-to-Time Converters (VTCs) with the same reference threshold voltage of Vth = 300 mV, are used. The dual-input concept allows for more robust performance under various input conditions, with the circuit being capable of dynamically selecting the highest accurate time output path or combining both paths for hybrid encoding with enhanced linearity.

The dual-ramp structure takes advantage of the fact that ramp amplitudes have varying crossing characteristics with the comparator threshold. The larger amplitude ramp (Vin) results in more rapid threshold crossings and finer resolution, while the smaller amplitude ramp (Vin2) is sensitive even when the input signal range is limited. As presented in Figure 4.8, both ramp waves and the fixed threshold voltage are graphed, which demonstrates how the points of intersection vary in both time as well as slope, which in turn influences the delay created by the VTC.

To create START and STOP pulses, both Vin and Vin2 go through the same T/H → Comparator → Buffer stages, creating two time-encoded digital pulses. The START signal is from Vin, while the STOP signal is from Vin2. These logic outputs are shown in Figure 4.9, with clear indication of the START and STOP signals, ascertaining their periodicity as well as their alignment with respect to the ramp input.

MATLAB processing of the VTC output verifies that the dual-ramp circuit efficiently translates analog voltage variations to discrete digital time pulses with great timing accuracy. The resolution estimated between the START and STOP transitions is about 4.14 ps, and the system exhibits stable, periodic transitions even for a 1 GHz ramp frequency. This demonstrates the reliability of the simple

design even with high frequency without the necessity of complex bias or bootstrapped circuitry. The full dual-VTC structure is schematically shown in Figure 4.10.

The work presents some original contributions. First, the work makes use of a minimalist CMOS-based T/H circuit that eschews bulky op-amps or bootstrapped switches, minimizing power consumption, area, and design costs. Second, the dual-ramp approach increases input dynamic range while supporting adaptive operation for amplitude variability. Third, the whole design is CMOS-compatible in terms of its operating voltages while achieving sub-5 ps resolution, its suitability for integration in low-power, high-speed Time-to-Digital Converters (TDCs) in advanced mixed-signal systems being demonstrated.

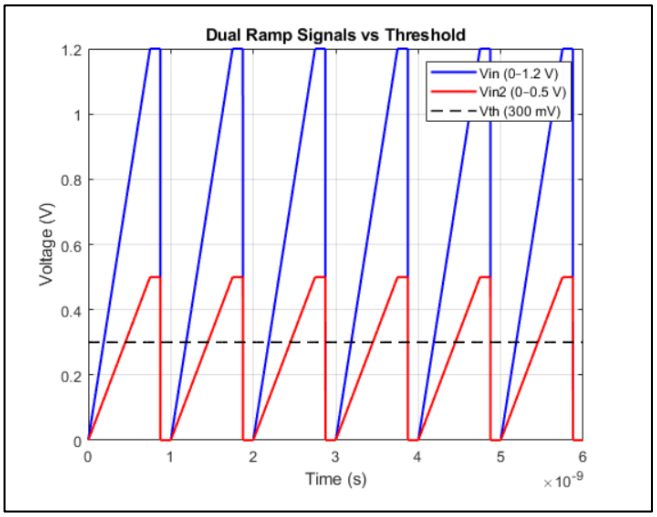


Figure 4.8 Dual Ramp Input Signals and Comparator Threshold

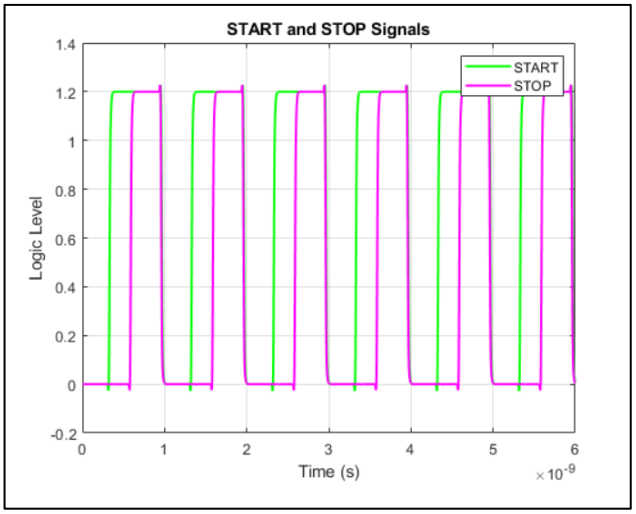


Figure 4.9 START and STOP Digital Outputs Generated by Dual VTC Circuits

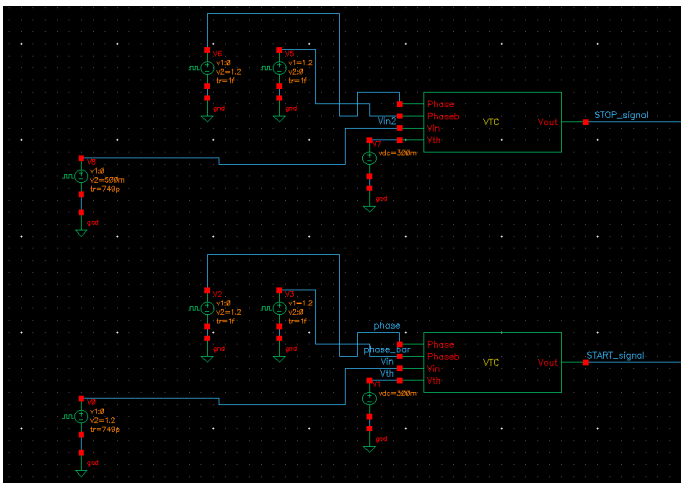


Figure 4.10 Schematic Diagram of Dual-Ramp VTC Architecture

4.6. PERFORMANCE ANALYSIS

This section presents the performance evaluation of the proposed dual-ramp Voltage-to-Time Converter (VTC) architecture. The analysis includes time-domain behavior, delay extraction, linearity characterization, and threshold sensitivity. All data were obtained from 6 ns transient simulations of the START

and STOP signal paths under GHz-range input ramp conditions. Table 4.1 includes a summary of performance analysis.

4.6.1. Delay Extraction and Resolution

To quantify the VTC's time-conversion precision, the delay from the instant the ramp input crosses the threshold voltage (V_{th}= 0.3 V) to the corresponding START or STOP signal transition was measured. The delay is calculated using:

$$\Delta t_i = t_{output,i} - t_{cross,i} \qquad (4.10)$$

where $t_{cross,i}$ is the time when the ramp crosses V_{th}, and $t_{output,i}$ is the time the START or STOP output rises.

From the simulation, the delay range for the START signal was found to be 0.02 ns to 0.05 ns, while for the STOP signal, it ranged from 0.04 ns to 0.05 ns, as shown in Figure 4.14. With a total of 6 quantization steps, the average resolution is computed as:

$$\text{Resolution}_{avg} = \frac{1}{N-1} \sum_{i=1}^{N-1} (\Delta t_{i+1} - \Delta t_i) \approx 4.14 \text{ ps} \qquad (4.11)$$

This verifies the capability of the VTC for producing fine, uniform timing intervals appropriate for sub-nanosecond Time-to-Digital Converters (TDCs). The suggested VTC attains around 4.11 ps of time resolution with 6 steps of quantization over a 30 ps delay span. Although existing high-speed realizations in scaled CMOS technologies have shown multi-GS/s performance, like the 5 GS/s VTC in the work of (Zhang, Xie, & Geiger, 2015), these are normally dependent upon aggressive pipelining and consume much larger silicon area. The current structure instead balances speed with simplicity, facilitating simple integration within small mixed-signal systems.

4.6.2. INL and DNL Characterization

The linearity of the quantized time outputs was analyzed using Differential Non-Linearity (DNL) and Integral Non-Linearity (INL). DNL reflects the deviation of individual steps from the ideal resolution:

$$DNL_i = \Delta t_i - Resolution_{avg} \quad (4.12)$$

INL is the cumulative deviation from ideal linearity:

$$INL_i = \sum_{k=1}^{i} DNL_k \quad (4.13)$$

As shown in Figure 4.12, the maximum observed DNL was 27.72 ps, and the maximum INL was 16.51 ps. These results are within acceptable limits for a comparator-based VTC operating under high-speed input conditions.

4.6.3. Monotonicity and Threshold Margin Analysis

VTC's delay ideally should decrease monotonically as input voltage goes up. The START and STOP delays were not necessarily monotonic, though, because of analog circuit constraints as well as comparator transition noise. This is normal for GHz-range implementations in which finite comparator response time as well as small ramp slopes introduce small jitters in terms of time.

In addition, the margin between the ramp input and the comparator threshold at the crossing point was calculated as:

$$Margin_i = V_{in,i} - V_{th} \quad (4.15)$$

The average threshold margin was determined as about −1 mV, ascertaining that the comparator transitions take place right about the threshold as desired with great accuracy and little overshoot or noise-driven deviation.

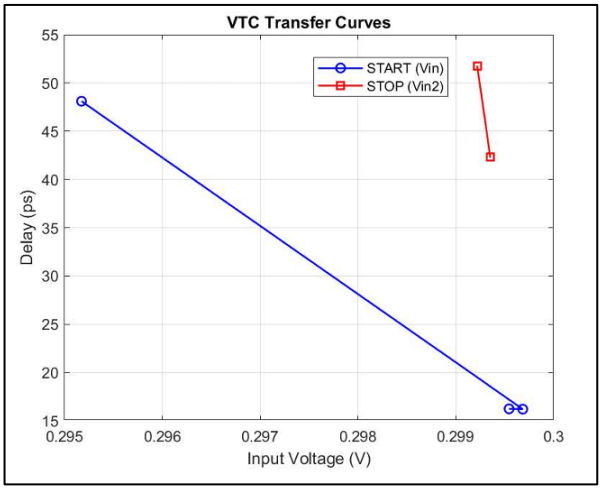


Figure 4.11 VTC transfer curve: Delay vs Input Voltage for START and STOP

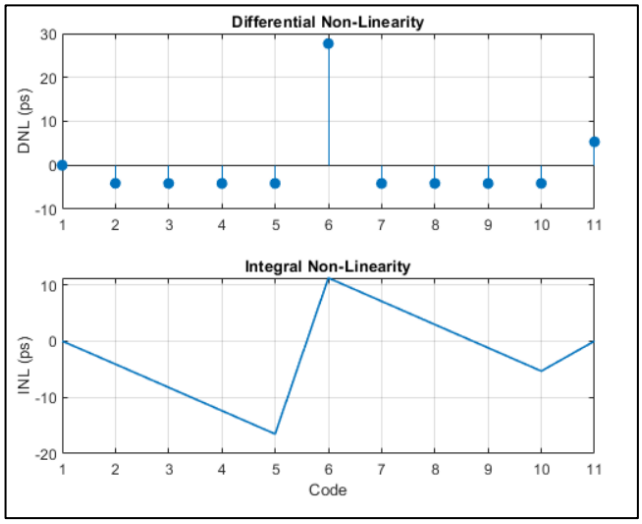


Figure 4.12 Differential and Integral Non-Linearity plots

Table 4.1 Summary of VTC Performance Metrics

Parameter	Value	Remarks
Input Ramp Range	0–1.2 V and 0–0.5 V	Dual-ramp inputs with shared comparator threshold (V_{th})
Comparator Threshold (V_{th})	0.3 V	Fixed for both ramps
Average Time Resolution	4.14 ps	Calculated from quantized delay steps
Total Quantization Steps	6	From extracted delay range
Delay Range	20 ps – 50 ps	Full VTC response window across both paths
Max Differential Non-Linearity (DNL)	27.72 ps	Variation in successive delay steps
Max Integral Non-Linearity (INL)	16.51 ps	Cumulative deviation from ideal linear response
Threshold Crossing Margin	−1 mV (avg)	Indicates precise comparator triggering
Monotonicity	Not strictly monotonic	Minor jitter near threshold crossings observed at GHz frequency
Input Frequency	1 GHz	Ramp signal frequency used in simulations

CHAPTER 5

5. VERNIER DELAY LINE (VDL) ARCHITECTURE

5.1. PRINCIPLE OF VERNIER DELAY LINE

The Vernier Delay Line (VDL) is the basic architecture used in Time-to-Digital Converters (TDCs) to provide high-resolution measurement of time. The principle is to pass the START and the STOP signals through two different chains of unit delays. As indicated in Figure 5.1 (Wang, Zhang, & Liu, 2017), the topmost chain provides the START signal with the delay of τ_1 per stage, and the bottom one provides the STOP signal with the delay of τ_2, where the difference in the delays, is given as $\Delta\tau=\tau_1-\tau_2$, is what actually determines the resolution of the system.

As the START pulse travels through its delay line, it slowly "catches up" to the STOP pulse, which travels through the lengthier delay chain. At every stage, the two signals are compared using the D-type flip-flops. As the START pulse reaches the logic high simultaneously as the STOP pulse crosses a rising edge (used as the clock input), the flip-flop registers the occurrence and produces the high logic level at the stage. The index of the earliest high output is the duration between the START and the STOP pulses.

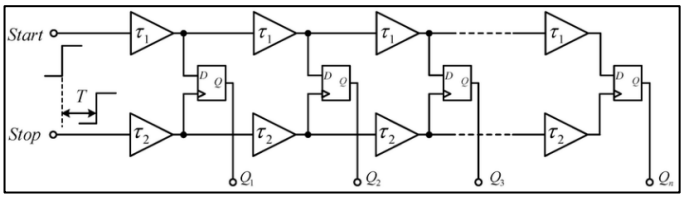


Figure 5.1 Time-to-Digital converter (TDC) based on a Vernier Delay Line (Wang et al., 2017)

In the paper, the START and the STOP path delay cells are implemented as cascaded CMOS buffers with three stages in each buffer. This arrangement has the advantage of providing unambiguous propagation delay with improved drive capability as well as improved signal integrity. This is simpler compared to other complicated structures such as differential pairs or the current-starved inverters and has the benefit of low power consumption and adequate resolution to achieve high-speed operation in the TDC.

The flip-flops at every stage of delay are implemented using 5-transistor True Single Phase Clocking (TSPC) logic-based D flip-flops. This small cell is selected for the fast switching action and low sensitivity to clock skew. The input START is supplied to the D input, and the input STOP is used as the clock input. The output is set to logic high only when both the D and CLK are set to high at the same instant. At that instant, the pass transistors within the internal stages turn on, enabling the input to go through to the output. In particular, when the clock is high and input D is additionally high, the first-stage PMOS and second-stage NMOS transistors are cut off, and the other transistors are conducting, setting the output to high. Therefore, the circuit is acting as a level-sensitive latch during the active clock phase, setting and transferring the input to the output only during this condition.

The detection mechanism does not depend on retaining the previous state of the output. Rather, the intention is to sense the moment at which the START and the STOP signals overlap to provide instantaneous pulse coincidence that is best suited for thermometer coding as well as for rapid evaluation of the signal.

The system's resolution, $\Delta\tau$, is expressed as:

$$\Delta\tau=\tau_{START}-\tau_{STOP} \qquad (5.1)$$

This resolution affects the smallest measurable amount of time. The chain of delays continues to propagate both signals stage by stage until there is a rising edge on the STOP signal and a logic HIGH on the START path.

5.2. BUFFER DESIGN AND DELAY CHARACTERIZATION

In the designed Vernier Delay Line TDC architecture, the delay element is realized through compact and efficient design with three cascaded CMOS Buffers as indicated in Figure 5.3. This design selection allows the START and the STOP signals to pass through uniformly timed delay stages with less distortion and jitter. The inverters are sized using a standard CMOS technology with PMOS and NMOS widths of 240 and 120 nm, respectively, and an equal channel length of 100 nm, powered at 1.2 V. This inverter string operates as the buffer cell where each buffer offers unit delay to facilitate the GHz range of fine time interpolation.

The simulation of the buffer transient was carried out using Cadence Virtuoso, and the resultant waveform as shown in Figure 5.2 indicates the START input and its delayed output after traversing the cascaded buffer. MATLAB was employed to analyze the waveforms of the signals exported from the simulation data to determine the buffer propagation delay accurately. The buffer propagation delay was calculated by finding the instant at which the input and the output signals passed 50% supply voltage using the formula:

$$t_{delay}=t_{out,50\%}-t_{in,50\%} \qquad (5.2)$$

In the implementation, the calculated propagation delay was about 64.15 picoseconds, which is consistent with the resolution target set for the TDC system.

Aside from latency, the input and the output rise times were also analyzed using the standard definition of the 10% to 90% rise time of the full swing voltage:

$$t_{rise}=t_{90\%}-t_{10\%} \qquad (5.3)$$

From the analysis, the rise time of the input signal was found to be about 21.88 picoseconds, and that of the output was found to be about 10.01 picoseconds. These values affirm that the buffer circuit not only provides the desired amount of the delay but also sharpens the edges of the signal as it is sent to the successive stage, adding to the timing accuracy along the Vernier chain.

The buffer structure schematic in Cadence is presented in Figure 5.4. The multi-stage buffer cell is superior to the single-inverter delay stages employed in standard time-to-digital circuits and offers improved control of the signal shape along with GHz-range operation capability. Three stages were decided to cascade in order to achieve the extension in the delay along with maintaining sharpness of the edge, which is essential in the case of Vernier-based interpolation techniques. It is motivated from existing work on CMOS-based buffer delay cells (Ismail, 2015) but has been optimized for higher-frequency operation.

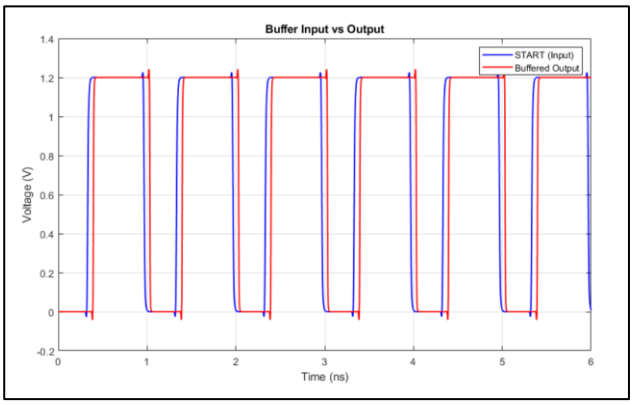


Figure 5.2 Transient waveform of START signal input and buffered output

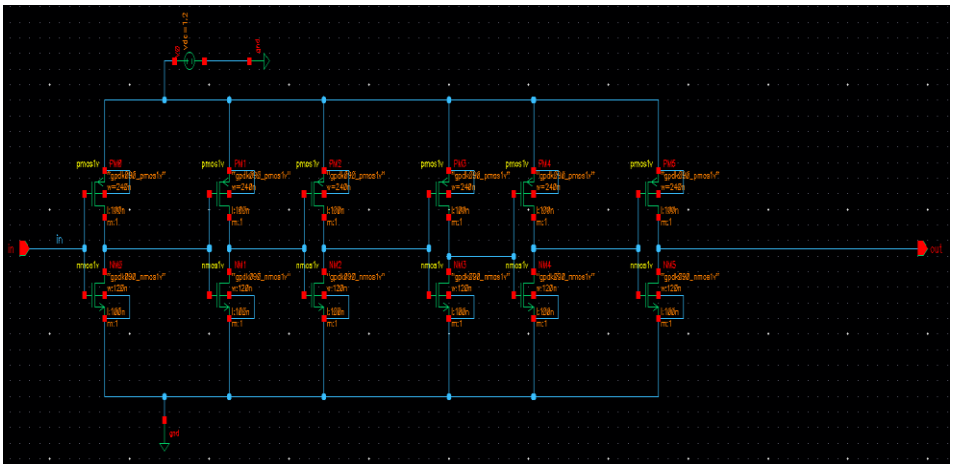


Figure 5.3 Schematic of three-stage CMOS buffer implemented in Cadence Virtuoso

5.3. FLIP-FLOP INTEGRATION AND EDGE DETECTION

In the given architecture, the flip-flop performs a crucial function in storing the phase correlation among the delayed START and stop signals in the Vernier Delay Line (VDL) Time-to-Digital Converter (TDC). In particular, the TSPC-based D Flip-Flop is used for edge detection between the two signals because it has high-speed and area-efficient architecture. TSPC (True Single Phase Clocking) logic family is a dynamic CMOS family that does away with the requirement for multiple phases of the clock and clock skews management and is very well-suited for GHz-range timing.

The flip-flop employed in this project is a 5-transistor TSPC-based D Flip-Flop consisting of three NMOS with W=835nm and two PMOS with W=1.665µm transistors as presented in Figure 5.5. It has a minimized area and high-speed switching-optimized structure. When the D input and also the clock signal are logic high, the PMOS transistors in the first stage and the NMOS transistors in the second stage are in a disabled condition, while the other transistors are conducting to enable a logic high to pass to the output (Table 5.1).

With the clock still active, the circuit transfers the D input to the Q output transparently (Deb, Sharma, & Dev, 2019).

In this configuration, the START pulse is fed to the D input, while the STOP pulse acts as the clock input to facilitate the detection of the intersection of the two delay paths. The configuration operates to form a thin Q output pulse to denote phase differences that have been recorded by the VDL. The flip-flop acts as a phase comparator and sends a high pulse when the START pulse is high in the rising phase of the STOP pulse. A waveform simulation of the operation of the TSPC D Flip-Flop appears in Figure 5.4, illustrating the START (D), STOP (CLK), and Q output signals. Simulated results confirm correct edge-triggered operation in which the rising edge of the STOP signal samples the current logic level of the START input and updates Q.

To quantitatively compare the performance of the flip-flop, a MATLAB analysis was utilized to determine the Clock-to-Q propagation delay. The MATLAB analysis picked up rising edges of the STOP signal and timed the delay until a considerable change was observed in the Q output. The delay was calculated using the formula:

$$t_{delay}=t_{Q(rising)}-t_{CLK(rising)} \qquad (5.4)$$

Where $t_{Q(rising)}$ represents the rising edge timestamp of the Q output transition and $t_{CLK(rising)}$ represents the rising edge of the input clock. Averaging for all such transitions, the average Clock-to-Q delay was about 19.37 ps, which validates the appropriateness of this flip-flop for sub-nanosecond TDC designs.

In addition, the flip-flop functional accuracy was checked using windowed edge matching. The test ensures that Q transitions always follow rising CLK edges in case D is high, verifying that the timing works as expected for encoding in the VDL.

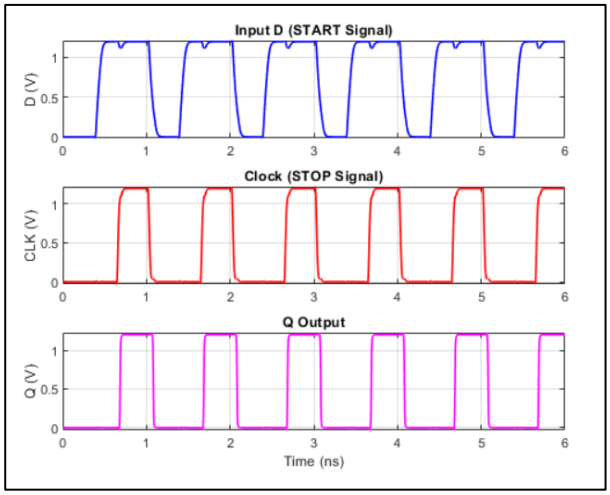


Figure 5.4 Timing Diagram of TSPC D Flip-Flop (D, CLK, Q)

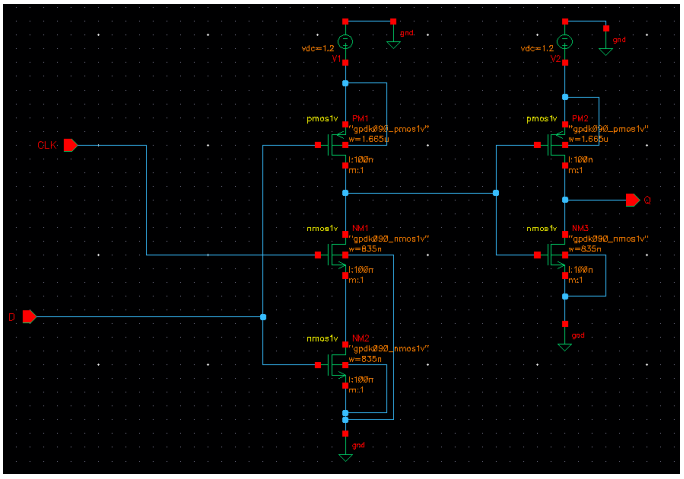


Figure 5.5 Schematic of 5-Transistor TSPC D Flip-Flop

Table 5.1 Truth Table of 5-T TSPC D flip flop (Deb, et al., 2019)

CLK	D	Q
1	0	0
1	1	1
0	0	0
0	1	0

5.4. VERNIER DELAY LINE TDC RESULTS AND EVALUATION

In this section, analysis and performance of the designed 15-bit Vernier Delay Line Time-to-Digital Converter (VDL-TDC) based on the waveform outputs, simulated data, and MATLAB analysis are presented. In the designed circuit, TSPC D flip-flops detect the overlapping pulse without retaining the logic level for achieving precise temporal detection of the narrow pulse.

5.4.1. Time-Domain Signal Behavior

Figure 5.6 shows the time-domain response of the input START and STOP signals, the delayed D and CLK signals in each stage, and the Q output pulses picked up by the flip-flop. The signals exhibit linear delay in the chain. Each Q output captures only the brief overlap between the delayed START (D) and STOP (CLK) signals, as in a Vernier configuration through pulse overlap detection.

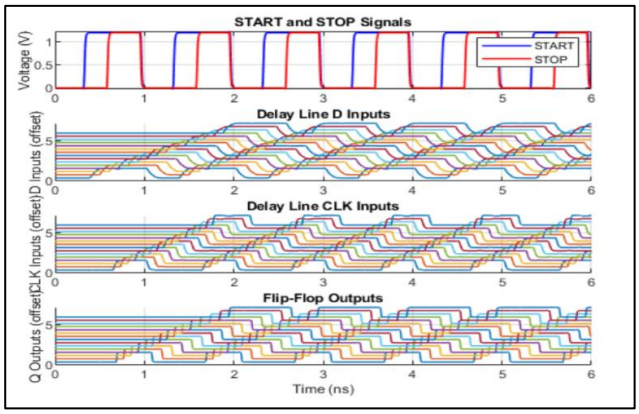


Figure 5.6 Time-domain visualization of START/STOP signals, D inputs, CLK inputs, and Q outputs across 15 VDL stages

5.4.2. Propagation Delay Analysis

Signal delay in the Vernier delay line was evaluated by monitoring the rising edge of the D, CLK, and Q signals. As shown in Figure 5.7, the delay for the three signals grows linearly, confirming the calibrated and symmetrical properties of the delay cells. This linearity ensures uniform resolution as well as reduced time-walk error.

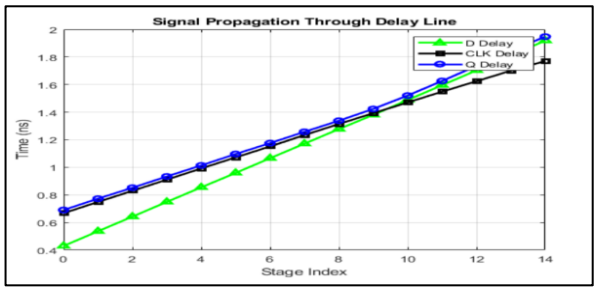


Figure 5.7 Propagation delay across delay line stages for D, CLK, and Q signals

Letting $\tau_{D(i)}$, $\tau_{CLK(i)}$, and $\tau_{Q(i)}$ represent the delay times for each signal at stage i, the delay propagation can be expressed as:

$$\tau_{D(i)}=t_{Di}-t_{START},\ \tau_{CLK(i)}=t_{CLKi}-t_{STOP},\ \tau_{Q(i)}=t_{Qi}-t_{START} \qquad (5.5), (5.6), (5.7)$$

Where t_{Di}, t_{CLKi}, and t_{Qi} are the rising edge timestamps at stage i.

5.4.3. Code Pattern and Histogram

In order to monitor the conversion behavior of the TDC, the Q outputs coded by the thermometer were decoded using their rising and falling edges. Figure 5.8 verifies that 16 distinct output codes were observed through the 15 stages. The algorithm for detection was dependent on pulse width and delay alignment instead of plain level thresholds to provide clear distinction even in the case of pulses of non-fixed width.

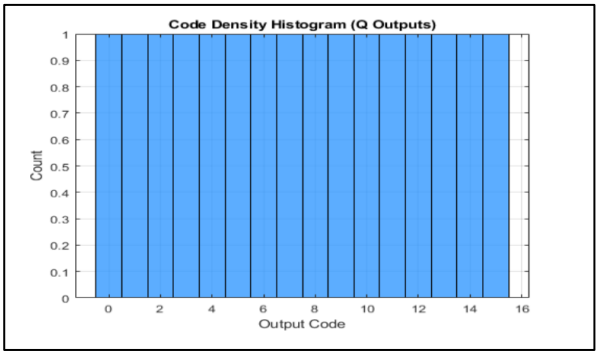


Figure 5.8 Histogram of decoded TDC output codes from 15-stage Vernier delay line

Each detected code corresponds to a specific time interval between the START and STOP signals, demonstrating the delay line's time resolution capabilities.

5.4.4. INL and DNL Characteristics

To further assess linearity, the spacing of the identified code transitions was used to calculate differential non-linearity (DNL) and integral non-linearity (INL) metrics. We used the following expressions:

$$DNL_i = \frac{\Delta t_i - \Delta t^-}{\Delta t^-},\ INL = \sum_{j=1}^{i} DNL_j \qquad (5.8), (5.9)$$

Where Δt_i is the code step size and Δt^- is the mean code step size (resolution).

Figure 5.9 shows that the maximum DNL was roughly 19.30 ps, and the maximum INL accumulated to 75.25 ps, demonstrating sufficient linearity for high-resolution applications.

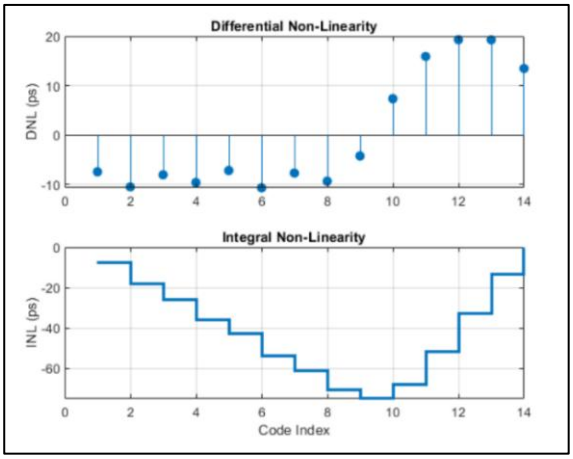


Figure 5.9 Differential and integral non-linearity for output code transitions

5.4.5. Performance Summary

Table 5.2 summarizes the performance characteristics of the Vernier Delay Line TDC, which were obtained using MATLAB to extract transition times and pulse lengths.

The dynamic range was defined as the total time span between the earliest and latest detected Q pulse (Table 5.2):

$$\text{Dynamic Range} = t_{Q14} - t_{Q0} \qquad (5.10)$$

The results show that the proposed TSPC-based Vernier Delay Line TDC can accurately resolve fine timing variations at 1 GHz input signal frequency. The increasing Q pulse widths, together with their linear delays, resulted in robust pulse discrimination and narrow overlap detection.

Table 5.2 Performance Summary of VDL Design and Simulation Results

Metric	Value	Details
Buffer Propagation Delay	64.15 ps	Delay measured from 50% VDD of input to output
Input Rise Time (Buffer)	21.88 ps	Measured from 10% to 90% of input signal
Output Rise Time (Buffer)	10.01 ps	Measured from 10% to 90% of output signal
Flip-Flop CLK-to-Q Delay	19.37 ps (average)	TSPC-based D Flip-Flop delay per stage
Number of VDL Stages	15 stages	D and CLK signals each propagate through 15 buffer-based delay cells
Detected Output Codes	16 codes (0 to 15)	From pulse overlap detection across 15 Q outputs
Resolution (Average Code Step)	89.79 ps	Mean delay difference between successive code transitions
Dynamic Range	1.26 ns	Time span between earliest and latest detected Q outputs
Minimum Pulse Width (Q Outputs)	395.95 ps	Narrowest overlap pulse captured by flip-flop chain
Maximum Pulse Width (Q Outputs)	698.94 ps	Widest overlap pulse captured by flip-flop chain
Average Pulse Width (Q Outputs)	585.93 ps	Mean pulse width from all detected Q pulses
Maximum DNL (Differential Non-Linearity)	19.30 ps	Maximum deviation from ideal step size
Maximum INL (Integral Non-Linearity)	75.25 ps	Cumulative deviation across code transitions

CHAPTER 6

6. ENCODER AND DAC DESIGN

6.1. ENCODER DESIGN AND SIMULATION RESULTS

In the current chapter, implementation of the encoder for Vernier Delay Line Time-to-Digital Converter (VDL-TDC) is discussed. The encoder is a principal element that converts the VDL-TDC thermometer code into a binary representation. The 15-4 multiplexer (MUX) with a series of XOR gates is implemented to design the encoder for maximum performance and highest reliability (Hussain et al., 2020). The method reduces bit transition errors and makes it suitable for GHz-level frequency applications.

6.1.1. MUX-Based Encoder Design

At the center of the encoder circuit design is a series of 2:1 multiplier MUXs (Dua & Rajput, 2020) that take their inputs directly from the 15 outputs of the VDL-TDC thermometer code ranging from Q0 to Q14. To implement the design, the inputs for one stage of the MUX are chosen as follows: A = Q11, B = Q3, and SEL = Q7. These inputs are processed by the MUX to generate the initial binary value. The MUX is designed using a CMOS-based circuit using 120 nm width NMOS and 240 nm width PMOS transistors with a constant length of 100 nm. This size is adopted for achieving the best combination of speed and energy efficiency for the circuit with maximum frequency performance.

The schematic of MUX that is necessary for the primary encoding step is shown in Figure 6.1. Complementary CMOS is used for the design to ensure that there is both high noise immunity and low power consumption, which are features that are necessary for GHz-speed applications. Figure 6.2 provides confirmation, from the simulation, of correct signal selection and switching

action. Figure 6.3 shows the CMOS circuit design of the inverter (NOT gate) that is shown in the MUX circuit design and Figure 6.4 shows the simulation of the inverter.

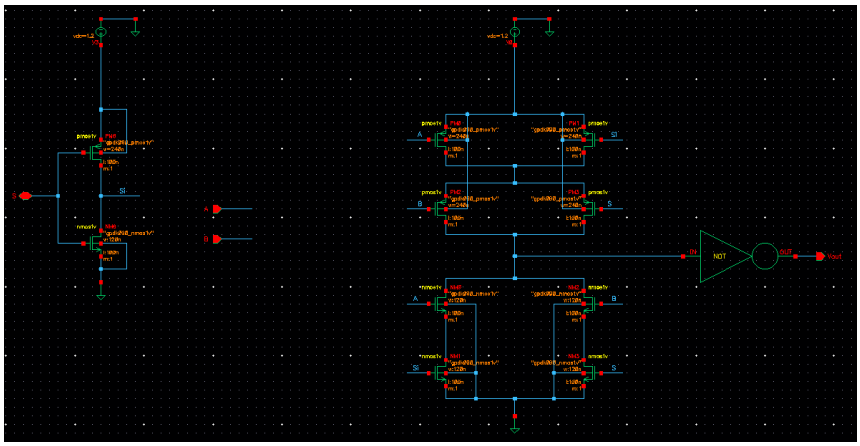


Figure 6.1 MUX Circuit Schematic for Encoder Design

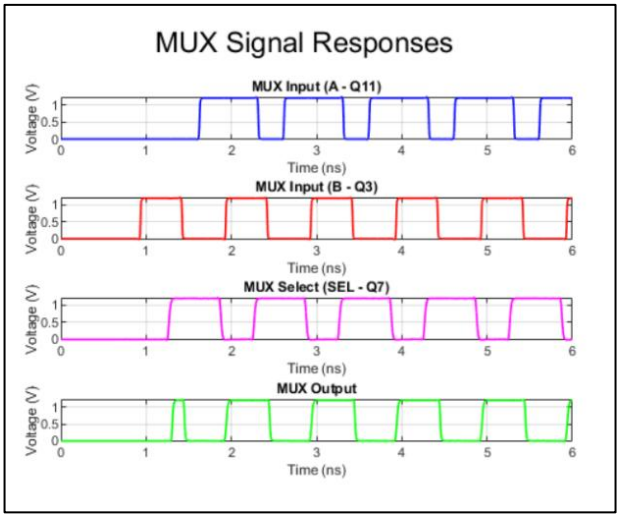


Figure 6.2 MUX Signal Responses (A = Q11, B = Q3, SEL = Q7)

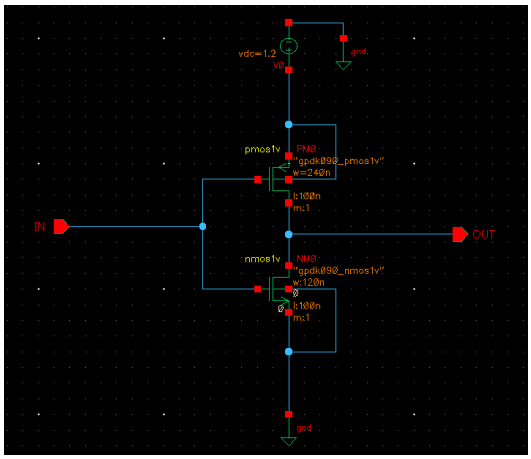


Figure 6.3 Inverter Circuit Schematic for MUX Design

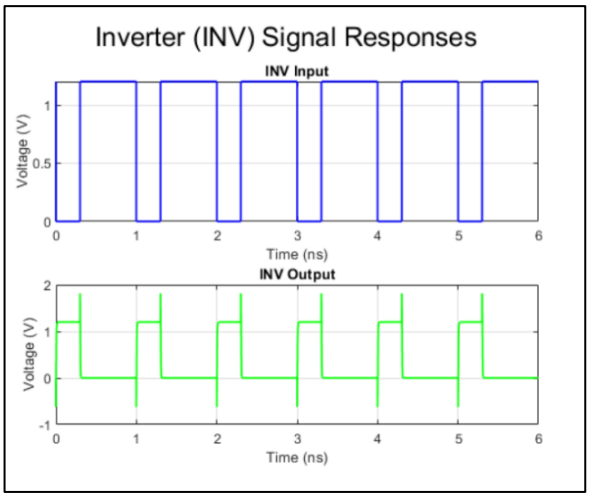


Figure 6.4 Inverter Signal Responses

6.1.2. The Rule of Buffers in The Encoder Circuit

Buffers at each MUX and XOR stage are implemented for stabilizing the signals and enhancing the performance of the circuit overall. The buffer circuit is implemented using the same NMOS (120 nm) and PMOS (240 nm) sizing for the transistors as is being used for the MUX circuit to provide consistent impedance matching and delay performance. This reduces the distortion and error caused by improper timing, thereby guaranteeing overall stability of the

encoder. Figure 6.5 presents the buffer circuit design, and Figure 6.6 illustrates the corresponding simulation results, which reveal improved signal quality.

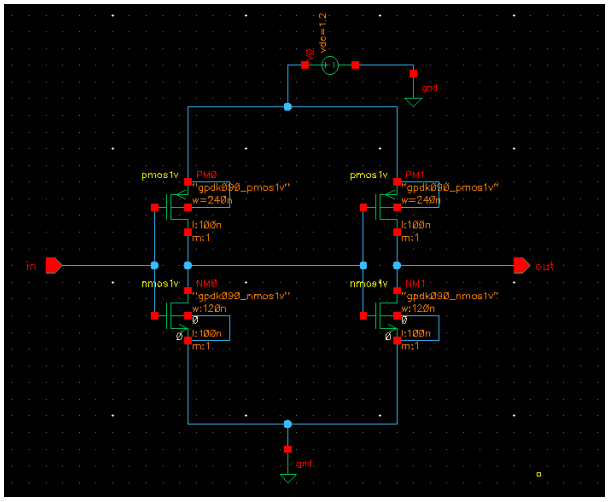


Figure 6.5 Buffer Circuit for Signal Stabilization

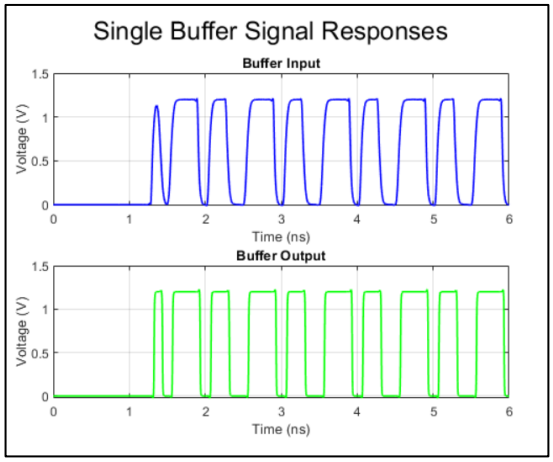


Figure 6.6 Single Buffer Signal Responses

6.1.3. Binary-to-Gray Code Conversion Using XOR Gates

To improve the performance of the encoder, the outputs from MUX stages are also converted into Gray code. This is done using CMOS XOR gates that minimize bit transition errors by allowing only one bit to transition at a time

when operating at a high speed (Figure 5.9 includes XOR gates connection approach). The XOR gate design using the CMOS logic comprises NMOS and PMOS transistors with a width of 120 nm and 240 nm for PMOS, with a constant length of 100 nm. This is a borrowed design from the work of Singh and Mehra (2014) that highlights the effectiveness of the CMOS XOR gates to ensure low-power consumption and propagation delay (Singh & Mehra, 2014).

The XOR circuit that is applied for binary to gray code conversion is illustrated in Figure 6.7, and the simulation outputs, validating the correct binary to gray code conversion, are depicted in Figure 6.8 (You & Jeon, 2019)

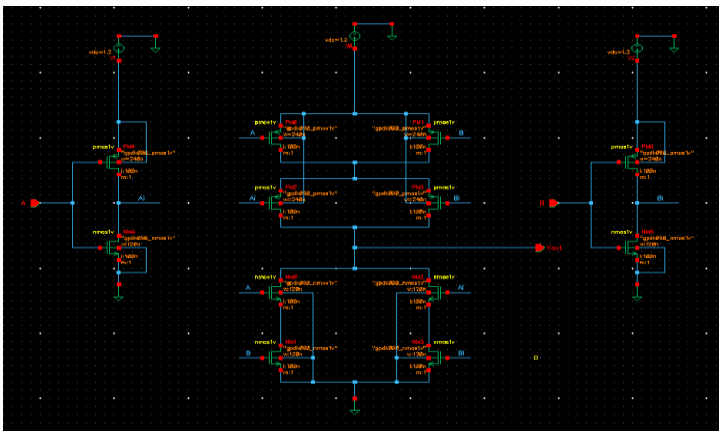


Figure 6.7 XOR Gate Circuit for Binary-to-Gray Code Conversion

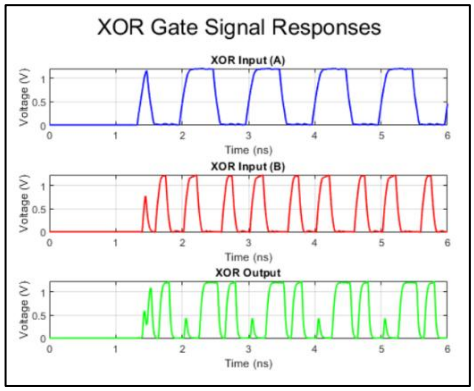


Figure 6.8 XOR Gate Signal Responses

Simulation results for the encoder circuit elements prove the efficacy of MUX-based design along with binary-to-gray coding. The application of buffers ensures extra stability for the signal, rendering the encoder highly compatible for GHz-speed implementation. This method reduces bit transition error and minimizes the consumption of power, aligning with design requirements for time-to-digital converters at high speeds. And you can see the full encoder circuit in Figure 6.9 and Figure 6.10.

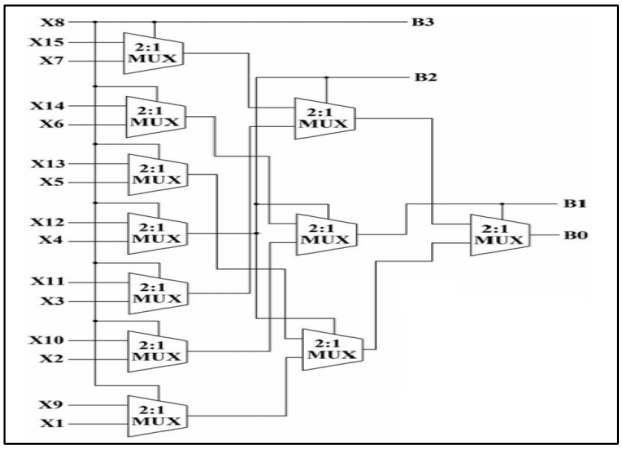


Figure 6.9 Original 15-4 MUX Based Encoder (Singh & Mehra, 2014)

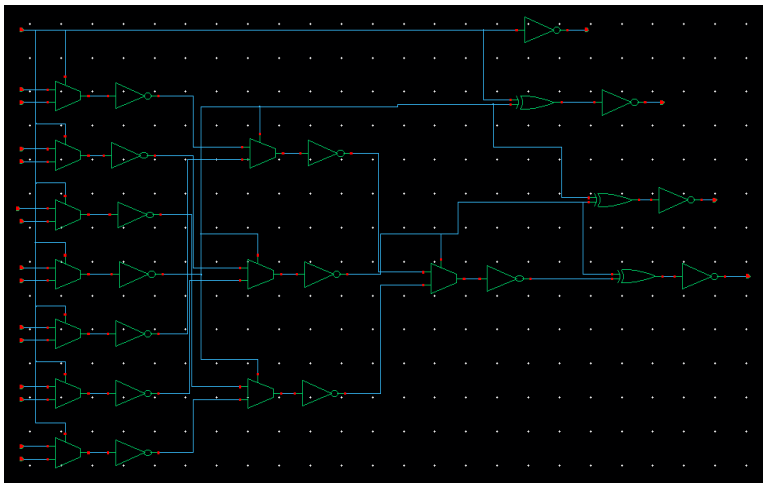


Figure 6.10 15-4 MUX Based Encoder Cadence Virtuoso

6.2. ENCODER PERFORMANCE ANALYSIS

It is the operation of the 4-bit encoder within the Vernier Delay Line Time-to-Digital Converter that is paramount to both the stability and accuracy of the system overall. Presented here is the differential non-linearity analysis for said encoder, fabricated with 100 nm CMOS for the purpose of sustaining a high frequency of operation at 1 GHz. Key to its specification for design is a resolution of 4 bits with a dynamic range of 1.2 V. Throughout the course of the analysis, the peak DNL recorded was 3.32 LSB, a reading not optimal but acceptable for the purposes of a high-speed application.

Differential Non-Linearity (DNL) quantifies the difference between the actual step size of the digital output and the step size that is ideal. The ideal transition of a 4-bit encoder should be constant between two consecutive digital codes. The DNL is computed as the difference between the actual step size and the ideal step size, divided by the ideal step size itself (Ben Mansour et al., 2017). This is given by:

$$\text{DNL}=\frac{V_{actual}-V_{ideal}}{V_{ideal}} \qquad (6.1)$$

where V_{actual} is the noticed step size and V_{ideal} the expected step size. The dynamic range divided by the total number of possible output levels yields the appropriate step size (LSB) for a 4-bit encoder. For an N-bit system, this is as follows:

$$\text{Ideal LSB}=\frac{\text{Dynamic Range}}{2^N-1}=\frac{1.2V}{15}\approx 0.08V \qquad (6.2)$$

This translates into the optimum voltage difference for subsequent digital codes being about 0.08 V. However, the maximum recorded DNL for the current encoder was 3.32 LSB, corresponding to a difference of about 0.27 V. The higher-than-optimal DNL may be attributed to a variety of reasons that include slight mismatches in the sizes of the transistors (120 nm NMOS, 240 nm PMOS, and 100 nm length) along with phase noise and 1 GHz jitter, which may create

uncertainty in the signal transition time. Parasitic capacitance within the buffer and MUX stages may create slight voltage drifts, with mismatched buffer loads creating slight delays, which add to the recorded DNL.

In spite of the higher DNL compared to the theoretically optimal, it is acceptable for that application. The encoder is intended for GHz-range TDC applications where some small-scale nonlinearities are acceptable provided that the overall stability and consistent periodic encoding are ensured by the system. The fact that the device employs both MUX-based encoding and binary-to-gray code, supplemented with buffers, ensures consistent encoding even at high speeds.

Figure 6.11 demonstrates the Encoder Inputs (Q0 to Q14) and Encoder Outputs (G0 to G3), clearly showing the periodic nature of the encoded signals and uniform step transitions. The corresponding DNL Analysis of the 4-Bit Encoder is presented in Figure 6.12, which highlights step size variability while confirming the overall stability of the encoding process.

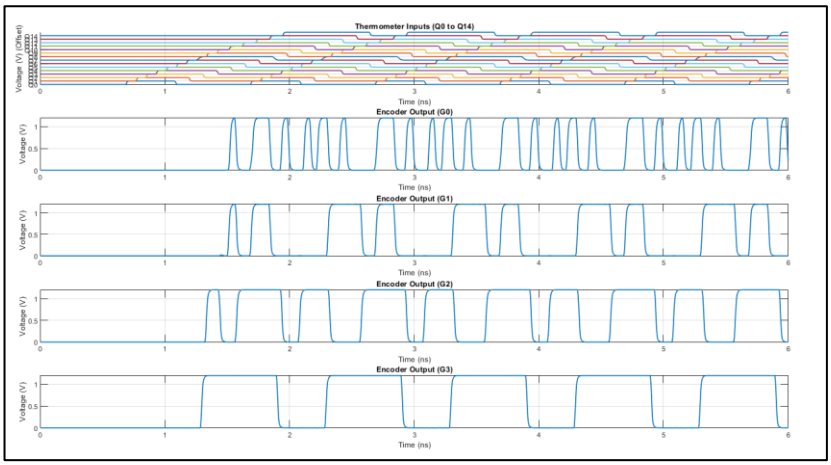


Figure 6.11 Thermometer Inputs (Q0 to Q14) and Encoder Outputs (G0 to G3)

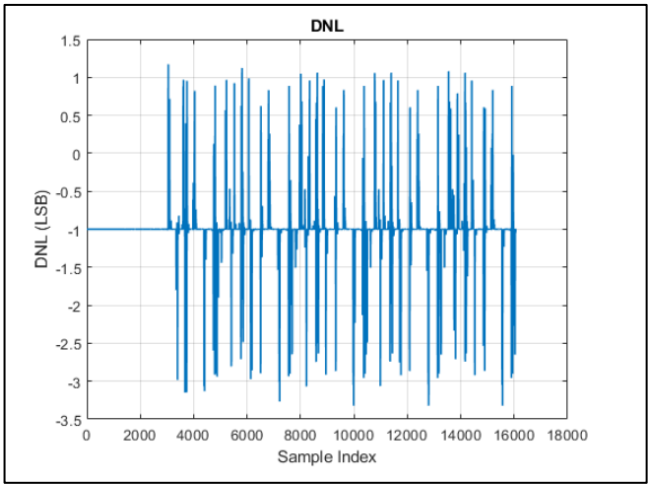


Figure 6.12 DNL Analysis of 4-Bit Encoder

In short, the 3.32 LSB DNL, although not ideal, is acceptable within the context of high-speed digital conversions. Transistor mismatch, timing jitter, and GHz-frequency-specific parasitic effects are the main contributing factors. Overall encoder performance, as tested using simulation, ensures that the encoder is functional and reliable for the meant application with a consistent encoding process even with the measured DNL deviations (Table 6.1).

6.3. DAC DESIGN

The 4-bit Digital-to-Analog Converter (DAC) implemented within the design is a resistor ladder network-based one, a common method of converting a digital signal to accurate analog outputs. The configuration is suitable for the Vernier Delay Line Time-to-Digital Converter (VDL-TDC) because it is simple, highly linear, and fast, properties which make it apt for reliable time-to-voltage conversion at GHz rates (Kerstetter, 2019).

Its inputs come from the encoder as outputs of Gray codes (B0 to B3). Inputs are processed by a series of stages using resistors, with each bit carrying a different weight, resulting in a proportionate analog voltage at the point of final output. In contrast to binary inputs, which are traditional, using Gray code is

helpful for the design because it reduces the occurrence of large transient voltage glitches at bit transition time, enhancing the stability of the overall signal.

6.3.1. Resistor Ladder DAC Structure

The building block for the DAC is a simple ladder stage, as shown by Figure 6.13. A stage is made up of a series-connected string of resistors that split the input voltage based on the weight of the input bit. The configuration comprises:

- IN Node: Accepts the gray code input (e.g., B0, B1, B2, B3) for the least significant bit (LSB) to most significant bit (MSB) of the 4-bit output.
- OUT Node: It is linked to the IN node of the subsequent stage, passing the partially processed analog signal along up the ladder.
- BOTTOM Node: Is connected to the OUT node of the previous stage or ground for the last stage, with the return signal path provided.
- Resistor Network: Contains precisely matched resistors (5 kΩ each) that ensure a consistent voltage division along with a high linearity.

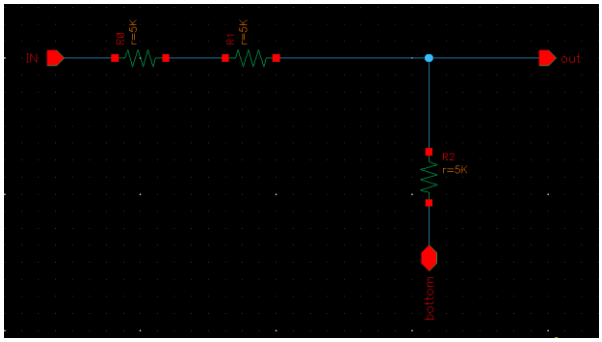


Figure 6.13 Single Resistor Ladder Stage for 4-Bit DAC

Such a ladder configuration is designed so that the contribution of each bit to the resulting output voltage is suitably weighted, with the most significant bit B3 creating a maximum voltage step and the least significant bit B0 creating a

minimum step. This binary-weighted ladder is essential for obtaining a proper digital-to-analog conversion.

6.3.2. Complete 4-Bit Ladder Assembly

A full 4-bit DAC is built using four of these ladder stages in series, as shown in Figure 6.14. The stages are configured so that:

- B0 (LSB): Provides the minimum voltage step.
- B3 (MSB): Supplies the biggest voltage step.

Each stage is an effective voltage divider that takes the weighted sum of each input bit and produces the final analog output. The B3 OUT node represents the overall DAC exit, V_{out}, with the aggregate analog voltage being a summing of the individual stages.

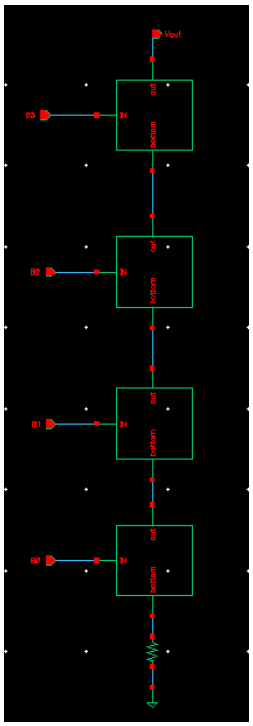


Figure 6.14 Intermediate 4-Bit DAC Assembly with Cascaded Ladder Stages

In order to keep the resistive loading of the DAC stages from interfering with the outputs of the Gray code encoder, a small one-stage buffer is inserted between each encoder stage and the DAC input. The buffer isolates the encoder from the ladder and minimizes loading effects while maintaining the integrity of the high-rate signals.

A fully assembled version of the four stages is presented in Figure 6.15. Here, each of the four stages is evidently wired using a cascaded arrangement, with the BOTTOM terminal on each stage joined with the OUT terminal of the previous stage for proper voltage division to take effect.

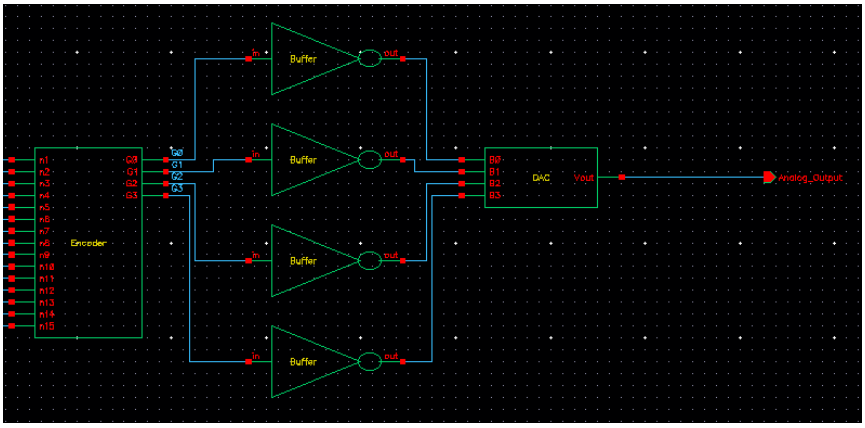


Figure 6.15 Complete 4-Bit DAC with Buffer Stages

This arrangement makes the analog output directly proportional to the input code, which is a linear response with step size dependent on the resistance ratios. The use of Gray code also reduces the risk of glitches with one bit changed at a time, thereby keeping the simultaneous transition effects on the output voltage to a minimum.

6.3.3. Design Considerations and Challenges

This ladder resistor DAC has several advantages, possibly including reduced glitching with Gray coding inputs, good linearity due to precisely trimmed resistors, and low power consumption due to its absence of an active

stage. But it has some issues of its own, such as vulnerability to resistor mismatch, which is likely to have a tangible impact on total linearity. Parasitic capacitor action within the ladder network could also result in signal distortion at higher frequency levels, potentially decreasing the effective bandwidth of the DAC.

Buffers added between the encoder and ladder add to the complexity of the design because the buffers need to be sized with some careful design to avoid extra delay and distortion. This design is still a strong solution for high-speed converting from digital to analog, but it is especially suitable for the GHz-level operation of the VDL-TDC.

6.4. 4-BIT DAC PERFORMANCE ANALYSIS

The performance of the 4-bit DAC was also tested against two parameters, Dynamic Range and Differential Nonlinearity (DNL), which are critical for measuring the precision and performance of the DAC in the TDC system, with the operation at GHz-level frequencies. The testing is carried out according to the simulation output, as seen from Figures 6.16, 6.17, and 6.18.

6.4.1. Dynamic Range and Differential Nonlinearity (DNL)

The Dynamic Range of the DAC is the value that indicates the difference in the maximum output voltage and the lowest output voltage. For this architecture, the measured value of the dynamic range is 1.10 V, which can be seen in Figure 6.16. This is slightly below the theoretical value of 1.20 V due to parasitic effects, delay of the buffer, and resistive mismatches in the resistor ladder. This is still an acceptable value for applications of GHz-speed TDC.

Differential Nonlinearity (DNL) is an important parameter to assess the step accuracy of the DAC. It quantifies the difference of the actual step sizes from the ideal step size, calculated with equation (6.2) for the ideal LSB and equation (6.1) for DNL, as described in the previous parts of this thesis. The maximum DNL value seen in the simulation is 1.00 LSB (Figure 6.17), which

represents the greatest worst-case deviation from the nominal step size of one least significant bit.

This performance is adequate for high-speed TDC applications, where high-accuracy time-to-digital conversion is needed with precise step sizes. The causes of the observed DNL are:

- Resistor mismatch: Step size variation may also take place due to physical resistor deviations in the ladder network.

- Parasitic Capacitance: Parasitic capacitances can occur due to layout, which causes timing delays and voltage overshoot.

- Buffer Loading Effects: Despite inserting small single-stage buffers at the inputs of the encoder and at the inputs of the DAC to mitigate the loading effects, because of small response differences of the buffers, DNL can still occur.

In spite of these challengess, the DAC performs satisfactorily in DNL, which makes it suitable for application in the targeted GHz-range TDC (Table 6.1).

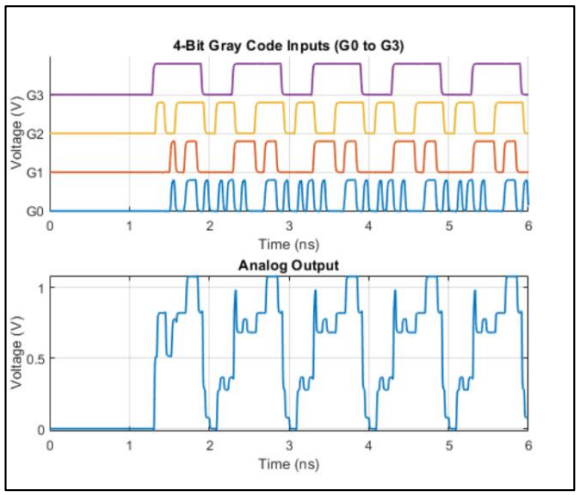


Figure 6.16 4-Bit Gray Code Inputs and Analog Output

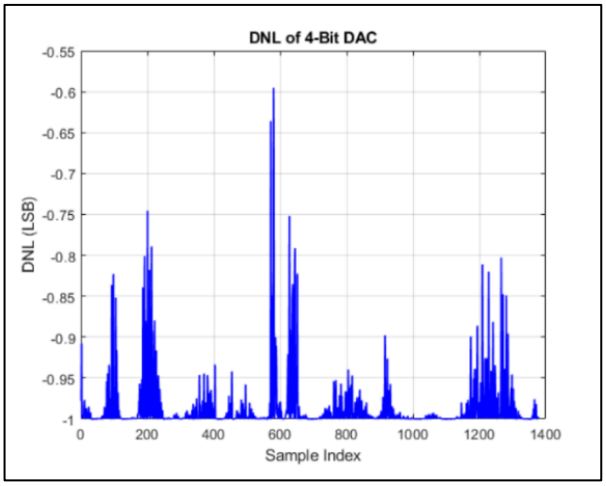


Figure 6.17 DNL of 4-Bit DAC

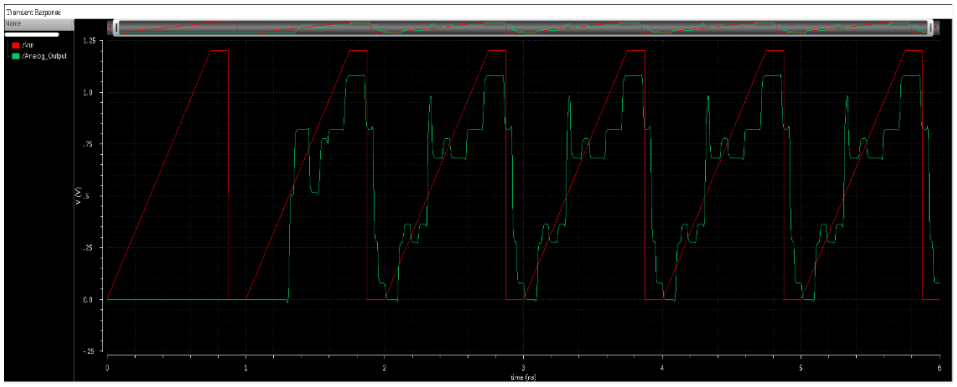


Figure 6.18 Analog Output vs. Ramp Input Response

Table 6.1 Summary of Encoder and DAC Performance Parameters

Parameter	Encoder (15-4 MUX-Based)	DAC (4-Bit Resistor Ladder)	Notes
Technology Node	100 nm		CMOS technology for high-speed operation
Resolution	4 Bits	4 Bits	Consistent across both blocks
Dynamic Range	1.20 V	1.10 V	Slight reduction in DAC due to parasitics
Maximum DNL	3.32 LSB	1.00 LSB	Within acceptable range for GHz applications
Signal Type	Gray Code	Analog	Post-encoded digital and analog stages
Operating Frequency	1 GHz	1 GHz	Based on the ramp signal frequency

CHAPTER 7

7. FULL SYSTEM ANALYSIS AND PERFORMANCE EVALUATION

The overall Time-to-Digital Converter (TDC) design is entirely analyzed, which encompasses Voltage-to-Time Converter (VTC), Vernier Delay Line (VDL), MUX-based encoder, and the final Digital-to-Analog Converter (DAC). The leading performance parameters, i.e., conversion time, resolution, maximum possible input frequency, power consumed, and differential nonlinearity (DNL), are addressed throughout this chapter. Simulation and measurement data are provided for confirming the performance of the total TDC system. Table 7.1 includes summary of the key performance metrics obtained from the complete TDC system analysis.

7.1. CONVERSION TIME AND RESOLUTION

The conversion time is an important parameter for TDC systems, which is the time taken to propagate the input signal from the VTC to the output of the encoder. It is considered to be the time interval from the rising edge of the start signal produced by the VTC to the MSB (G3) output of the encoder. According to the given simulation, the conversion time was observed to be about 964.04 ps (Figure 7.1).

The resolution of the TDC can often be expressed as the smallest time difference the system can measure. This can be calculated from the formula:

$$\text{Resolution}=\frac{t_{range}}{2^n} \qquad (7.1)$$

where n is the output bit number, and t_{range} is the range of the full-scale time. With the measured conversion time and the 4-bit output of the encoder, the resolution is about 60.25 ps, validating the high precision with which the system

is able to resolve fine time intervals. The resolution is adequate for use with GHz-level applications, and the TDC is hence qualified for high-speed timing measurements.

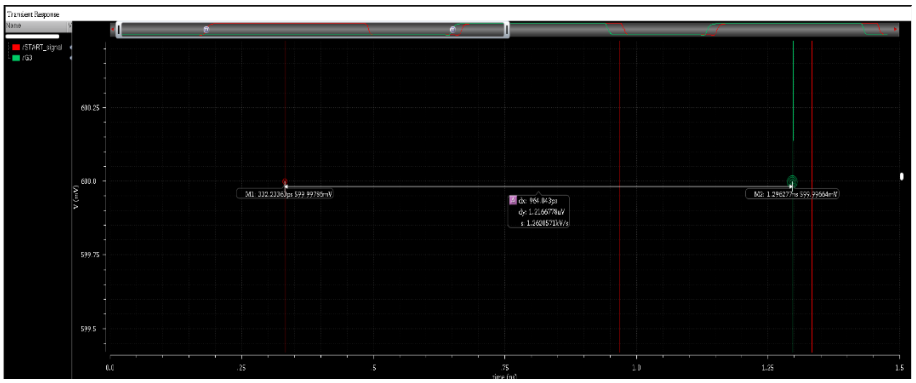


Figure 7.1 Conversion Time Measurement (Start Signal to MSB)

7.2. MAXIMUM INPUT FREQUENCY

The highest input frequency the TDC can operate is established by how quickly it can complete an entire conversion cycle in a single input period. Based on the measured conversion time, an upper input frequency can be approximated as:

$$f_{max} = \frac{1}{\text{Conversion Time}} \qquad (7.2)$$

Replacing the measured conversion time of 964.04 ps, the highest input frequency is about 1.04 GHz, which agrees quite well with the target operational goal of this design, which is 1 GHz.

7.3. POWER CONSUMPTION

The total power consumption of the whole TDC system was further tested to determine its efficiency. The static power value obtained was 137.0 µW, while the average power value obtained was 1.059 mW, as seen in Figures 7.2 (a) and

(b). These values can be applied to approximate the value of the dynamic power from:

$$P_{dynamic}=P_{avg}-P_{static} \qquad (7.3)$$

generating an approximate value of around 922 µW, which reflects the delay elements, encoders, and the power usage of the DAC. This power pattern reflects the power and performance balancing of the design.

(a) (b)

Figure 7.2 (a) Average Power Measurement (Total System) (b) Static Power Measurement (Total System)

7.4. DIFFERENTIAL NONLINEARITY (DNL) AND DYNAMIC RANGE

The differential nonlinearity, or DNL, of the complete TDC system was also evaluated. DNL measures the difference between actual step sizes and the perfectly uniform LSB value. The DNL, as determined from the output data taken during sampling, resulted in an absolute maximum DNL of 2.50 LSB, as illustrated in Figure 7.3.

The TDC's dynamic range was also assessed as part of the total performance evaluation. The measured dynamic range is 1.10 V, which is the complete range of voltage values that the TDC can resolve with accuracy. This is an important parameter for ensuring high signal integrity over the complete input range.

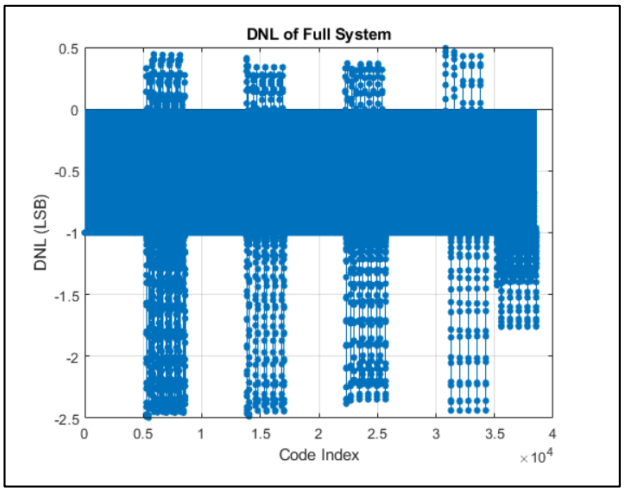


Figure 7.3 DNL of Full System

Table 7.1 Summary of The Key Performance Metrics Obtained from The Complete TDC System Analysis

Parameter	Value	Notes
Conversion Time	964.04 ps	Measured from START signal to MSB transition
Resolution	60.25 ps	Based on 4-bit quantization
Maximum Input Frequency	1.04 GHz	Calculated from conversion time
Average Power Consumption	1.059 mW	Includes both dynamic and static power
Static Power Consumption	137.0 µW	Measured with no active switching
Dynamic Power Consumption	922.0 µW	Total average power - static power
Dynamic Range	1.10 V	Limited by DAC performance
Maximum DNL	2.50 LSB	Acceptable for high-speed TDC

CHAPTER 8

8. ALTERNATIVE VERNIER DELAY LINE TDC SYSTEM DESIGN

In this chapter, an alternate Vernier Delay Line (VDL) Time-to-Digital Converter (TDC) system is presented to handle a different stop signal generation configuration. Contrary to the earlier described form, which made use of a double VTC approach, this alternate form makes use of a single VTC for the start and a Vpulse generator for the stop. The Vpulse signal is a 500 ps period, 250 ps pulse with voltages V1 = 1.2 V and V2 = 0 V. This provides a distinct timing relationship, two rising edges for every start signal period: a first rising edge before the start pulse and a second rising edge within the pulse.

To support this different signal arrangement, edge-triggered True Single-Phase Clock (TSPC) D Flip-Flops are implemented in the VDL, with VDL previously using 5-transistor TSPC D Flip-Flops. This change supports reliable edge detection without a reliance on pulse overlap, which is more suited for the new timing architecture. The identical buffer structure from the previous VDL is reused for maintaining stable signal propagation throughout the delay line.

In addition, the encoder circuitry is also modified so that it employs an alternate MUX-based scheme, producing a direct binary code output, unlike previously using Gray code. This is done with a view toward supporting the revised timing performance made available by the novel pulse structure. The DAC stage is kept unchanged with the previous design for ease of integration and for maintaining a consistent system overall.

The chapter will cover the design and implementation of the novel flip-flop and MUX-based encoder, followed by performance characterization of the new VDL, encoder, and entire TDC system. This involves measurement of updated DNL, dynamic range, and overall timing.

8.1. VERNIER DELAY LINE DESIGN WITH TSPC D FLIP-FLOPS

In this section, presentation of the alternative Vernier Delay Line (VDL) TDC system's design and analysis is provided. In contrast to the former approach, which used a dual VTC (Voltage-to-Time Converter) structure, there is the use of a single VTC for the start signal and a Vpulse generator for the stop signal. The Vpulse generator is designed with a period of 500 ps and a 250 ps wide pulse, switching between voltage values of 1.2V and 0V. Two rising edges are introduced within a single start signal period with this approach: one before and within the pulse, respectively. Consequently, use of edge-triggered TSPC D Flip-Flops is needed in the VDL, which is capable of precisely recording multiple rising edges.

8.1.1. Start and Stop Signal Generation

The start signal is created by a single VTC, and for a stop signal, a Vpulse generator is utilized. The Vpulse is programmed with a 500 ps period and 250 ps of a pulse, which produces two distinct rising edges for every period of a start signal. Such rising edges are essential for precise timing capture inside the delay line since they initiate the TSPC D Flip-Flops. Such a timing mechanism is essential for enabling precise edge detection, an important criterion for high-resolution TDC performance.

The generated start and stop signals are shown in Figure 8.1, which accentuates two rising edges that are of utmost importance for this alternative delay line arrangement.

Figure 8.1 START and STOP Signals

8.1.2. TSPC D Flip-Flop Architecture and Operation

The TSPC D Flip-Flop implemented within our design is derived from a modified topology resembling current high-speed flip-flop circuitry (Bhattu, 2025). This flip-flop is optimized for effective capturing of timing data related to rising edges of the start signal. Figure 8.2 gives the flip-flop's schematics. In it, PMOS and NMOS transistors of a width of W=120nm and length of L=100nm are utilized, which are optimized for speed, power, and stability.

The TSPC D Flip-Flop is particularly suited for application within this design based on its edge-triggering capability, which allows it to accurately deal with the high-speed edges of signals created through the Vpulse generator. This is needed for achieving high resolution with accurate timing for GHz-level TDCs.

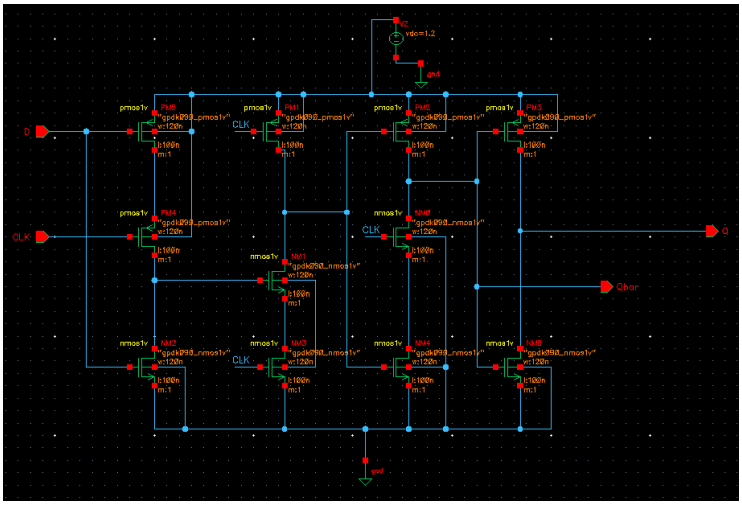


Figure 8.2 Schematic of TSPC D Flip-Flop for Alternative VDL Design

8.1.3. Edge Detection and Timing Analysis

Edge detection within VDL is an important consideration for ensuring accuracy of time-to-digital conversion. Precise detection of such edges is needed by virtue of the start signal's dual rising edge behavior. The simulation output,

which illustrates the timing accuracy using TSPC D Flip-Flop, is given by Figure 8.3.

Figure 8.3 depicts the delay line input and output, the D inputs, CLK inputs, and Q outputs, with each signal displaced for clarity. The progression of timing through the delay line is shown in Figure 8.4, again demonstrating a linear delay progression through every delay line stage.

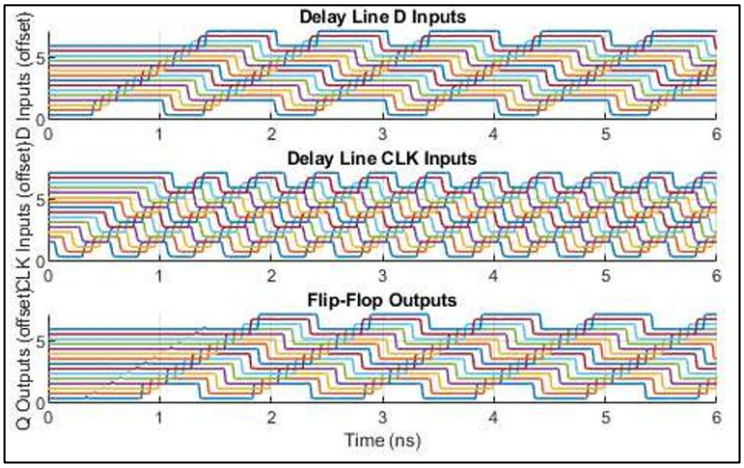


Figure 8.3 Delay Line D Inputs, CLK Inputs, and Q Outputs

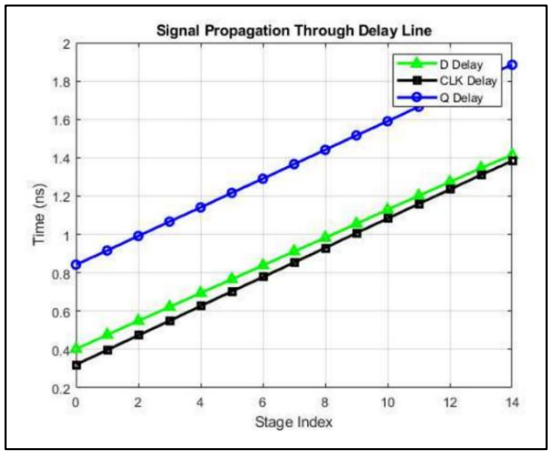


Figure 8.4 Signal Propagation Through Delay Line

8.1.4. VDL Results and Analysis

The delay line's performance metrics were assessed using simulation. Full code coverage is confirmed by the fact that the system is able to detect 16 different codes. The delay line resolution is calculated as 74.45 ps with a dynamic range of 1.04 ns. The minimum and maximum pulse widths are 516.72 ps and 534.03 ps, with an average of 524.99 ps.

In terms of non-linearity, Differential Non-Linearity (DNL) and Integral Non-Linearity (INL) values were 6.18 ps for each. These are shown in Figure 8.5, with the uniformity of DNL over the code range being clearly visible. The histogram shown above in Figure 8.6 validates that all 16 output codes are distributed equally, proving even quantization and with minimal non-linearity.

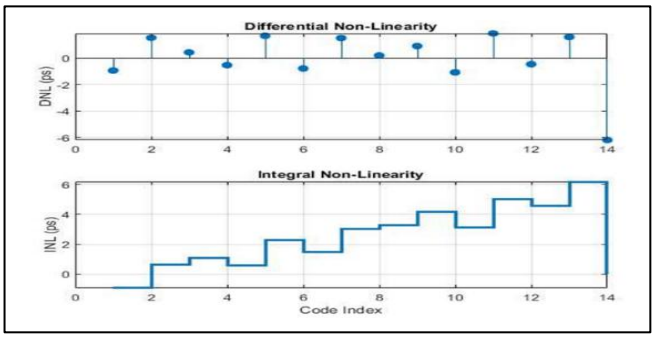


Figure 8.5 Differential Non-Linearity and Integral Non-Linearity

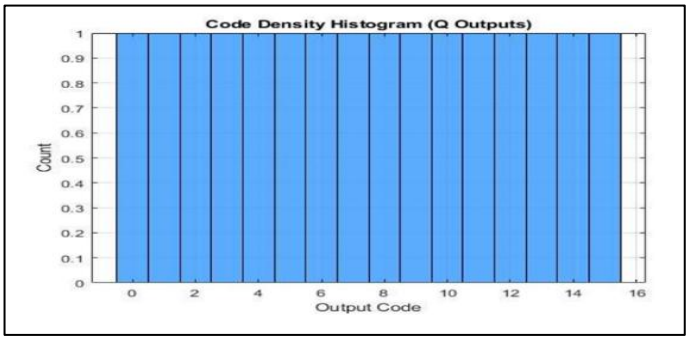


Figure 8.6 Code Density Histogram (Q Outputs)

Table 8.1 Performance Analysis of Vernier Delay Line Design with TSPC D Flip-Flops

Parameter	Value	Notes
Start Signal Frequency	1 GHz	Generated using a single VTC
Stop Signal Frequency	2 GHz	Generated using Vpulse with 500 ps period and 250 ps pulse width
Detected Codes	16 out of 16	Full code coverage
Minimum Pulse Width	516.72 ps	Measured at the Q outputs
Maximum Pulse Width	534.03 ps	Measured at the Q outputs
Average Pulse Width	524.99 ps	Measured at the Q outputs
Resolution (Mean Step)	74.45 ps	Calculated from Q outputs
Dynamic Range	1.04 ns	Calculated from the delay line timing
Maximum DNL	6.18 ps	Differential non-linearity
Maximum INL	6.18 ps	Integral non-linearity
Delay Line Stages	15	Based on the number of flip-flop stages
Transistor Dimensions	W=120 nm, L=100 nm	Used in the TSPC D Flip-Flop design

8.2. MUX-BASED BINARY ENCODER DESIGN

In this section, an explanation of the MUX-based binary encoder for the alternate VDL TDC architecture is given. The encoder is fabricated using a CMOS process with PMOS transistors with a channel length (L) of 100 nm and a channel width (W) of 240 nm, and NMOS transistors with a channel length (L) of 100 nm and a channel width (W) of 120 nm. This arrangement supports faster switching rates and reduced parasitic capacitance, improving overall encoder timing performance.

8.2.1. Binary Output Encoding Strategy

The MUX-based encoder architecture is derived from Figure 8.7, which uses a structured mix of multiplexers and inverters to convert thermometer-coded input into a minimal redundancy gray code output. The architecture is tree-based, with each tree of MUX decreasing input code length until a reduced output of a 4-bit (B0 to B3) is produced from 15 thermometer-coded inputs (Q0 to Q14) (V. Sivakumar Reddy et al., 2020).

The encoder circuit, as indicated by Figure 8.8, uses single-buffering stages for minimizing the effects of MUX output loading. The buffers stabilize signals after processing by the XOR gates, which convert into final bits. The XOR gates are converting gray code to binary based on (VLSI Verify, n.d.), but they are implemented with CMOS logic with the identical transistor size for the MUXes so that there is consistency with the signal path.

The strategy indeed minimizes the total delay in the encoder without compromising on high-speed operation, a necessary requirement for accurate TDC performance at GHz-level frequencies.

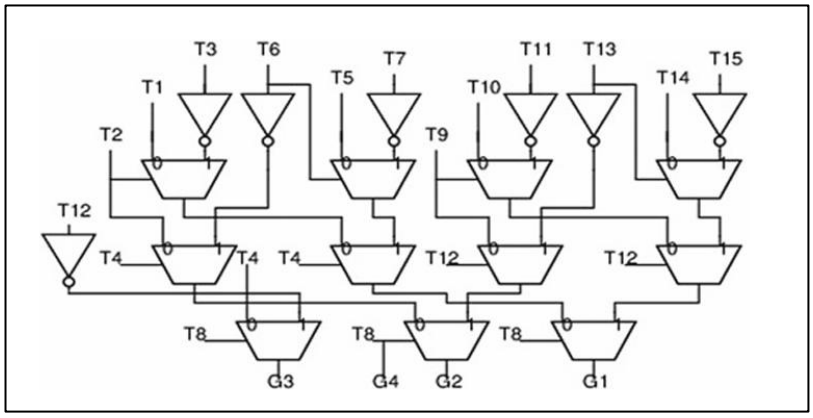


Figure 8.7 Tree MUX-Based Encoder Structure (V. Sivakumar Reddy et al., 2020)

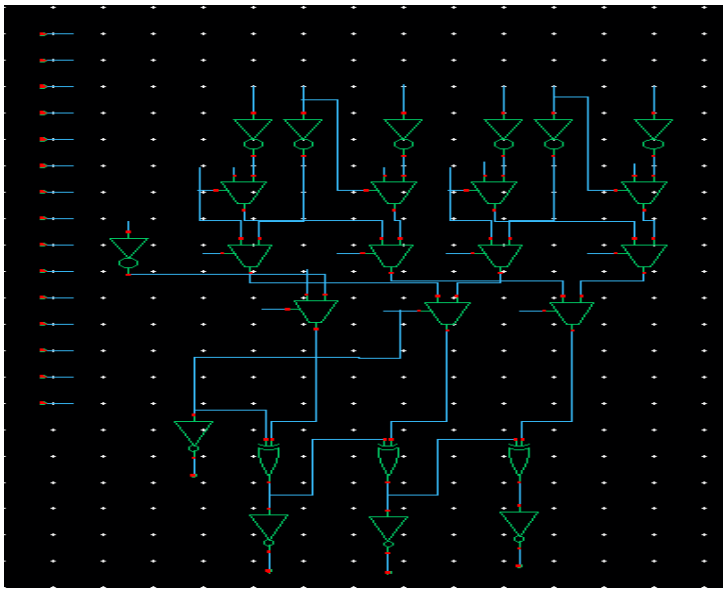


Figure 8.8 Schematic of Tree MUX-Based Encoder in Cadence Virtuoso

8.2.2. Encoder Performance Analysis

Performance analysis of the developed encoder DNL is shown in Figure 8.9. The measurement corresponds with the ability of the encoder to manage high-speed input signals from the VDL, with a dynamic range of 1.2V and a peak differential non-linearity (DNL) of 3.14 LSB. While this is a high non-linearity, it is within acceptable limits for use in the targeted application, since the minimal buffer delays, anticipated jitter within the MUX blocks, and single-buffer impact are all dealt with satisfactorily. Timing responses, presented in Figure 8.10, emphasize quick signal transitioning achieved through optimized CMOS architecture. In addition, incorporating single-stage buffers into this architecture serves to enhances binary output stability. This strategy provides an even trade-off of speed against accuracy so that the encoder is compatible with high-frequency TDC operations.

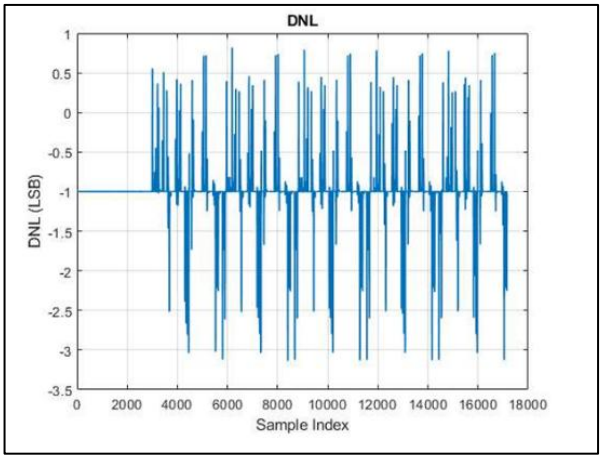


Figure 8.9 Tree MUX-Based Binary Encoder DNL

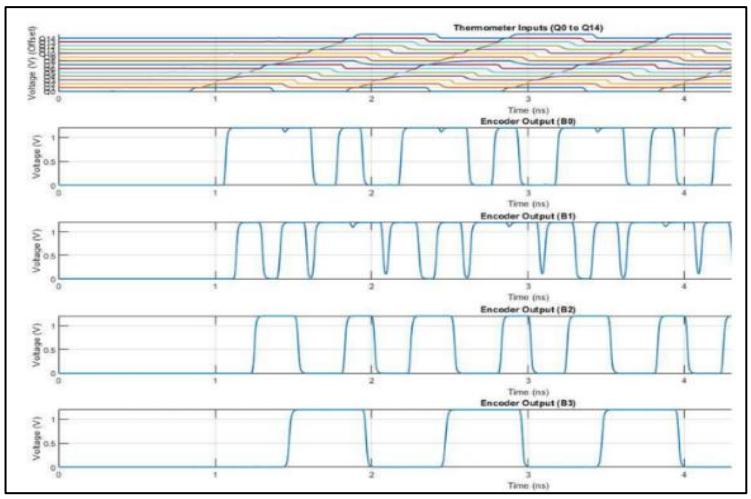


Figure 8.10 Encoder Outputs (B0 to B3)

8.3. DAC PERFORMANCE ANALYSIS AND SIGNAL RECONSTRUCTION

The performance of the DAC within the alternative VDL TDC system is explained herein. The design of DAC itself has already been given within Chapter 6, wherein architecture, working principles, and design aspects were covered comprehensively. Hence, within this section, performance analysis and

signal integrity of the DAC within the present alternative VDL TDC are explained. The DAC input is taken from 4-bit binary output of the MUX-based encoder, which is explained within Section 8.2.

The main objective of this section is assessing the non-linearity, dynamic range, and signal fidelity of the DAC. The analysis encompasses the input and output waveforms taken during the recording, emphasizing DAC's expected vs. actual performance. This is especially important because DAC is the last part of the TDC process, with any deviation directly impacting the quality and accuracy of the overall process.

The 4-bit binary outputs (B0-B3) were fed into the DAC, which generated the analog signal of Figure 8.11. Figure 8.11 depicts both the digital code input and accompanying DAC output, which is a visible translation of digital into analog space. The output registered a dynamic range of 1.09V, almost matching ideal VDD of 1.2V, which is a measure of using available voltage swing.

In addition, measurement of DAC's DNL, which is plotted as indicated by Figure 8.12, verified that peak DNL of DAC is 1 LSB, which is an acceptable value for use for such an application. This is consistent with overall design objectives for precise and reliable digital-to-analog conversion. The comparison of the overall DAC output with the input ramp signal is shown in Figure 8.13, with direct graphical assurance of the DAC performance towards simulating the targeted analog signal. Such a comparison is an assurance of minimal signal distortion and low noise operation of the DAC, which supports the adopted design for the integrated VDL TDC system.

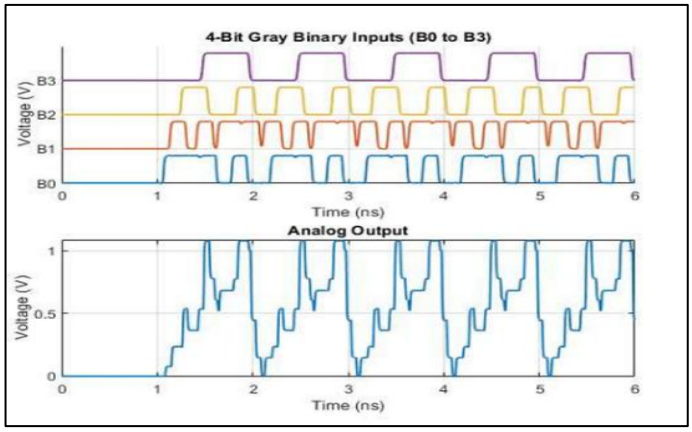


Figure 8.11 4-Bit Binary Inputs (B0 to B3) and Analog Output

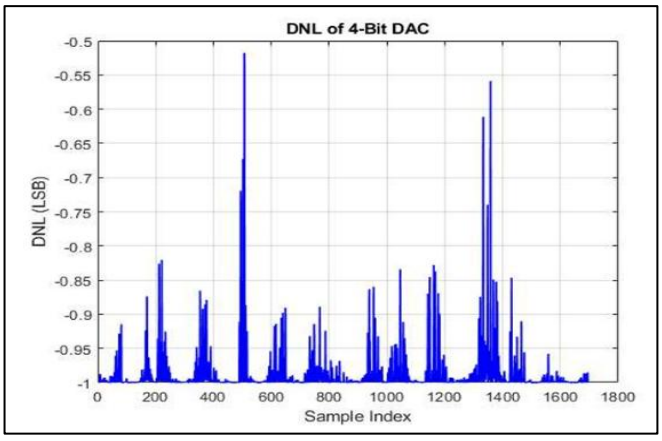


Figure 8.12 DNL of 4-Bit DAC

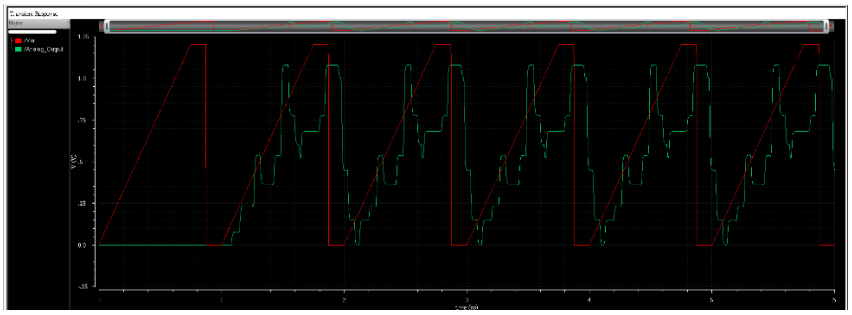


Figure 8.13 Ramp Input vs DAC Analog Output

Table 8.2 Summary Table for Encoder and DAC Results

Parameter	Encoder (15-4 MUX-Based)	DAC (4-Bit Resistor Ladder)	Notes
Technology Node	100 nm		CMOS technology for high-speed operation
Resolution	4 Bits	4 Bits	Consistent across both blocks
Dynamic Range	1.20 V	1.09 V	DAC dynamic range is lower due to voltage drop
Maximum DNL	3.14 LSB	1.00 LSB	Within acceptable range for GHz applications
Signal Type	Binary Code	Analog	Outputs

8.4. FULL SYSTEM PERFORMANCE ANALYSIS

The overall performance of the system is an important aspect for deciding on the functionality and reliability of the suggested TDC architecture. Total system's conversion time, resolution, dynamic range, power consumption, and differential non-linearity (DNL) are considered within this section. The figures offered are utilized for arriving at conclusions here without replicating previously established formulas of Chapter 7 since all equations are already formulated.

8.4.1. Timing and Resolution Analysis

The alternative VDL TDC system's conversion time, taken as the period elapsing from the first rising edge of the START signal until arrival of the MSB of the encoder output, was 506.518 ps (Figure 8.14). Assuming this conversion time, the system's maximum achievable operation frequency is calculated to be about 1.97 GHz. This is a measure of the fast speed with which the system is

able to process input signals with accuracy without any timing errors, and is appropriate for high-speed time measurement tasks.

The system resolution, or minimum representable time resolution, is by definition given by the conversion time over the number of 4-bit levels and, for our case, is 31.657 ps. This resolution is a representation of the ability of the system to distinguish tiny differences of time within input signals, making it especially desirable for precise measurement of time.

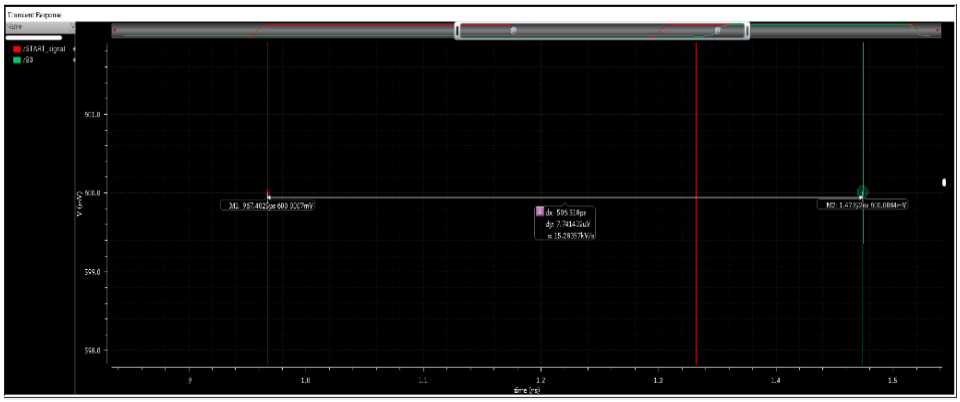
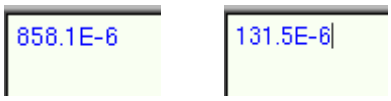


Figure 8.14 Full System Conversion Time Measurement

8.4.2. Power Consumption Analysis

The full system's power consumption analysis was implemented by determining the average and static power consumption. The overall average power of the system amounted to 858.1 µW, while the static power consumption equaled 131.5 µW (Figure 8.15 (a) and (b)), leaving a dynamic power consumption of 726.6 µW. The dynamic power represents the switching power inherently needed for fast signal conversion and is an important parameter for assessing the efficiency of the TDC system.



(a) (b)

Figure 8.15 (a) Average Power Measurement (Total Alternative System) (b) Static Power Measurement (Total Alternative System)

8.4.3. Dynamic Range and Nonlinearity Analysis

The total dynamic range of the overall system was verified at 1.09 V, which is consistent with the output range of the measured DAC. The peak of DNL, derived from the last system DNL plot (Figure 8.16), was 3.00 LSB. This is a reflection of the nonlinearity caused by the combined influence of the VTC, delay line, encoder, and DAC. Although slightly worse than ideal, an acceptable value for this range for the system's target applications for high speed and low latency.

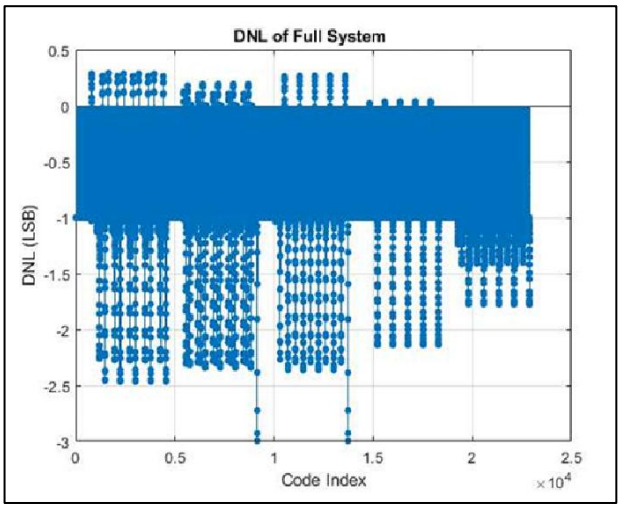


Figure 8.16 DNL of Full System

In general, this performance assessment verifies that the novel VDL TDC architecture supports high-speed operation with accurate timing resolution,

acceptable power efficiency, and reasonable signal integrity, which is well suited for demanding timing measurement tasks.

Table 8.3 Summary Table for Full System Performance

Parameter	Value	Notes
Conversion Time	506.518 ps	Measured from START signal to MSB transition
Maximum Operating Frequency	1.97 GHz	Calculated from conversion time
Resolution	31.657 ps	Based on 4-bit quantization
Dynamic Range	1.09 V	Limited by DAC performance
Max DNL	3.00 LSB	Acceptable for high-speed TDC application
Total Average Power	858.1 µW	Includes both dynamic and static power
Static Power	131.5 µW	Measured with no active switching
Dynamic Power	726.6 µW	Total average power - static power

CONCLUSION AND SUGGESTIONS

This thesis presents two new Time-to-Digital Converter (TDC) architectures targeted to high-speed, low-power and high-resolution operations optimized to cater to varying requirements. The first architecture employs a Vernier Delay Line (VDL) and MUX-based encoder using a dual-ramp Voltage-to-Time Converter (VTC), and the second presents a new and simplified alternative VDL system employing a single VTC and a Vpulse generator to achieve higher timing precision. Both methods address the challenges of achieving GHz-level time resolution using low-power devices in small CMOS design and contribute to useful advancements in the fields of digital signal processing and medical imaging.

The core TDC design incorporates double-ramp VTC architecture with two separate ramp signals (0–1.2 V for START path and 0–0.5 V for STOP path) providing better dynamic range and higher timing resolution. This mechanism de-couples start and stop signals practically to enhance the accuracy and adjustability of measurements to different signal levels. The VTC circuits employ passive CMOS Track-and-Hold (T/H) stages with high-speed sampling having an average resolution of close to 4.14 ps. Cascode current mirror biasing utilized by the presented design stabilizes the current sources under GHz operation to achieve precise voltage-to-time conversion without complicated calibrations.

The main design incorporates a 15-stage Vernier Delay Line, including a combination of three-stage buffer chains for START and STOP signals and 5-transistor TSPC D flip-flops for effective detection of overlapped pulses. This architecture provides fast propagation with less jitter, with average buffer propagation delay of 64.15 ps and flip-flop clock-to-Q delay of 19.37 ps. The MUX-based encoder, backed by single-stage buffers and CMOS XOR gates, supplies strong binary-to-Gray code translation with reduced transitioning errors at high speeds. The entire system, including the resistor ladder DAC, delivers a

dynamic range of 1.10 V and a resolution of 60.25 ps, validating it for GHz-speed applications.

Conversely, the alternative TDC architecture offered by Chapter 8 is a reduced-complexity approach that also preserves high timing accuracy. Its second architecture substitutes a single VTC for the dual-ramp VTC structure for the START signal and a Vpulse for the STOP signal. This modification allows for precise edge detection with edge-triggered TSPC D flip-flops, obviating the need for steady ramp signals. Having a Vpulse generator imposes unique timing relationships that accommodate, for instance, being able to detect two rising edges within a single period of START. Its resolution is veritably increased, with a 506.518 ps conversion time and a 1.97 GHz maximal working frequency, with a 31.657 ps resolution. Though having a reduced architecture, it is still competitive with a dynamic range of 1.09 V and less power consumption (858.1 μW average power), making it a reasonable option for high-frequency, low-latency TDC applications.

Limitations and Optimization Possibilities: While both methods yield high-speed performance, each has its own drawbacks. The execution of double-ramp VTCs on the main design incurs additional design complexity and power overhead, but the other technique's simpler single VTC methodology incurs more non-linearity (DNL up to 3.00 LSB) owing to decreased timing control flexibility. Both methods' test errors in dynamic range owing to parasitic capacitances and non-matched component values reveal scope for further improvement. Improved matching, advanced calibration circuits, or even machine learning-based error correction techniques would have the promise to yield higher linearity and resolution, especially on buffer and DAC blocks.

Future Directions for Research: Adaptive thresholding, digital calibration, and multi-ramp architecture are all potential future areas of research for further increasing timing accuracy and dynamic range. Also, further integration with future technologies such as silicon photonics or quantum computing could push their use into ultra-high-speed communications and future generations of medical imagers. More sophisticated transistor geometries, such

as FinFETs or nanosheet FETs, could also provide substantial reductions in power consumption and performance, bringing these designs into alignment with future GHz-scale digital logic.

In conclusion, this thesis showcases the flexibility and potential of both TDC architectures, which are optimized for certain performance vs. complexity trade-offs. These two designs together present a solid platform for future advancements for high-speed digital timing and signal conversion, adding valuable knowledge to mixed-signal circuit design.

REFERENCES

Ben Mansour, I., Maghrebi, R., Sifi, N., & Touayar, O. (2017). Design and implementation of a platform for experimental testing and validation of analog-to-digital converters: Static and dynamic parameters. *International Journal of Metrology and Quality Engineering, 8*, 12. https://doi.org/10.1051/ijmqe/2017012

Bhattu, H. N. (2025). *Design of TSPC D Flip-Flop for High-Speed VDL Applications* [YouTube video]. YouTube. https://www.youtube.com/watch?v=xiqgDBEu5FA

Chen, Q., Boon, C. C., & Liang, Y. (2022). A 0.6 V 4 GS/s −56.4 dB THD voltage-to-time converter in 28 nm CMOS. *IEEE Access, 10*, 88558–88566. https://doi.org/10.1109/ACCESS.2022.3200678

Chunn, A., & Sarin, R. K. (2013). Comparison of thermometer-to-binary encoders for flash ADCs. In *Proceedings of IEEE INDICON 2013* 1–6. https://doi.org/10.1109/INDCON.2013.6726089

Deb, A., Sharma, S., & Dev, A. (2019). Analysis of various TSPC based D flip flops. *International Journal of Recent Technology and Engineering (IJRTE), 8*(1), 1716–1718. https://www.ijrte.org/download/volume-8-issue-1/

Dua, T., & Rajput, A. (2020). 2:1 multiplexer using different design styles: Comparative analysis. *Journal of Advancements in Robotics, 7*(3), 5–13.

Elgreatly, A., Dessouki, A., Mostafa, H., Abdalla, R., & El-Rabaie, E. (2020). A novel highly linear voltage-to-time converter (VTC) circuit for time-based analog-to-digital converters (ADC) using body biasing. *Electronics, 9*(12), 2033. https://doi.org/10.3390/electronics9122033

Granja, R. F. F. (2018). *11.7b time-to-digital converter with 0.82ps resolution in 130nm CMOS technology* Unpublished Master's thesis, Universidade de Lisboa Instituto Superior Técnico. https://fenix.tecnico.ulisboa.pt

Greenwald, E., Maier, C., Wang, Q., Beaulieu, R., Etienne-Cummings, R., Cauwenberghs, G., & Thakor, N. (2017). A CMOS current steering neurostimulation array with integrated DAC calibration and charge balancing. *IEEE Transactions on Biomedical Circuits and Systems, 11*(2), 324–335. https://doi.org/10.1109/TBCAS.2016.2609854

Gupta, Y., & Saini, S. (2014). Thermometer to Gray encoders. In M. Fakhfakh, E. Tlelo-Cuautle, & M. H. Fino (Eds.), *Performance optimization techniques in analog, mixed-signal, and radio-frequency circuit design* 323–335. IGI Global. https://doi.org/10.4018/978-1-4666-6627-6.ch013

Henzler, S., Koeppe, S., Kamp, W., Mulatz, H., & Schmitt-Landsiedel, D. (2008). 90nm 4.7ps-resolution 0.7-LSB single-shot precision and 19pJ-per-shot local passive interpolation time-to-digital converter with on-chip characterization. In *2008 IEEE International Solid-State Circuits Conference - Digest of Technical Papers* 548–635. IEEE. https://doi.org/10.1109/ISSCC.2008.4523300

Hussain, S., Kumar, R., & Trivedi, G. (2020). Methodology and comparative design of an efficient 4-bit encoder with bubble error corrector for 1-GSPS flash type ADC. *IET Circuits, Devices & Systems, 14*(5), 629–639.

Ismail, M. W. E. (2015). *Design of time-based analog to digital converter (TB-ADC): New design methodology for voltage-to-time-converter (VTC) circuits* Unpublished Master's thesis, Cairo University, Egypt.

Kerstetter, J. (2019). *Lab 2: Digital to Analog Converter (DAC) Design*. CMOSedu.

https://cmosedu.com/jbaker/courses/ee421L/f19/students/kerstett/lab_2/lab_2.htm

Latha, P., Sivakumar, R., & Pavithra, I. P. (2018). Implementation of MUX-based encoder for time-to-digital converters architecture. *International Journal of Engineering and Techniques, 4*(3), 520–526.

Liu, H., Xu, L., & Tian, T. (2020). A monolithic 12-bit digitally calibrated D/A converter. *IEICE Electronics Express, 17*(23), 1–6. https://doi.org/10.1587/elex.17.20200371

Liu, H., Xu, L., & Tian, T. (2021). A monolithic 12-bit digitally calibrated D/A converter. *IEICE Electronics Express, 18*(2), 20200371. https://doi.org/10.1587/elex.17.20200371

Marche, D., & Savaria, Y. (2010). Modeling R-2R segmented-ladder DACs. *IEEE Transactions on Circuits and Systems I: Regular Papers, 57*(1), 31–43. https://doi.org/10.1109/TCSI.2009.2019396

Sharma, S. (2018). *Low-power high-performance SAR ADC with redundancy and digital error-correction* Unpublished Doctoral thesis, Nanyang Technological University. https://hdl.handle.net/10356/89791

Shete, V. V., Gokhale, A. D., & Kale, S. S. (2019). Voltage-to-time converter architecture with a two-step delay technique. *Microelectronics Journal, 88*, 101–107. https://doi.org/10.1016/j.mejo.2019.04.003

Siddiqui, W. H. (2018). *Voltage-to-time converter for high-speed time-based analog-to-digital converters* Unpublished Master's thesis, Aalto University Aaltodoc. https://aaltodoc.aalto.fi/handle/123456789/30376

Singh, P., & Mehra, R. (2014). Design analysis of XOR gates using CMOS & pass transistor logic. *International Journal of Engineering Science Invention Research & Development, 1*(1), 21–24.

Sivakumar Reddy, V., et al. (2020). *Soft computing and signal processing*. Springer. https://doi.org/10.1007/978-981-33-6912-2

Szyduczyński, J., Kościelnik, D., & Miśkowicz, M. (2023). Time-to-digital conversion techniques: A survey of recent developments. *Measurement, 214*, 112762. https://doi.org/10.1016/j.measurement.2023.112762

VLSI Verify. (n.d.). Gray to binary code converter using XOR. https://vlsiverify.com/verilog/verilog-codes/gray-to-binary/

Wang, H., Zhang, M., & Liu, Y. (2017). High-resolution digital-to-time converter implemented in an FPGA chip. *Applied Sciences, 7*(1), 52. https://doi.org/10.3390/app7010052

Wang, Y., Xie, W., Chen, H., & Li, D. D.-U. (2022). High-resolution time-to-digital converters (TDCs) with a bidirectional encoder. *Measurement, 196*, 111248. https://doi.org/10.1016/j.measurement.2022.111248

Wang, Y., Xie, W., Chen, H., & Li, D. D.-U. (2023). High-resolution time-to-digital converters (TDCs) with a bidirectional encoder. *Measurement, 206*, 112258. https://doi.org/10.1016/j.measurement.2022.112258

Yadav, N., Kim, Y., Alashi, M., & Choi, K. K. (2020). Design of a voltage to time converter with high conversion gain for reliable and secure autonomous vehicles. *Electronics, 9*(3), 384. https://doi.org/10.3390/electronics9030384

You, Y.-W., & Jeon, J.-C. (2019). Design of extendable binary to Gray code converter using quantum-dot cellular automata. *International Journal of Mechanical Engineering and Technology, 10*(2), 587–596.

Zhang, X., Xie, Y., & Geiger, R. L. (2015). A 5GS/s voltage-to-time converter in 90nm CMOS. *IEEE Transactions on Circuits and Systems I: Regular Papers, 62*(3), 746–754. https://doi.org/10.1109/TCSI.2014.2362514

APPENDICES

APPENDIX A: MATLAB CODES

```
clc; close all; clear;
% Load the data
filename = 'T_H.csv';
data = readtable(filename, 'VariableNamingRule', 'preserve');
% Extract signals
time_vin = data{:,1};        % /Vin X
vin = data{:,2};             % /Vin Y
time_phase = data{:,3};      % /phase X
phase = data{:,4};           % /phase Y
time_phase_bar = data{:,5};  % /phase_bar X
phase_bar = data{:,6};       % /phase_bar Y
time_sampled = data{:,7};    % /sampled_sig X
sampled_sig = data{:,8};     % /sampled_sig Y

% Interpolate Vin to sampled_sig timestamps
vin_interp = interp1(time_vin, vin, time_sampled, 'linear');

% Calculate sampling error
sample_error = vin_interp - sampled_sig;
rmse = sqrt(mean((sample_error).^2));
max_error = max(abs(sample_error));

% Time-domain Plots
figure('Name', 'T/H Waveform Analysis');
subplot(4,1,1);
plot(time_vin, vin, 'r');
title('Input Ramp (Vin)');
ylabel('Voltage (V)');
grid on;

subplot(4,1,2);
```

```
    plot(time_phase, phase, 'g'); hold on;
    plot(time_phase_bar, phase_bar, 'm');
    title('Clock Phases');
    ylabel('Voltage (V)');
    legend('Phase', 'Phase\_bar');
    grid on;

    subplot(4,1,3);
    plot(time_sampled, sampled_sig, 'c');
    title('Sampled Signal');
    ylabel('Voltage (V)');
    grid on;

    subplot(4,1,4);
    plot(time_sampled, sample_error, 'Color', [1 0.5 0]);
    title(sprintf('Sample Error (RMSE = %.4f V, Max = %.4f V)',
rmse, max_error));
    xlabel('Time (ns)');
    ylabel('Error (V)');
    grid on;

    % Linearity Analysis
    figure('Name', 'Linearity of Sampled Signal');
    scatter(vin_interp, sampled_sig, 'b'); hold on;
    P = polyfit(vin_interp, sampled_sig, 1);
    fit_line = polyval(P, vin_interp);
    plot(vin_interp, fit_line, 'r', 'LineWidth', 1.5);
    xlabel('Interpolated Vin (V)');
    ylabel('Sampled Signal (V)');
    title(sprintf('Linearity Fit: y = %.2fx + %.2f', P(1), P(2)));
    legend('Data Points', 'Linear Fit');
    grid on;

    % Display metrics in Command Window
    fprintf('Track-and-Hold Sampling Error Analysis:\n');
    fprintf('  RMSE       = %.4f V\n', rmse);
```

```
    fprintf('  Max Error  = %.4f V\n', max_error);
    r_squared = 1 - sum((sampled_sig - fit_line).^2) /
sum((sampled_sig - mean(sampled_sig)).^2);
    fprintf('  R-squared  = %.4f\n', r_squared);

    clc; close all; clear;
    % Load the data
    filename = 'comp.csv';
    data = readtable(filename, 'VariableNamingRule', 'preserve');

    % Extract time and signal values
    time = data{:, '/sampled_sig X'} * 1e9; % Convert to ns
    Vin = data{:, '/sampled_sig Y'};
    Vth = data{:, '/Vth Y'};
    Vout = data{:, '/comp_out Y'};

    % Plot input vs threshold vs output
    figure;
    subplot(3,1,1);
    plot(time, Vin, 'b', 'LineWidth', 1.5); hold on;
    plot(time, Vth, 'k--', 'LineWidth', 1.5);
    ylabel('Voltage (V)');
    title('Comparator Input vs Threshold');
    legend('Vin (Sampled)', 'V_{th}');
    grid on;

    subplot(3,1,2);
    plot(time, Vout, 'r', 'LineWidth', 1.5);
    ylabel('Comparator Output');
    ylim([-0.1 1.2]);
    title('Comparator Digital Output');
    grid on;

    % Delay Analysis
    % Detect where Vin crosses Vth and where comp_out switches
```

```
    cross_idx = find(Vin(1:end-1) < Vth(1:end-1) & Vin(2:end) >=
Vth(2:end));
    switch_idx = find(Vout(1:end-1) < 0.5 & Vout(2:end) >= 0.5);

    % Estimate first switching delay
    if ~isempty(cross_idx) && ~isempty(switch_idx)
        t_cross = time(cross_idx(1));
        t_out = time(switch_idx(find(switch_idx > cross_idx(1),
1)));
        delay_ps = (t_out - t_cross) * 1e3; % in picoseconds
    else
        delay_ps = NaN;
    end

    subplot(3,1,3);
    plot(time, Vin - Vth, 'm', 'LineWidth', 1.5);
    ylabel('Vin - Vth (V)');
    xlabel('Time (ns)');
    title(sprintf('Threshold Crossing Margin (Delay ≈ %.2f ps)',
delay_ps));
    grid on;

    % Display delay in console
    fprintf('Calculated Comparator Delay ≈ %.2f ps\n', delay_ps);

    clc; clear; close all;

    filename = 'enhanced_VTC.csv';
    data = readtable(filename, 'VariableNamingRule', 'preserve');
    % Extract time and signal values
    t_vin1 = data.("/Vin X");
    vin1 = data.("/Vin Y");

    t_vin2 = data.("/Vin2 X");
    vin2 = data.("/Vin2 Y");
```

```
    t_vth = data.("/Vth X");
    vth = data.("/Vth Y");

    t_start = data.("/START_signal X");
    start_sig = data.("/START_signal Y");

    t_stop = data.("/STOP_signal X");
    stop_sig = data.("/STOP_signal Y");

    % Interpolate to common time base
    t_common = intersect(t_start, t_stop);
    vin1_interp = interp1(t_vin1, vin1, t_common);
    vin2_interp = interp1(t_vin2, vin2, t_common);
    vth_interp = interp1(t_vth, vth, t_common);
    start_interp = interp1(t_start, start_sig, t_common);
    stop_interp = interp1(t_stop, stop_sig, t_common);

    % Detect rising edges of START and STOP
    start_edges = find(diff(start_interp) > 0.5);
    stop_edges = find(diff(stop_interp) > 0.5);

    % delays from START to STOP
    delays = [];
    start_times = [];
    stop_times = [];

    for i = 1:length(start_edges)
        start_idx = start_edges(i);
        next_stops = stop_edges(stop_edges > start_idx);
        if ~isempty(next_stops)
            stop_idx = next_stops(1);
            delay_ps = (t_common(stop_idx) - t_common(start_idx)) *
1e12; % ps
            delays(end+1) = delay_ps;
            start_times(end+1) = t_common(start_idx);
            stop_times(end+1) = t_common(stop_idx);
```

```
    end
end

% Plot Vin1, Vin2, and Vth
figure;
plot(t_vin1, vin1, 'b', 'LineWidth', 1.5); hold on;
plot(t_vin2, vin2, 'r', 'LineWidth', 1.5);
plot(t_vth, vth, 'k--', 'LineWidth', 1.2);
xlabel('Time (s)');
ylabel('Voltage (V)');
title('Dual Ramp Signals vs Threshold');
legend('Vin (0–1.2 V)', 'Vin2 (0–0.5 V)', 'Vth (300 mV)');
grid on;

% Plot START and STOP signals
figure;
plot(t_start, start_sig, 'g', 'LineWidth', 1.5); hold on;
plot(t_stop, stop_sig, 'm', 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Logic Level');
title('START and STOP Signals');
legend('START', 'STOP');
grid on;
clc; clear; close all;

filename = 'enhanced_VTC.csv';
data = readtable(filename, 'VariableNamingRule', 'preserve');

% Extract signals
t = data.('/Vin X') * 1e9;  % Convert time to ns
Vin = data.('/Vin Y');
Vin2 = data.('/Vin2 Y');
Vth = data.('/Vth Y');
phase = data.('/phase Y');
phase_bar = data.('/phase_bar Y');
START = data.('/START_signal Y');
```

```
    STOP = data.('/STOP_signal Y');

    % 1. Time-Domain Plots
    figure('Name','VTC Signal Overview','Position',[100 100 1000
700]);
    subplot(4,1,1);
    plot(t, Vin, 'b', t, Vin2, 'r', t, Vth, 'k--','LineWidth',1.2);
    legend('Vin (0-1.2V)','Vin2 (0-0.5V)','Vth');
    ylabel('Voltage (V)'); title('Input Ramps and Vth'); grid on;

    subplot(4,1,2);
    plot(t, phase, 'b', t, phase_bar, 'r','LineWidth',1.2);
    legend('phase','phase\\_bar');
    ylabel('Logic'); title('Phase and Phase Bar'); grid on;

    subplot(4,1,3);
    plot(t, START, 'g','LineWidth',1.2);
    ylabel('START Output'); title('START Signal (VTC for Vin)');
grid on;

    subplot(4,1,4);
    plot(t, STOP, 'm','LineWidth',1.2);
    ylabel('STOP Output'); xlabel('Time (ns)');
    title('STOP Signal (VTC for Vin2)'); grid on;

    % 2. Transfer Curve Extraction
    % Find delay from Vin/Vin2 crossing Vth to START/STOP edges
    cross_start = find((Vin(1:end-1) < Vth(1:end-1)) & (Vin(2:end)
>= Vth(2:end)));
    cross_stop = find((Vin2(1:end-1) < Vth(1:end-1)) & (Vin2(2:end)
>= Vth(2:end)));
    t_cross_start = t(cross_start);
    t_cross_stop = t(cross_stop);

    delays_start = NaN(size(t_cross_start));
    delays_stop = NaN(size(t_cross_stop));
```

```
    for i = 1:length(cross_start)
        idx = cross_start(i);
        out_idx = idx + find(diff(START(idx:end)) > 0, 1, 'first');
        if ~isempty(out_idx)
            delays_start(i) = t(out_idx) - t(idx);
        end
    end

    for i = 1:length(cross_stop)
        idx = cross_stop(i);
        out_idx = idx + find(diff(STOP(idx:end)) > 0, 1, 'first');
        if ~isempty(out_idx)
            delays_stop(i) = t(out_idx) - t(idx);
        end
    end

    vin_cross_vals = Vin(cross_start(~isnan(delays_start)));
    vin2_cross_vals = Vin2(cross_stop(~isnan(delays_stop)));
    delay_start_ns = delays_start(~isnan(delays_start));
    delay_stop_ns = delays_stop(~isnan(delays_stop));

    % 3. Transfer Curve Plot
    figure('Name','VTC Transfer Curves');
    plot(vin_cross_vals, delay_start_ns*1e3, 'bo-',
'LineWidth',1.2); hold on;
    plot(vin2_cross_vals, delay_stop_ns*1e3, 'rs-',
'LineWidth',1.2);
    xlabel('Input Voltage (V)');
    ylabel('Delay (ps)');
    title('VTC Transfer Curves');
    legend('START (Vin)','STOP (Vin2)');
    grid on;

    % 4. INL/DNL (Basic)
```

```
    all_delays = [delay_start_ns delay_stop_ns]*1e3;  % Convert to
ps
    all_delays = sort(all_delays);
    step_sizes = diff(all_delays);
    step_sizes = step_sizes(:);                  % Ensure column
vector
    mean_step = mean(step_sizes);
    DNL = [0; step_sizes - mean_step];           % Column format for
cumsum
    INL = cumsum(DNL);

    figure('Name','INL/DNL');
    subplot(2,1,1);
    stem(1:length(DNL), DNL, 'filled');
    ylabel('DNL (ps)'); title('Differential Non-Linearity'); grid
on;

    subplot(2,1,2);
    plot(INL, 'LineWidth',1.2);
    ylabel('INL (ps)'); xlabel('Code'); title('Integral Non-
Linearity'); grid on;

    % 5. Monotonicity Check
    isMonotonicStart = all(diff(delay_start_ns) <= 0);
    isMonotonicStop = all(diff(delay_stop_ns) <= 0);
    fprintf('\n--- Monotonicity Check ---\n');
    fprintf('START Delay Monotonic Decrease: %d\n',
isMonotonicStart);
    fprintf('STOP  Delay Monotonic Decrease: %d\n',
isMonotonicStop);

    % 6. Threshold Crossing Margin
    % Difference between crossing Vin and comparator threshold
    margin_start = vin_cross_vals - Vth(1);
    margin_stop = vin2_cross_vals - Vth(1);
```

```
    fprintf('\n--- Threshold Crossing Margin (Vin - Vth) ---\n');
    fprintf('START Mean Margin: %.3f V, Max: %.3f V\n',
mean(margin_start), max(margin_start));
    fprintf('STOP  Mean Margin: %.3f V, Max: %.3f V\n',
mean(margin_stop), max(margin_stop));

    % 7. Summary
    fprintf('\n--- VTC Delay Summary ---\n');
    fprintf('START path: Delay range %.2f - %.2f ns\n',
min(delay_start_ns), max(delay_start_ns));
    fprintf('STOP  path: Delay range %.2f - %.2f ns\n',
min(delay_stop_ns), max(delay_stop_ns));
    fprintf('Total steps: %d\n', length(all_delays));
    fprintf('Resolution (mean step): %.2f ps\n', mean_step);
    fprintf('Max DNL: %.2f ps\n', max(abs(DNL)));
    fprintf('Max INL: %.2f ps\n', max(abs(INL)));

    clc; clear; close all;

    % Load CSV data
    filename = 'Buffer.csv';
    data = readtable(filename, 'VariableNamingRule','preserve');

    % Extract and scale time
    t_in = data.('/START_signal X') * 1e9;  % ns
    vin = data.('/START_signal Y');
    t_out = data.('/Output X') * 1e9;       % ns
    vout = data.('/Output Y');

    % 1. Time-Domain Plot
    figure('Name','Buffer Input vs Output','Position',[100 100 900
500]);
    plot(t_in, vin, 'b', 'LineWidth', 1.3); hold on;
    plot(t_out, vout, 'r', 'LineWidth', 1.3);
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
```

```
    legend('START (Input)', 'Buffered Output');
    title('Buffer Input vs Output');
    grid on;

    % 2. Interpolate for Delay Matching
    % Interpolate output on input timeline for easier delay
computation
    vout_interp = interp1(t_out, vout, t_in, 'linear', 'extrap');

    % Find 50% VDD points
    Vmid = 0.6;
    idx_in = find(vin(1:end-1) < Vmid & vin(2:end) >= Vmid, 1);
    idx_out = find(vout_interp(1:end-1) < Vmid & vout_interp(2:end)
>= Vmid, 1);

    if ~isempty(idx_in) && ~isempty(idx_out)
        t_delay = t_in(idx_out) - t_in(idx_in);
        fprintf('Buffer propagation delay ≈ %.2f ps\n', t_delay *
1e3);
    else
        warning('Could not find valid transition points for delay
estimation.');
        t_delay = NaN;
    end

    % 3. Rise/Fall Times
    % Rise time: 10% to 90% (Input)
    v10 = 0.1 * max(vin);
    v90 = 0.9 * max(vin);
    idx10 = find(vin >= v10, 1, 'first');
    idx90 = find(vin >= v90, 1, 'first');
    trise_in = t_in(idx90) - t_in(idx10);

    % Rise time: Output
    v10_out = 0.1 * max(vout);
    v90_out = 0.9 * max(vout);
```

```
idx10_out = find(vout >= v10_out, 1, 'first');
idx90_out = find(vout >= v90_out, 1, 'first');
trise_out = t_out(idx90_out) - t_out(idx10_out);

fprintf('Input Rise Time ≈ %.2f ps\n', trise_in * 1e3);
fprintf('Output Rise Time ≈ %.2f ps\n', trise_out * 1e3);

clc; clear; close all;

% Load data
filename = 'TSPCFF.csv';
data = readtable(filename, 'VariableNamingRule','preserve');

% Convert time to ns
t = data.('/D X') * 1e9;
D   = data.('/D Y');
CLK = data.('/CLK Y');
Q   = data.('/Q Y');

% 1. Time-Domain Plot
figure('Name','TSPC D Flip-Flop Timing');
subplot(3,1,1); plot(t, D, 'b', 'LineWidth', 1.2); ylabel('D
(V)');
title('Input D (START Signal)'); grid on;

subplot(3,1,2); plot(t, CLK, 'r', 'LineWidth', 1.2); ylabel('CLK
(V)');
title('Clock (STOP Signal)'); grid on;

subplot(3,1,3); plot(t, Q, 'm', 'LineWidth', 1.2); ylabel('Q
(V)'); xlabel('Time (ns)');
title('Q Output'); grid on;

% 2. CLK-to-Q Delay Measurement
clk_thresh = 0.6;
```

```
    clk_edges = find(CLK(1:end-1) < clk_thresh & CLK(2:end) >=
clk_thresh);
    min_delay = 1e-3;  % 1 ps
    delays = [];

    for i = 1:length(clk_edges)
        idx_clk = clk_edges(i);
        t_clk = t(idx_clk);
        window = idx_clk:idx_clk+100;
        dQ = diff(Q(window));
        q_change = find(abs(dQ) > 0.05, 1);  % significant Q change

        if ~isempty(q_change)
            idx_q = window(q_change + 1);
            delay = t(idx_q) - t_clk;
            if delay > min_delay
                delays(end+1) = delay;
            end
        end
    end

    if ~isempty(delays)
        avg_delay = mean(delays);
        fprintf('Average CLK-to-Q Delay: %.2f ps\n', avg_delay *
1e3);
    else
        warning('No valid CLK-to-Q delay transitions found.');
    end
    clc; clear; close all;

    filename = 'finalvdl.csv';
    data = readtable(filename, 'VariableNamingRule','preserve');
    t = data{:, '/START X'} * 1e9; % ns
    N = 15;

    % --- Extract signals ---
```

```
    START = data{:,'/START Y'};
    STOP  = data{:,'/STOP Y'};
    D     = zeros(length(t), N);
    CLK   = zeros(length(t), N);
    Q     = zeros(length(t), N);

    rising_times = NaN(1, N);
    falling_times = NaN(1, N);
    pulse_widths = NaN(1, N);
    rising_D = NaN(1, N);
    rising_CLK = NaN(1, N);

    for k = 1:N
        D(:,k)   = data{:, sprintf('/D%d Y', k-1)};
        CLK(:,k) = data{:, sprintf('/CLK%d Y', k-1)};
        Q(:,k)   = data{:, sprintf('/Q%d Y', k-1)};

        % Q rising/falling
        q_sig = Q(:,k);
        rise_idx = find(q_sig(1:end-1) < 0.6 & q_sig(2:end) >= 0.6,
1);
        fall_idx = find(q_sig(1:end-1) > 0.6 & q_sig(2:end) <= 0.6,
1);

        if ~isempty(rise_idx)
            rising_times(k) = t(rise_idx);
        end
        if ~isempty(fall_idx)
            falling_times(k) = t(fall_idx);
        end
        if ~isnan(rising_times(k)) && ~isnan(falling_times(k))
            pulse_widths(k) = falling_times(k) - rising_times(k);
        end

        % D & CLK rising
        d_sig = D(:,k);
```

```
        clk_sig = CLK(:,k);
        d_idx = find(d_sig(1:end-1) < 0.6 & d_sig(2:end) >= 0.6, 1);
        clk_idx = find(clk_sig(1:end-1) < 0.6 & clk_sig(2:end) >=
0.6, 1);

        if ~isempty(d_idx)
            rising_D(k) = t(d_idx);
        end
        if ~isempty(clk_idx)
            rising_CLK(k) = t(clk_idx);
        end
    end

    % 1. Multi-Panel Signal Visualization
    figure('Name','All Delay Line Signals','Color','w');

    subplot(4,1,1);
    plot(t, START, 'b', t, STOP, 'r', 'LineWidth', 1.2);
    legend('START','STOP');
    ylabel('Voltage (V)');
    title('START and STOP Signals');
    grid on;

    subplot(4,1,2); hold on;
    for k = 1:N
        plot(t, D(:,k) + k*0.4, 'LineWidth', 1);
    end
    ylabel('D Inputs (offset)');
    title('Delay Line D Inputs'); grid on;

    subplot(4,1,3); hold on;
    for k = 1:N
        plot(t, CLK(:,k) + k*0.4, 'LineWidth', 1);
    end
    ylabel('CLK Inputs (offset)');
    title('Delay Line CLK Inputs'); grid on;
```

```
subplot(4,1,4); hold on;
for k = 1:N
    plot(t, Q(:,k) + k*0.4, 'LineWidth', 1);
end
ylabel('Q Outputs (offset)');
xlabel('Time (ns)');
title('Flip-Flop Outputs'); grid on;

% 2. Propagation Delay Plot (Q, D, CLK)
figure('Name','Propagation Delays');
plot(0:N-1, rising_D, 'g^-','LineWidth',2); hold on;
plot(0:N-1, rising_CLK, 'ks-','LineWidth',2);
plot(0:N-1, rising_times, 'bo-','LineWidth',2);
legend('D Delay','CLK Delay','Q Delay');
xlabel('Stage Index');
ylabel('Time (ns)');
title('Signal Propagation Through Delay Line');
grid on;

% 3. Thermometer Code Detection (Including Code 0)
valid_rising = ~isnan(rising_times);
sorted_rises = sort(rising_times(valid_rising));
code_times = [sorted_rises(1) - 0.01, sorted_rises + 1e-3];

thermo_codes = zeros(length(code_times), 1);
for i = 1:length(code_times)
    thermo_codes(i) = sum(rising_times <= code_times(i));
end

% 4. Histogram of Codes
figure('Name','Code Density Histogram');
histogram(thermo_codes, 'BinMethod','integers', ...
    'FaceColor',[0.2 0.6 1], 'FaceAlpha',0.8, 'EdgeColor','k');
xlabel('Output Code'); ylabel('Count');
title('Code Density Histogram (Q Outputs)'); grid on;
```

```
% 5. INL/DNL Analysis
step_sizes = diff(sorted_rises);
mean_step = mean(step_sizes);
DNL = step_sizes - mean_step;
INL = cumsum(DNL);

figure('Name','INL and DNL');
subplot(2,1,1);
stem(1:length(DNL), DNL*1e3, 'filled');
ylabel('DNL (ps)');
title('Differential Non-Linearity'); grid on;

subplot(2,1,2);
stairs(1:length(INL), INL*1e3, 'LineWidth', 2);
ylabel('INL (ps)'); xlabel('Code Index');
title('Integral Non-Linearity'); grid on;

% 6. Performance Summary
dynamic_range = max(rising_times) - min(rising_times);

fprintf('\n--- Vernier Delay Line TDC Summary ---\n');
fprintf('Detected Codes: %d out of %d possible\n', ...
    length(unique(thermo_codes)), N+1);
fprintf('Min Pulse Width: %.2f ps\n', min(pulse_widths)*1e3);
fprintf('Max Pulse Width: %.2f ps\n', max(pulse_widths)*1e3);
fprintf('Avg Pulse Width: %.2f ps\n', mean(pulse_widths)*1e3);
fprintf('Resolution (mean step): %.2f ps\n', mean_step*1e3);
fprintf('Dynamic Range: %.2f ns\n', dynamic_range);
fprintf('Max DNL: %.2f ps\n', max(abs(DNL))*1e3);
fprintf('Max INL: %.2f ps\n', max(abs(INL))*1e3);
fprintf('-------------------------------------\n');
% 7. Detect Rising and Falling Edges + Delays and Pulse Widths
threshold = 0.6;
rising_times = NaN(1, N);
falling_times = NaN(1, N);
```

```
    pulse_widths = NaN(1, N);

    for k = 1:N
        sig = Q(:,k);
        rise_idx = find(sig(1:end-1) < threshold & sig(2:end) >=
threshold, 1);
        fall_idx = find(sig(1:end-1) >= threshold & sig(2:end) <
threshold, 1);
        if ~isempty(rise_idx)
            rising_times(k) = t(rise_idx);
        end
        if ~isempty(fall_idx)
            falling_times(k) = t(fall_idx);
        end
        if ~isnan(rising_times(k)) && ~isnan(falling_times(k))
            pulse_widths(k) = falling_times(k) - rising_times(k);
        end
    end

    % 8. Plot Q Outputs with Edges
    figure('Name','Q Output Pulses with Edge Detection');
    hold on;
    colors = lines(N);
    offset = 0;
    for k = 1:N
        plot(t, Q(:,k) + offset, 'Color', colors(k,:),
'LineWidth',1.2);
        if ~isnan(rising_times(k))
            plot(rising_times(k), 1 + offset, 'ko',
'MarkerFaceColor','g');
        end
        if ~isnan(falling_times(k))
            plot(falling_times(k), offset, 'ks',
'MarkerFaceColor','r');
        end
        offset = offset + 1.5;
```

```
    end
    xlabel('Time (ns)');
    ylabel('Q Outputs (offset)');
    title('Q Signals with Detected Rising/Falling Edges');
    grid on;

    clc; clear; close all;

    % INV Figure
    inv_data = readtable('inv.csv', 'VariableNamingRule',
'preserve');
    figure('Name', 'INV Signals', 'Color', 'w');
    sgtitle('Inverter (INV) Signal Responses', 'FontSize', 18);

    subplot(2,1,1);
    plot(inv_data.('/INV_INPUT X') * 1e9, inv_data.('/INV_INPUT Y'),
'b', 'LineWidth', 1.2); hold on;
    title('INV Input');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    subplot(2,1,2);
    plot(inv_data.('/INV_OUT X') * 1e9, inv_data.('/INV_OUT Y'),
'g', 'LineWidth', 1.2); hold on;
    title('INV Output');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    % MUX Figure
    mux_data = readtable('mux.csv', 'VariableNamingRule',
'preserve');
    figure('Name', 'MUX Signals', 'Color', 'w');
    sgtitle('MUX Signal Responses', 'FontSize', 18);
```

```
    subplot(4,1,1);
    plot(mux_data.('/A X') * 1e9, mux_data.('/A Y'), 'b',
'LineWidth', 1.2); hold on;
    title('MUX Input (A - Q11)');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    subplot(4,1,2);
    plot(mux_data.('/B X') * 1e9, mux_data.('/B Y'), 'r',
'LineWidth', 1.2); hold on;
    title('MUX Input (B - Q3)');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    subplot(4,1,3);
    plot(mux_data.('/SEL X') * 1e9, mux_data.('/SEL Y'), 'm',
'LineWidth', 1.2); hold on;
    title('MUX Select (SEL - Q7)');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    subplot(4,1,4);
    plot(mux_data.('/OUT X') * 1e9, mux_data.('/OUT Y'), 'g',
'LineWidth', 1.2); hold on;
    title('MUX Output');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    % SingleBuffer Figure
    buffer_data = readtable('SingleBuffer.csv',
'VariableNamingRule', 'preserve');
    figure('Name', 'Single Buffer Signals', 'Color', 'w');
```

```
    sgtitle('Single Buffer Signal Responses', 'FontSize', 18);

    subplot(2,1,1);
    plot(buffer_data.('/Buffer_INPUT X') * 1e9,
buffer_data.('/Buffer_INPUT Y'), 'b', 'LineWidth', 1.2); hold on;
    title('Buffer Input');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    subplot(2,1,2);
    plot(buffer_data.('/Buffer_OUTPUT X') * 1e9,
buffer_data.('/Buffer_OUTPUT Y'), 'g', 'LineWidth', 1.2); hold on;
    title('Buffer Output');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    % XOR Figure
    xor_data = readtable('xor.csv', 'VariableNamingRule',
'preserve');
    figure('Name', 'XOR Signals', 'Color', 'w');
    sgtitle('XOR Gate Signal Responses', 'FontSize', 18);

    subplot(3,1,1);
    plot(xor_data.('/XOR_A X') * 1e9, xor_data.('/XOR_A Y'), 'b',
'LineWidth', 1.2); hold on;
    title('XOR Input (A)');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    subplot(3,1,2);
    plot(xor_data.('/XOR_B X') * 1e9, xor_data.('/XOR_B Y'), 'r',
'LineWidth', 1.2); hold on;
    title('XOR Input (B)');
```

```
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    subplot(3,1,3);
    plot(xor_data.('/XOR_OUT X') * 1e9, xor_data.('/XOR_OUT Y'),
'g', 'LineWidth', 1.2); hold on;
    title('XOR Output');
    xlabel('Time (ns)');
    ylabel('Voltage (V)');
    grid on;

    clc; close all; clear;
    % Load the CSV file with preserved variable names
    filename = 'muxencoder.csv';
    data = readtable(filename, 'VariableNamingRule', 'preserve');

    % Extract time (in nanoseconds)
    t = data{:, '/n0 X'} * 1e9; % Convert to nanoseconds

    % Number of thermometer codes and encoder outputs
    N = 15;
    M = 4;

    % Extract thermometer code signals (Q0 to Q14)
    n = zeros(length(t), N);
    for k = 1:N
        n(:,k) = min(max(data{:, sprintf('/n%d Y', k-1)}, 0), 1.2);
% Clip to 0-1.2V
    end

    % Extract encoder outputs (G0 to G3)
    Gout = zeros(length(t), M);
    for k = 1:M
        Gout(:,k) = min(max(data{:, sprintf('/G%d Y', k-1)}, 0),
1.2); % Clip to 0-1.2V
```

```
    end

    % ----- PLOT: Thermometer Inputs and Encoder Outputs -----
    figure;
    set(gcf, 'Position', [100, 100, 1600, 2400]);

    % Plot thermometer inputs (Q0 to Q14)
    subplot(M+1, 1, 1);
    hold on;
    for k = 1:N
        plot(t, n(:,k) + (k-1) * 1.2, 'LineWidth', 1.2); % Offset
for visual separation
    end
    xlabel('Time (ns)');
    ylabel('Voltage (V) (Offset)');
    title('Thermometer Inputs (Q0 to Q14)');
    yticks(0:1.2:(N-1)*1.2);
    yticklabels(arrayfun(@(x)sprintf('Q%d',x),0:N-
1,'UniformOutput',0));
    ylim([0 (N)*1.2]);
    grid on;
    hold off;

    % Plot each encoder output (G0 to G3) in separate subplots
    for j = 1:M
        subplot(M+1, 1, j+1);
        plot(t, Gout(:,j), 'LineWidth', 1.2);
        xlabel('Time (ns)');
        ylabel('Voltage (V)');
        title(['Encoder Output (G', num2str(j-1), ')']);
        ylim([0 1.2]);
        grid on;
    end

    % ----- DNL Calculation (Improved Period Method) -----
    % Calculate the ideal LSB size
```

```
ideal_LSB = 1.2 / (2^M - 1);
DNL_all = [];

% Calculate DNL for each 4-bit period without flattening
num_samples = size(Gout, 1);
period_length = 2^M;
num_periods = floor(num_samples / period_length);

for p = 1:num_periods
    % Extract the current 4-bit period
    start_idx = (p-1) * period_length + 1;
    end_idx = p * period_length;
    current_period = Gout(start_idx:end_idx, :);

    % Calculate step sizes within this period
    for b = 1:M
        steps = diff(current_period(:, b));
        DNL = (steps - ideal_LSB) / ideal_LSB;
        DNL_all = [DNL_all; DNL];
    end
end

% Calculate maximum DNL
max_DNL = max(abs(DNL_all));

% Print the results
fprintf('Dynamic Range: %.2f V\n', max(Gout(:)) - min(Gout(:)));
fprintf('Maximum DNL: %.2f LSB\n', max_DNL);

% ----- PLOT: DNL -----
figure;
plot(DNL_all, 'LineWidth', 1.2);
title('DNL');
xlabel('Sample Index');
ylabel('DNL (LSB)');
grid on;
```

```
clc; close all; clear;

% Load the DAC CSV file
filename = 'dac.csv';
data = readtable(filename, 'VariableNamingRule', 'preserve');

% Extract time (in nanoseconds)
time = data{:,'/Analog_Output X'} * 1e9;  % Convert to ns

% Extract 4-bit Gray code inputs
G0 = data{:,'/G0 Y'};
G1 = data{:,'/G1 Y'};
G2 = data{:,'/G2 Y'};
G3 = data{:,'/G3 Y'};

% Extract analog output
analog_output = data{:,'/Analog_Output Y'};

% Calculate Dynamic Range
dynamic_range = max(analog_output) - min(analog_output);

% Calculate DNL
% Extract unique levels to reduce noise impact
unique_levels = unique(round(analog_output, 5));
ideal_LSB = dynamic_range / (2^4 - 1);
dnl = (diff(unique_levels) - ideal_LSB) / ideal_LSB;
max_DNL = max(abs(dnl));

% Plot the 4-bit Gray code inputs and analog output
figure;
subplot(2, 1, 1);
hold on;
plot(time, G0, 'LineWidth', 1.2);
plot(time, G1 + 1.5, 'LineWidth', 1.2);
plot(time, G2 + 3.0, 'LineWidth', 1.2);
```

```
plot(time, G3 + 4.5, 'LineWidth', 1.2);
title('4-Bit Gray Code Inputs (G0 to G3)');
xlabel('Time (ns)');
ylabel('Voltage (V)');
yticks([0, 1.5, 3.0, 4.5]);
yticklabels({'G0', 'G1', 'G2', 'G3'});
grid on;
hold off;

subplot(2, 1, 2);
plot(time, analog_output, 'LineWidth', 1.2);
title('Analog Output');
xlabel('Time (ns)');
ylabel('Voltage (V)');
grid on;

% Plot the DNL in a separate figure
figure;
plot(dnl, 'LineWidth', 1.2, 'Color', 'blue');
title('DNL of 4-Bit DAC');
xlabel('Sample Index');
ylabel('DNL (LSB)');
grid on;

% Print the analysis results
fprintf('Dynamic Range: %.2f V\n', dynamic_range);
fprintf('Maximum DNL: %.2f LSB\n', max_DNL);

clc; close all; clear;
% Load the full system CSV file
filename = 'fullvdltdc1.csv'
data = readtable(filename, 'VariableNamingRule', 'preserve');
t = data{:, '/START_signal X'} * 1e9;  % Convert to ns

% Extract Encoder and DAC signals as a single vector
ENC = data{:, 45:52};  % G0 to G3
```

```
DAC = data{:,'/Analog_Output Y'};

ENC = min(max(ENC, 0));
DAC = min(max(DAC, 0));

% Combine Encoder and DAC for full system analysis
combined_output = [ENC(:); DAC(:)];

% Apply Moving Average Filter
ma_window_size = 7;
filtered_output = movmean(combined_output, ma_window_size);

% Apply Median Filter
med_window_size = 5;
filtered_output = medfilt1(filtered_output, med_window_size);

% Apply Savitzky-Golay Filter (Preserves edges better than
moving average)
sg_order = 2;
sg_frame_size = 15;
filtered_output = sgolayfilt(filtered_output, sg_order,
sg_frame_size);

% Baseline Correction
baseline_offset = min(filtered_output);
filtered_output_corrected = filtered_output - baseline_offset;
filtered_output_corrected = min(max(filtered_output_corrected,
0), 1.2);

% Calculate Dynamic Range
dynamic_range = max(filtered_output_corrected) -
min(filtered_output_corrected);

% Calculate overall DNL (in LSB)
ideal_LSB_full = dynamic_range / (2^4 - 1);
step_sizes_full = diff(filtered_output_corrected);
```

```
DNL_full = (step_sizes_full - ideal_LSB_full) / ideal_LSB_full;
max_DNL_full = max(abs(DNL_full));

% Plot DNL for Full System
figure;
stem(1:length(DNL_full), DNL_full, 'filled', 'MarkerSize', 4);
title('DNL of Full System');
xlabel('Code Index');
ylabel('DNL (LSB)');
grid on;

% Print the results in LSB and Dynamic Range
fprintf('\n--- Full System DNL Summary ---\n');
fprintf('Max DNL: %.2f LSB\n', max_DNL_full);
fprintf('Dynamic Range: %.2f V\n', dynamic_range);
fprintf('--------------------------------\n');
```

CURRICULUM VITAE