Time-to-Digital Converter for Wideband Time-Based Analog-to-Digital Converters

Okko Järvinen

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 26.3.2018

Thesis supervisor:

Prof. Jussi Ryynänen

Thesis advisor:

D.Sc. Marko Kosunen

Aalto University
School of Electrical
Engineering

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE
MASTER'S THESIS

Author: Okko Järvinen

Title: Time-to-Digital Converter for Wideband Time-Based Analog-to-Digital Converters

Date: 26.3.2018 Language: English Number of pages: 8+65

Department of Electronics and Nanoengineering

Professorship: Electronic Circuit Design

Supervisor: Prof. Jussi Ryynänen

Advisor: D.Sc. Marko Kosunen

Modern deeply scaled semiconductor processes make the design of voltage-domain circuits increasingly challenging. On the contrary, the area and power consumption of digital circuits are improving with every new process node. Consequently, digital solutions are designed in place of their purely analog counterparts in applications such as analog-to-digital (A/D) conversion. Time-based analog-to-digital converters (ADC) employ digital-intensive architectures by processing analog quantities in time-domain. The quantization step of the time-based A/D-conversion is carried out by a time-to-digital converter (TDC).

A free-running ring oscillator -based TDC design is presented for use in wideband time-based ADCs. The proposed architecture aims to maximize time resolution and full-scale range, and to achieve error resilient conversion performance with minimized power and area consumptions. The time resolution is maximized by employing a high-frequency multipath ring oscillator, and the full-scale range is extended using a high-speed gray counter. The error resilience is achieved by custom sense-amplifier -based sampling flip-flops, gray coded counter and a digital error correction algorithm for counter sampling error correction. The implemented design achieves up to 9-bit effective resolution at 250 MS/s with 4.3 milliwatt power consumption.

Keywords: analog-to-digital converter, time-to-digital converter, time-based, time-domain, multipath ring oscillator, gray counter

AALTO-YLIOPISTO
SÄHKÖTEKNIIKAN KORKEAKOULU

DIPLOMITYÖN
TIIVISTELMÄ

Tekijä: Okko Järvinen

Työn nimi: Aika-digitaalimuunnin laajakaistaisiin aikapohjaisiin analogia-digitaalimuuntimiin

Päivämäärä: 26.3.2018 Kieli: Englanti Sivumäärä: 8+65

Elektroniikan ja nanotekniikan laitos

Professuuri: Mikro- ja nanoelektroniikkasuunnittelu

Työn valvoja: Prof. Jussi Ryynänen

Työn ohjaaja: TkT Marko Kosunen

Modernien puolijohdeteknologioiden skaalautumisen seurauksena jännitetason piirien suunnittelu tulee entistä haasteellisemmaksi. Toisaalta digitaalisten piirirakenteiden pinta-ala sekä tehonkulutus pienenevät prosessikehityksen myötä. Tästä syystä digitaalisia ratkaisuja suunnitellaan vastaavien puhtaasti analogisien rakenteiden tilalle. Analogia-digitaalimuunnos (A/D-muunnos) voidaan toteuttaa jännitetason sijaan aikatasossa käyttämällä aikapohjaisia A/D-muuntimia, jotka ovat rakenteeltaan pääosin digitaalisia. Kvantisointivaihe aikapohjaisessa A/D-muuntimessa toteutetaan aika-digitaalimuuntimella.

Työ esittelee vapaasti oskilloivaan silmukkaoskillaattoriin perustuvan aika-digitaalimuuntimen, joka on suunniteltu käytettäväksi laajakaistaisessa aikapohjaisessa A/D-muuntimessa. Esitelty rakenne pyrkii maksimoimaan muuntimen aikaresoluution sekä muunnosalueen, sekä saavuttamaan virhesietoisen muunnostoiminnan minimoidulla tehon sekä pinta-alan kulutuksella. Aikaresoluutio on maksimoitu hyödyntämällä suuritaajuista monipolkuista silmukkaoskillaattoria, ja muunnosalue on maksimoitu nopealla Gray-koodi -laskuripiirillä. Muunnosprosessin virhesietoisuus on saavutettu toteuttamalla näytteistys herkillä kiikkuelementeillä, hyödyntämällä Gray-koodattua laskuria, sekä jälkiprosessoimalla laskurin näytteistetyt arvot virheenkorjausalgoritmilla. Esitelty muunnintoteutus saavuttaa 9 bitin efektiivisen resoluution 250 MS/s näytetaajuudella ja 4.3 milliwatin tehonkulutuksella.

Avainsanat: analogia-digitaalimuunnin, aika-digitaalimuunnin, aikapohjainen, A/D-muunnin

Preface

I would like to thank my supervisor Prof. Jussi Ryynänen firstly for having enough faith in me to let me work on this project, and secondly for having patience with me and helping me keep this project on track. I'm deeply grateful for the guidance given by my advisor D.Sc. Marko Kosunen regarding technical details of the work, as well as thesis writing process in general. Thank you friends and co-workers for making the work environment friendly and pleasant, and for sharing ideas and knowledge. I'm happy and grateful to be a part of the Electronic Circuit Design group. Finally, I want to show my sincerest gratitude to my family who have supported me throughout my life and who always take interest in what I'm up to – thank you.

Otaniemi, 26.3.2018

Okko T. I. Järvinen

Contents

Symbols and Abbreviations

Symbols

$A_{distortion}$	Integrated distortion amplitude
A_{noise}	Integrated noise amplitude
A_{signal}	Signal amplitude
$\Delta[n]$	Pulse widths of the time signal
ΔLSB	Width of an LSB
D_{out}	Digital output bus
ϕ	Clock phase
f_{in}	Input signal frequency
f_{osc}	Oscillator frequency
f_s	Sampling frequency
FOM_S	Schreier figure-of-merit
FOM_W	Walden figure-of-merit
ρ	Slope of the VTC output pulse width versus input voltage
σ_j	Timing jitter
σ_{ss}	Single-shot precision
σ_{tie}	Time interval error jitter
S	Synchronization period in MTBF calculation
τ	Delay element delay
τ_C	Counter path buffer delay
τ_{coarse}	Coarse path delay in Vernier TDCs
τ_{fine}	Fine path delay in Vernier TDCs
τ_{min}	Minimum element delay in SAR TDC
$\tau_{min} + \tau_k$	Adjustable delay line in SAR TDC
t_{begin}	Begin time instant of the time signal
$t_{conversion}$	Conversion portion of the sampling period
t_{end}	End time instant of the time signal
$t_{F,max}$	Maximum time required by fine conversion in VDR
t_{hold}	Hold time of a flip-flop
t_{max}	Maximum time interval at the VTC output
$t_{sampling}$	Sampling portion of the sampling period
t_{setup}	Setup time of a flip-flop
T_D	Delay between oscillator zero phase and counter increment
T_G	Counter output glitch zone width
T_{meta}	Metastability window
T_s	Sampling period
V_{dd}	Positive voltage supply
V_{fs}	Input full-scale voltage
V_{in}	Analog input voltage signal
$x[n]$	N'th input sample

Abbreviations

A/D	Analog-to-digital
ADC	Analog-to-digital converter
ADPLL	All-digital phase locked loop
BW	Bandwidth
DFF	D-flip-flop
DL	Delay line
DNL	Differential nonlinearity
ENOB	Effective number of bits
FF	Flip-flop
FOM	Figure-of-merit
FS	Full-scale
GRO	Gated ring oscillator
IC	Integrated circuit
INL	Integral nonlinearity
IoT	Internet-of-Things
LSB	Least significant bit
MPRO	Multipath ring oscillator
MSB	Most significant bit
MTBF	Mean time between failures
NMOS	N-type metal-oxide semiconductor
PI	Phase interpolation
PL	Pipeline
PMOS	P-type metal-oxide semiconductor
PT-TA	Pulse-train time amplifier
PVT	Process, voltage, temperature
PWM	Pulse width modulation
RO	Ring oscillator
S/H	Sample-and-hold
S/s	Samples per second
SA	Sense-amplifier
SAFF	Sense-amplifier flip-flop
SAR	Successive approximation register
SFDR	Spurious-free dynamic range
SNR	Signal-to-noise ratio
SNDR	Signal-to-noise-and-distortion ratio
T/D	Time-to-digital
TA	Time amplifier
TB	Time-based
TDC	Time-to-digital converter
TFF	Toggle flip-flop
TI	Time-interleaving
TIE	Time interval error
TOF	Time-of-flight
TSPC	True single phase clock
VCO	Voltage-controlled oscillator
VDL	Vernier delay line
VDR	Vernier delay ring
VHDL	VHSIC hardware description language
VHSIC	Very high speed integrated circuit
VTC	Voltage-to-time converter

1 Introduction

The amount of data being transmitted and processed by modern devices is constantly increasing. The rapid development of deeply scaled integrated circuit (IC) process technologies is a key driver in this development. Consequently, modern telecommunications applications require circuits with ever increasing performance demands. The physical world is analog by nature, whereas electronic processing systems are digital. The disconnect between these two fundamental domains is bridged by analog-to-digital converters (ADC). The ADCs are key components in applications, such as telecommunications, data acquisition, measurement and sensor systems. With each new developed semiconductor process node, the feature size of the transistor decreases. Additionally, the supply voltages of these circuits keep decreasing, which reduces the voltage headroom of analog circuit structures. On the other hand, digital circuits benefit from process scaling in terms of improved power efficiency and speed. This trend emphasizes the importance of efficient digital and mixed-signal solutions designed to replace their analog counterparts. The processing of analog quantities is typically performed in voltage-domain. However, the analog quantities can also be represented in another analog domain – time.

Time-domain circuits can employ digital architectures in processing analog quantities. Analog-to-digital (A/D) conversion can also be carried out in time-domain. Thus, time-based ADCs (TBADC) provide an digital-intensive solution to the necessary conversion step in the flow of information. One of the most critical parts of the time-based ADC is the time-to-digital converter (TDC). The TDC circuits operate completely in time-domain, and they are responsible for producing a digital representation of the time signals, in which the analog information is carried by time intervals. As a consequence to the digital-intensive architecture, TDCs and TBADCs can take advantage of the rapid development of newer sub-micron processes by performing analog signal processing operations in the time-domain.

The objective of this thesis is to design a high-speed TDC for a wideband time-based ADC. In order to reach this objective, several proven TDC concepts are evaluated and compared on a qualitative level. The rest of the thesis is divided into 5 sections. In Section 2, various basic concepts related to A/D-conversion are presented along with common metrics used for characterizing ADCs, and ADC development trends are discussed. In Section 3, time-based ADCs are discussed on a general level, with the aim to highlight the difference between voltage-domain and time-domain conversion. In Section 4, several TDC concepts are presented and their limitations and attributes are discussed. In Section 5, the high-speed TDC design is presented. The designed TDC is simulated and characterized, and the shortcomings and possibilities of the architecture are presented. Finally, in Section 6, the results of the thesis are summarized.

2 Background

2.1 Analog-to-Digital Conversion

Analog-to-digital converter is a commonly used circuit, which converts analog signals into digital representation. All systems which contain analog sections together with digital processing circuitry require ADCs. The performance requirements of modern ADCs are increasingly demanding, and the required performance metrics depend heavily on the application. The data rate of the converter is defined by the bit depth of the output as well as the sampling frequency (f_s). In applications such as mobile radio receivers, a moderately high data rate is required with low power consumption. The next generation of mobile devices creates demand for higher bandwidth (BW) and lower power converters. Sensors and audio devices are typical applications for high resolution converters. The concept of standalone mobile sensor networks introduced as Internet-of-Things (IoT) increases the demand for power efficient converters [1].

Analog-to-digital conversion can be divided into a sampling phase and a conversion phase. In the sampling phase, the current value of the analog signal is sampled and stored for conversion. The voltage is typically stored into a capacitor known as a sampling capacitor. Once the sampling phase is finished, the sampled voltage is quantized. There are numerous ways of creating the digital representation of the sampled voltage value. The concept of quantization is based on seeking the nearest available discrete binary value, which corresponds to the sampled analog quantity. This will effectively quantize the analog signal into digital binary values. Fig. 1 presents the quantization of a ramp signal.

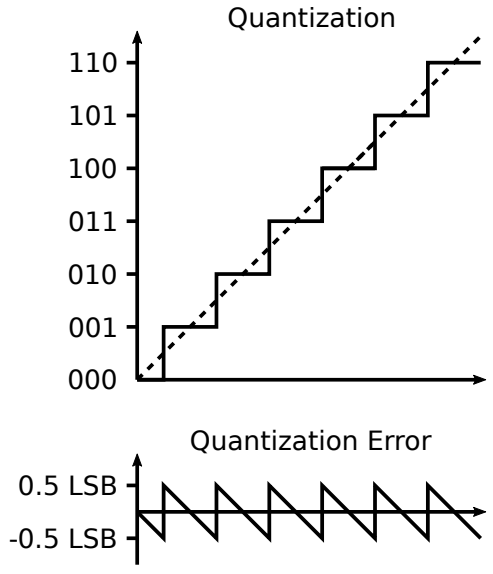


Figure 1: The quantization of an analog signal (dashed line) producing a digital output code (solid line). The inherent error caused by the finite quantization step is presented in the lower figure.

2.2 ADC Characterization

The quantization of an analog signal will introduce quantization noise. The noise is caused by the finite resolution of the quantized representation, as the discontinuities between the discrete samples result in additional frequency components in the output spectrum. Thus, the output of an ADC will contain intrinsic noise caused by the quantization. Furthermore, the non-ideal conversion process will introduce additional harmonics and distortion components, which will limit the performance of the converter.

Numerous metrics are used for characterizing the performance and linearity of an ADC. The typical metrics include signal-to-noise ratio (SNR), signal-to-noise-and-distortion ratio (SNDR), spurious-free dynamic range (SFDR), effective number of bits (ENOB), integral and differential nonlinearity (INL and DNL), as well as figures-of-merit (FOM) derived from these.

Signal-to-noise ratio is defined as the ratio between the signal (desired frequency component at the output) amplitude and the noise floor level. It can be calculated as

$$\text{SNR} = 20 \cdot \log\left(\frac{A_{signal}}{A_{noise}}\right) \quad \text{[dB]}, \tag{1}$$

where A_{signal} is the signal amplitude and A_{noise} is the integrated noise level.

Signal-to-noise-and-distortion ratio is defined similarly to SNR. However, the harmonic distortion components are included in the calculation. The value of SNDR can be calculated as

$$\text{SNDR} = 20 \cdot \log\left(\frac{A_{signal}}{A_{noise} + A_{distortion}}\right) \quad \text{[dB]}, \tag{2}$$

where A_{signal} is the signal amplitude, A_{noise} is the integrated noise level and $A_{distortion}$ is the level of integrated distortion components.

The spurious-free dynamic range is defined as the difference between the signal level and the highest spurious (unwanted) frequency component level. SFDR is typically expressed in decibels in respect to the carrier amplitude (dBc).

The effective number of bits at the output of the ADC is dependent on the SNDR of the converter. ENOB can be calculated as

$$\text{ENOB} = \frac{\text{SNDR} - 1.76}{6.02} \quad \text{[bits]}, \tag{3}$$

where the number 1.76 is the quantization noise power of a quantized sinusoid and 6.02 is the power corresponding to one quantization bit according to $20 \cdot \log(2) \approx 6.02$ [2].

Integral nonlinearity of an ADC is defined as the deviation of the output transfer curve from a straight line. The INL is calculated after removing both offset and gain errors from the ADC output. The straight line used as reference can be chosen either by connecting the end points of the output curve or by using a best-fit line. INL is typically given either as a single value or as a plot with INL values for all output codes. In the case of a single value, the given value is the maximum deviation. Differential nonlinearity of an ADC refers to the deviation of an output code step

from the ideal one LSB step. For an ADC, if the DNL is between ±0.5 LSB for all codes, no output codes are skipped. Like for INL, DNL is typically taken after removing the offset and gain errors from the output. [2] The DNL can be calculated as

$$\mathrm{DNL}[n] = \frac{D[n+1] - D[n]}{\Delta\mathrm{LSB}_{ideal}} \quad [\mathrm{LSB}], \tag{4}$$

where $D[n]$ is the n-th output code of the ADC and $\Delta\mathrm{LSB}_{ideal}$ is the ideal width of the LSB. The INL can be then calculated as the cumulative sum of the calculated DNL values.

In Section 2.4, different figures-of-merit are introduced for comparing various ADCs. The state-of-the-art trends can be derived using these figures. The two commonly used FOM metrics are the Walden FOM and Schreier FOM. The Walden FOM gives the energy used for one conversion step [3]. It can be calculated as

$$\mathrm{FOM}_W = \frac{P}{f_s \cdot 2^{\mathrm{ENOB}}} \quad [\mathrm{J\ /\ conversion\ step}], \tag{5}$$

where P is the overall power consumption of the ADC, f_s is the sampling frequency and ENOB is the effective number of bits. On the other hand, the Schreier FOM denotes a decibel value, which accounts for SNDR, bandwidth and power consumption. It can be calculated as

$$\mathrm{FOM}_S = \mathrm{SNDR} + 10 \cdot \log_{10}\left(\frac{\mathrm{BW}}{P}\right) \quad [\mathrm{dB}], \tag{6}$$

where SNDR is the signal-to-noise-and-distortion ratio, BW is the bandwidth and P is the power consumption of the ADC.

2.3 Common ADC Architectures

Various different ADC architectures are needed since the range of applications and their requirements are numerous. ADC architectures can be vaguely categorized based on their conversion rate and resolution. For each range of operation speed and accuracy, popular architectures exist. Some of the most common and frequently used architectures are briefly introduced below. Time-based ADCs are omitted from this section, since they are presented in Section 3.

2.3.1 Delta-Sigma (ΔΣ)

The ΔΣ (Delta-Sigma) ADC is an oversampling converter that achieves very high resolution at the expense of sample rate. The dynamic range of an ADC can be increased by oversampling, which effectively distributes the quantization noise power over an increased frequency range. However, doubling the sampling frequency will only improve the dynamic range by 3 decibels. The ΔΣ ADC modulators provide so-called noise shaping by utilizing a feedback loop in the modulator. In noise shaping, the quantization noise of the ADC is effectively high-pass filtered. Combined with the band-limited input signal, noise shaping and oversampling offer high dynamic range for the conversion.

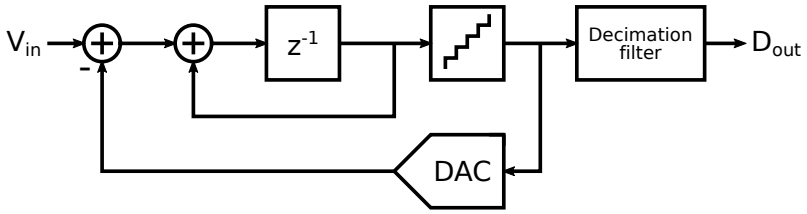


Figure 2: Block diagram of a $\Delta\Sigma$ ADC with first order modulator. [2]

A first-order $\Delta\Sigma$ modulator is presented in Fig. 2. A typical $\Delta\Sigma$ ADC consists of a low-pass filter, sample-and-hold (S/H), $\Delta\Sigma$ modulator and a digital decimation filter. The low-pass filter at the input limits the input band to less than half of the oversampling frequency. The S/H circuit samples the input signal and keeps the sample constant during conversion. The $\Delta\Sigma$ modulator converts the analog sample into a low-resolution digital signal, which is noise shaped. The decimation filter at the output takes the low-resolution signals from the modulator and converts them into a low-frequency high-resolution digital output. [2] $\Delta\Sigma$ ADCs are suited for low-speed and high-resolution applications since the noise shaping and oversampling provide the high resolution, while they increase the overall conversion time.

2.3.2 Successive Approximation Register

The successive approximation register (SAR) ADC is one of the most popular ADC architectures. SAR ADCs provide mid-range speed and resolution for great energy efficiency. The conversion algorithm employed by SAR ADCs resembles binary search algorithm. The sampled input value is being compared to a voltage reference, that is used to split the search range. The voltage reference is updated after each comparison cycle. The algorithm will first resolve the MSB of the conversion by comparing the input value to the half-range reference. The result of the first step is the MSB bit, which informs whether the input sample is in the upper or lower region of the full range. The voltage reference is then updated and the following bit is resolved. For an N-bit conversion, the algorithm requires N cycles.

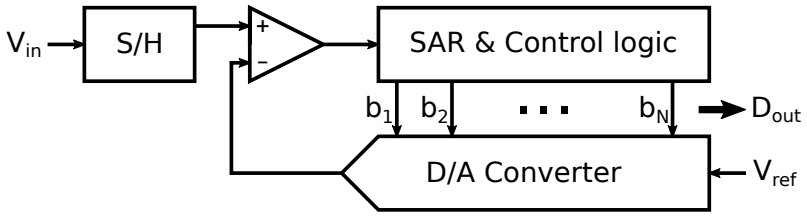


Figure 3: Block diagram of a SAR ADC. [2]

A block diagram of a SAR ADC is presented in Fig. 3. The SAR ADC consists of a S/H circuit at the input, a comparator, the successive approximation register and a digital-to-analog converter (DAC) for generating the comparator reference.

In general, the DAC design is critical to the accuracy and speed of the ADC. [2] Also, the comparator performance is critical, since the comparison has to be accurate enough to resolve the LSB. The resolution of SAR ADCs is typically limited to approximately 10 bits due to the required capacitor sizes and comparator accuracy [4].

2.3.3 Flash ADC

Flash ADCs (also known as parallel ADCs) are the fastest type of ADC available. A 2-bit Flash ADC block diagram is presented in Fig. 4. The conversion is carried out by parallel comparators, which have the comparison reference provided by for example a resistor string. The resistor string divides the full voltage range into multiple voltage bins. As the input sample is fed into the comparators, the samples are compared to each respective voltage level. All of the comparators which have a reference voltage higher than the input sample will output a logic high, while the rest of the comparator outputs will be low. This effectively quantizes the analog voltage into thermometer encoded bits. The thermometer encoded output is then typically converted into a one-hot code, where only one of the bits is logic high. Additionally, bubble error correction can be implemented into this conversion in order to correct any erroneous bits. The corrected output is finally encoded into binary.

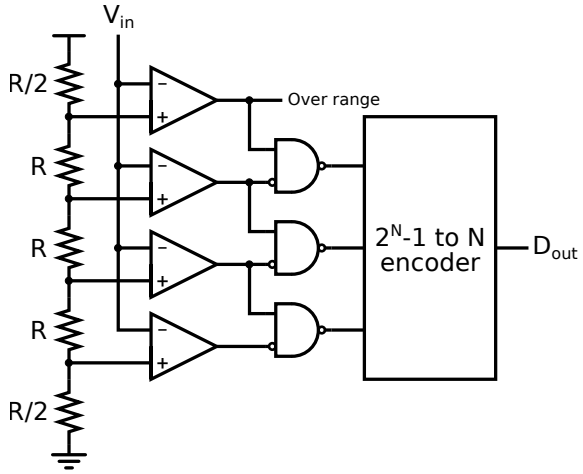


Figure 4: Block diagram of a 2-bit Flash ADC. [2]

While Flash ADCs are certainly fast, they require a large area and consume significant power, since in order to carry out an N-bit conversion, 2^N comparators are needed. [2] This effectively limits the typical bit range to less than 6 or 7 bits. Thus, Flash ADCs are typically used for high frequency, low resolution applications.

2.3.4 Pipeline

The A/D conversion can be carried out in multiple stages. A pipelined ADC architecture utilizes multiple cascaded low-resolution converter stages. Between the stages, the residue of the previous conversion stage is amplified. In pipelining, all

stages are functioning simultaneously after the initial latency. The first converter stage will start the following conversion step immediately after it has passed the conversion result to the following stage. Consequently, the sample rate of a pipelined ADC will be the same as the sample rate of a single converter stage. However, pipelining introduces latency to the conversion. Pipelined ADCs can utilize any type of ADC architectures in the pipeline stages. Thus, pipelined ADC is actually a technique, which can be used to improve the overall performance of the ADC.

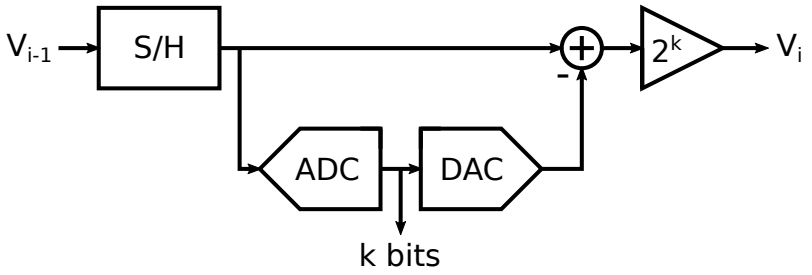


Figure 5: Block diagram of a k-bit pipeline stage of a pipelined ADC. [2]

A k-bit pipeline converter stage is presented in Fig. 5. It consists of a S/H, low-resolution ADC, low-resolution DAC and an amplifier. The sampled signal is converted into digital by the ADC block and passed into a shift register. The digital word is converted back into an analog value and subtracted from the original analog value. This creates a residue value, which is then amplified by 2^k, where k is the number of bits in the A/D and D/A conversion of a single stage. The amplified residue is passed onto the next cascaded pipeline stage, which repeats the same process. After N stages, a digital word with $N \cdot k$ bit resolution is available. Pipelined ADCs can achieve high-speed conversion with increased resolution at the expense of latency. [2]

2.3.5 Time-Interleaving

In addition to pipelining, ADC performance can be increased using time-interleaving (TI). Similarly to pipelining, in time-interleaving, any ADC architecture can be used in the time-interleaved branches. Time-interleaving refers to using multiple parallel ADC branches, which are clocked with clock signals, that are equally distributed in time-domain. Each time-interleaved branch operates at a fraction of the total sampling frequency of the TIADC, which eases the design requirements of individual stages. A block diagram of a four-times time-interleaved ADC is presented in Fig. 6.

Time-interleaving can be used to achieve high-speed and high-resolution ADCs, similarly to pipelining. However, this method requires careful calibration of timing, gain and offset mismatches between the time-interleaved branches. Mismatches will produce strong spurious tones inside the signal band at f_s/M intervals, where M is the number of time-interleaved branches. [2]

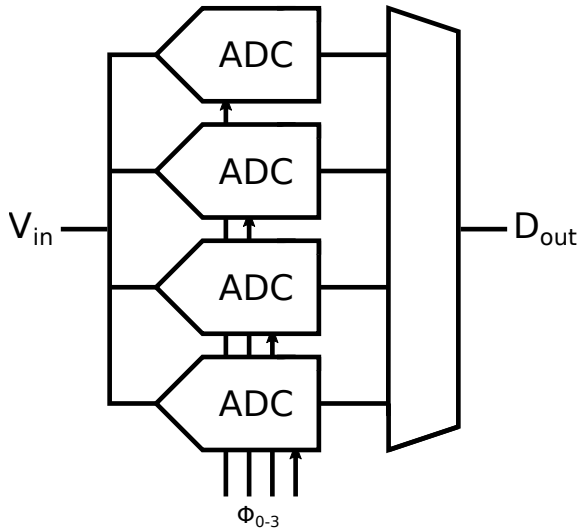


Figure 6: Block diagram of a four-times time-interleaved ADC structure.

2.4 Converter Trends

Comparing various ADCs can be challenging since the performance requirements and metrics vary significantly. However, various figures of merit have been derived in order to facilitate the comparison. The two commonly used FOMs are explained in Section 2.2. These figures can be used to follow the state-of-the-art of modern converters. The aperture plot of ADCs published in ISSCC and VLSI is presented in Fig. 7. The figure describes the speed-resolution trade-off of ADCs. However, it does not account for the power consumption, which is critical since the number of demanding mobile applications is increasing at a rapid rate. Thus, the Walden and Schreier FOMs describe the performance of ADCs in a more well rounded manner.

The Walden FOM has been plotted for various published converters in Fig. 8. The Walden FOM describes the energy consumed by a single conversion step. Typically, low to medium resolution ADCs do well in this figure-of-merit. In the figure, the envelope curve has been plotted for the year 2017, which represents the state-of-the-art for the published converters. In 2017, the state-of-the-art in terms of Walden FOM is approximately 1.2 femtojoules per conversion step for a converter with 100 megahertz sampling frequency. Out of the 32 included converters from 2017, 20 are SAR ADCs. Many of these ADCs also utilize pipelining or time-interleaving. All of the top 5 Walden FOMs published in ISSCC are SAR ADCs.

The Schreier FOM is presented in Fig. 9. This FOM is more suitable for higher resolution ADCs compared to the Walden FOM, since the Schreier FOM assumes that the circuit is limited by thermal noise [5]. Thus, for higher resolution designs, which are most likely limited by thermal noise, the power trade-off is more realistic. This is one of the reasons why different types of FOMs are needed – various ADCs are simply too different from each other. Similarly to the Walden FOM, the state-of-the-art in terms of Schreier FOM is represented by the envelope curve in the figure. In 2017, the state-of-the-art for an ADC with 100 megahertz sampling frequency is

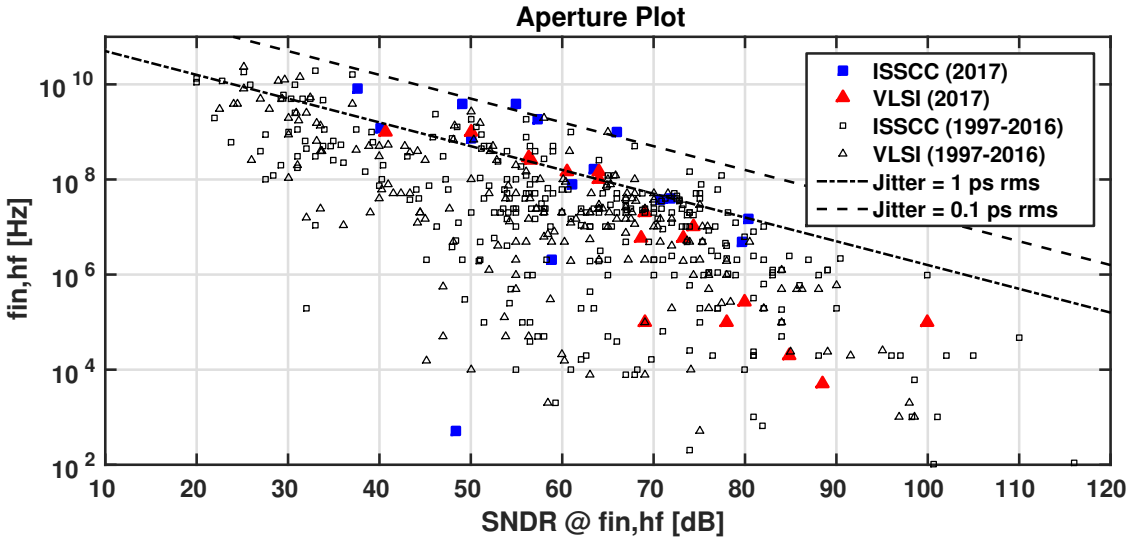


Figure 7: Aperture plot illustrating the speed-resolution trade-off of ADCs. [5]

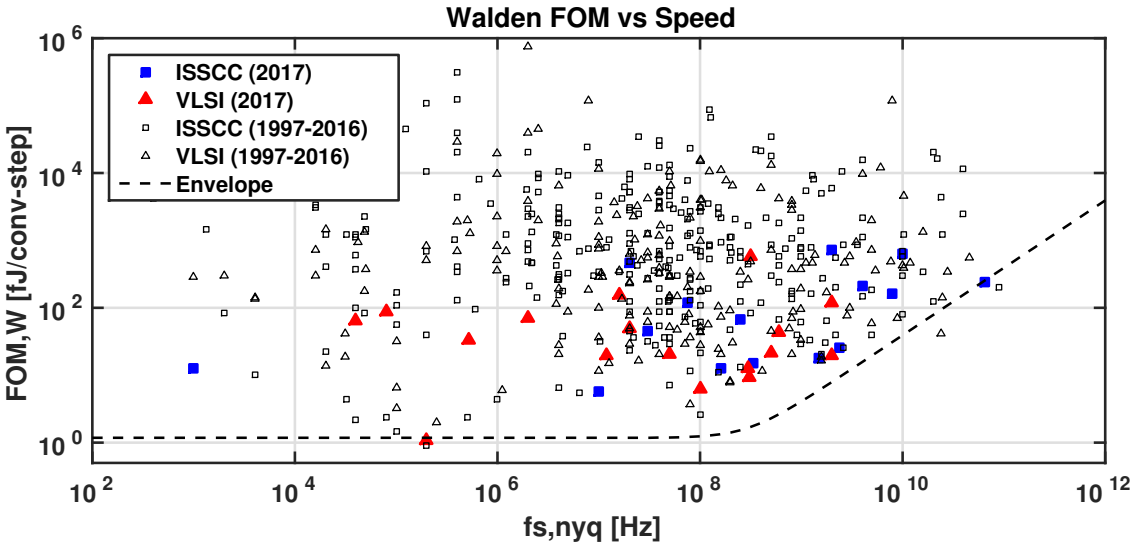


Figure 8: Walden figure-of-merit plotted agains the Nyquist sampling frequency of the ADC. [5]

approximately 177 dB. Once again, SAR ADCs are well represented in this FOM with 3 out of the top 5 ISSCC ADCs being SAR. The two remaining ADCs are $\Delta\Sigma$ converters.

Based on these figures, the SAR architecture seems to produce the highest performing ADCs today. However, this does not imply that other ADC architectures would become redundant. Plenty of applications require ADC metrics that are not favorable for SAR ADCs, such as very high resolution or very wide bandwidth. Additionally, the power consumption plays a key role in the FOMs, and thus the

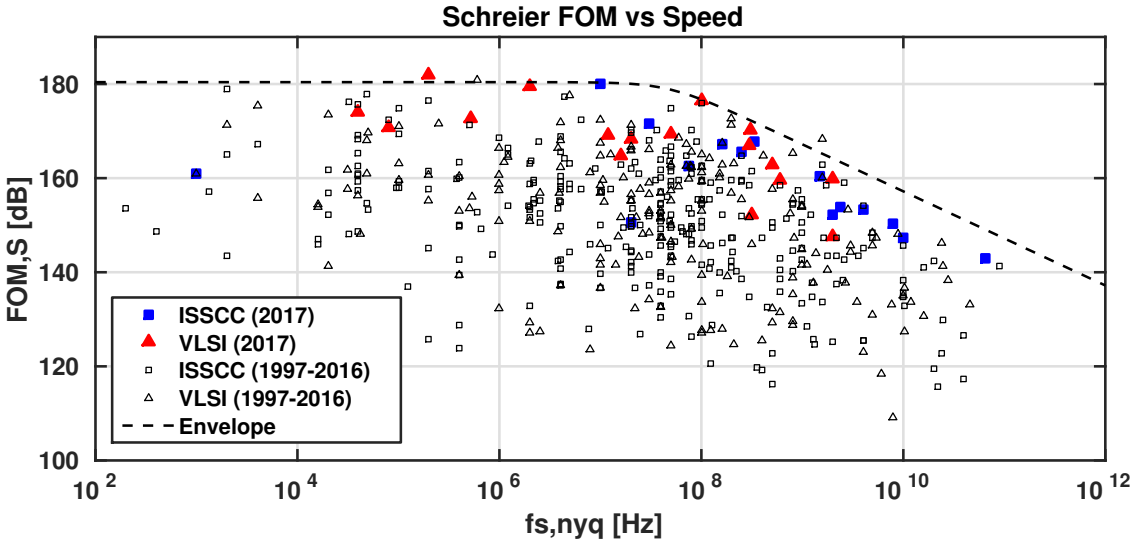


Figure 9: Schreier figure-of-merit plotted against the Nyquist sampling frequency of the ADC. [5]

SAR ADCs produce impressive values. It should be noted that the way the published ADCs are measured and reported may vary significantly, which can skew the results. For example, the reported power consumption might not include the power consumed by necessary driving or calibration circuits. Consequently, the absolute results of the figures can be interpreted in various ways, but the trends plotted should be indicative of the state-of-the-art performance.

In the ADC survey [5], only three time-based ADCs are reported, excluding SAR ADCs used with a time-to-digital converter [6–8]. The Walden FOMs achieved by these converters are 98 fJ per conversion (2009), 38.2 fJ per conversion (2013) and 210 fJ per conversion (2014), respectively. The first converter has a bandwidth of 300 kHz, the second ADC operates at 70 MS/s and the third ADC has a high sampling rate at 2.2 GS/s. These converters achieve competitive FOMs with time-based architectures.

The recent ADC development trends seem to highlight SAR ADCs. However, pushing the SAR state-of-the-art is becoming increasingly difficult. The SAR resolution is limited by capacitor matching and comparator noise, and the sample rate is limited by the bit-by-bit resolving cyclic algorithm, where the clock frequency has to be the sample rate multiplied by the number of bits [9]. Time-based ADCs could potentially alleviate some of these issues by employing digital-intensive designs. Taking into account the growing interest in time-based A/D-conversion over the past 5 years [10] combined with the increasing performance of digital circuits, time-based converters could approach state-of-the-art FOMs in the near future.

3 Time-Based Analog-to-Digital Conversion

In modern sub-micron processes, voltage domain analog-to-digital conversion becomes increasingly difficult, which is mainly caused by decreasing supply voltages [11]. On the other hand, modern processes allow for reduced signal transition times due to smaller feature sizes. This will improve the performance of digital circuits and improve the time resolution of delay lines. Additionally, area and power consumption of these circuits decreases. [12] Since time-based ADCs employ highly digital designs, time domain A/D-conversion will benefit from technology scaling. This partially explains the increasing interest towards time-domain signal processing [10]. The digital TBADC designs allow for rapid design cycles by utilizing CAD tools and automated design flow, which is important with the growing number of standalone portable systems [13].

3.1 Operation Principle

Time-based A/D-conversion introduces an additional conversion step compared to voltage-domain A/D-conversion. The analog input signal is first converted into a timing signal with a voltage-to-time converter (VTC), in which the information is encoded in the width of the signal pulses. The timing signal resembles a pulse width modulation (PWM) signal. The timing signal is then further converted into digital codes by a time-to-digital converter. In this two-step conversion, the sampling is performed by the VTC and the information is quantized in time-domain by the TDC. Time-based converters can implement many of the well-known voltage-domain A/D-conversion methods, such as pipelining, successive approximation and flash conversion. The key difference is the information being encoded into timing signals instead of voltages before quantization. This introduces additional challenges in the design of these time-based ADCs. For instance, time intervals cannot be stored for processing like voltages can be stored in capacitors. However, methods have been designed for processing time signals in ways that are analogous to the voltage-domain conversion procedures.

3.2 Core Architecture

The two main sections of a time-based ADC are the VTC and the TDC. The top level architecture concept of a time-based ADC is presented in Fig. 10. In the figure, the analog input is converted into the timing signal by the VTC. The two outputs of the VTC, the start- and stop-signals, represent the beginning and ending edges of a pulse. The analog information is encoded in the timing between these two signal edges. The TDC quantizes the time interval and produces coded digital outputs. In this case, the outputs are a binary counter value and a thermometer reading from a delay ring. Finally, the digital codes are encoded into a binary value corresponding to the original analog input. Below the signal names are illustrations of the respective waveforms.

Time-domain A/D-conversion can also be implemented without the VTC. The

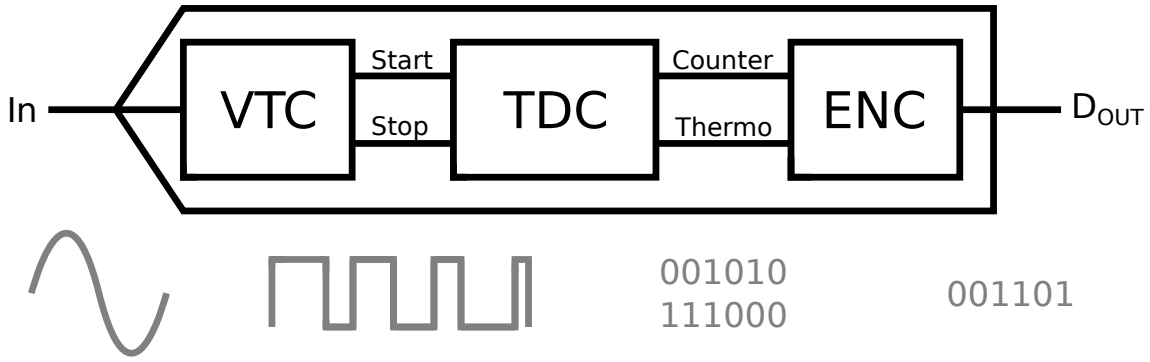


Figure 10: Basic architecture of a time-based ADC.

analog voltage signal can be used to control for example an oscillator frequency, which is then converted into a digital word using a TDC. Here, the voltage-controlled oscillator (VCO) acts as voltage-to-frequency converter. The frequency can be considered as a timing signal with slowly varying pulse width, similarly to the VTC. The VCO can be considered as an integrator, and thus first order noise shaping can be utilized with the VCO. In a VCO-based ADC, the quantized value is the difference between two synchronous samples, and the quantization noise of the first sample will be included in the second sample as well. This essentially $\Delta\Sigma$ modulates the signal and provides noise-shaping. Consequently, VCO-based ADCs can also effectively utilize oversampling in order to achieve improved dynamic range. [11] In addition to noise shaping, VCO-based ADC has inherent anti-alias filtering [14]. Regardless of the attractive properties of VCO-based ADCs, oversampling and oscillator gating limits the bandwidth of the converter, which might make them sub-optimal for wideband applications. Since the focus of this thesis is wideband TBADCs, the use of a VCO as a voltage-to-frequency converter is not inspected further.

3.2.1 Voltage-to-Time Converter

The design of the VTC is critical to the performance of the TBADC. The VTC should allow for high enough linearity and bandwidth for the TDC to operate at the desired point. A commonly used topology for voltage-to-time conversion is the current-starved inverter (CSI) VTC, which is presented in Fig. 11. The VTC operation is based on controlling the discharge current of the capacitor. In the beginning state, the clock and stop signals are low and the ramp capacitor is charged. Once the clock signal transitions from low to high, the capacitor will begin to discharge through the NMOS-transistors of the first stage. The discharging time is proportional to the input voltage at the gate of the NMOS-transistor. Once the capacitor voltage crosses the second inverter threshold, there is a rising edge at the Stop-signal. Thus, the time between the clock and stop rising edges is proportional to the input voltage. Constant voltage Vc at the gate of the second pull-down NMOS-transistor ensures that the inverter remains always operational even with input voltage lower than the threshold of the NMOS.

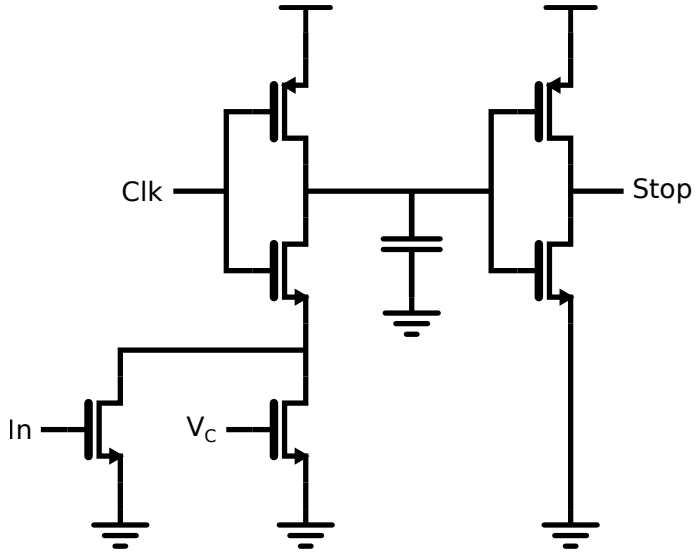


Figure 11: A current-starved inverter -based voltage-to-time converter. [15]

The current-starved architecture provides high bandwidth and low power consumption. Additionally, this circuit doesn't require a dedicated sample-and-hold circuit. However, the common-source connected NMOS-transistor at the analog input is inherently nonlinear. Thus, this topology cannot reach high linearity required for high-resolution time-based ADCs [16]. Current-starved inverter VTC is commonly used in high-speed low-resolution applications [10].

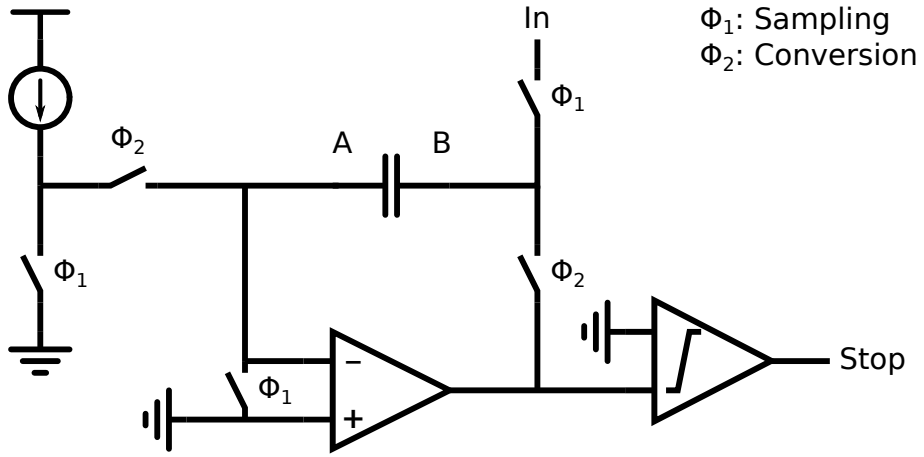


Figure 12: A comparator-based VTC utilizing an operational amplifier and a comparator.

Fig. 12 presents an alternative VTC architecture with improved linearity. The architecture is based on ramp generator and a comparator. The ramp linearity is improved by using an operational amplifier connected in feedback, which will maintain a constant voltage over the bias current source, effectively improving the

linearity [17]. Without the operational amplifier, the voltage over the current source would change together with the ramp voltage, which would introduce nonlinearity to the ramp. Thus, the operational amplifier provides improved linearity for the voltage-to-time conversion.

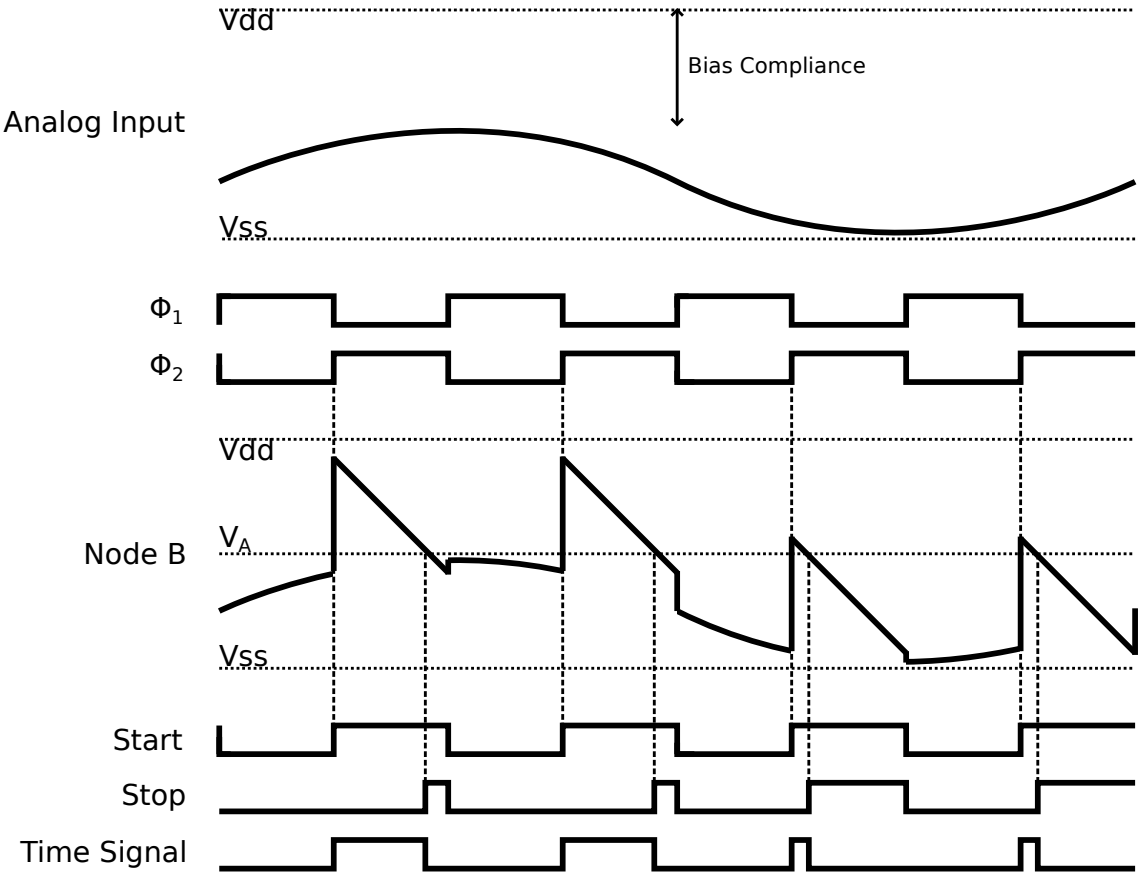


Figure 13: The timing diagram of the operational amplifier VTC (Fig. 12).

The timing diagram describing the operation of the operational amplifier -based VTC is presented in Fig. 13. The topmost waveform is the analog input of the VTC, for which the full-scale range of the input signal is limited to half V_{dd}. The two clock phases Φ_1 and Φ_2 correspond to the sampling and conversion phases according to Fig. 12. The voltage at node B of the circuit is depicted next. During the first phase, the node voltage follows the input. During the conversion phase, the node will see a descending ramp according to the bias current. Since the operational amplifier forces the voltage at node A to be constant, the ramp slopes downwards. Once the voltage of node B surpasses the constant voltage of node A, the comparator output will trigger. The start signal is the inverse of the ADC clock (also Φ_1). The stop signal is the output of the comparator. Effectively, the analog input information is encoded into the time difference between the two rising edges, which is denoted as time signal in the figure.

Utilizing a differential topology will provide larger input full-scale range, as well as

common mode noise rejection. However, the power consumption of a differential VTC is significantly higher than the single-ended topology. For high performance TBADCs, the improved VTC performance is critical, which is why differential architectures are favorable in general.

The ramp signal input relaxes the resolution requirement of the VTC comparator compared to equivalent comparators used in SAR ADCs. The monotonic ramp will cause the comparator to switch close to the ideal switching point even in slightly non-ideal conditions.

3.2.2 Time-to-Digital Converter

The basic function of a time-to-digital converter is to measure the time interval between two asynchronous input signal edges, which are denoted as the start- and stop-signals. The output of the TDC is an n-bit digital word, corresponding to the elapsed time between the two inputs. Time-to-digital converters are used in various applications, such as all-digital phase-locked loops (ADPLL), time-of-flight (TOF) applications, laser range finding [18], frequency synthesis, measurement devices [19] and time-based ADCs. In this thesis, TDCs are discussed in the context of ADCs. Various common time-to-digital converter architecture concepts are presented in Section 4, and a design of a 250 MS/s TDC is presented in Section 5.

3.3 Mathematical Model

The mathematical model for a time-based ADC can be defined in parts. In basic terms, the time-based ADC of this example consists of an ideal sample-and-hold, the VTC and the TDC as a simple quantizer. The transfer function of the VTC can be expressed as a timing signal, which is dependent on the magnitude of the input sample $x[n]$. The input sample is a held value of the analog input signal

$$x[n] = V_{in}(nT_s), \tag{7}$$

where n is the index of the sample and T_s is the sampling clock period. The time signal can be written as a piecewise square-wave function

$$\tau(n,t) = \begin{cases} 1, & t_{begin}[n] \leq t < t_{end}[n] \\ 0. \end{cases} \tag{8}$$

In 8, $t_{begin}[n]$ and $t_{end}[n]$ are time instants of the begin and end points of each pulse, respectively. The expression for the synchronous beginning time instant is

$$t_{begin}[n] = nT_s + t_{sampling}, \tag{9}$$

where n is the index of the sample, T_s is the sampling clock period and $t_{sampling}$ is the time taken by sampling. The end time point of the pulse is correlated to the

input sample magnitude

$$\begin{aligned} t_{end}[n] &= t_{begin}[n] + \frac{x[n]}{V_{fs}} \cdot \frac{T_s \cdot t_{conversion}}{t_{conversion} + t_{sampling}} \\ &= nT_s + t_{sampling} + \frac{x[n]}{V_{fs}} \cdot t_{conversion}, \end{aligned} \tag{10}$$

where $t_{conversion}$ is the portion of the sampling period used for the conversion, $x[n]$ is the magnitude of the input sample and V_{fs} is the maximum input voltage. Now, the complete time signal $\tau(n, t)$ can be plotted by superimposing all pulses corresponding to each sample. The time signal is plotted in Fig. 14. In the figure, the sampling frequency is 250 megahertz and the sampling time is 500 picoseconds. The output of the VTC can be written also as purely discrete-time signal describing the pulse widths of the timing signal

$$\Delta[n] = t_{end}[n] - t_{begin}[n] = \frac{x[n]}{V_{fs}} \cdot t_{conversion}. \tag{11}$$

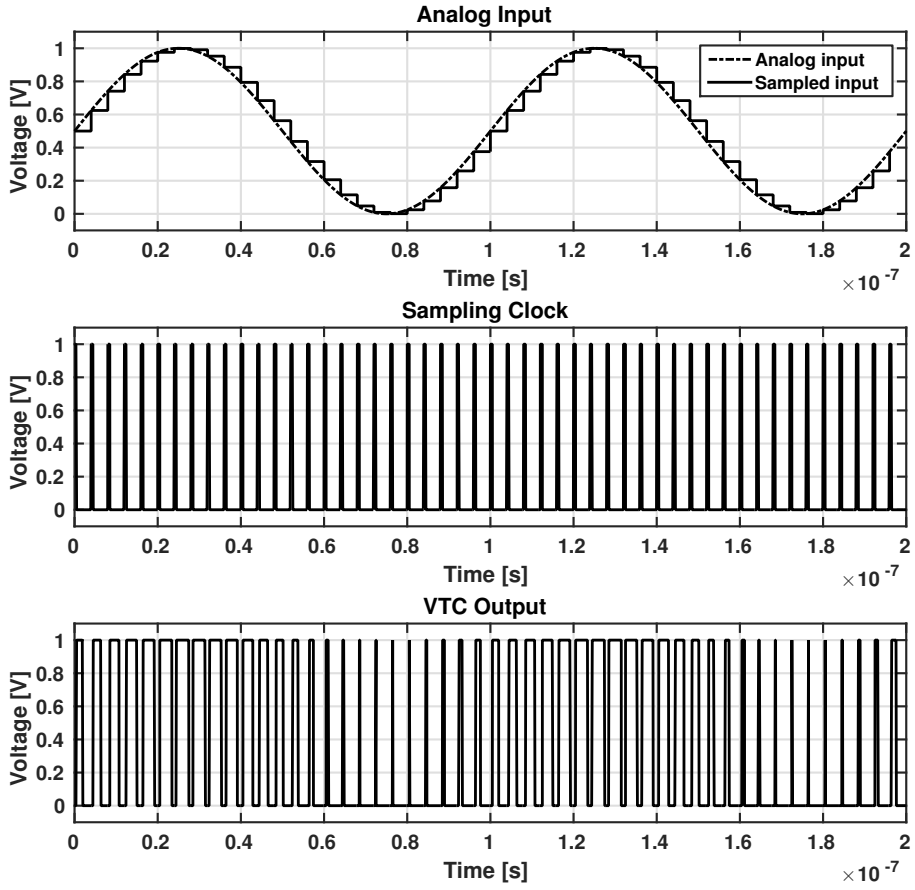


Figure 14: The input to output transfer characteristic of the VTC in time-domain.

The TDC, in this model, is a simple quantizer, which quantizes the width of the time signal pulses. The input of the TDC is the analog pulse width of the time signal.

In this example, the maximum time interval is set to be

$$t_{max} = max(\Delta[n]) = t_{conversion}. \tag{12}$$

The quantization step or the LSB will be the maximum time divided by the number of steps

$$\text{LSB} = \frac{t_{max}}{2^N}, \tag{13}$$

where N is the number of bits in the quantizer. The unit of the LSB is time, although the LSB unit can be converted to voltage as well. The quantizer function can now be written as

$$\begin{aligned} D[n] &= \left\lfloor \frac{\Delta[n]}{\text{LSB}} + \frac{1}{2} \right\rfloor \\ &= \left\lfloor \frac{\Delta[n]}{t_{max}} \cdot 2^N + \frac{1}{2} \right\rfloor, \end{aligned} \tag{14}$$

where the brackets denote rounding down to the nearest integer. This function will map the analog pulse width to an integer value between 0 and $2^N - 1$.

The previously presented equations for the separate stages of the conversion are

$$\begin{aligned} x[n] &= V_{in}(nT_s), \\ \Delta[n] &= \frac{x[n]}{V_{fs}} \cdot t_{conversion}, \\ D[n] &= \left\lfloor \frac{\Delta[n]}{t_{max}} \cdot 2^N + \frac{1}{2} \right\rfloor. \end{aligned} \tag{15}$$

Combining these yields a generic ADC transfer equation, where essentially the input is held and then quantized. The intermediate timing signal simply projects the analog input voltage to time-domain, and in the quantizer the full-scale is determined by the sampling clock frequency instead of the input full-scale voltage.

Variation of the input voltage caused by noise sources can be seen as variation of the pulse widths of the VTC output. Additionally, jitter noise affects the timing pulse widths. In the TDC, the noise contributions are similar - voltage source noise and noise caused by process variations cause shifting of the time signal edges. Timing skew and timing jitter impose a limit on the TBADC resolution, where the spread of the Gaussian distributed jitter ($3 \cdot \sigma_j$) has to be less than the LSB of the ADC. The LSB time of an N-bit ADC is

$$\text{LSB} = \frac{V_{fs} \cdot \rho}{2^N}, \tag{16}$$

where V_{fs} is the input full-scale voltage of the ADC, ρ is the slope of the VTC timing pulse width versus input voltage curve, and N is the number of bits [20]. Now, since the spread of the jitter has to be smaller than the LSB, the upper limit for the resolution is

$$N < \frac{1}{\ln 2} \cdot \ln\left(\frac{V_{fs} \cdot \rho}{3 \cdot \sigma_j}\right). \tag{17}$$

Similarly, maximum tolerable timing jitter can be calculated for a desired resolution. For example, the LSB time of the design presented in Section 5 is approximately 3.4 picoseconds, and the corresponding maximum value of timing jitter is

$$\sigma_{J,max} = \frac{\text{LSB}}{3} \approx 1.1\ ps. \tag{18}$$

The jitter levels for the specific design can be simulated to ensure sufficiently low jitter levels.

4 Time-to-Digital Conversion

4.1 Delay Line

A simplified schematic of a delay line TDC (also known as Flash TDC) is presented in Fig. 15. The circuit consists of the delay line and readout flip-flops. The start signal starts the conversion process with a low-to-high transition. The signal edge will travel in the delay line, setting the note voltages to a high value. As the stop-signal arrives, the flip-flops will register the node values in the delay line. For a wide pulse width at the input, the start signal will travel further in the delay line, which in turn results in more logic high bits at the output. This simple process will effectively quantize the timing signal pulse width. The output bits, denoted as D0-D3, will form a thermometer coded digital output.

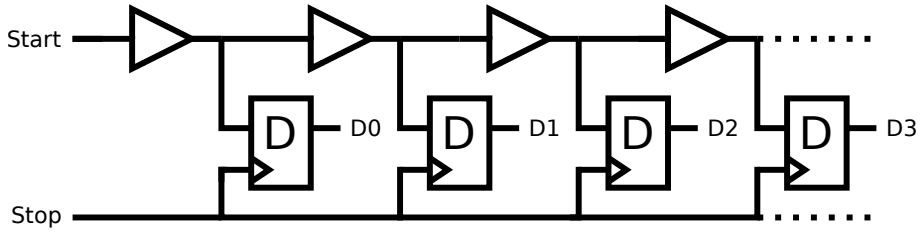


Figure 15: A delay line TDC, consisting of a chain of delay buffers connected to D-flip-flops (DFF).

This architecture has some fundamental limitations. First, the time resolution of the conversion is limited by the propagation delay of a single delay element. Even in modern sub-micron processes, the minimum transition delays are too large for achieving fine time resolution. For example, in modern 28 nanometer process, a typical propagation delay for a delay element is over 12 picoseconds [12]. Second, the required number of delay stages increases exponentially with the desired bit depth. For instance, a 12-bit delay line TDC would require 4096 delay stages. The case is similar in Flash ADCs, where the required number of comparators is 2^N, which makes converters of over 7 bits inconvenient. Additionally, as the number of cascaded elements increases, nonlinearity accumulates due to element mismatches. Thus, the INL for significantly long delay chains degrades [19].

4.2 Ring Oscillator

The limited full-scale (FS) input range and limited linearity of the delay line TDC can be improved by employing a ring oscillator (RO) -based TDC architecture. In this method, a counter is utilized together with an oscillator. In a simple case, the counter by itself can carry out the time-to-digital conversion. As the counter counts the cycles of an oscillator signal, the accumulated counter value will correspond to the elapsed time. Thus, by stopping the counter at a time specified by the stop signal, the elapsed time can be read as the counter value. [11] However, the counter alone

will result in a coarse conversion result, as the period of the signal being counted is relatively large. For example, a 10 gigahertz oscillator signal would result in a 100 picosecond LSB. The time resolution can be improved by encoding the multiple phases of a RO together with the counter.

The example ring oscillator TDC is presented in Fig. 16. The ring oscillator is composed of three inverting delay-stages, where the first stage controls the oscillator operation. When the start-signal is low, the gate blocks the feedback path and prevents the ring from oscillating. Once the start-signal transitions from low value to a high value, the oscillator will start to oscillate. The thermometer bits are read using both rising-edge-triggered and falling-edge-triggered DFFs, due to the alternating phase of the ring nodes. The three physical delay stages of the oscillator will produce six phases, resulting in a 6-bit thermometer output. In binary encoding, the thermometer code corresponds to 2.5 bits.

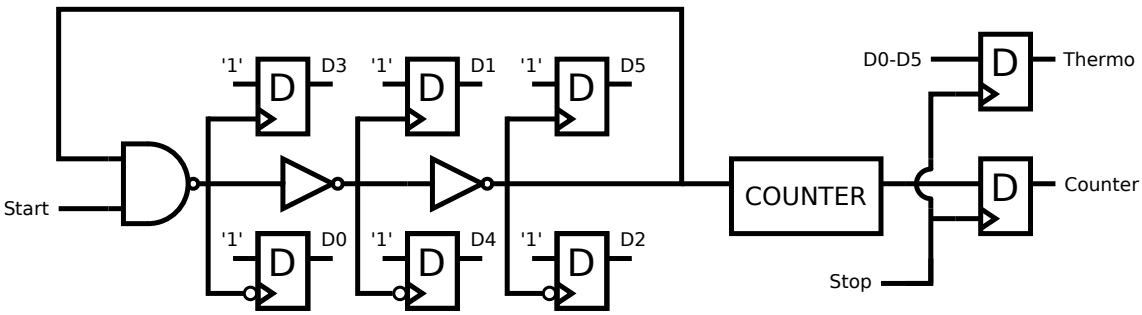


Figure 16: A ring oscillator TDC, consisting of a ring oscillator, D-flip-flops for thermometer code and a counter for coarse conversion.

Similarly to the delay line TDC, the start signal will start the conversion. The RO will start oscillating and the counter counts the oscillator cycles. On each cycle of the ring, the thermometer flip-flops are reset to their initial value (low bit). The low-to-high transition of the start-signal will cause the first node of the ring to have a high-to-low transition. Thus, the falling-edge-triggered DFF will receive the first bit. Each consequent stage voltage will transition in the opposite direction due to the inverting stages. This is reflected in the connection of the DFFs. On the rising edge of the stop-signal, the counter value and the thermometer bits are read using additional flip-flops. The final output code of the TDC is

$$D_{out} = N \cdot Counter + Thermo, \tag{19}$$

where N is the number of phases in the ring (twice the physical stages), $Counter$ is the final counter value and $Thermo$ is the thermometer code value.

This architecture has the same time resolution limitation as the delay line TDC, which is the delay of a single delay element. However, the full-scale input range is not limited by the amount of physical delay stages anymore. The TDC can continue converting for as long as the bit depth of the counter allows. This will result in reduced hardware use compared to the delay line architecture. Using a ring oscillator,

however, might introduce additional problems, such as settling time of the oscillator, resetting of the thermometer flip-flops and matching the delay paths of the counter and thermometer bits. Since the ring oscillator is started and stopped separately for every sample, the settling time of the oscillator causes nonlinearity in the conversion. Furthermore, asynchronous sampling of the counter bits might result in additional errors caused by transition zones in binary counters as well as metastable bits. These issues should be accounted for in the design of the TDC.

A more efficient ring oscillator -based TDC can be implemented by sampling the ring oscillator directly instead of using additional flip-flops. However, this requires a more sophisticated encoding of the sampled oscillator phase. The presented example architecture was chosen, because it is directly derived from the presented delay line architecture.

4.3 Vernier

For the previous two architectures, the resolution of the conversion is limited by the minimum delay of the delay elements. The time resolution can be significantly improved using the Vernier method, where two parallel delay lines are utilized. The delay elements of the two chains have slightly different delays. Consequently, the timing resolution will be defined by the difference of the two delays. [18]

4.3.1 Vernier Delay Line

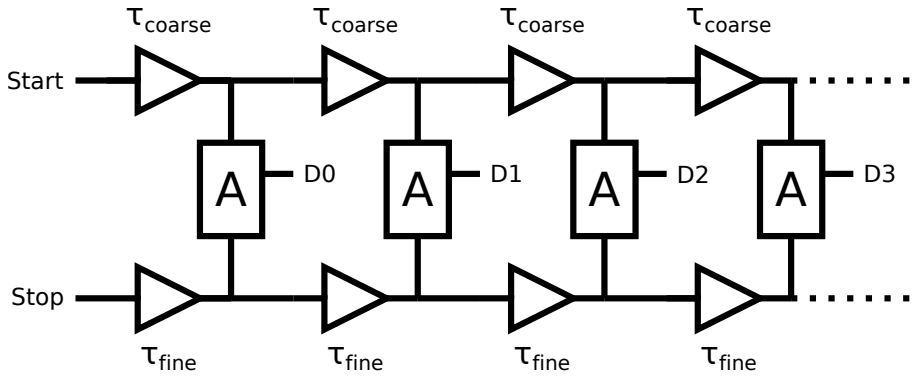


Figure 17: A Vernier delay line. The coarse path delays τ_{coarse} are slightly greater than the fine path delays τ_{fine}.

A schematic of a Vernier delay line (VDL) TDC is presented in Fig. 17. The top delay path is called the coarse delay path, or the slow path, and the lower delay path is the fine delay path, or the fast path. In between the two delay lines, arbiter circuits are used to determine which delay path is leading. The operation of the arbiter resembles the function of a DFF in this case: while the slow path is leading the arbiters output logic high bits, and after the fast path surpasses the slow path, the rest of the output bits are low. The arbiters are crucial for the operation, as they should be able to detect the phase difference of the signals very accurately. The

sub-gate delay results from the fact that the time difference between the start and stop signals diminishes by $\tau_{coarse} - \tau_{fine}$ for every stage of propagation. Thus, the final value of the thermometer code denotes the number of stages taken for the faster signal path to overtake the slow path. Vernier delay line requires even greater number of delay stages compared to a single delay line, since the amount of remaining time to be measured reduces only by a fraction of a single gate delay, instead of the full gate delay, for every stage. Again, it is possible to utilize a ring structure with a counter to increase the input full-scale range.

4.3.2 Vernier Delay Ring

The transition from a delay line to a ring oscillator -based architecture is analogous to the transition from a Vernier delay line to a Vernier delay ring. In Vernier delay ring, the difference in the slow and fast ring delays results in slightly differing oscillation frequencies.

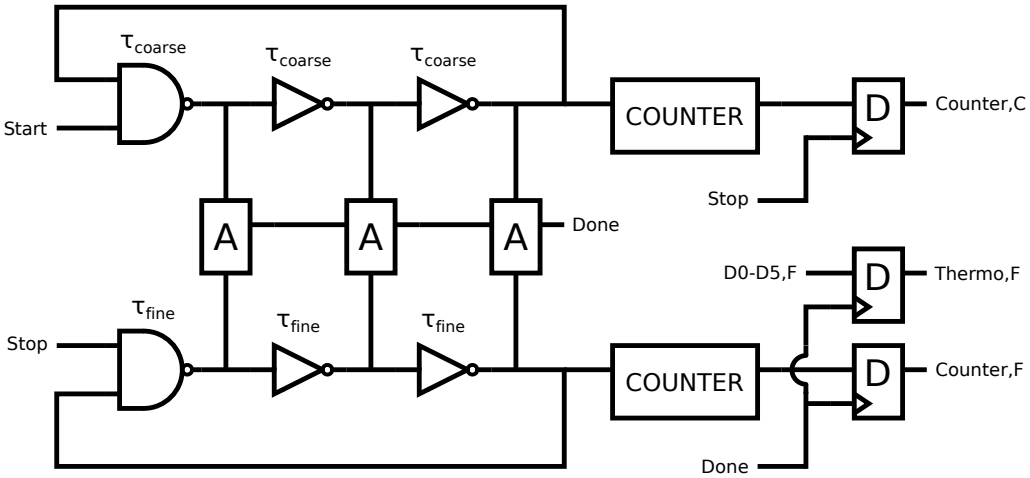


Figure 18: A Vernier delay-ring. The thermometer code readout DFFs are left out of the figure for clarity.

The block diagram of the Vernier delay ring is presented in Fig. 18. The arbiter circuits will be the same as in the Vernier delay line. Counters are utilized for both rings. The circuit operates in two conversion phases denoted as coarse conversion and fine conversion. The coarse conversion phase starts when the start-signal opens the NAND-gate of the coarse (slow) oscillator. The oscillator works in the same way as the single ring oscillator TDC in Section 4.2. However, the thermometer code is not utilized for the coarse ring as the coarse conversion will result from the counter value only. The second phase begins with the rising edge of the stop-signal. This marks the end of the coarse conversion, and the coarse output is registered by the DFFs. The coarse ring keeps oscillating as the fine ring starts oscillating. The situation is now analogous to the Vernier delay line, where the two signals are propagating at different speeds in parallel delay lines. Once the phases of the rings align, the arbiters connected between the rings will detect the event and output a

pulse marked as the done-signal. This signal marks the end of conversion and latch the fine ring thermometer codes as well as the counter value. [21]

The Vernier delay ring TDC has three outputs: coarse counter, fine counter and thermometer code. The measured time can be compiled from these three values and the delay values of the stages as

$$T_{in} = N \cdot Counter, C \cdot \tau_{coarse} + (Counter, F + Thermo, F) \cdot (\tau_{coarse} - \tau_{fine}), \quad (20)$$

where N is the number of phases, $Counter, C$ is the coarse counter value, $Counter, F$ is the fine counter value, $Thermo, F$ is the thermometer code and τ_{coarse} and τ_{fine} are the coarse and fine stage delays, respectively. The digital output code can be derived as the coarse conversion result summed with the fine conversion value scaled with the stage delays

$$D_{out} = N \cdot Counter, C + (Counter, F + Thermo, F) \cdot \frac{\tau_{coarse} - \tau_{fine}}{\tau_{coarse}}. \quad (21)$$

Since the conversion result is sensitive to the ratio of the two delays, the design of a Vernier delay ring architecture requires effective calibration.

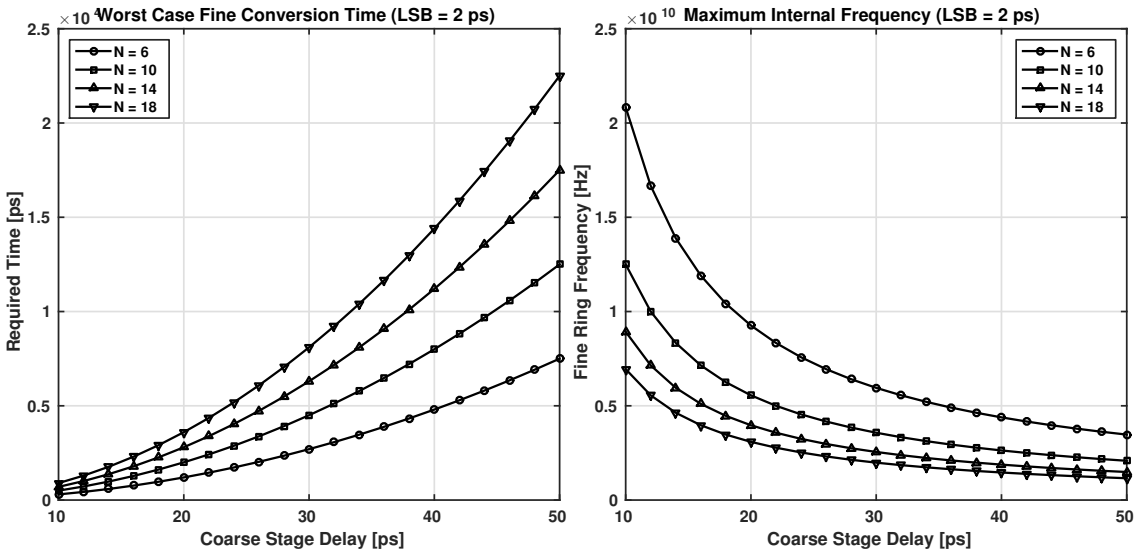
Vernier delay ring TDC offers solutions to the issues of limited time resolution and full-scale range. However, the design of the structure is complex, since the operation is based on the ratio of the delays. Furthermore, the required conversion time is increased compared to a simple delay line or delay ring. In order to minimize the required conversion time, the oscillator frequencies should be maximized. The minimum oscillator frequency will limit the achievable sample rate of the TDC. Additionally, the size of the LSB, which is defined by the difference of the coarse and fine delays, will have significant effect on the total required conversion time, once again limiting the speed and resolution of the converter. In the worst case, the time interval left for fine conversion is one complete period of the coarse oscillator, and the resulting maximum time required by fine conversion is

$$t_{F,max} = \frac{T_{coarse}}{LSB} \cdot \tau_{coarse} = \frac{N \cdot \tau_{coarse}}{LSB} \cdot \tau_{coarse} = \frac{N \cdot \tau_{coarse}^2}{LSB}, \quad (22)$$

where T_{coarse} is the period of the coarse ring, N is the number of phases in the rings, LSB is the desired least significant bit and τ_{coarse} is the coarse ring stage delay. The equation is presented as a graph in Fig. 19a. On the other hand, the frequency of the oscillators depends on the absolute value of the delays and the number of delay stages. The maximum frequency of the system is determined by the fine ring as

$$f_{max} = \frac{1}{T_{fine}} = \frac{1}{N \cdot \tau_{fine}}. \quad (23)$$

The trade-off is visualized in Fig. 19b. Based on 22, the total required conversion time for a 14-phase (7-stage) with 2 picosecond LSB and 20 picosecond coarse stage delay is presented in Fig. 20. Based on these values and 23, the required oscillator frequency is 4 gigahertz. In order to reach a full-scale range of 10 bits with these values, the theoretical limit of the sample rate of the TDC is approximately 200 megahertz.



(a) (b)

Figure 19: (a) The worst case time required by the fine conversion phase as a function of the coarse stage delay. (b) The maximum fine ring frequency as a function of the coarse stage delay. Here the LSB is set to 2 picoseconds and the number of phases is denoted by N.

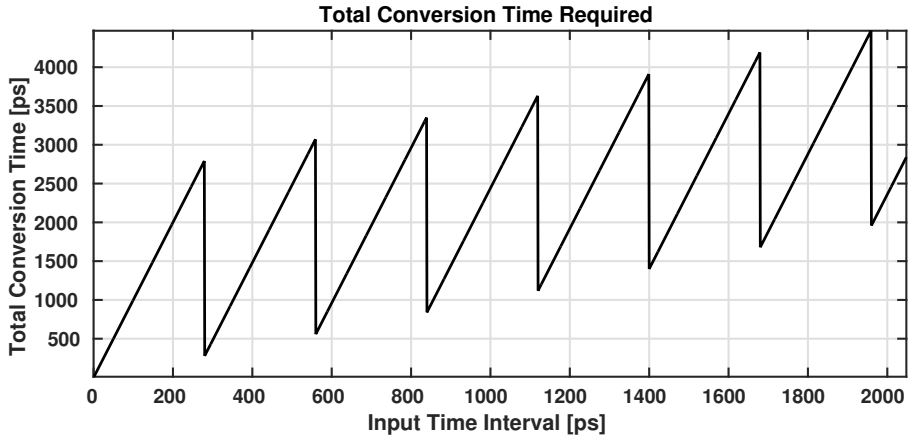


Figure 20: The total required conversion time, when the number of phases is 14, LSB is 2 picoseconds and coarse stage delay is 20 picoseconds.

4.4 Pipeline

In addition to the Vernier method presented in Section 4.3, sub-gate delays can be achieved using multi-phase conversion, such as pipelining. The basic concept of pipelining is presented in Section 2.3.4. Pipelining can be utilized in time-to-digital converters as well. In time-domain pipelining, the gain between the stages is implemented by stretching the timing signal pulse using a time amplifier (TA). The

pipeline stages can be implemented as simple delay lines, each resolving for example 2.5 bits. The coarse conversion of each stage will leave a residue time pulse that gets amplified between the stages. The operation is analogous to voltage-domain pipelined ADCs.

With time-domain multi-phase conversion, the handling of data between the stages can be challenging. In voltage-domain converters, the voltage can be stored in a capacitor for example, whereas in TDCs the residue time pulse has to be generated in real time. Thus, implementing synchronous multi-phase converters in time-domain is difficult. Virtual timing pulse storing implementation techniques have been implemented using gated delay lines [22] [23]. Storing the time pulse between the stages allows for synchronous pipeline operation in TDCs.

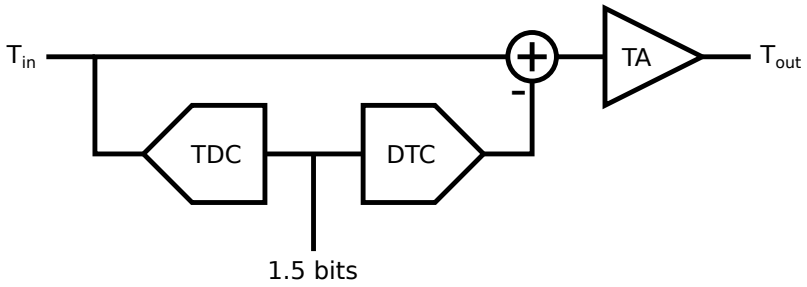


Figure 21: A 1.5-bit time-domain pipeline stage.

A simple concept of a time-domain pipeline stage is presented in Fig. 21. The input pulse is quantized using a coarse TDC, and the output bits are stored in a shift register. The converted portion is subtracted from the original pulse and the residue is amplified with a time amplifier. The residue generation can be implemented with multiplexing or gating the delay line, for example [23].

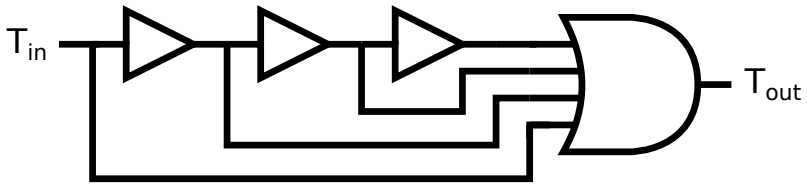


Figure 22: A pulse-train time amplifier. [23]

The time amplification can be realized using voltage-to-time converters, where the bias current of the VTC is controlled based on the generated residue. Using a VTC as a time amplifier requires constant A/D and D/A conversion. However, digital methods for amplifying timing pulses have been presented. In the case of a gated delay line, where the supply voltages of the delay elements can be switched on and off, a pulse-train time amplifier (PT-TA) can be used. An example PT-TA with a gain of four is presented in Fig. 22. As the short input pulse T_{in} enters the OR-gate, the output voltage T_{out} will follow the narrow pulse. The second input will receive the same pulse delayed. The delay has to be larger than the width of the

pulse. Once the second input receives the pulse, the OR-gate will output the pulse again. Thus, the output of the OR-gate will see four times the single narrow residue pulse. The train of four pulses can be used to switch the supply voltage of a delay line. Consequently, each time the supply voltage of the delay elements gets activated, the pre-set signal edge will propagate in the delay line. After the four pulses are loaded into the delay line, the supplies can be switched on and the output pulse width will be the inverse of the sum of the four input pulses. Thus, the pulse-train time amplifier together with the gated delay line implements the residue generation and time amplification. [23]

In multi-phase conversion, the mismatches between the conversion stages will introduce nonlinearity to the output. Also, any errors in early stages will get amplified as the data moves along the pipeline. Consequently, extensive error correction and calibration are required. However, pipelined TDCs are capable of achieving good FOMs, since the sub-gate delay resolution is achieved with the expense of latency.

4.5 Successive Approximation

Successive approximation register ADCs perform extremely well in terms of Walden FOM [5]. The successive approximation method, presented in Section 2.3.2, can also be applied to time-based conversion. Similarly to the pipeline ADC case, the time-domain SAR operation is directly analogous to the corresponding voltage-domain operation. In time-domain, the data values are compared by determining which of the two signal edges is leading, instead of using a voltage comparator. A SAR TDC can be implemented either as a chain or as a cyclical structure, like most multi-phase conversion methods.

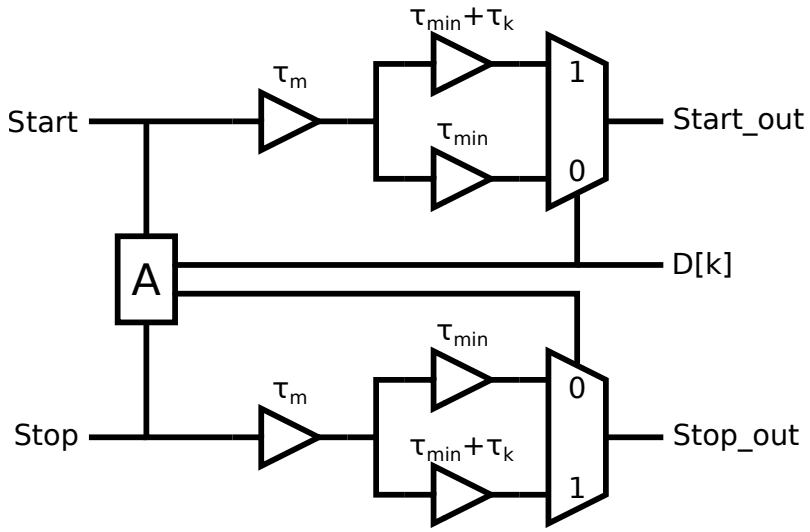


Figure 23: A 1-bit SAR TDC stage. [24]

A chain type SAR TDC can be implemented by cascading delay stages presented in Fig. 23. In the figure, τ_{min} is a delay element with the minimum delay. The

delay $\tau_{min} + \tau_k$ refers to an adjustable delay line with a power of two multiple higher delay depending on the bit k. The output bit of the stage is denoted as $D[k]$. The operation is based on comparing the two input signal edges, start and stop, in order to determine which one is leading. The comparison is performed with an arbiter circuit. Depending on output of the arbiter, the multiplexers adjust the delay paths seen by the signals. The multiplexers are controlled in a way that the leading signal will see the longer delay. The delay τ_m is added onto both delay paths to accommodate the metastability time of the arbiter.

The adjustable delays of consequent stages are designed so that each stage has half the delay of the previous stage. Thus, the LSB stage will have the smallest delay, and the delay is doubled for every bit. In the first stage, the leading signal will get delayed by $2^k \cdot \tau_{min}$ in respect to the lagging signal. The delay is half of the input full-scale range. In the second stage, if the same signal is still leading, it will get shifted further back. Otherwise the new leading signal will get shifted. The arbiter output gives the output bit k as well. This arrangement effectively implements the binary search algorithm for the time difference between the two edges, which corresponds to the SAR conversion algorithm.

Implementing a SAR TDC using the presented method solves the issue of large number of elements required by delay line TDC. However, the SAR chain has some drawbacks. The required delays are quite large, which in turn necessitates extensive calibration hardware and results in increased area use. Also, the delay matching between stages is critical. Furthermore, the speed of the conversion is reduced by the arbiters and the required multiplexing. The SAR TDC could also be implemented as a cyclical architecture by employing similar logic and adjusting the delay with for example capacitor banks [25].

4.6 Phase Interpolation

In addition to the previously presented methods, sub-gate delay resolution can also be achieved using phase interpolation. Phase interpolation can be applied to either delay lines or ring oscillators. The interpolation can be implemented using resistor strings or buffers with shorted outputs. The aim of phase interpolation is to increase the number of available phases, while keeping the number of physical delay stages unchanged, which will effectively reduce the delay between consequent stages, and thus improve the time resolution.

A two-fold phase interpolation is presented in Fig. 24. The interpolation is implemented using buffers with their outputs connected together. In the figure, the horizontal buffers represent the actual delay elements of the delay line or ring oscillator, and the vertical buffers implement the interpolation. The phases Φ_0, Φ_1 and Φ_2 are the physical phases of the chain, and the phases denoted with Φ_d represent the interpolated phases, which are delayed in respect to the main phase. Ideally, the phase Φ_{1d} would be located in the mid-point between Φ_1 and Φ_2. Due to the connected outputs of the interpolation buffers, the delayed phase is shifted less than one buffer delay, which allows the time separation to be lower than the minimum gate delay of the process. The degree of interpolation can be increased beyond two.

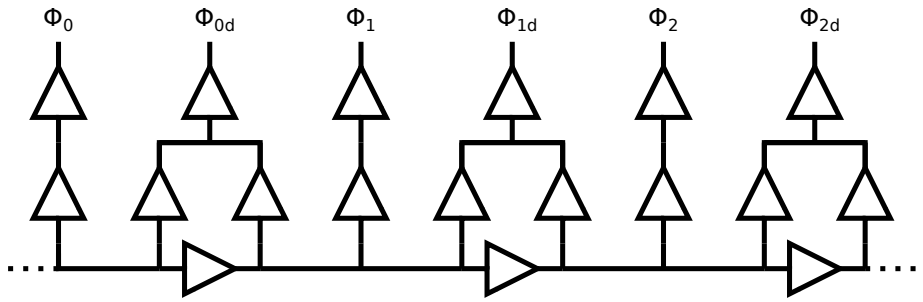


Figure 24: Two-fold phase interpolation implemented with buffers. [26]

The interpolated phases are not perfectly comparable to the physical phases. The outputs of the interpolation buffers are working against each other in the beginning of the transition, while at the later half of the transition they are contributing together to the transition. This results in a mismatch in the slew rates between the interpolated and real phases. Additionally, connecting buffer outputs in this manner allows a momentary connection between the supply rails, which results in an increase in power consumption. [26] The mid-point of the transitions can be aligned between two consequent real phases, but the rise and fall times cannot be matched. Furthermore, this type of interpolation does not account for PVT variations. Like stated previously, phase interpolation can be utilized in conjunction with simpler T/D-conversion methods, such as the ring oscillator or the simple delay line. The presented method can be used to achieve improved time resolution, and thus an improved figure-of-merit.

4.7 Summary

The TDC architectures and conversion methods presented in this section are simplified examples. They illustrate various more or less common methods for quantizing time signals. In general, the delay line type architectures work well for lower resolution designs. The lower resolution allows for simpler design with reasonable area consumption. The simplest delay line TDC (Section 4.1) is commonly used and it is one of the simplest architectures to implement. The delay line TDC can be utilized in multi-phase converters as a low-resolution conversion stage as well. The ring oscillator -based architecture (Section 4.2) offers improved full-scale range with efficient hardware use. The complexity of the implementation increases slightly, as the added counter has to aligned with the LSBs in time-domain (in this example at least). Without any additional changes, the resolution of conversion remains the same. The rest of the presented architectures provide techniques for improving the time resolution. The Vernier delay line (Section 4.3.1) and delay ring (Section 4.3.2) TDCs are suitable for high-resolution applications. However, the required conversion time is increased compared to the delay line architecture, which limits the maximum speed of the conversion. Multi-phase conversion architectures such as the pipelined TDC (Section 4.4) offer improved time resolution by executing multiple lower resolution

conversion steps in cascade, which results in increased overall conversion rate and resolution. The SAR TDC (Section 4.5) resolves one bit at a time, compared to the one LSB at a time of the direct delay line architecture. In theory, due to the more effective algorithm, the area consumption should be decreased. However, the SAR TDC will require significantly large programmable delays for higher number of bits. Phase interpolation is a relatively low complexity way of improving the time resolution of the flash type TDCs. Additionally, negative skew can be employed in ring oscillator -based designs to further improve the time resolution, which is discussed in Section 5.

Table 1: Qualitative comparison table for different TDC architectures.

	DL	RO	VDL	VDR	PL	SAR	PI
Speed	3	3	1	1	5	2	4
Time Resolution	3	3	5	5	4	3	4
Complexity	3	4	4	5	5	4	4
Power	3	3	4	4	4	4	4
Area	3	1	4	2	2	4	2

A qualitative comparison of the presented TDC architectures is presented in Table 1. The columns of the table correspond to the architectures presented in this chapter, while the rows display qualitative metrics of each inspected design. The first column in the table is used as a comparison reference. The metrics for the simple delay line TDC are set to 3, and the scale ranges from 1 to 5. The values given for the other architectures denote their qualitative performance in comparison to the simple delay line. The values do not represent any quantitative measurements and they are purely relative and subjective. The speed metric describes the architectures applicability to higher data rate converters. The time resolution row compares the achievable LSB time interval between the architectures. For these two metrics, higher values indicate better performance. The last three metrics describe the inherent complexity of the design, as well as the power and area consumptions, respectively. The power and area metrics indicate the overhead caused by the design architecture in the respective metrics. For the final three rows, lower values correspond to more favorable characteristics. The phase interpolation column refers to a ring oscillator -based architecture used together with phase interpolation.

Based on the comparison, the Vernier architectures provide good performance in low-frequency applications, while the pipeline architecture appears favorable for high-speed applications. However, the complexity of the implementation is significant for these converter types. A potentially simpler way of achieving competitive FOMs is the ring oscillator -based architecture used together with some form of phase interpolation. Thus, a RO-based architecture was chosen for further examination in the following section.

5 Time-to-Digital Converter Design

5.1 Design Requirements

The target performance of the device is 10 bits and 250 MS/s with minimized power and area consumption. The presented TDC is designed for use as a part of a time-based ADC, which consists of a VTC and a TDC according to Section 3.2. The design of the VTC is not presented in this thesis. The presented TDC is designed for high parallelism, as the TDC design should be usable in a wideband ADC application by employing time-interleaving.

In terms of the overall performance, the TDC has three main parameters: time resolution, full-scale range and sampling performance (error mitigation). The time resolution together with the full-scale range defines the maximum speed and resolution of the converter, while the sampling error mitigation is critical for maximizing dynamic performance and linearity. These are the key aspects considered in designing the TDC.

In order to reach 10-bit resolution at 250 MS/s sample rate with 87.5 % conversion duty cycle (3.5 nanoseconds maximum conversion time out of 4 nanosecond period), the time resolution has to be approximately 3.4 picoseconds. However, the gate delay in modern processes is approximately 10 picoseconds [12]. Thus, a simple gate-delay limited TDC architecture will not suffice. Additionally, for 10-bit resolution the chain-based TDC architectures discussed in Section 4 become increasingly nonlinear. Oscillator-based architectures are more suitable for larger range TDCs.

5.2 System Design

5.2.1 Architecture

As discussed in Section 5.1, the three main parameters of the design are time resolution, full-scale range and error resilience of the sampling. A ring oscillator -based TDC architecture is chosen in order to achieve linear operation up to 10-bit resolution with minimized area. The required sub-gate time resolution is achieved using a multipath topology in the oscillator, and a high-speed counter is employed in order to increase the full-scale range of the TDC. The sampling of the oscillator and counter is performed using custom high-performance flip-flops, and an error correction algorithm is implemented for providing error-free counter sampling.

The system-level diagram of the TDC is presented in Fig. 25. The oscillator is pictured on the left hand side of the figure. The 32 phases of the oscillator are connected to a multiplexer, which selects a single phase-tap as the clock signal of the 5-bit gray counter. The instantaneous state of the oscillator and the value of the counter are sampled by the sense-amplifier flip-flops (SAFF) at the start and stop time instants. These sampled values are encoded into binary, and an error correction algorithm is implemented for correcting counter sampling errors. The final 10-bit digital output code is formed by taking the difference of the begin and end states for both the oscillator and the counter, and combining the two quantities.

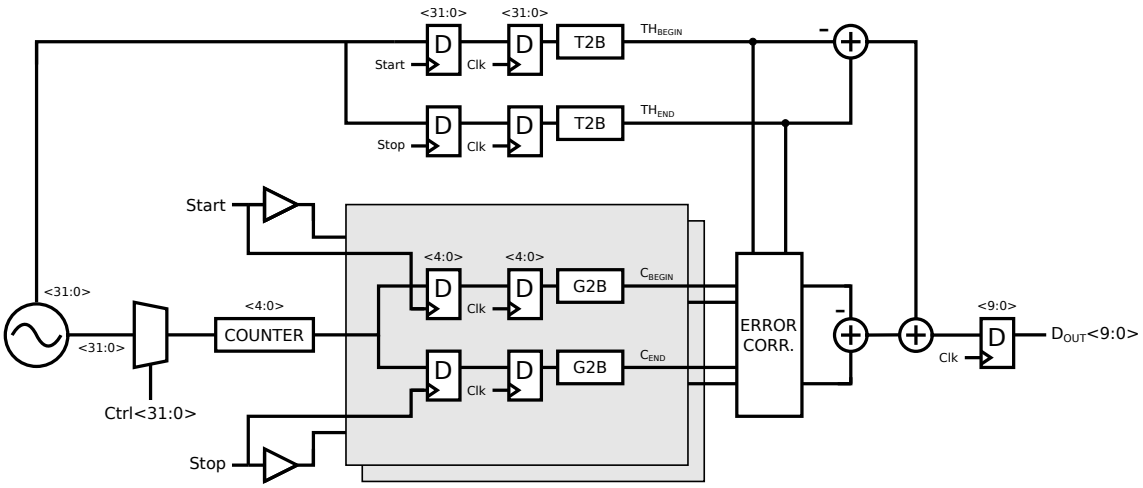


Figure 25: Block-level design of the TDC.

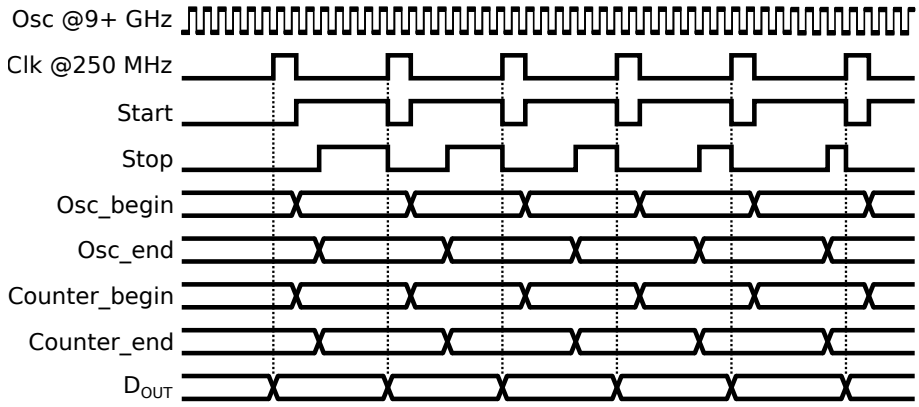


Figure 26: Timing diagram describing the TDC operation.

Fig. 26 presents the timing diagram of the TDC operation. The operation speed of the TDC is determined by the sampling clock, which in this case is designed to have 12.5 % sampling duty cycle and 87.5 % conversion duty cycle at 250 megahertz frequency. This is performed in order to maximize the input full-scale range for the TDC. In a generic case, where the clock signal has 50 % duty cycle, the full-scale input range would simply be reduced. Thus, the operation of the TDC does not critically depend on the duty cycle of the TDC clock.

The operation principle is the same as the ring oscillator architecture presented in Section 4.2. The oscillator phase keeps increasing over time and as the phase cycle resets, the counter is incremented. The main difference is the use of a free-running oscillator instead of one with an enable gate that is used to start and stop the oscillation. Now, the phase of the ring and the counter is sampled twice, and the elapsed time interval is the difference between the begin and end states as

$$D_{out} = N_{phases} \cdot (Counter\_end - Counter\_begin) + (Osc\_end - Osc\_begin), \quad (24)$$

where N_{phases} is the number of oscillator phase taps (32 in this case), and $Counter\_begin$, $Counter\_end$, Osc_begin, Osc_end are the counter and oscillator begin and end states, respectively.

Potentially more optimal results could be reached for non-time-interleaved ADC structure by either using a VCO as a voltage-to-frequency converter or by using a gated ring oscillator (GRO). In both cases, the end point of one sample would act as the begin point of the next sample, which enables the inherent noise shaping quality of the VCO to be utilized. However, since wideband operation is the goal of this design, the adjacent samples are kept isolated from each other. This allows the oscillator and counter to be shared between parallel sampling sections. Thus in time-interleaved systems, the time-interleaved branches can share the common TDC core (oscillator and counter). Consequently, this should result in decreased power and area consumption. Another advantage of asynchronously sampling a free-running oscillator is that it effectively spreads out any mismatch errors of the oscillator delay elements, which converts the mismatch errors into white noise [10].

5.2.2 Operating Conditions

The frequency relationship between the oscillator and the sampling frequency, together with the counter and backend bit depths, introduces limitations on the TDC operation. In this design, the output is 10 bits with 5 bits extracted from the oscillator phase and 5 bits extracted from the counter. In order to prevent clipping of the output, the 5-bit counter cannot count more than one full bit sequence during a conversion step. Given the desired sampling frequency, the frequency relationship provides the maximum oscillator frequency for the oscillator. The sampling time of the VTC is fixed at 500 picoseconds, and the rest of the period is used for conversion (in an ideal case). Thus, the maximum conversion time is 3.5 nanoseconds for 250 megahertz sampling frequency. Consequently, the maximum allowed oscillator frequency in this case is

$$f_{osc} \leq \frac{2^N}{T_{conv}} \approx 9.14 \text{ GHz}, \tag{25}$$

where N is the counter bit depth and T_{conv} is the maximum available conversion time. In reality, the VTC will not be able to output the maximal time interval. Rather, there will be overhead time taken by the VTC itself, which relaxes the oscillation frequency requirement. However, a higher frequency is then required in order to maximize the full-scale range of the TDC. The ideal operating case is when the oscillator frequency is as high as possible, while still obeying the above condition. Lower oscillator frequency will cause slight loss of output range.

The above condition is "artificially" created by limiting the TDC hardware to 10 bits. The clipping condition could be relaxed by adding one more bit to the counter and backend. However, in nominal conditions, only a portion of the hardware would be used, which is why in this design the oscillator is assumed to be locked to a specific frequency. A future improvement would be to increase the range to prevent clipping even in extreme operating conditions.

The error correction algorithm has a few constraints for error-free operation, which

are defined by the oscillator frequency and the counter path delay. The constraints are met by adjusting the counter path delay with the multiplexer, and by adjusting the sampling delay with delay buffers drawn on the start and stop paths in Fig. 25. The operation of the algorithm, as well as the constraints are discussed in 5.3.4.

5.2.3 Matlab Model

A Matlab model is implemented describing the functionality of the TDC. The model consists of a quantized ramp signal representing the incrementing oscillator phase and voltage-to-time conversion in accordance to 9 and 10. The model seeks the start and stop sampling time instants and reads the corresponding oscillator phases. The beginning phase is subtracted from the end phase and the integer number of full cycles gives the counter value. The model has finite accuracy due to the significant size needed for the time vectors. The finite time-step will introduce additional quantization error to the simulation.

The spectrum of a single-tone sinusoid is plotted in Fig. 27. The sampling rate used in the model is 250 megahertz. The oscillator frequency is 9.12 gigahertz and the maximum VTC output interval is 3 nanoseconds. Consequently, the range of output codes is 875 codes or 9.77 bits. Without any added noise sources, the model gives an SNDR of approximately 57.5 decibels, which corresponds to an ENOB of 9.25. The ideal model has roughly 0.5 bits of degradation in the ENOB compared to the theoretical maximum.

Input uncorrelated spurious tones can be observed from the figure. The spurs are a consequence of the interaction between oscillator and sampling frequencies. With specific ratios, the sampling appears to downconvert tones of the oscillator frequency into the signal band. Since the simulation is noiseless, the aliased tones will cause a spur. In the presence of noise sources, the tones should smear into the noise floor.

Another source of non-idealities in the simulation are the combined quantization errors of the start and stop samples. Each sample has its own quantization error, which is analog by nature. When the two samples are subtracted, the two errors are combined. In a portion of the cases, the magnitude of the combined error surpasses one LSB, which appears as variation of the LSB. In the ideal model, this phenomenon is also present. The error is correlated with the ratio of the oscillator frequency and the sampling frequency, as well as the magnitude of the input sample. The presence of the error implies that perfectly monotonous operation cannot be guaranteed. In a signal-processing application, however, this should not cause issues as the signal characteristics are still preserved.

The single-shot precision σ_{ss} of the TDC model is presented in Fig. 28. In the simulation, a DC input value is sampled 8192 times, and the single-shot precision of the TDC for each input is calculated as the standard deviation of the digital output codes. This process is repeated over the whole input range. Ideally, the standard deviation would be zero, but due to the above mentioned combined quantization error, this is not the case. The single-shot precision has a periodic correlation to the input amplitude. The peaks of the single-shot precision are repeating every LSB. The maximum deviation points correspond to values, for which the sampling instants

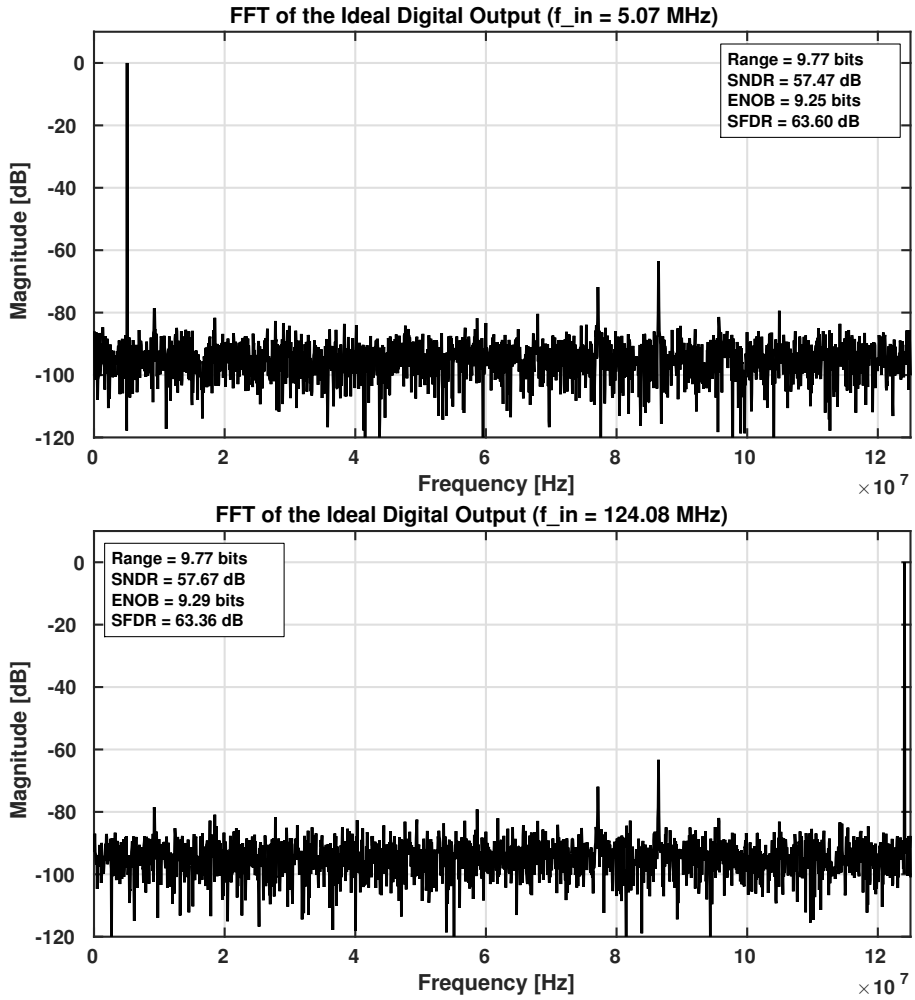


Figure 27: FFT of the digital output for a single-tone sinusoid produced by the ideal Matlab model. In the top figure, the input frequency is 5 megahertz, and in the bottom figure, the input frequency is 124 megahertz. The sampling frequency is 250 megahertz.

of the start and stop samples are located close to the center of the LSB time bin (on average). Thus, the output LSB will flutter between two values more frequently because of the larger quantization error. The minimum points correspond to input levels, for which the samples have low quantization errors on average. The average single-shot precision is close to 0.5 LSB over the whole range. For this architecture, even in the close-to-ideal simulation, the single-shot precision is non-zero. However, in a more realistic system, noise sources and non-idealities cause limited precision regardless.

The correlation of the dynamic performance and the input frequency is plotted in Fig. 29. The Matlab model is simulated with various introduced noise sources and the SNDR is calculated over a range of input frequencies. Without any noise sources, the SNDR is constant over the whole bandwidth. Likewise, the effect of phase noise

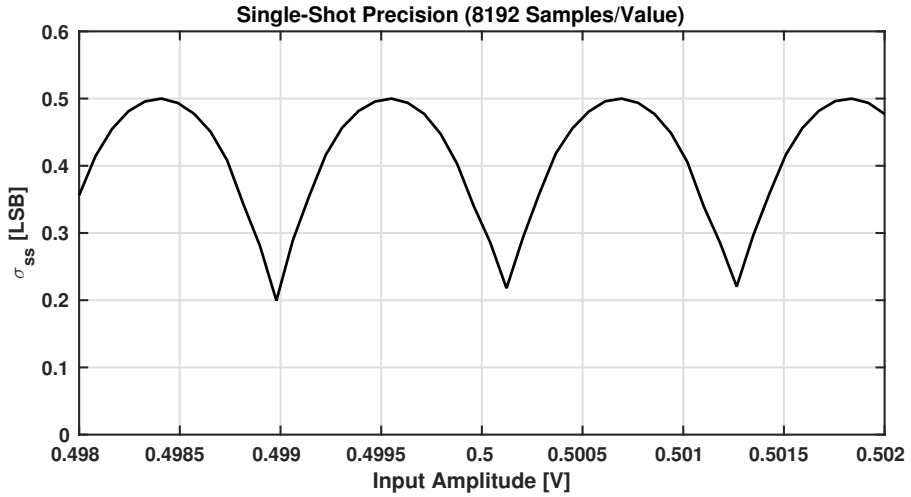


Figure 28: The single-shot precision (standard deviation σ_{ss}) of the digital output code as a function of the input DC amplitude.

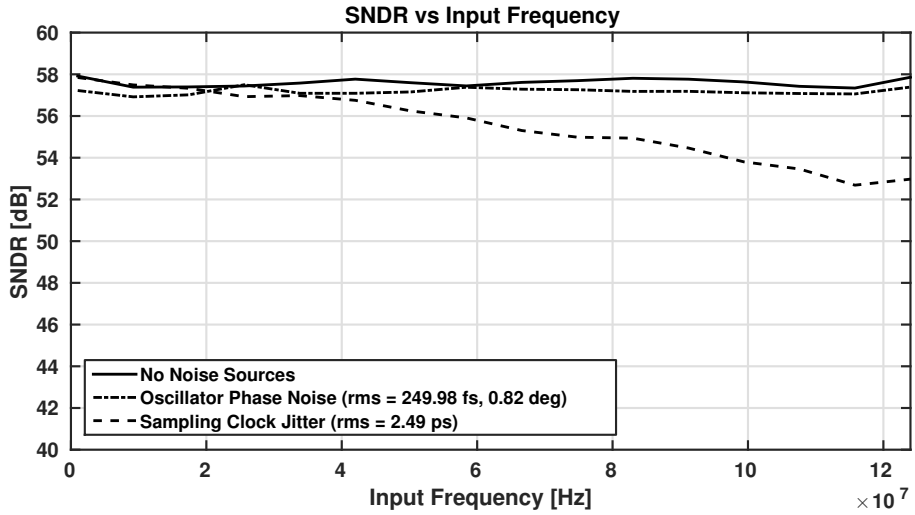


Figure 29: The dependence of the ideal model SNDR on the input signal frequency in the presence of noise sources.

is not input dependent. Sampling clock jitter, however, is input dependent, which is the case for all ADCs. With higher input frequencies, the signal changes its value faster. Thus, when the sampling instant is deviated from the ideal, a larger error can occur due to the sampling clock jitter. This can be clearly seen as steady degradation in the SNDR when the input frequency is increased. It should be noted that the sampling clock jitter does not cause any input correlated noise in the TDC, since the TDC simply quantizes the already sampled value.

5.3 Pre-Layout Design

5.3.1 Multipath Ring Oscillator

As mentioned in Section 5.1, one of the main metrics of the TDC is the time resolution or the LSB time interval. In the selected RO-based architecture (Fig. 25), the metric is defined by two parameters: the frequency and the number of delay stages of the oscillator. In a ring-oscillator, the relationship between the frequency and number of stages is

$$f_{osc} = \frac{1}{2N \cdot \tau}, \tag{26}$$

where N is the number of delay elements and τ is the delay of a single element, which is defined by the process. In a single-ring oscillator, the achievable time resolution is directly defined by *tau*, and despite the low propagation delays of modern processes, the time resolution is still in the order of 10 picoseconds. In order to reach the required time resolution of 3.4 picoseconds mentioned in Section 5.1, a more complex oscillator topology is used.

The internal oscillator of the TDC is a 32-stage multipath ring oscillator (MPRO). The usage of multiple oscillation paths provides higher oscillation frequency with larger number of phase-taps. An 8-stage multipath oscillator is presented in Fig. 30. The first phase-tap, which is numbered as zero in the schematic, is driven by phases 2 and 7, which implements a skew of five stages. The increase in oscillation frequency is the consequence of two inverters driving the same node simultaneously. Single-ring ring oscillators require an odd number of inverting stages for the oscillation to be possible, whereas in multipath oscillators an even number of stages can be used, since the 180 degree phase shift is provided by the multipath arrangement. In this application, the number of stages is 32, which is convenient, since the number of phases corresponds to exactly five bits. The MPRO architecture used in the TDC is an extended version of the one presented in Fig. 30. The trade-off of MPROs is the increased power consumption compared to single-ring ROs. When two inverters are driving the same node simultaneously, a conducting path is formed between the supply rails, which consumes additional power.

Multipath ring oscillators can oscillate in various oscillation modes. For each MPRO topology and configuration, one or more dominant modes of oscillation exist. The dominant mode determines the frequency of oscillation, as well as the number and order of unique phases. So called mode gain can be calculated for each oscillation mode, and the mode with the lowest gain is dominant. In this particular topology, the dominant mode is determined by the ratio of gains between the main path inverters, which are numbered in Fig. 30, and the corresponding feedback inverters. All main path inverters are the same size, and all feedback inverters are the same size. The inverters have sizing ratio of 2:1 between the PMOS and the NMOS transistors, and the main path inverters are twice the size of the feedback inverters. This ratio provides a dominant mode, in which all 32 phase-taps are unique. When the number of stages increases, the mode gains become more closely spaced, which makes the occurrence of unwanted oscillation modes more likely. This limits the maximum feasible number of stages. The actual mode of oscillation is dependent

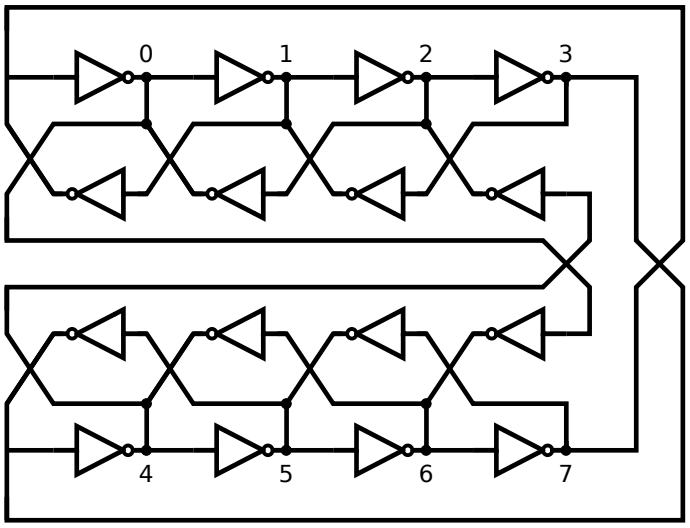


Figure 30: The simplified schematic of the multipath oscillator. [27]

on the initial conditions during startup. [27] The 32-stage MPRO is sensitive to these initial conditions, as well as PVT variations. Thus, the starting and tuning of the oscillator should be considered carefully. The relative sizes of the main and feedback path inverters should be adjustable as well in a physical implementation. Mathematical analysis of the MPRO is beyond the scope of this thesis.

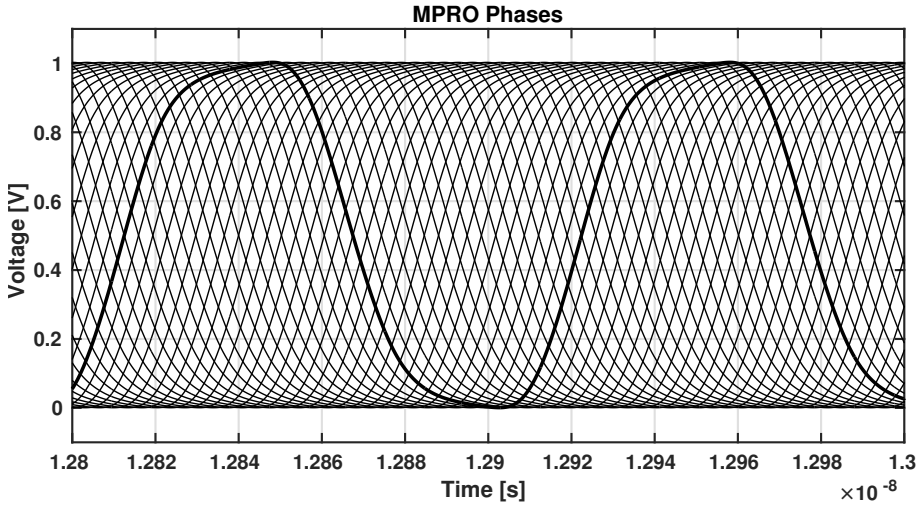


Figure 31: Transient simulation of the 32 individual phase-taps of the MPRO.

The 32 phases of the MPRO are plotted in a transient simulation in Fig. 31. The oscillator is running at over 9 gigahertz frequency. Consequently, the time-domain spacing between the phases is approximately 3.4 picoseconds, which determines the time resolution of the TDC. In this TDC, the propagating rising edge will mark the enumerated instantaneous phase of the MPRO. The ordering of the phases is critical to the encoding of the phase presented in Section 5.3.4.

As mentioned in Section 5.2.2, the oscillator frequency should be set at a sufficiently high value, while still staying below a certain frequency in order to avoid clipping. In this implementation, the frequency of the MPRO is adjusted with digitally controlled loads set for each stage of the MPRO. Controlling the load capacitance does not affect the mode gains, which is why it should provide robust frequency control for the MPRO. However, the power efficiency is reduced, since the current consumption is unaffected. Variance in the oscillation frequency is visible as nonlinearity in the conversion output. However, in this design implementation, frequency locking is not implemented, because the effect of frequency drift is only visible over longer time spans, whereas the TDC is characterized over a shorter time. Thus, the frequency will remain approximately constant during the characterization. Stabilizing the oscillator frequency from a stable frequency reference is beyond the scope of this thesis.

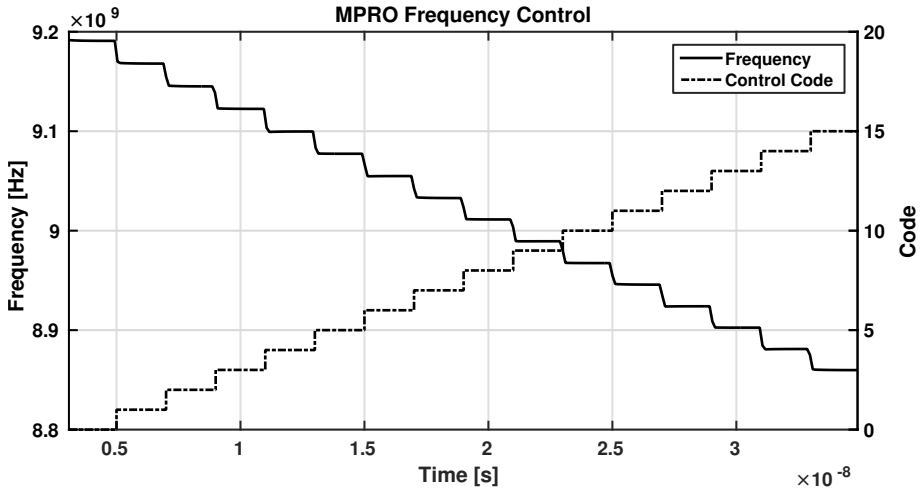


Figure 32: The MPRO frequency over time when the control word is stepped from 0 to 15.

The effect of the frequency tuning is presented in Fig. 32. A 4-bit control word is used to adjust the load of the delay elements in the MPRO. The maximum value 15 (1111) corresponds to the highest load and consequently the lowest frequency. Vice versa, the lowest control value 0 corresponds to the maximum frequency. The digitally controlled load is implemented with binary weighted NMOS-transistors. The sources and drains of the transistors are connected together, and they are further connected to the intermediate nodes of the MPRO. The control bits are connected to the gates of the transistors. The LSB of the tunable capacitance is determined by the minimum size NMOS-transistor. In this case, the implementation provides approximately 350 megahertz of tuning range with roughly 20 megahertz resolution. The tuning is implemented only for setting a coarse frequency for simulation purposes. For a proper frequency compensation application, the tuning range and resolution should be higher.

5.3.2 Gray Counter

The purpose of the counter in this TDC is to extend the full-scale range. Since the number of stages in the MPRO cannot be arbitrarily large, additional bits should be extracted from the integer cycles of the oscillator. Minimizing the time resolution is one of the key goals in the design, and that is partially achieved by maximizing the oscillator frequency. Thus, a high-speed counter circuit is required. Another important design parameter of the TDC is the error tolerance of the sampling. Asynchronously sampling a binary counter can result in large sampling errors caused by partial sampling during the transition of the counter bits. Thus, a gray code counter should be utilized, in which only one bit transitions at a time. The use of a gray counter does not prevent sampling errors on its own, but it limits the magnitude of the error and the transition time of the counter.

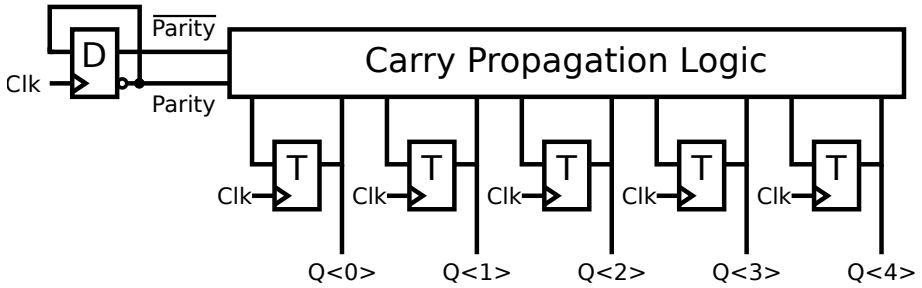


Figure 33: The gray counter block diagram.

Fig. 33 shows the block diagram of the high-speed gray counter used in the TDC. The previously mentioned high speed requirement is achieved by using parallel carry propagation logic, pseudo-NMOS logic gates and dynamic true single phase clock (TSPC) toggle flip-flops (TFF). The carry propagation logic shortens the critical path of the counter carry path compared to common bit-by-bit counting architectures. The dynamic TSPC flip-flops enable high-frequency operation, and the pseudo-NMOS logic gates perform at high speeds even in high fan-in logic gates.

The toggle flip-flop used in the counter is presented in Fig. 34. The flip-flop has two main inputs: the toggle signal and the clock signal. A high logic state of the toggle signal allows the output bit to be toggled with the rising edge of the clock. A low logic state on the other hand keeps the output constant irrespective of the clock signal. The flip-flop can be asynchronously set or reset to a predetermined output state using the set and reset inputs. In normal operation, the reset voltage should be high and the set voltage should be low. Inverting the reset bit will force the output to zero, and inverting the set bit will force the output to one. Two minimum size transistors are placed after the three clocked stages. During the low clock phase, these transistors form a static dual-inverter locking loop together with the first stage, which will prevent any changes of the output during the low clock phase. During the high clock phase, the dual-inverter locking loop is broken and these two transistors won't affect the operation. The sizing of the transistors is carried out according to [28].

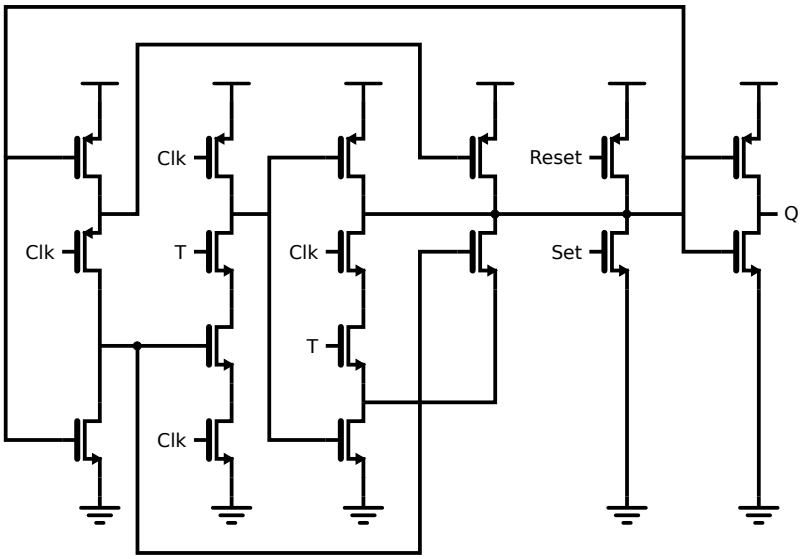


Figure 34: A TSPC-based toggle flip-flop. [28]

Table 2: The truth table of the carry propagation logic.

Clk	NMOS_n	T_{prev}	T
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	T_{prev}
1	0	1	T_{prev}
1	1	0	0
1	1	1	0

The carry propagation logic produces toggle inputs for each TFF based on the states of the previous bits. The logic is presented here using a truth table of a single clocked pseudo-NMOS NOR-gate and a logic function expression for the pull-down network of each stage. The truth table of a single pseudo-NMOS logic gate is presented in Table 2. The toggle output T of the logic gate depends on clock, the pull-down network input NMOS_n and the previous state of the output. Because the gates are implemented using pseudo-NMOS logic, the actual logic inputs are given only to the pull-down network NMOS_n. The logic functions for each bit are

expressed as

$$\begin{aligned}
\mathrm{NMOS}_0 &= Parity, \\
\mathrm{NMOS}_1 &= \overline{Parity} + \overline{Q_0}, \\
\mathrm{NMOS}_2 &= \overline{Parity} + Q_0 + \overline{Q_1}, \\
\mathrm{NMOS}_3 &= \overline{Parity} + Q_0 + Q_1 + \overline{Q_2}, \\
\mathrm{NMOS}_4 &= \overline{Parity} + Q_0 + Q_1 + Q_2,
\end{aligned} \tag{27}$$

where $Parity$ is the parity bit generated by the divide-by-2 circuit in Fig. 33 and Q_n is the n'th bit of the counter. The toggle signals for each TFF are created based on these logic functions together with Table 2.

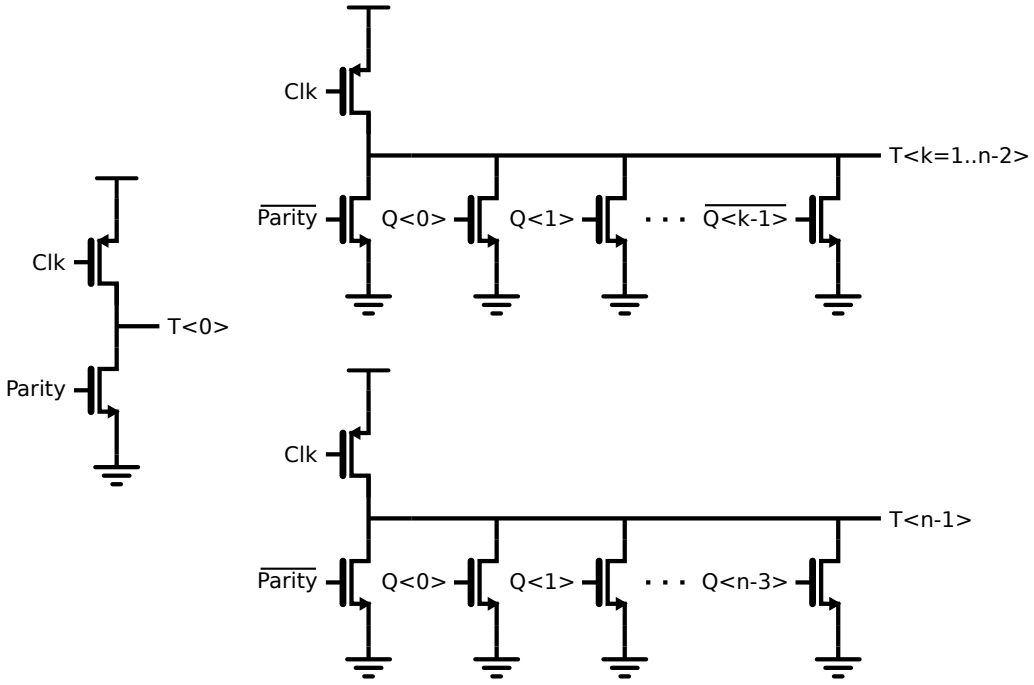


Figure 35: The carry propagation logic for the gray counter. [29]

Fig. 35 shows the high-speed pseudo-NMOS circuit implementation of the carry propagation logic. The leftmost pseudo-NMOS logic gate implements the toggle signal for bit 0. The corresponding logic function is the first row of 27. The top circuit implements the toggle logic for bits 1, 2 and 3. They correspond to the respective logic functions. The lower circuit implements the final logic function.

Fig. 36 shows a transient simulation of the implemented gray counter running at 15 gigahertz frequency. The two design goals for the counter can be verified from the simulation result. First, the achieved operation speed is sufficiently high for counting the cycles of the MPRO running at approximately 9 gigahertz. Second, the gray code sequence has only one bit transitioning at a time, which eases the sampling error mitigation.

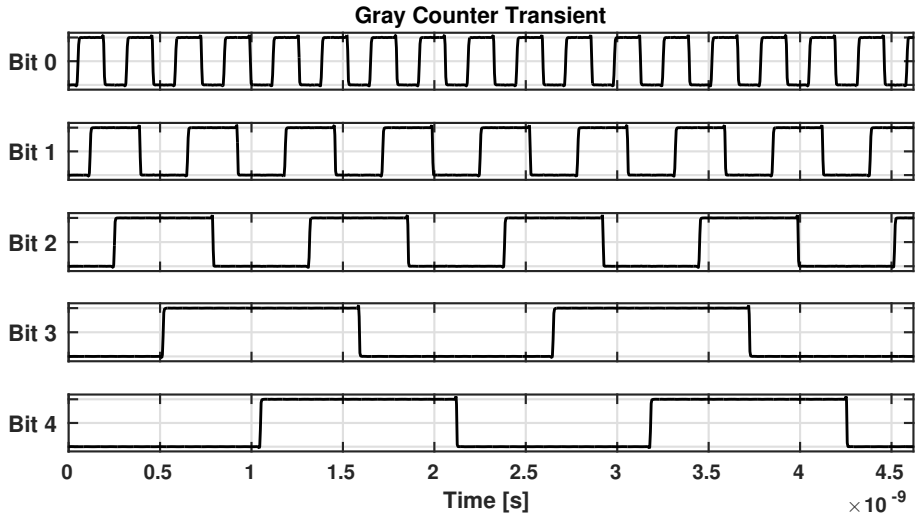


Figure 36: Transient simulation of the implemented gray counter running at 15 GHz clock frequency in nominal conditions.

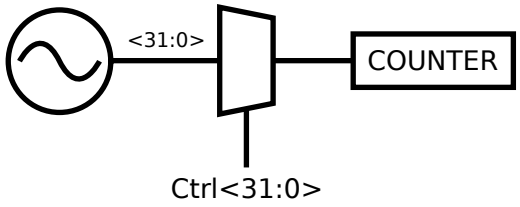


Figure 37: The counter correction multiplexer.

The clock signal of the counter is a single phase of the MPRO, which is selected using a 32-bit multiplexer, as shown in Fig. 37. The selection is related to calibrating the counter correction presented in Section 5.3.4.

The schematic of the multiplexer is presented in Fig. 38. The 32-bit multiplexer is implemented with 32 transmission gates and a 32-input OR-gate. The transmission gate inputs are buffered with simple inverters, and the outputs of the transmission gates are connected to the OR-gate. The OR-gate is implemented using pseudo-NMOS logic because of the large number of inputs. Also, grounding switches are added between the transmission gates and the OR-gate in order to prevent undefined logic states. Without the OR-gate, the time constant of the series switch resistance of the transmission gates and the large capacitance of the output node is too large to operate at the required frequency. Thus, the OR-gate is utilized in order to avoid the large time constant.

5.3.3 Sampling Flip-flop

An integral part of error resilient sampling is a high-performance flip-flop circuit. In time-based asynchronous designs, the importance of sampling performance is

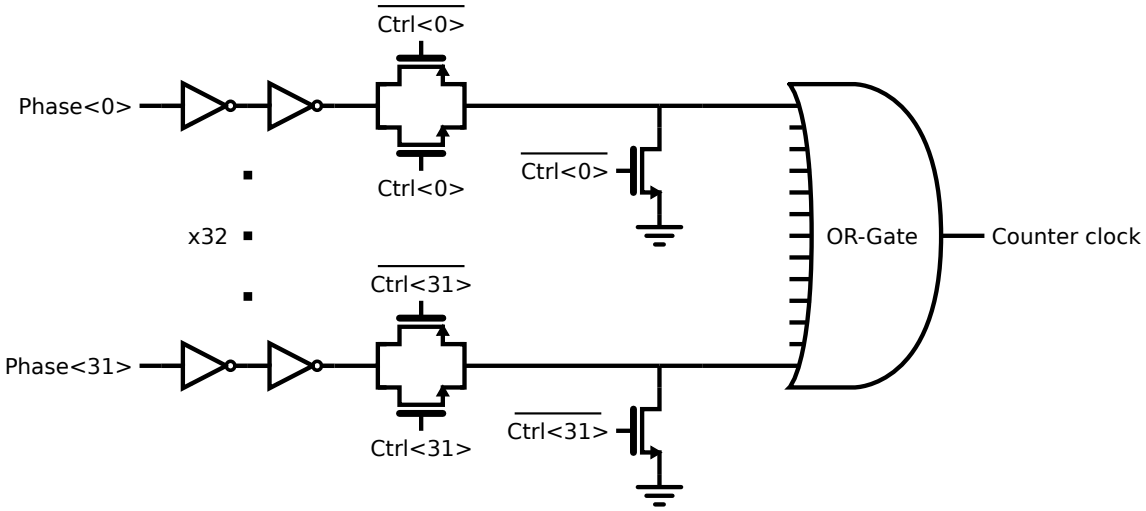


Figure 38: Schematic of the multiplexer.

emphasized, since any non-idealities in sampling directly effect the processing of the signal. The flip-flop performance is evaluated here in terms of resilience to metastability induced sampling errors in asynchronous sampling, which is analyzed through the setup and hold times of the flip-flop. The flip-flop designs can be evaluated using various parameters, such as energy, delay, area and clock load [30]. Additionally, error-robustness and metastability performance can be included in the design trade-offs [31].

Static master-slave flip-flops are based on positive feedback elements and they offer isolation from noise at the output, glitch-free output and PVT variation resistant switching capability [32]. However, the maximum operation speed or the width of the metastability window often limits the performance in demanding applications. In high-performance applications, some of these characteristics can be sacrificed in order to achieve higher performance where needed. In this design, the high-frequency oscillator phases are sampled asynchronously, which makes setup and hold time violations inevitable. Thus, a flip-flop with as robust metastability performance as possible should be selected for sampling. In the following analysis, two flip-flop types are compared: a dynamic TSPC flip-flop and a sense-amplifier flip-flop (SAFF). The maximum operating speed of the TSPC flip-flops is known to be high, and the SAFFs should provide high performance while sampling near the metastable window.

A rising-edge triggered TSPC flip-flop is presented in Fig. 39. The TSPC flip-flop is a dynamic flip-flop, where the state of the flip-flop is stored in the internal node capacitance, as opposed to using a feedback loop like in static flip-flops. This allows for high-speed operation of the device. Furthermore, the use of a true single phase clock avoids problems related to generating complementary clock signals. Additionally, the TSPC flip-flop has minimal amount of transistors, which results in area savings. However, the dynamic logic CMOS circuits are more sensitive to noise, because the internal dynamic node, which holds the bit charge, might suffer from noise coupling. This can result in problems with reliability. [34] Like discussed in Section 5.3.2,

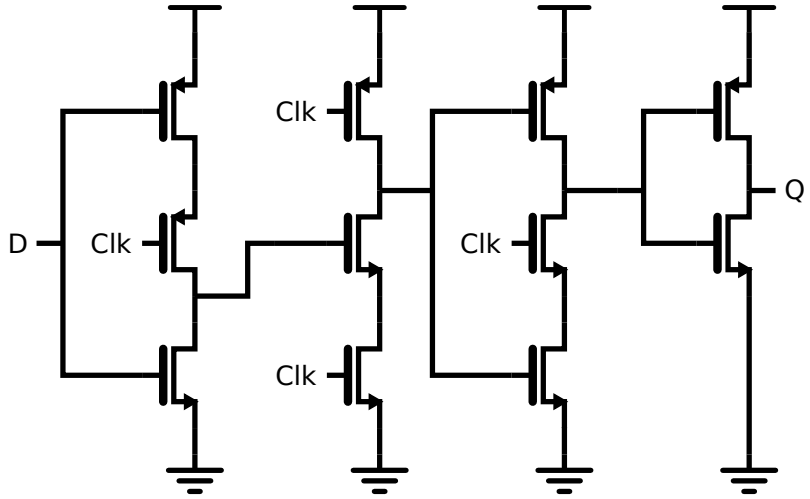


Figure 39: Rising-edge triggered TSPC flip-flop schematic. [33]

the implemented gray counter utilizes TSPC flip-flops for high-frequency operation. Although TSPC flip-flops can operate at high frequencies, they do not have any internal mechanism for resolving a metastable state, unlike cross-coupled latch -based systems.

Current-controlled latch sense-amplifiers provide a fast and low-power solution for memory elements [35]. Sense-amplifier flip-flops are usually implemented using a StrongARM latch as the regenerative comparator input stage, because it provides zero static power consumption [36]. Sense-amplifiers can sense a small differential current at the input and produce a large differential output voltage. Furthermore, once the latch has regenerated, the sense current is stopped automatically, which results in zero static power consumption.

The two stages of the SAFF used in this TDC are presented in Fig. 40a and 40b, respectively. The SAFF offers many attractive qualities, such as switching capability independent from circuit sizing, near-zero setup time, low hold time, low clock load and TSPC operation [32]. In the low phase of the clock signal, the $\overline{\mathrm{S}}$ and $\overline{\mathrm{R}}$ nodes are pre-charged to V_{dd}. The cross-coupled inverter connection also opens up the two NMOS-transistors below. Since $\overline{\mathrm{S}}$ and $\overline{\mathrm{R}}$ are both high, the latch stage keeps the flip-flop state constant. The rising edge of the clock signal will allow either $\overline{\mathrm{S}}$ or $\overline{\mathrm{R}}$ to be pulled to ground through one of the input transistors. The other node will remain charged to V_{dd}. As soon as one of the SA outputs falls to zero, the corresponding NMOS-transistor of the inverter switches off. Thus, the DFF operation is provided and no further change in the D input will change the state. The NMOS transistor driven by V_{dd} guarantees a grounding path for either output in case D changes during the high clock phase. The latch stage in Fig. 40b stores the state determined by the SA stage.

Instead of a normal NAND SR-latch, in this version, the output latch is a hybrid between a NAND SR-latch and a N-C^2-circuit. Without the two stacked NMOS-transistors on the left hand side in Fig. 40b, the latch would be a NAND SR-latch.

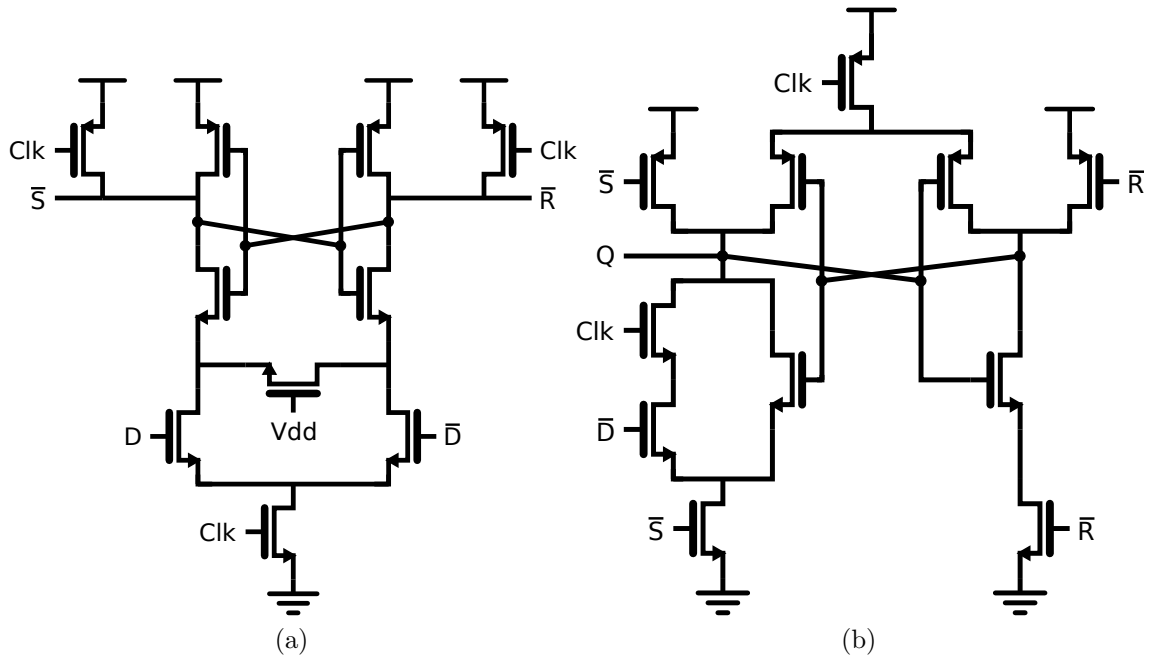


Figure 40: (a) The SAFF sense-amplifier stage. (b) The SAFF latch stage. The sizing has been optimized for a single output according to [32].

The modification reduces the high-to-low delay of the latch. The new delay is only one delay stage compared to the three delay stages of a NAND SR-latch -based SAFF. Furthermore, in this SAFF design, the latch stage is optimized for a single output. In a normal complementary output case, the latch would also be symmetrical. Now, however, only one side has the added N-C^2-circuit, which allows for reduced area and power consumption. [32]

Fig. 41 shows a transient simulation characterizing the two presented flip-flops. The clock rising edge is used to trigger both flip-flops, and the rising edge time instant is kept constant. Both of the outputs are preset to zero and they are both loaded by a minimum size inverter. The input node D sees a rising edge, which is moved between successive simulations. The output Q of both flip-flops is plotted in all simulation runs. Both flip-flops respond quickly after the clock edge. However, it can be seen that the SAFF has superior performance in resolving the metastable state of the flip-flop. The output state of the SAFF is quickly resolved, whereas the TSPC enters a metastable state easily and remains there for a significant amount of time. The TSPC flip-flop, unlike the SAFF, has no inherent mechanism for overcoming the metastable state. Additionally, the TSPC flip-flop output shows slight glitching on the falling edges of the clock as well (not visible here). The fast resolving of the metastable state shown by the SAFF makes it a suitable choice for the sampling flip-flop in this TDC.

The clock-to-q delays of the two flip-flops are plotted in Fig. 42. The x-axis values represent the time difference between a signal edge at the input D and the clock

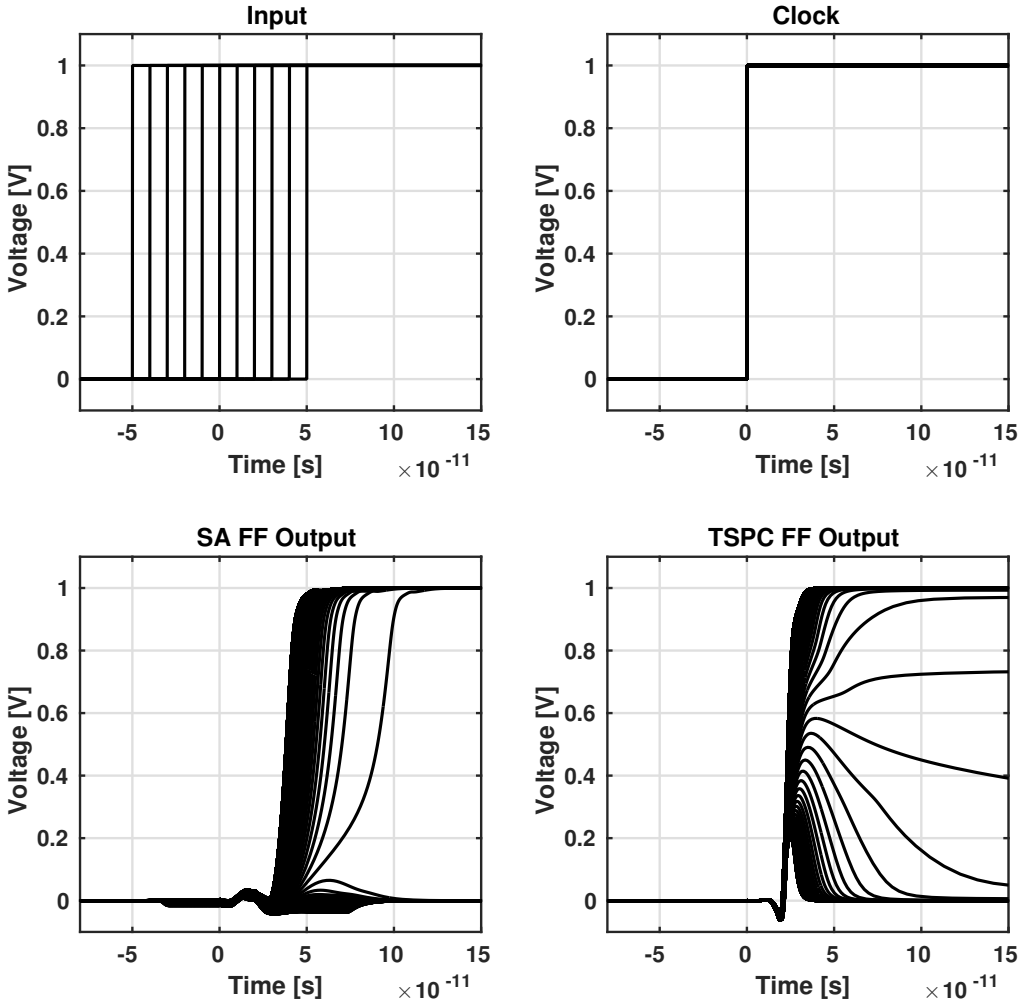


Figure 41: Transient simulation of the SA and TSCP flip-flops.

signal rising edge. At the zero point, the two edges are aligned, while negative values imply the input leading the clock and the positive values imply the input lagging behind the clock. The delays are measured according to the transient simulation in Fig. 41 from 90% level of the clock to the 90% or 10% level of the output for rising and falling edges, respectively. As the clock-to-q delay gets its maximum value, the output corresponding to the following input edge time is not registered at the output anymore. For both rising and falling edges, the setup time instant t_{setup} is taken as the point where the clock-to-q delay has increased by 10% from the minimum value. The hold time instant t_{hold} is the final point where the output registered the correct value (at 90% of the range). [37]

According to these definitions, the metastability window of the SAFF for a rising edge input is

$$T_{meta} = t_{setup,lh} + t_{hold,lh} = -1.7 + 3.9 = 2.2 \quad \text{[ps]}. \tag{28}$$

It should be noted that in this simulation, a threshold value of 90% was used to

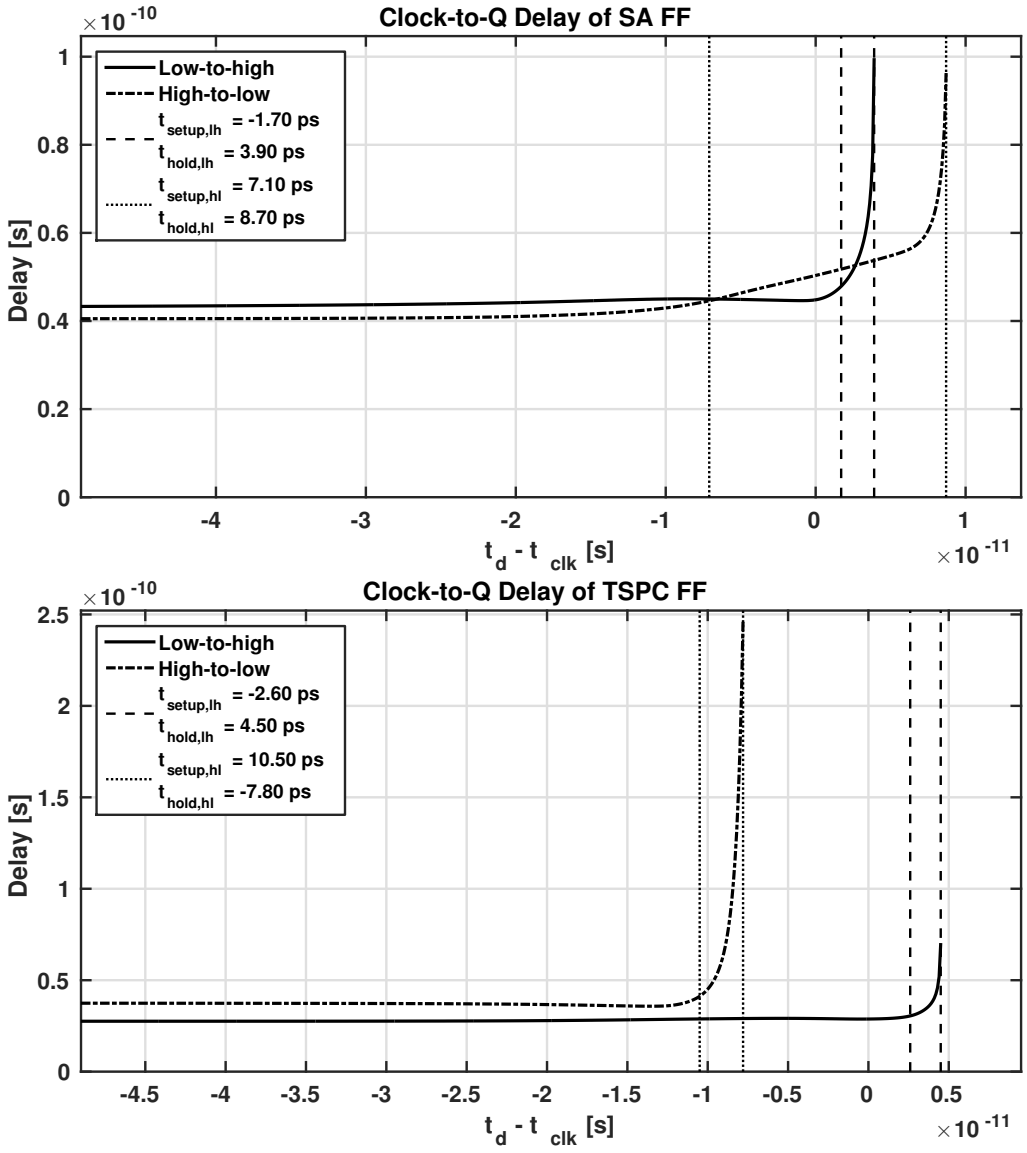


Figure 42: The clock-to-q delays of the SA and TSPC flip-flops.

look for the rising edge at the output. Thus, the hold time for the rising edge might appear smaller than it is in reality. However, for the SAFF, the metastable states seem to resolve quickly, which implies a reduced discrepancy caused by the threshold. The reliability of a flip-flop can be assessed by calculating the mean time between failures (MTBF) for the circuit. The value for MTBF can be calculated as

$$\text{MTBF} = \frac{e^{S/\tau}}{T_{meta} \cdot f_{clk} \cdot f_d}, \tag{29}$$

where S is the synchronization period, τ is the maximum time it takes to resolve the metastable state, T_{meta} is the width of the metastability window and f_{clk} and f_d are the clock and input frequencies, respectively [38]. The resolving time τ is measured

from the clock edge to the point where the output reaches one or zero. For this SAFF flip-flop implementation, the simulated τ is approximately 130 picoseconds. The synchronization period is one clock period and the input frequency is set by the application. In the case of this TDC, the input frequency should be the oscillator frequency. Calculating the MTBF for this flip-flop in this application yields

$$\text{MTBF} \approx 1.58 \cdot 10^7 \text{ s}. \tag{30}$$

The calculated value translates to roughly half a year of mean time between failure, which is very low in conventional terms. However, in an asynchronous system like this TDC, violation of the metastability window is expected. In this system, the metastability of the sampling flip-flops will cause only LSB errors, since the sampled phase is encoded in such a way that only one bit transition at a time represents the data. LSB errors caused by metastability are unavoidable. However, the probability of the errors occurring can be minimized by using a robust flip-flop. For the TSPC flip-flop, the MTBF value would be very low, as there is no real mechanism for overcoming the metastable state. Thus, the SAFF is used for sampling instead of the dynamic TSPC flip-flop.

5.3.4 Encoding and Error Correction

The previously presented MPRO, counter and sampling flip-flop account for providing the required time resolution, full-scale range, and robust sampling. However, the error-free conversion result is not guaranteed by these sections alone, but digital error correction and encoding is required.

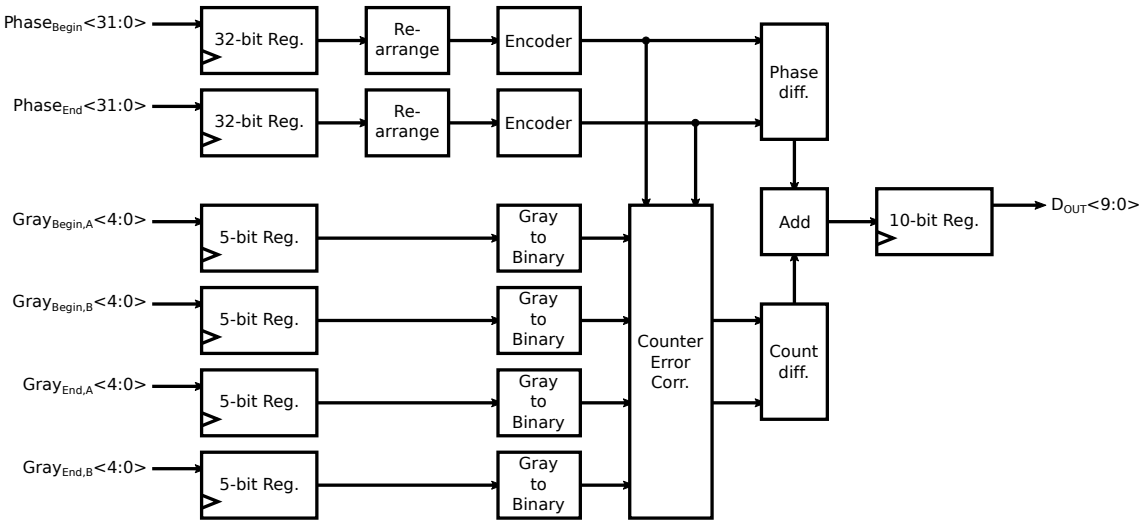


Figure 43: The block diagram of the digital backend structure.

The encoding and error correction is carried out by the digital backend of the TDC, which is presented in Fig. 43. The backend has two main data paths: the oscillator state - or thermometer code - path and the counter path. The oscillator

phase is registered with two 32-bit registers, one for the begin state and one for the end state. The final counter value is derived from four registered values. The error correction algorithm is based on taking two samples of the counters separated in time-domain, which are denoted as A and B samples in the figure, corresponding to the delayed asynchronous samples shown in Fig. 25. Additionally, similarly to the oscillator phase sampling, the counter state is sampled both at the beginning of the time interval, as well as the end of the time interval. Altogether this results in four 5-bit registers. All registers in the digital backend are clocked by the sampling clock.

The sampled raw 32-bit state of the MPRO is registered in the 32-bit input registers. The sampled state is rearranged into a more coherent pattern, where all high bits are grouped together. If the node 0 is set to be the reference phase-tap, the order of the rising edges can be observed as

$$i[n] = (N - 3n) \ \% \ 32, \quad n = 0..N - 1, \tag{31}$$

where N is the number of stages, which is 32 in this case. The equation re-maps the indices of the phase-taps. Using such reordering, the MPRO phases can be rearranged into a coherent code, where a string of ones moves into one direction. The coherent code can be easily converted into one-hot encoded word, where only one of the bits is high. The index of the high bit denotes the enumerated phase value. An example of the rearranging process is presented in Fig. 44. In the figure, the left column is the index of the bit, the phase column is the sampled oscillator phase, the following column is the result of reordering the indices according to 31 and the final column is the corresponding one-hot code.

In this representation, the group of ones propagates in one direction and loops around in a circular manner. The coherent code is then further converted into one-hot code. In this conversion, bubble error correction is utilized in order to mitigate the effects of temporal phase reordering, which might occur due to non-idealities. The one-hot encoded oscillator phase is converted to binary. In this design, the falling edge propagating in the oscillator is ignored. The falling edges could also be used in the encoding process, which would result in an improvement in time resolution. However, tracking only the rising edge results in significantly simpler encoding process. Furthermore, the rise and fall times of the oscillator phases do not have to be matched. Also, only one bit should transition from low to high at a time. Erroneous sampling of the leading bit due to metastability results in only a single LSB error. After converting the 32-bit oscillator phase into 5-bit binary value, the begin and end values are subtracted. The subtraction allows overflowing of the bits, which results in a correct difference value into a specific direction. The differentiated 5-bit phase value gives the 5 lowest significance bits of the converter output.

Each counter input contains a 5-bit gray encoded counter value. The values are converted into binary codes with a simple combinational gray to binary converter. The conversion is performed for all four 5-bit counter inputs. The counter error correction -block implements an error correction algorithm, which is a modification of [39]. The implemented algorithm is based on comparing two counter samples separated in time-domain and deducing the correct sample based on the sampled instantaneous oscillator phase.

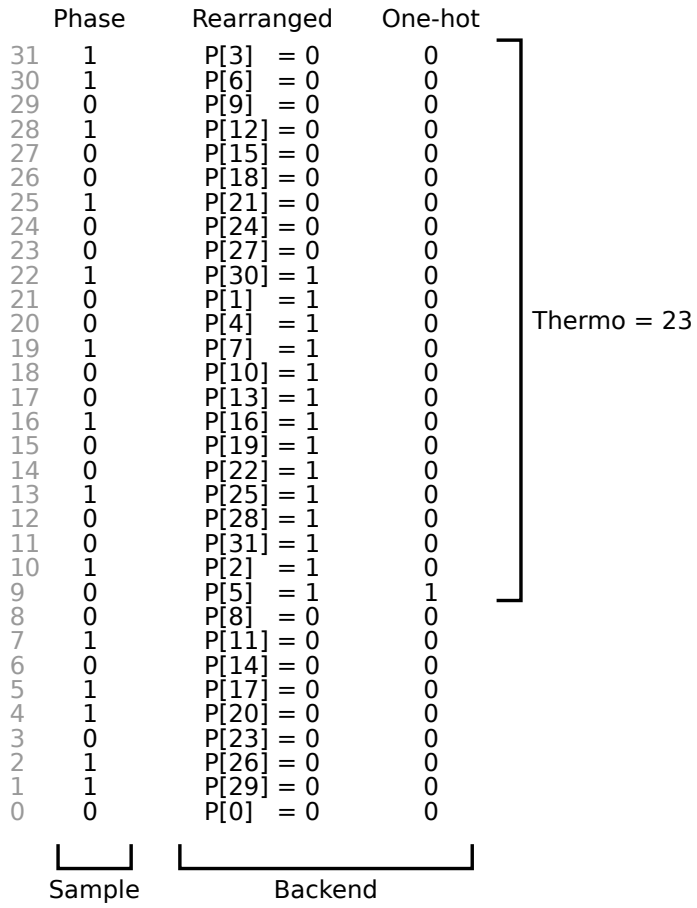


Figure 44: The rearranging of the sampled MPRO phases.

In Fig. 45, the operation of the correction algorithm is illustrated through five sampling cases. The delay between the zero-phase of the oscillator and the corresponding counter increment is denoted as T_D, and the transition time at the counter output (glitch time) is denoted as T_G. The time separation between the two counter samples is denoted as τ_C. The five sampling cases and the corresponding corrected counter values are presented in Table 3. Sample A refers to the leftmost sample in each case in Fig. 45, and sample B refers to the delayed sample. The correct sample is chosen based on the instantaneous phase of the oscillator, which is determined by sampling the oscillator at the sample A instant. The first sample is assumed to be correct, unless the phase of the oscillator is low at the sampling instant and sample B is higher than sample A. In such a case, sample B can be assumed to be the settled state of the counter.

The correction algorithm has a few limitations to its operation. The time separation between the two samples is determined by the sample offset delay τ_{C} by the buffer in the start and stop paths in Fig. 25. The length of the delay buffer has to be set within margins, which are set by the counter path delay $\mathrm{T_D}$ and the oscillator frequency. The delay $\mathrm{T_D}$ is the delay between the oscillator zero phase and the next counter tick. The counter tick can be shifted by changing the selection

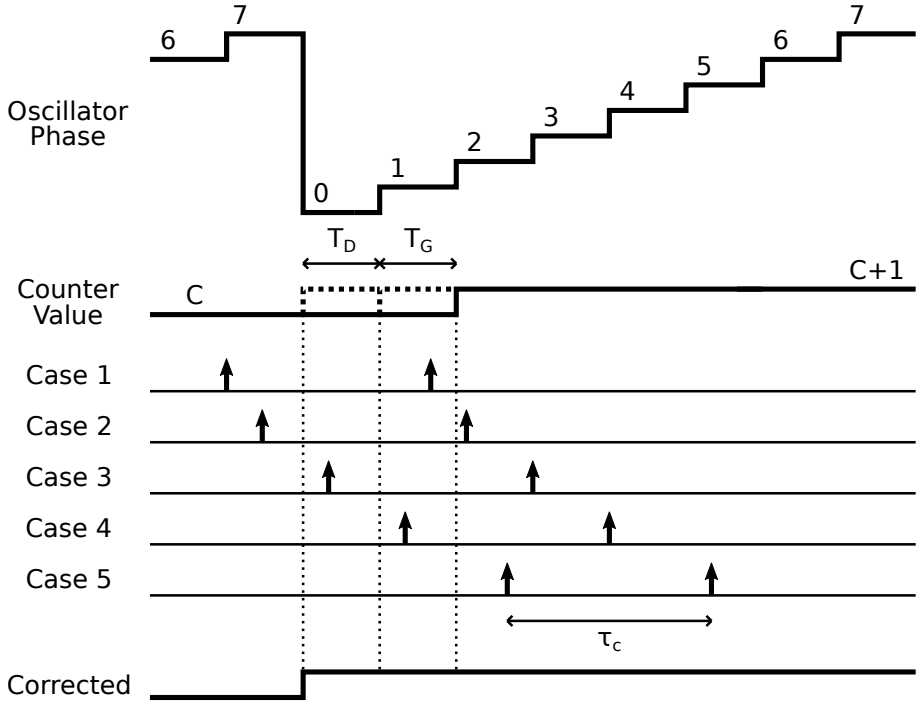


Figure 45: A timing diagram of the delayed double sampling of the counter. The five sampling cases show all the possible cases between the two samples. [39]

Table 3: The example sampling cases from Fig. 45.

Case	Osc. Phase	Sample A	Sample B	Selected
1	High	C	C	C
2	High	C	C+1	C
3	Low	C	C+1	C+1
4	Low	C	C+1	C+1
5	Low	C+1	C+1	C+1

of the phase-tap driving the counter. The value of T_D can be adjusted with LSB precision, which should be enough to meet the margins. The counter should lag behind the oscillator zero phase, which results in

$$T_D > 0. \tag{32}$$

Additionally, buffer τ_C has to obey the margins set as

$$2 \cdot \max(T_G, T_D) < \tau_C < T_D + \frac{T_{osc}}{2}, \tag{33}$$

where T_{osc} is the period of oscillation of the oscillator. Only one of the two samples should be in either erroneous region at once. This requirement results in the lower margin, where the spacing has to be larger than either erroneous region. The upper

margin is placed to prevent the second sample from entering the erroneous zone while the first sample is still in the first half-period of the oscillator. The erroneous region ($T_D + T_G$) has to be smaller than the half-period of the oscillator. If these margins are obeyed, there should be no counter sampling errors present at the output.

The above mentioned conditions were confirmed with a simulation. In the simulation the oscillator period is

$$f_{osc} = 9.122 \text{ GHz} \Rightarrow T_{osc} = 109.6 \text{ ps}. \tag{34}$$

The simulated counter delays T_D are presented in Table 4. These values were simulated for a specific selected phase-tap by comparing the bit rising edges to the zero phase phase-tap rising edge. Also, the counter bits have uneven loading caused by the internal counter structure, which skews the bit delays slightly. Taking the skew into account when calculating the margins results in counter delay values of

$$T_D = 11.3...13.1 \; ps. \tag{35}$$

Now, the τ_C buffer delay has to obey the condition in 33. In the equation, the worst case values are used from the simulated T_D, which means using the maximum value of T_D for the lower bound and the minimum for the upper bound. Consequently, the margins become

$$26.2 \; ps < \tau_C < 66.0 \; ps. \tag{36}$$

The τ_C delay line has 3 buffers (2 inverters each), and the simulated delay value is 39.8 picoseconds. The buffer delay should be kept approximately constant, while the counter path delay should be adjusted using the multiplexer.

Table 4: The phase-to-output delays for each bit in the gray counter.

Bit	T_D [ps]
0	13.1
1	12.3
2	11.8
3	11.8
4	11.3

Using this configuration, the sampled counter values are error-free. Increasing the oscillator frequency will reduce the available margin. Also, any skew between the counter bit delays makes the margins stricter. However, the algorithm is somewhat resistant to PVT variations. As the supply voltage reduces, the buffer delays grow longer, but the oscillation frequency drops as well. Consequently, the margin values and buffer delays move in the same direction. Thus, to an extent the margins should be preserved even in varying conditions. [39] In the original implementation in [39], the counter was sampled three times in stead of two, which was implemented to avoid ambiguous errors caused by asynchronously sampling of a binary counter. In this design, the correction is implemented with two samples, because a gray counter

is used. In essence, the second sample is simply a counter sample, which is assumed to be correctly settled when the sampling occurs in a specific region.

In addition to the presented counter sampling error correction algorithm, additional digital logic has to be applied to prevent erroneous counter values. After applying the previously presented algorithm to each individual counter sample (begin and end), they can be assumed to be correctly sampled. Now, depending on the oscillator phase during the sampling instants of these samples, the counter difference has to be adjusted. The additional correction step is based on whether the begin and end phase samples are in the opposite half-periods of the oscillator. Also, the phase difference dictates the correction decision. A truth table describing the correction step is presented in Table 5. In the table, a high bit means that the corresponding value is in the upper half-period of the oscillator (≥ 16). Vice versa, the zero-bit means the value is in the lower half-period (< 16). In the correction column, zero-bit means no action is taken. For the cases with a one-bit in the correction column, the current counter difference value is decreased by one. Applying the additional counter correction prevents erroneous extra counts, which can occur when sampling the free-running oscillator.

Table 5: The truth table describing the additional counter correction step.

Begin	End	Diff	Correction
0	0	1	0
0	0	0	1
0	1	1	0
0	1	0	0
1	0	1	1
1	0	0	1
1	1	1	0
1	1	0	1

After encoding the instantaneous oscillator phase and the counter value, the final section of the backend concatenates the two 5-bit values. The direct phase difference gives the lower 5 bits and the corrected counter difference value gives the 5 MSBs. The 10-bit output code is then registered to the output register by the ADC clock.

5.4 System Simulations

The designed TDC is simulated with a single mixed-mode simulation in 28 nanometer process. In the simulation, the analog input signal is generated in Eldo and converted into a timing signal by an ideal VTC circuit model. The ideal timing signal is then quantized by the designed TDC, and the quantized bits are encoded into a binary output code by the digital backend (Section 5.3.4), which is implemented as a functional VHDL block. These analog and digital blocks are simulated together using Questa ADMS, which runs Eldo for the Spice-netlists. Essentially, the TDC is

characterized as if it was a full ADC with an ideal front-end. The circuit is simulated as a pre-layout version, but additional capacitance is added to the nodes of the MPRO to simulate the increased load caused by layout parasitics. The digital backend is has power and timing performance estimated from a post-layout stage.

5.4.1 General Characteristics

First, the TDC is characterized with a noise-free simulation. The purpose is to show the performance of the architecture with only structural non-idealities present. The performance metrics of the TDC are extracted from a single-tone sinusoid simulation. The main metrics considered are the SNDR, SFDR, INL, DNL and the power consumption.

The single-sided FFT of the digital output for a low frequency and a Nyquist rate signal is presented in Fig. 46. The input frequencies and the sampling frequency are chosen such that coherent sampling can be achieved. In coherent sampling, the ratio of the sampling frequency and the input frequency should be a prime number divided by the desired number of FFT samples. The ratio results in reduced spectral leakage and smearing of the tones. Thus, in Fig. 46, only a rectangular window is required.

The SNDR and SFDR are calculated from the time-domain data using Matlab. According to the simulation, the TDC reaches SNDR of 57.10 dB and 57.56 dB for low and high input frequencies, respectively. The corresponding ENOB values are 9.19 bits and 9.27 bits, out of the full-scale range of 9.82 bits. The SFDR of the TDC is limited in this simulation by a spur around 9 GHz. The spur is caused by the interaction between the oscillator and sampling frequencies. The calculated SFDR values are 63.28 dB and 63.20 dB for the low and high input frequencies, respectively.

The differential and integral nonlinearities are presented in Figs. 47 and 48, respectively. These static nonlinearity metrics are calculated from the single-tone sinusoid using the histogram method. The sinusoid simulation was used for the INL and DNL instead of a typical ramp simulation, since simulating a ramp response for a low-slope ramp takes significant amount of time. The simulated dataset contains 2048 samples. Due to the sinusoid signal used in conjunction with the histogram method, the resolution of the figures is quite low in the center of the code range, which can result in exaggerated values. The maximum measured positive INL value is 0.83 LSBs, and the maximum measured positive DNL value is 1.25 LSBs.

The power consumption of the TDC is presented in Fig. 49. The majority of the total current is consumed by the MPRO, which consumes approximately 2.2 milliamperes from a 1 volt power supply. The second high-frequency circuit in the TDC is the gray counter, which consumes approximately 1.5 milliwatts of power. The power consumption of the sampling flip-flops was originally over 1.5 milliwatts. However, by gating the input node of the flip-flops and opening the sense-amplifier stages to the high-frequency input only around the start and stop signal edges allowed for significant power savings. These current consumption estimates are calculated as the average of the extracted transient current drawn from the power supply for each respective circuit block. The power consumption estimate of the digital backend is

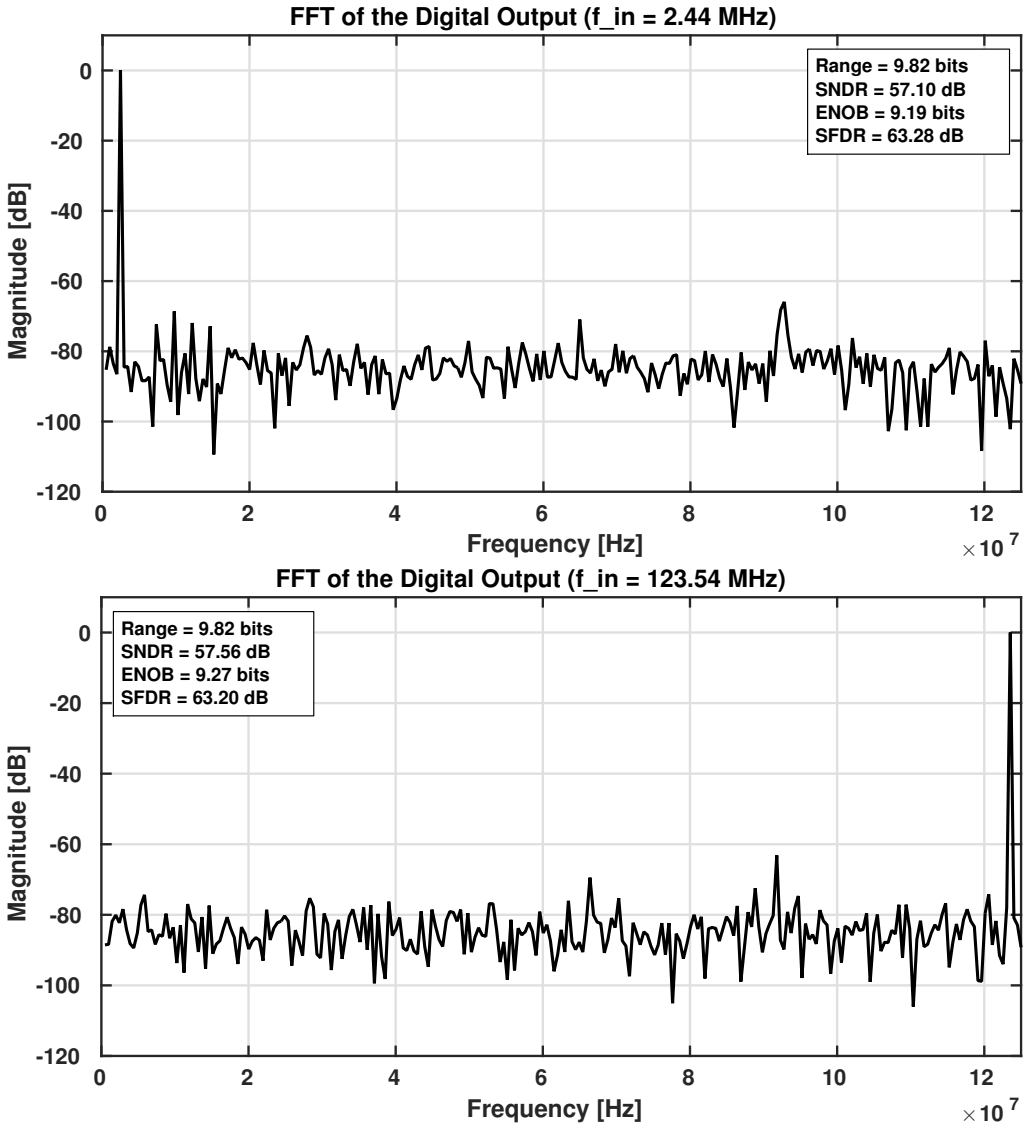


Figure 46: Single-tone 512-point FFT simulation of the TDC with low and high frequency input signals.

produced by the static timing analysis with extracted parasitics.

5.4.2 Corner Simulation

The operation of the TDC is characterized in three different PVT corners. The input signal used in the simulation is a low-frequency single-tone sinusoid. The nominal corner results are the same as in Subsection 5.4.1. The results of the corner simulations are presented in Table 6. The chosen operating corners for the simulation are from 0 to 85 degrees Celcius, ±5 % supply variation, and TT, SS and FF process corners. The column headers of the table describe the operating corner of each case.

In all simulations, the full-scale range of the ideal VTC is kept constant, where

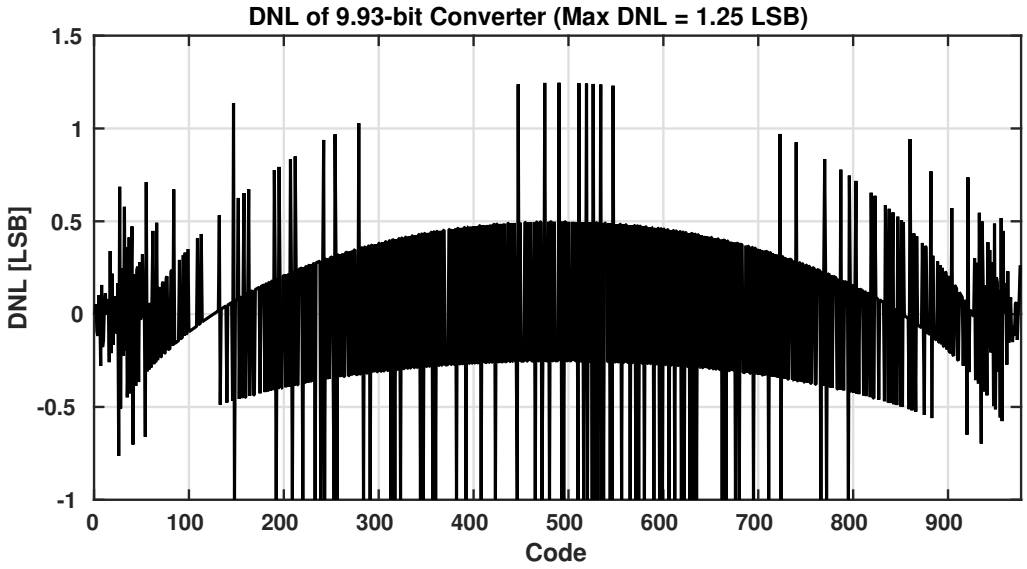


Figure 47: Differential nonlinearity.

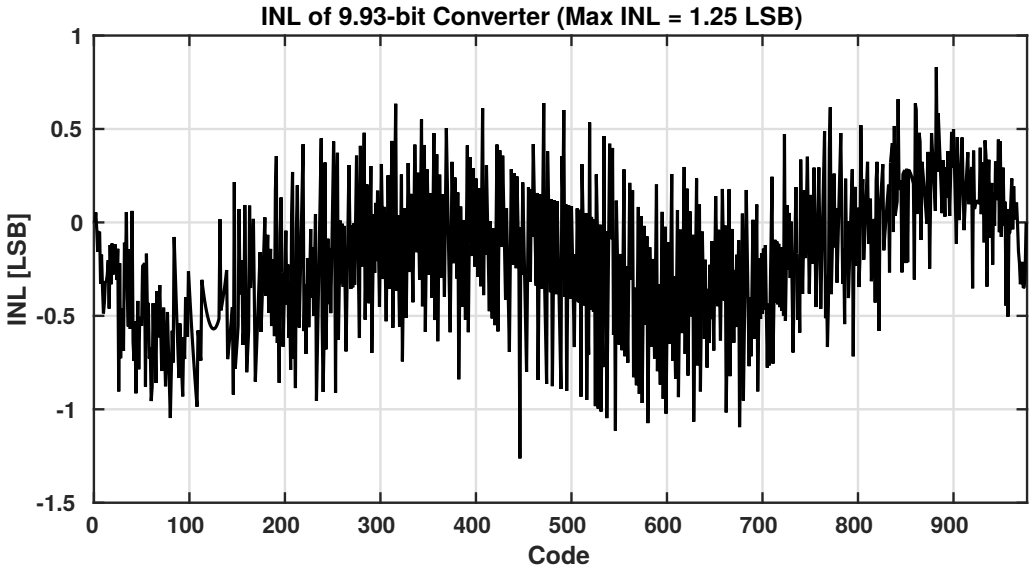


Figure 48: Integral nonlinearity.

the maximum pulse width of the VTC is set to 3.07 nanoseconds. From Table 6, it can be observed that the oscillator frequency changes significantly with the corners. In the slow corner, the range and ENOB are degraded due to the lower frequency. On the other hand, in the fast corner, the TDC output shows significant clipping. The clipping is caused by the counter bit depth limitation, which is a consequence of a design decision. For the sake of comparison, the clipping output codes have been corrected in Matlab. In a future version, the counter depth should be large enough for clipping-free operation in all corners.

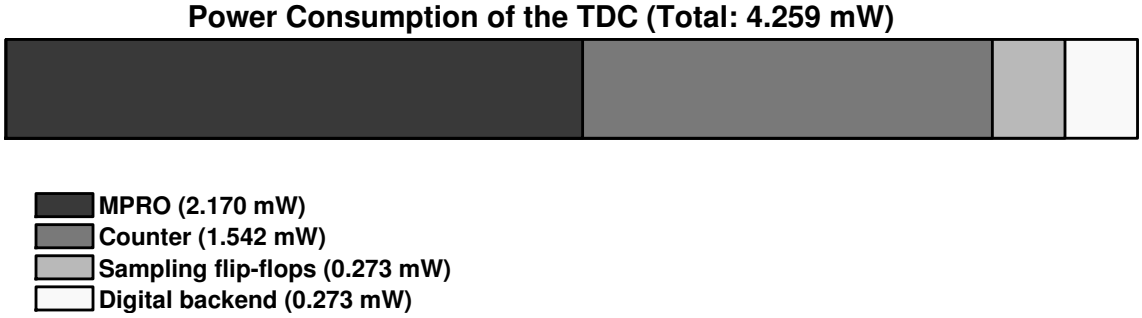


Figure 49: Power consumption of the TDC.

Table 6: Corner simulation results for a low-frequency input signal.

	SS/0.95V/85°C	TT/1.0V/27°C	FF/1.05V/0°C
SNDR [dB]	54.63	57.10	63.13
ENOB [bits]	8.78	9.19	10.19
SFDR [dB]	59.45	63.28	71.92
f_{osc} [GHz]	5.71	9.17	13.83
Range [bits]	9.13	9.82	10.41
LSB [ps]	5.48	3.39	2.26
Power [mW]	2.50	4.26	6.81

The counter was observed to be sensitive to PVT variations. The corners with either fast PMOS and slow NMOS or vice versa were not simulated here. These mixed corners might cause additional issues because of the pseudo-NMOS logic used in the counter. The counter calibration via the selector switch did not require any adjustments over the simulated corners, despite the oscillator frequency changing drastically. As was mentioned in Section 5.3.4, the double sampling error correction should be tolerant of process corners, which was confirmed by these simulations.

5.4.3 Transient Noise Simulation

The TDC was simulated with transient noise using Eldo noise transient simulation. In the simulation, noise sources from 1 hertz to 10 gigahertz are introduced to the simulation. As was mentioned previously, the spurious components visible in the TDC spectrum are caused by the relationship between the oscillator frequency and the sampling frequency. Thus, with variance in the oscillator frequency, the relationship should be somewhat broken and the spur should be smeared to the noise floor.

An FFT of the TDC output codes with transient noise is presented in Fig. 50. From the spectrum it can be observed that the spurs have been effectively reduced. With all other conditions equal to the previous simulations, the SFDR has increased from 63.20 decibels to 69.27 decibels, which confirms that the contribution of noise

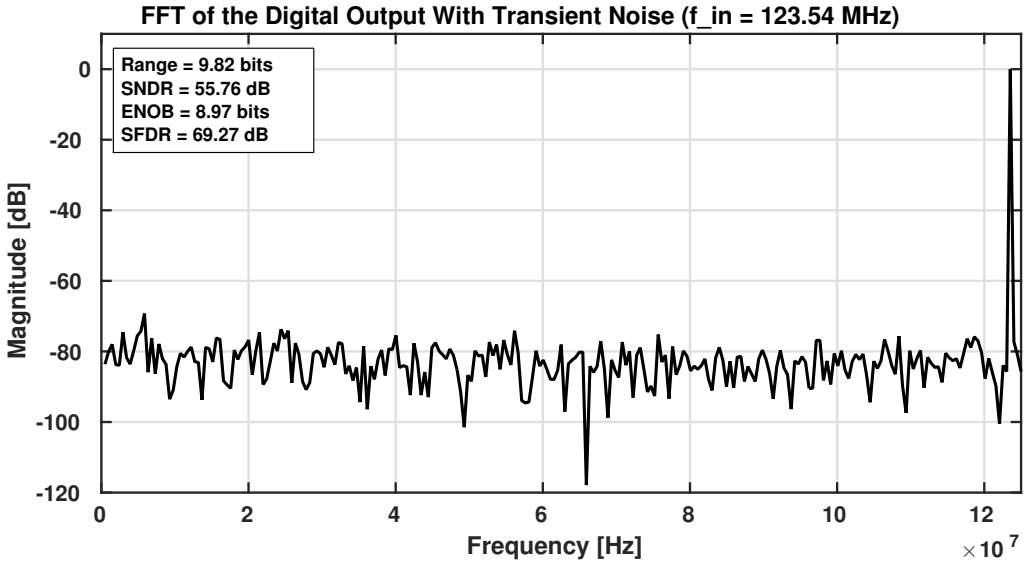


Figure 50: Spectrum of the TDC with transient noise.

effectively reduces the spurs caused by the fundamental frequency of the oscillator. Moreover, oscillator noise can even be seen as beneficial in this case. The ENOB of the TDC decreased only slightly and the circuit seems rather insensitive to noise. The calculated time interval error (TIE) jitter of the MPRO is $\sigma_{tie} = 0.304$ picoseconds. According to 18 in Section 3.3, the maximum allowed jitter is close to one picosecond.

5.4.4 Performance Comparison

A performance comparison is presented in Table 7. The table compares TDCs of similar operating ranges to the design presented in this thesis. The metrics considered in the table do not include the dynamic performance metrics (SNDR and SFDR), since these are often not reported for pure TDCs. In this thesis, however, the TDC was characterized with an ideal VTC front-end. Thus, the TDC was characterized as a pseudo-ADC. It should be noted that all simulation results presented in this thesis are pre-layout simulations. Thus, the power consumption and linearity metrics may be optimistic.

Furthermore, the unconventional sampling clock signal used in this design skews the results slightly. The clocking scheme is based on maximizing the full-scale range and thus producing higher FOM for the conversion. The operation has been confirmed with a real VTC design. The maximum input time interval used in the simulations in this thesis is constant at 3.07 nanoseconds, which corresponds to a sampling rate of approximately 165 megahertz. However, the clock used in this design can be considered as a design choice made to relax the design of the TDC, since the feasibility of such solution was confirmed in a VTC design. For the sake of comparison, however, this should be noted.

The figures-of-merit discussed in Section 2.4 are not used for the TDC comparison, since the VTC is a major source of nonlinearity and power consumption. Thus, the

comparison to full ADC designs would be misleading. Furthermore, the dynamic metrics are often not reported for standalone TDC designs, because of the undefined front-end sampling device. For this reason, the FOM comparison has been omitted here.

Table 7: The performance summary and comparison.

	2009 [40]	2012 [41]	2012 [42]	2014 [23]	This work
Architecture	2D-Vernier	SAR	Two-Step	Pipeline	MPRO
LSB [ps]	4.8	9.77	3.75	1.12	3.4
Resolution [bits]	7	10	7	9	10
Sample rate [MHz]	50	80	200	250	250 *
DNL [LSB]	1.0	1.4	2.3	1.7	1.25
INL [LSB]	3.3	2	0.9	0.6	1.25
Power [mW]	1.7	9.6	3.6	15.4	4.3

* The conversion duty cycle used in the simulations is 87.5 % for 250 MHz sample rate with 3.07 ns maximum input time interval.

6 Conclusion

Time-domain circuits can implement various signal processing applications using digital circuit structures, which benefit from technology scaling in deep sub-micron processes. Modern telecommunications applications require high-performance A/D-converters, and mobile applications place great importance on power consumption as well. Thus, power efficient high-performance converter circuits are key factors in the development of today's communications systems. Time-based ADCs can be used to carry out A/D-conversion in time-domain. Consequently, these time-based ADC architectures emphasize digital circuitry, which results in minimized area and power consumption in deeply scaled processes. An integral part of the TBADCs is the time-to-digital converter. By improving the design of the TDCs, the performance and power efficiency of TBADCs can be significantly improved.

In a time-based ADC, the TDC affects the main metrics of the ADC by defining the time resolution, full-scale range, and dynamic quantization performance by mitigating sampling errors. The time resolution is maximized by maximizing the oscillator frequency and number of phase taps in a ring oscillator -based TDC. The full-scale range can be increased by employing counters or multi-phase conversion. In this thesis, a ring oscillator -based TDC design was presented. The goal of the work was to design a high-speed TDC with 10-bit resolution, which can be used in a wide bandwidth time-based ADC. Using a free-running oscillator architecture allows efficient time-interleaving, which makes the proposed design suitable for wideband ADCs.

The chosen architecture is based on a multi-path ring oscillator (Section 5.3.1), which effectively improves the time resolution of the conversion by providing increased oscillation frequency. The designed MPRO consists of 32 stages and it oscillates at up to 10 gigahertz frequency, thus providing time resolution of approximately 3.4 picoseconds. The achieved time resolution enables conversion up to 10 bits at the desired frequency.

The full-scale range of the TDC is increased from the 5 bits of the MPRO to the full 10 bits by employing a counter. Due to the asynchronous sampling and the high frequency of the MPRO, a high-speed gray counter was implemented (Section 5.3.2). The counter speed was achieved by employing parallel carry propagation logic implemented using pseudo-NMOS elements, and dynamic TSPC flip-flops. As a future improvement, the frequency requirement of the counter could be relaxed by increasing the size of the MPRO from 5 bits to 6 bits, which would effectively half the frequency while maintaining the overall number of bits. This could allow the use of static CMOS elements in place of the pseudo-NMOs elements as well.

The error-free conversion output is achieved by sampling the oscillator and counter states with high-performance sense-amplifier flip-flops (Section 5.3.3) and correcting the counter sampling errors in the digital backend (Section 5.3.4). The implemented counter sampling error correction algorithm is based on selecting an error-free sample out of two counter samples based on the oscillator phase. The oscillator phase is encoded to binary via a thermometer-like code and corrected for temporal reordering using rudimentary bubble error correction. In future implementations, the falling

edge could be encoded as well in addition to the rising edge, which would provide an extra bit from the encoded oscillator phase. Additionally, an improved bubble correction method could be implemented to reduce noise and metastability induced errors in sampled oscillator phase.

The designed circuit was simulated and characterized in Section 5.4. In nominal conditions, the TDC reaches an SNDR of 57.10 decibels, which corresponds to an ENOB of 9.19 bits, with a full-scale range of 9.82 bits. The SFDR of the TDC is simulated at 63.28 decibels in nominal operating conditions. Spurious components were observed in the output spectra in the noise-free simulations. However, as was shown in Section 5.4.3, the addition of noise sources effectively smeared the spurs into the noise-floor. Consequently, the SFDR of the converter was improved by 6 decibels, resulting in SFDR of 69.27 decibels. The power consumption of the TDC was simulated at 4.3 milliwatts in the nominal operating conditions.

The target of this thesis was to design a TDC to be used in a wideband ADC application. The target performance for the TDC was 10 bits and 250 MS/s with minimized power and area consumption. Based on the system simulations, the TDC achieves linear operation close to 10 bits with time resolution of approximately 3.4 picoseconds. The ring oscillator -based architecture used together with a counter provides minimized area consumption, and the pre-layout simulations show the architecture to be power efficient as well. Based on the work done in this thesis, oscillator-based TDCs can provide high-performance quantizers for time-based ADCs.

References

[1] J. E. Park, Y. H. Hwang, and D. K. Jeong, "A 0.4-to-1 v voltage scalable *deltasigma* adc with two-step hybrid integrator for iot sensor applications in 65-nm lp cmos," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 12, pp. 1417–1421, Dec 2017.

[2] D. Johns and K. Martin, *Analog Integrated Circuit Design.* John Wiley & Sons Inc., 1997.

[3] R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, pp. 539–550, Apr 1999.

[4] C. C. Lee and M. P. Flynn, "A sar-assisted two-stage pipeline adc," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 4, pp. 859–869, April 2011.

[5] B. Murmann. Adc performance survey 1997-2017. [Online]. Available: http://web.stanford.edu/~murmann/adcsurvey.html

[6] S. Naraghi, M. Courcy, and M. P. Flynn, "A 9b 14 μw 0.06mm2 ppm adc in 90nm digital cmos," in *2009 IEEE International Solid-State Circuits Conference - Digest of Technical Papers*, Feb 2009, pp. 168–169,169a.

[7] T. Oh, H. Venkatram, and U. K. Moon, "A 70ms/s 69.3db sndr 38.2fj/conversion-step time-based pipelined adc," in *2013 Symposium on VLSI Circuits*, June 2013, pp. C96–C97.

[8] M. Miyahara, I. Mano, M. Nakayama, K. Okada, and A. Matsuzawa, "22.6 a 2.2gs/s 7b 27.4mw time-based folding-flash adc with resistively averaged voltage-to-time amplifiers," in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2014, pp. 388–389.

[9] Y. Song, C. H. Chan, Y. Zhu, L. Geng, S. P. U, and R. P. Martins, "Passive noise shaping in sar adc with improved efficiency," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 2, pp. 416–420, Feb 2018.

[10] S. Zhu, B. Xu, B. Wu, K. Soppimath, and Y. Chiu, "A skew-free 10 gs/s 6 bit cmos adc with compact time-domain signal folding and inherent dem," *IEEE Journal of Solid-State Circuits*, vol. 51, no. 8, pp. 1785–1796, Aug 2016.

[11] J. Kim, T. K. Jang, Y. G. Yoon, and S. Cho, "Analysis and design of voltage-controlled oscillator based analog-to-digital converter," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 18–30, Jan 2010.

[12] J. P. Jansson, P. Keränen, J. Kostamovaara, and A. Baschirotto, "Cmos technology scaling advantages in time domain signal processing," in *2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, May 2017, pp. 1–5.

[13] V. Unnikrishnan and M. Vesterbacka, "Mixed-signal design using digital cad," in *2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, July 2016, pp. 6–11.

[14] V. Unnikrishnan and M. Vesterbacka, "Time-mode analog-to-digital conversion using standard cells," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 12, pp. 3348–3357, Dec 2014.

[15] A. R. Macpherson, J. W. Haslett, and L. Belostotski, "A 5gs/s 4-bit time-based single-channel cmos adc for radio astronomy," in *Proceedings of the IEEE 2013 Custom Integrated Circuits Conference*, Sept 2013, pp. 1–4.

[16] H. Mostafa and Y. I. Ismail, "Highly-linear voltage-to-time converter (vtc) circuit for time-based analog-to-digital converters (t-adcs)," in *2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS)*, Dec 2013, pp. 149–152.

[17] B. Provost and E. Sanchez-Sinencio, "On-chip ramp generators for mixed-signal bist and adc self-test," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 2, pp. 263–273, Feb 2003.

[18] P. Dudek, S. Szczepanski, and J. V. Hatfield, "A high-resolution cmos time-to-digital converter utilizing a vernier delay line," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 2, pp. 240–247, Feb 2000.

[19] J. P. Jansson, A. Mantyniemi, and J. Kostamovaara, "A cmos time-to-digital converter with better than 10 ps single-shot precision," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 6, pp. 1286–1296, June 2006.

[20] H. Mostafa and Y. I. Ismail, "Modeling the limitations imposed by the timing jitter/skew on the time-based adc performance," in *2014 International Conference on Engineering and Technology (ICET)*, April 2014, pp. 1–5.

[21] J. Yu, F. F. Dai, and R. C. Jaeger, "A 12-bit vernier ring time-to-digital converter in 0.13 μm cmos technology," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 4, pp. 830–842, April 2010.

[22] K. Kim, Y. H. Kim, W. Yu, and S. Cho, "A 7 bit, 3.75 ps resolution two-step time-to-digital converter in 65 nm cmos using pulse-train time amplifier," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 4, pp. 1009–1017, April 2013.

[23] K. Kim, W. Yu, and S. Cho, "A 9 bit, 1.12 ps resolution 2.5 b/stage pipelined time-to-digital converter in 65 nm cmos using time-register," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 4, pp. 1007–1016, April 2014.

[24] M. A. Abas, G. Russell, and D. J. Kinniment, "Built-in time measurement circuits – a comparative design study," *IET Computers Digital Techniques*, vol. 1, no. 2, pp. 87–97, March 2007.

[25] A. Mantyniemi, T. Rahkonen, and J. Kostamovaara, "A cmos time-to-digital converter (tdc) based on a cyclic time domain successive approximation interpolation method," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 11, pp. 3067–3078, Nov 2009.

[26] S. Fairbanks, "High precision timing using self-timed circuits," University of Cambridge, Computer Laboratory, Tech. Rep. UCAM-CL-TR-738, Jan. 2009. [Online]. Available: http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-738.pdf

[27] A. A. Hafez and C. K. K. Yang, "Design and optimization of multipath ring oscillators," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 10, pp. 2332–2345, Oct 2011.

[28] J. R. Yuan, "Efficient cmos counter circuits," *Electronics Letters*, vol. 24, no. 21, pp. 1311–1313, Oct 1988.

[29] P. R. Thotat and A. K. Mal, "A high speed counter for analog-to-digital converters," in *2016 International Conference on Microelectronics, Computing and Communications (MicroCom)*, Jan 2016, pp. 1–5.

[30] M. Alioto, E. Consoli, and G. Palumbo, "Analysis and comparison in the energy-delay-area domain of nanometer cmos flip-flops: Part i - methodology and design strategies," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 5, pp. 725–736, May 2011.

[31] D. Rennie, D. Li, M. Sachdev, B. Bhuva, S. Jagannathan, S. Wen, and R. Wong, "Performance, metastability and soft-error robustness tradeoffs for flip-flops in 40nm cmos," in *2011 IEEE Custom Integrated Circuits Conference (CICC)*, Sept 2011, pp. 1–4.

[32] A. G. M. Strollo, D. D. Caro, E. Napoli, and N. Petra, "A novel high-speed sense-amplifier-based flip-flop," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 11, pp. 1266–1274, Nov 2005.

[33] J. Yuan and C. Svensson, "High-speed cmos circuit technique," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 1, pp. 62–70, Feb 1989.

[34] P. Larsson and C. Svensson, "Noise in digital dynamic cmos circuits," *IEEE Journal of Solid-State Circuits*, vol. 29, no. 6, pp. 655–662, Jun 1994.

[35] T. Kobayashi, K. Nogami, T. Shirotori, and Y. Fujimoto, "A current-controlled latch sense amplifier and a static power-saving input buffer for low-power architecture," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 4, pp. 523–527, Apr 1993.

[36] A. Abidi and H. Xu, "Understanding the regenerative comparator circuit," in *Proceedings of the IEEE 2014 Custom Integrated Circuits Conference*, Sept 2014, pp. 1–8.

[37] V. Stojanovic and V. G. Oklobdzija, "Comparative analysis of master-slave latches and flip-flops for high-performance and low-power systems," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 4, pp. 536–548, Apr 1999.

[38] R. Ginosar, "Metastability and synchronizers: A tutorial," *IEEE Design Test of Computers*, vol. 28, no. 5, pp. 23–35, Sept 2011.

[39] K. C. Choi, S. W. Lee, B. C. Lee, and W. Y. Choi, "A time-to-digital converter based on a multiphase reference clock and a binary counter with a novel sampling error corrector," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 59, no. 3, pp. 143–147, March 2012.

[40] A. Liscidini, L. Vercesi, and R. Castello, "Time to digital converter based on a 2-dimensions vernier architecture," in *2009 IEEE Custom Integrated Circuits Conference*, Sept 2009, pp. 45–48.

[41] H. Chung, H. Ishikuro, and T. Kuroda, "A 10-bit 80-ms/s decision-select successive approximation tdc in 65-nm cmos," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 5, pp. 1232–1241, May 2012.

[42] K. Kim, Y. Kim, W. Yu, and S. Cho, "A 7b, 3.75ps resolution two-step time-to-digital converter in 65nm cmos using pulse-train time amplifier," in *2012 Symposium on VLSI Circuits (VLSIC)*, June 2012, pp. 192–193.