

Design of readout channels for time-of-flight image sensors based on a 28-nm FPGA

by
Mojtaba Parsakordasiabi

A dissertation submitted to the Department of Physics in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the
UNIVERSITY OF SEVILLE

December 2022

© University of Seville 2022. All rights reserved.

Author
Mojtaba Parsakordasiabi
Department of Physics
June 2022

Certified by.
Ricardo Carmona-Galán
Tenured Scientist
Thesis Supervisor

Certified by.
Ángel Rodríguez-Vázquez
Full Professor
Thesis Supervisor

Acknowledgements

First and foremost, I would like to express my gratitude to Professor Ricardo Carmona-Galán, who has been extremely patient and supportive during my PhD. Thank you for believing in me during this whole difficult but beautiful journey. You are a brilliant leader. It is my pleasure to acknowledge Professor Ángel Rodríguez-Vázquez for his continuous support to the completion of this project. Your broad experience, knowledge, and achievements on time metrology systems have been very helpful to me. Thank you also for reading my papers and giving me valuable feedback. I am also grateful to Doctor Ion Vornicu, who devotes his time to assisting me and improving the quality of my work. I was lucky to work with such a complete team.

I am deeply grateful to Doctor Teresa Cervero-García, for giving me a very inspiring experience during my stay at Barcelona Supercomputing Center.

I would also like to express my gratitude to Instituto de Microelectrónica de Sevilla (IMSE-CNM) for hosting me. This institute has seemed like a second home to me, and I have had the privilege of sharing unforgettable experiences with great colleagues. I would like to thank everyone who has worked in PA-21, especially Delia, for making our office a pleasant place to work every day. I could not have had a better working environment.

Last but not least, I would like to express my heartfelt gratitude to my family. Many thanks to my lovely wife, Sahar, for your unending love and support throughout my PhD. You have been my best buddy, with whom I have been able to share all of my stressful and frustrating moments. My beloved parents, thank you for your unconditional support, love, and commitment over the years, as well as your special attention to my education. Thank you for always being at my side and helping me through different stages of my life.

This work was supported by EU H2020 MSCA through Project ACHIEVE-ITN (Grant No 765866), by the Spanish MINECO and European Region Development Fund (ERDF/FEDER) through Project RTI2018-097088-B-C31 and by the US Office of Naval Research through Grant No. N00014-19-1-2156.

Acronyms

ASIC Application-Specific Integrated Circuit

BRAM Block Random-Access Memory

CDT Code Density Test

CLB Configurable Logic Block

DNL Differential Non-Linearity

FLIM Fluorescence Lifetime Imaging Microscopy

FPGA Field-Programmable-Gate-Array

INL Integral Non-Linearity

LiDAR Light Detection and Ranging

LUT Lookup Table

MMCM Mixed-Mode Clock Manager

PET Positron Emission Tomography

SPAD Single-Photon Avalanche Diode

SSP Single-Shot Precision

T2B Thermometer-to-Binary

TDC Time-to-Digital Converter

TDL Tapped-Delay-Line

ToF Time-of-Flight

Abstract

This thesis presents a contribution to the design of readout channels for time-of-flight image sensors. Specifically, the focus has been on the development of time-to-digital converters (TDCs) based on a 28-nm field-programmable-gate-array (FPGA). TDCs are used in a wide range of applications where time measurement is required. This thesis proposes the study of FPGA-based TDCs to optimize their performance in terms of resolution, measurement throughput, precision, linearity, resources usage, and power consumption. As a result, in this project, we focus on the following objectives:

- Reaching high-resolution TDCs required in many applications
- Reducing the TDC resources usage while preserving the other specifications of TDC for multi-channel configuration
- Maximizing the measurement throughput to achieve high-speed high-detection efficiency ToF sensors
- Improving the TDC linearity to reach high-accuracy measurements

Pushing these requirements to the limit is challenging, although it is constantly required by many applications. This thesis presents three FPGA-based TDC architectures delivering high performance with low resource usage. The first proposed FPGA-based TDC presents an architecture to achieve high performance with low usage of resources. It consists of a synchronizing input stage, a tuned tapped delay line (TDL), a combinatory encoder of ones and zeros counters, and an online calibration stage. The second architecture presents a new approach for dead-time minimization while preserving low resource usage and high resolution in FPGA-based TDC. This architecture consists of a toggling input stage, a TDL, a dual-mode counter-based encoder, a coarse counter, and a bin width calibration stage. The minimum dead-time of TDL TDCs is two clock cycles. This architecture reduced dead-time to one clock cycle. The last proposed FPGA-based TDC presents a dual-mode TDL —propagating 1's and 0's in alternating measurement cycles— architecture that complies with the mentioned specifications. The dead-time of the proposed TDC is one system clock cycle by using a toggling input stage and a dual-mode counter-based encoder. To improve the TDC linearity, the TDL sampling sequence is tuned separately for each operating mode. The presented architecture employs a low-resources dual-mode combinatory encoder of one- and zero-counters to remove the bubbles and cover both operating modes. A dual-mode bin-width calibration has been carried out to improve the TDC performance in each mode.

The proposed architectures have been evaluated and characterized on a 28-nm Xilinx Artix-7 FPGA. The presented results are the evidence of the validity of the approach to reach high performance while maintaining a low use of resources and low power consumption.

TABLE OF CONTENTS

1. READOUT CHANNELS FOR DIRECT TIME-OF-FLIGHT IMAGE SENSORS.....	17
1.1 INTRODUCTION	17
1.2 OBJECTIVES	19
1.3 PERFORMANCE METRICS	19
1.3.1 Resolution	19
1.3.2 Measurement Range.....	20
1.3.3 Nonlinearity Parameters.....	20
1.3.4 Precision	20
1.3.5 Dead-time.....	20
1.3.6 Power Consumption.....	20
1.3.7 Resources Usage.....	21
1.4 RESULTS	21
1.5 CONCLUSION.....	25
1.6 THESIS ORGANIZATION	26
2. A LOW-RESOURCES TDC FOR MULTI-CHANNEL DIRECT TOF READOUT BASED ON A 28-NM FPGA	28
2.1 ABSTRACT	28
2.2 INTRODUCTION	29
2.3 TDC ARCHITECTURE	34
2.4 EXPERIMENTAL RESULTS.....	39
2.4.1 Measurements	39
2.4.2 Comparison.....	47
2.5 CONCLUSIONS.....	49
3. A NOVEL APPROACH FOR MEASUREMENT THROUGHPUT MAXIMIZATION IN FPGA-BASED TDCS.....	51
3.1 ABSTRACT	51
3.2 INTRODUCTION	52
3.3 TDC ARCHITECTURE	54
3.4 TOGGLING INPUT STAGE	56
3.5 DUAL-MODE COUNTER-BASED ENCODER.....	58
3.6 EXPERIMENTAL RESULTS.....	60
3.7 SYSTEM FEATURES AND COMPARISON.....	66
3.8 CONCLUSION.....	68
4. AN EFFICIENT TDC USING A DUAL-MODE RESOURCE-SAVING METHOD EVALUATED IN A 28-NM FPGA	69
4.1 ABSTRACT	69
4.2 INTRODUCTION	70
4.3 REVIEW OF FPGA-BASED TDC TECHNIQUES	72

4.3.1	<i>Fine time interpolation</i>	72
4.3.2	<i>Input stage</i>	75
4.3.3	<i>Thermometer-to-binary encoder</i>	76
4.3.4	<i>Online calibration</i>	77
4.3.5	<i>Dead-time</i>	78
4.4	PROPOSED TDC ARCHITECTURE	78
4.4.1	<i>Target device technology</i>	80
4.4.2	<i>Dual-mode tuned TDL</i>	81
4.4.3	<i>Toggling input stage</i>	82
4.4.4	<i>Dual-mode combinatory counter-based encoder</i>	83
4.4.5	<i>Dual-mode bin width calibrator</i>	85
4.5	PERFORMANCE EVALUATION	86
4.5.1	<i>Experimental results</i>	86
4.5.2	<i>Comparison with state-of-the-art FPGA-based TDCs</i>	96
4.6	CONCLUSION	97
	BIBLIOGRAPHY	100
	SCIENTIFIC PUBLICATIONS	107
	JOURNAL PAPERS	107
	CONFERENCE PAPERS	107
	WORKSHOPS	107
	MEDIA APPEARANCES	108

LIST OF FIGURES

Figure 2.1. Architecture of the proposed FPGA-based TDC.....	35
Figure 2.2. Simplified diagram of the structure of the CARRY4 block.	35
Figure 2.3. Combinatory encoder of ones and zeros counters.....	38
Figure 2.4. Evaluation board of the proposed TDC.....	40
Figure 2.5. Measured bin widths of a TDC with “SCSS” sampling pattern resulted from code density test.	40
Figure 2.6. DNL results of “SCSS” and “CCCC” sampling patterns.....	42
Figure 2.7. INL results of “SCSS” and “CCCC” sampling patterns.	43
Figure 2.8. Bin width distributions of “SCSS” and “CCCC” sampling patterns.....	43
Figure 2.9. (a) Comparison of the TDC calibration table content and the ideal transfer function. (b) The differences between the codes.	45
Figure 2.10. Measurement histogram of a constant time interval.....	45
Figure 2.11. RMS precision of the different time intervals.	45
Figure 2.12. RMS precision variations with temperature.	46
Figure 2.13. Comparison of FoM_TDC.	49
Figure 3.1. Simplified schematic of CARRY4 structure	55
Figure 3.2. The simplified architecture of the proposed TDC	56
Figure 3.3. (a) Toggling input stage (b) Timing diagram.....	57
Figure 3.4. Dual-mode counter-based encoder	59
Figure 3.5. Encoder controller	60
Figure 3.6. Bin widths of the proposed TDC.....	61
Figure 3.7. Bin widths distribution of the proposed TDC	62
Figure 3.8. DNL of the uncalibrated TDC.....	63
Figure 3.9. INL of the uncalibrated TDC	63
Figure 3.10. (a) TDC calibrated and ideal transfer functions (b) Their differences	64
Figure 3.11. DNL of the calibrated TDC.....	65

Figure 3.12. INL of the calibrated TDC	65
Figure 3.13. RMS precision for for different time intervals	66
Figure 4.1. Architecture of the proposed TDC	80
Figure 4.2. Calibration procedure in each operating mode.....	86
Figure 4.3. DNL and INL of “SCSC” in (a) “1” (b) “0” propagation modes.....	89
Figure 4.4. (a) Estimated bin widths and (b) bin width histogram of ‘SCSC’ sequence in “1” propagation mode.....	90
Figure 4.5. (a) Estimated bin widths and (b) bin width histogram of ‘SCSC’ sequence in “0” propagation mode.....	91
Figure 4.6. The content of the calibration tables (a) in “1” propagation mode (b) in “0” propagation mode.....	92
Figure 4.7. DNL and INL of the calibrated TDC for (a) in “1” propagation mode (b) in “0” propagation mode.....	93
Figure 4.8. TDC precision	94
Figure 4.9. TDC precision variations over temperature	95

LIST OF TABLES

Table 1.1. Characteristics of the TDC presented in [1]	22
Table 1.2. Resources and power consumption of one channel presented in [1].....	22
Table 1.3. Characteristics of the TDC presented in [2]	23
Table 1.4. Resources and power consumption of one channel presented in [2].....	23
Table 1.5. Characteristics of the TDC presented in [3]	24
Table 1.6. Resources and power consumption of one channel presented in [3].....	24
Table 2.1. Summary of the ones and zeros combinatory counters encoder.....	38
Table 2.2. Sampling patterns comparison.....	42
Table 2.3. Resources usage and power consumption of one TDC channel.....	46
Table 2.4. Characteristics of the proposed TDC.....	46
Table 2.5. Comparison with the state-of-the-art FPGA-based TDCs.....	47
Table 3.1. Resources and power consumption for one TDC channel.....	66
Table 3.2. Characteristics of the proposed TDC.....	67
Table 3.3. Comparison with the other FPGA-based TDCs	67
Table 4.1. Fine Interpolators Comparison	75
Table 4.2. Comparison of FPGA Platforms.....	81
Table 4.3. Characteristics of the encoder.....	85
Table 4.4. DNL and INL of uncalibrated TDC for ‘CCCC’, ‘SCSS’, and ‘SCSC’ in both operating modes	87
Table 4.5. DNL and INL of calibrated TDC for ‘CCCC’, ‘SCSS’, and ‘SCSC’ in both operating modes	89
Table 4.6. Characteristics of the proposed TDC.....	95
Table 4.7. Resources and power consumption of one channel.....	96
Table 4.8. Comparison with the state-of-the-art FPGA-based TDCs.....	98

To the soul of my beloved nephew, Abolfazl.

1. Readout channels for direct time-of-flight image sensors

1.1 Introduction

Artificial vision is not restricted to light-intensity maps. Depth sensing technologies are now reaching the level of maturity required for reliable deployment in real scenarios. They provide an enriched representation of the scene, thereby broadening the application of vision systems. At sensor level, techniques like photon-counting are enabling radical departures from classical vision algorithms. New circuit structures improve the accuracy of depth measurements and are notably reducing the form factor of typically bulky systems. A promising alternative to realize single-sensor depth estimation in CMOS technology are avalanche diodes, also known as single-photon avalanche diodes (SPAD). SPAD is a photodetector made up of a p-n junction that has been reverse-biased above its breakdown voltage. A single photon hitting the junction in this unstable state, known as Geiger-mode, can spark a self-sustaining avalanche, resulting in a measurable current flow. The accompanying electronics can then time or count these observed single photon events, which are produced as digital pulses. SPADs are desirable photodetector solutions for time-dependent applications like 3D imaging because of their high sensitivity, quick response time, and low timing

jitter.

One of the most relevant characteristics in 3D imaging and ranging applications based on SPAD is the Time-of-Flight (ToF) capability. ToF sensors can capture distances by estimating the travel time of an emitted signal. ToF sensors are classified into two groups. Indirect time-of-flight (iToF) sensors measure light intensity in predetermined time frames and calculate depth as a post-processing step, whereas direct time-of-flight (dToF) sensors directly measure the ToF of a light pulse.

A dToF sensor works by emitting pulses of light and measuring the time between emission and detection using a timing circuit, commonly a time-to-digital converter (TDC). In 3D imaging applications, the light pulse is emitted toward the scene, scatters off the object, and is detected by the sensor.

This project is centered around the design of a readout channel for dToF image sensors. The Time-to-Digital Converter (TDC) is a central component of a readout channel.

TDCs are used to measure time intervals and have long been employed in ToF applications. TDC is a device that converts a time interval into a binary number. In modern TDCs, it is critical to increase the TDC resolution. For instance, high-resolution TDCs are highly demanded for 3D imaging. In addition, real-time operations require high throughput. Another important characteristic is that simultaneous measurement of a huge number of parallel detector modules is required, which means that the required accuracy and speed should be achieved with a reduced number of resources. Under these circumstances, timing resolution scales with the gate delay, therefore, advances in CMOS fabrication technology help realize higher-resolution TDCs. However, compared to application-specific integrated circuits (ASICs), field-programmable gate arrays (FPGAs) offer more flexibility, shorter time-to-market, and lower development costs. For these reasons,

FPGAs are suitable platforms for the implementation of fully digital TDCs. Additionally, the FPGAs fabricated in the finest silicon technologies have the intrinsic delay of the carry elements of sub-ten picoseconds. This makes a good fine-time interpolator, rendering FPGAs an interesting option for high-resolution TDCs. Furthermore, considering the available resources on each FPGA, they are a suitable option also to implement multi-channel TDCs.

Most FPGA-based TDCs are based on the Nutt interpolation method. They employ a coarse counter with a few nanoseconds resolution, running at the system clock frequency, and a fine time interpolator that allows reaching sub-clock-period resolution. In this way, a long time interval of up to several hundreds of microseconds can be covered while achieving a fine resolution down to a few picoseconds.

1.2 Objectives

This project focuses on achieving high-resolution TDCs, reducing TDC resources usage while maintaining other TDC specifications for multi-channel architectures, maximizing measurement throughput to reach high-speed high-detection efficiency ToF sensors, and improving TDC linearity to achieve high-accuracy measurements.

1.3 Performance Metrics

The principal characteristics to be sought for in FPGA-based TDCs are resolution, measurement range, nonlinearity, precision, measurement throughput, resources usage, and power consumption.

1.3.1 Resolution

Resolution is the minimum input time interval that the TDC can measure. In an

input-output curve, being the input a time interval between two signals, and the output a digital number that represents the magnitude of the interval, the resolution is the magnitude of the least significant bit (LSB).

1.3.2 Measurement Range

The measurement range is the longest time interval that the TDC can measure before overflowing.

1.3.3 Nonlinearity Parameters

Nonlinearity deviates the input-output curve from its ideal characteristic. It is usually due to many factors, like delay errors and mismatch, signal cross-talk and process, voltage and temperature (PVT) variations. The parameters that characterize nonlinearity INL and DNL (integral and differential nonlinearities) are usually referenced to the LSB.

1.3.4 Precision

The TDC precision, also known as single-shot precision (SSP) or standard deviation, is a parameter that specifies how far a measurement might deviate from its predicted value. Due to noise sources, like the jitter in the clocks and delay lines, the same input interval renders different output values.

1.3.5 Dead-time

The time that TDC needs to perform a measurement and be able to accept a new input is defined as dead-time. Measurement throughput is the inverse of the dead-time.

1.3.6 Power Consumption

Digital devices' power consumption is generally divided into two categories:

dynamic and static. Dynamic power is proportional to the clock frequency, but static power is proportional to the technology employed. When comparing two systems, the total power consumption may be used. Because of this technological reliance, the clock frequency and fabrication technology must be regarded when comparing the power consumption of two architectures.

1.3.7 Resources Usage

In FPGA-based TDCs, the quantity of utilized FPGA resources such as flip-flops (FFs), lookup tables (LUTs), and block RAMs (BRAMs) is commonly employed as a parameter for system size.

1.4 Results

An Artix-7 FPGA (XC7A200T-1FBG484), embedded in an Opal Kelly XEM7310 board, has been used to evaluate the performance of the proposed TDCs. The application programming interface (API) components of the Opal Kelly are employed to send the measurement results to a host PC through a USB connection. To generate an uncorrelated signal for the code density test, the CFGMCLK port of the STARTUPE2 primitive is employed. This signal is generated by the internal oscillator of the Artix-7 and therefore, it does not have any correlation with the system clock. The code density test is performed by measuring more than one hundred thousand events. The presented architectures employ a 250 MHz clock frequency and can measure input intervals beyond 260 μ s. We have provided full electrical characterization, including power consumption and resource usage estimation. These parameters are important in portable systems for distance ranging applications based on direct ToF, which requires multiple parallel channels.

In [1], we have tested all the sampling patterns to find out the one rendering the

highest linearity. We have employed a combinatory ones and zeros counters encoder to achieve high immunity to bubbles in the TDL, composed of different sampling elements with opposite logic states. The obtained resolution and single-shot precision are 22.2 ps and 26.04 ps, respectively. The measurement throughput is 125 MSa/s. The experimental results of the TDC show a DNL in the range of $[-0.953, 1.185]$ LSB, and an INL within $[-2.750, 1.238]$ LSB. Moreover, the proposed architecture requires low FPGA resources.

Table 1.1 and Table 1.2 display the measured characteristics and the use of resources and power consumption of one single TDC channel—extracted from the post place & route report—of the proposed architecture in [1].

Table 1.1. Characteristics of the TDC presented in [1]

Parameter	Value/Range	Unit
Clock Freq.	250	MHz
LSB	22.2	ps
Meas. Range	262.14	μ s
DNL	$[-2.750, 1.238]$	LSB
INL	$[-0.953, 1.185]$	LSB
Dead-Time	2	Clock Cycle
Readout Speed	125	MS/s
SSP	26.04	ps

Table 1.2. Resources and power consumption of one channel presented in [1]

Resource	Available	Utilization	Utilization (%)
LUT	133,800	216	0.16
FF	267,600	638	0.24
BRAM	365	2.50	0.68
Total Power		164 mW	
Dynamic Power		33 mW	

In [2], to minimize the dead-time of TDCs with low resource usage, a novel approach has been introduced. The dead time is reduced to one clock cycle, i.e., 4

ns for a 250-MHz system clock frequency. The LSB size and single-shot precision are 22.1 ps and 28.43 ps, respectively. This architecture can measure time intervals with 250 MSa/s conversion rate. These results show that the proposed architecture is suitable for applications requiring high-throughput in multiple channels. The measurement results of the proposed TDC show [-0.80, 1.34] LSB DNL and [-0.73, 1.97] LSB INL.

Table 1.3 and Table 1.4 display the measured characteristics and the use of resources and power consumption of one single TDC channel —extracted from the post place & route report— of the proposed architecture in [2].

Table 1.3. Characteristics of the TDC presented in [2]

Parameter	Value/Range	Unit
Clock Freq.	250	MHz
LSB	22.1	ps
Meas. Range	262.14	μs
DNL	[-0.80 1.34]	LSB
INL	[-0.73 1.97]	LSB
Dead-Time	1	Clock Cycle
Readout Speed	250	MS/s
SSP	28.43	ps

Table 1.4. Resources and power consumption of one channel presented in [2]

Resource	Available	Utilization	Utilization (%)
LUT	133,800	228	0.17
FF	267,600	678	0.25
BRAM	365	2.50	0.68
Total Power		171 mW	
Dynamic Power		40 mW	

In [3], we have proposed an FPGA-based TDC based on a dead-time-minimizing and resource-saving approach. The dead-time is reduced to one clock cycle by using a toggling input stage and a dual-mode combinatory encoder of 1's and 0's

counters. The presented encoder is robust against bubble errors while using low resources. To improve the linearity, the most linear sampling sequence is exploited and the encoder outputs are calibrated using bin-width calibration. The measured LSB resolution and TDC precision are 22.1 ps and 22.35 ps, respectively. This architecture can measure time intervals with 250 MSa/s measurement throughput. The experimental results have shown a DNL within [-0.71 1.05] LSB and an INL within [-0.85 0.86] LSB for the propagation of 1's. DNL and INL are within [-0.73 1.06] LSB and [-1.17 0.04] LSB, respectively, for the propagation of 0's. Table 1.5 and Table 1.6 display the measured characteristics and the use of resources and power consumption of one single TDC channel —extracted from the post place & route report— of the proposed architecture in [3].

Table 1.5. Characteristics of the TDC presented in [3]

Parameter	Value/Range	Unit
Clock Freq.	250	MHz
LSB	22.1	ps
Meas. Range	262.14	μ s
DNL (“1” mode)	[-0.71 1.05]	LSB
INL (“1” mode)	[-0.85 0.86]	LSB
DNL (“0” mode)	[-0.73 1.06]	LSB
INL (“0” mode)	[-1.17 0.04]	LSB
Dead-Time	1	Clock Cycle
Readout Speed	250	MS/s
SSP	22.35	ps

Table 1.6. Resources and power consumption of one channel presented in [3]

Resource	Available	Utilization	Utilization (%)
LUT	133,800	228	0.17
FF	267,600	678	0.25
BRAM	365	2.50	0.68

Total Power	164 mW
Dynamic Power	33 mW

1.5 Conclusion

This thesis represents three FPGA-based TDC architectures delivering high performance with low resource usage. The proposed architectures have been evaluated and characterized on a 28-nm Xilinx Artix-7 FPGA. The presented TDCs could be implemented in other types of Xilinx FPGAs by applying a few changes to the HDL code.

The architecture reported in [1] shows the following strengths:

- The selection of the best S-C combination improves linearity. By searching for the most uniform configuration of the delay line, the linearity improves without time resolution degradation and additional dead time and resource usage.
- The online calibration resulted from a code density test, improving accuracy even further.
- The synchronization module consists of only two FFs, efficiently shaping any input pulse;
- The ones-zeros encoder requires low resource usage; it features a mere 8-ns propagation time. Moreover, it is robust against bubble errors, without requiring any additional correction logic.
- Reference frequency optimization for short TDL. It is adapted to the FPGA speed grade. Measurement throughput is 125 Msamples/s.

The architectures described in [2][3] improve the measurement throughput, linearity, and precision by employing the additional following design options:

- The implementation of dual-mode —propagating 1’s and 0’s in alternating measurement cycles— tuned TDL with improved the linearity in both operating modes. This improvement is achieved by testing all the possible sampling patterns and finding the configuration that provides the most linear output.
- Linearization needs to be achieved without incurring extra dead-time, time resolution degradation, and unreasonable additional resource utilization.
- The reduction of the dead-time to one single clock period by using a toggling input stage that eliminates the need to reset the TDL. Measurement throughput is 250 Msamples/s.
- The minimization of bubble errors by using a pipelined encoding of the count of 1’s and 0’s present in the TDL.

The presented results are the evidence of the validity of the approach to deliver high performance while maintaining a low use of resources and low power consumption. The proposed TDCs feature a high-throughput, high-precision, and low-resources therefore is well-suited for high-speed high-accuracy multi-channel applications such as LiDAR and ToF-PET systems.

1.6 Thesis Organization

The following chapters present three [1]-[3] of our five publications [1]-[5]. In the first presented paper [1], which was published in MDPI Sensors journal, we propose a low resources FPGA-based TDC architecture to achieve high performance. This TDC can be employed for multi-channel direct ToF applications. The proposed architecture consists of a synchronizing input stage, a tuned tapped delay line (TDL), a combinatory encoder of ones and zeros counters, and an online calibration stage.

The second presented paper [2], which was published in 7th International

Conference on Event-Based Control, Communication, and Signal Processing (EBCCSP), proposes a new approach for dead-time minimization while preserving low resource usage and high resolution in FPGA-based TDCs. The presented architecture consists of a toggling input stage, a tapped delay line (TDL), a dual-mode counter-based encoder, a coarse counter, and a bin width calibration stage. The minimum dead-time of TDL TDCs is two clock cycles. The proposed architecture reduced dead-time to one clock cycle.

The last presented paper [3], which was published in IEEE Transactions on Instrumentation and Measurement, proposes a dual-mode TDL —propagating 1's and 0's in alternating measurement cycles— architecture for an FPGA-based TDC. The dead-time of the proposed TDC is reduced to one system clock cycle by using a toggling input stage and a dual-mode counter-based encoder. To improve the TDC linearity, the TDL sampling sequence is tuned separately for each operating mode. The presented architecture employs a low-resources dual-mode combinatory encoder of one- and zero-counters to remove the bubbles and cover both operating modes. A dual-mode bin-width calibration has been carried out to improve the TDC performance in each mode.

2. A Low-Resources TDC for Multi-Channel Direct ToF Readout Based on a 28-nm FPGA

This work has been published in:

M. Parsakordasiabi, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “A Low-Resources TDC for Multi-Channel Direct ToF Readout Based on a 28-nm FPGA,” Sensors, vol. 21, no. 1, p.308, 2021.

The article belongs to the Special Issue **SPAD Image Sensors**.

Impact Factor: 3.576 (2nd flagship journal in the field of FPGA-based TDCs)

Quartile: Q1 (Instruments & Instrumentation)

Number of Citations (Google Scholar): 15 (2022, December 23)

Article Full-text Views: 2936 (2022, December 23)

2.1 Abstract

In this paper, we present a proposed field programmable gate array (FPGA)-based time-to-digital converter (TDC) architecture to achieve high performance with low usage of resources. This TDC can be employed for multi-channel direct Time-of-Flight (ToF) applications. The proposed architecture consists of a synchronizing input stage, a tuned tapped delay line (TDL), a combinatory encoder of ones and zeros counters, and an online calibration stage. The experimental

results of the TDC in an Artix-7 FPGA show a differential non-linearity (DNL) in the range of $[-0.953, 1.185]$ LSB, and an integral non-linearity (INL) within $[-2.750, 1.238]$ LSB. The measured LSB size and precision are 22.2 ps and 26.04 ps, respectively. Moreover, the proposed architecture requires low FPGA resources.

2.2 Introduction

Time-to-Digital Converters (TDCs) play a key role in a broad range of applications that require time measurement. One of the most relevant characteristics in three-dimensional (3D) imaging and ranging applications based on Single-Photon Avalanche Diode (SPAD) is the direct Time-of-Flight (ToF) capability [6]. To this end, high-resolution TDCs are highly demanded for 3D imaging [7][8], Fluorescence Lifetime Imaging Microscopy (FLIM) [9][9], and Positron Emission Tomography (PET) [11][12]. Furthermore, the increase in the number of detector modules and the requirement for real-time acquisition have led to the widespread utilization of multi-channel TDCs.

In recent years, Field-Programmable Gate Arrays (FPGAs) have been considered as an interesting implementation platform for fully-digital TDCs because of their flexibility, faster development phase, and lower implementation cost than Application-Specific Integrated Circuits (ASICs). Additionally, FPGA's carry elements, whose intrinsic propagation delays can be used as a sort of fine time interpolator, have made FPGAs a suitable solution to implement high-resolution TDCs [13][14].

Different techniques for implementing TDCs on FPGA have been introduced in recent years [4][14], depending on the application's specific requirements. Seeking to expand the measurable time interval and achieve higher time resolutions, the Nutt method, which combines a coarse counter and a time

interpolator, is the most extended technique in FPGA-based TDCs [15][16]. There are different approaches in the literature to implement the time interpolator, such as Tapped Delay Lines (TDLs) [15][17], Vernier Delay Lines (VDLs) [18][19], multiple clock phases [20][21], delay-line loop-shrinking [22], and stochastic TDCs such as a matrix of counters [23].

As a straightforward time interpolator, a TDL [15][17] employs the carry elements of the FPGAs as delay elements. The intrinsic propagation delay of the delay elements determines the resolution. In VDLs [18][19], which employ more resources, the resolution is determined by the difference of the delays in two different chains of delay elements. With the increasing improvement of FPGAs manufacturing process, both of these methods can achieve a sub-hundred-picosecond resolution. Multiple-phase clock interpolators [20][21] use different clock phases of the reference clock to reach sub-clock resolutions. Since only a few different phases of the main clock are usually available, the best achievable resolution in this method is limited. Another time interpolator is based on the delay-line loop-shrinking technique [22]. It consists of two delay-line loops which are similar in architecture and delay cells but different in routing and placement. These differences determine the resolution in this method. The main weakness of this approach is that the dead-time depends on the length of the interval. It may not be an appropriate technique for applications that cover long time intervals. Finally, in the method based on a matrix of counters [23] as a stochastic TDC, the delay cells are the routing resources that are built of metal tracks and insensitive to the drift of FPGA core voltage and ambient temperature. Although this method can reach high resolution, it employs more resources than the others, and thus, this method is not a suitable time interpolator for multi-channel purposes. In addition, it uses a large area to build the routing paths.

There are also other time interpolation techniques with better performance, such as wave union TDC [24], multi-chain TDL [25], dual-phase TDL [26], and Ring-Oscillator-based (RO-based) multi-measurement TDL [27]. To improve the TDL resolution without additional delay lines and reduce the nonlinearity, wave union TDCs measure multiple transitions generated by wave union launchers. In multi-chain TDL TDCs, each channel has more than one TDL, and the output code of each channel is obtained by averaging all output codes of TDLs. Dual-phase TDL TDCs consider two TDLs for each channel, and each of the TDLs covers a half of the clock period. This interpolation method replaces a long delay line with two shorter delay lines to minimize the clock skew. RO-based multi-measurement TDL uses a ring oscillator to improve time resolution. These methods enhance the linearity of the TDC at the expense of high resources consumption and/or additional dead-time. For high-resolution multi-channel applications that require as few resources as possible for each channel while achieving sub-hundred-picosecond resolution, TDL is the best choice. Won and Lee [28] improved the linearity of the TDL in FPGAs by introducing a tuned sampling pattern that selects different outputs of the carry elements as the outputs of the delay line. In their proposed TDC, changing the sampling pattern requires more resources.

Another challenging block of an FPGA-based TDC is the thermometer-to-binary (T2B) encoder, which converts the delay line state to a binary code. Traditional encoding approaches generate the output code by finding the transition point in the delay line (“one-hot” binary encoder), but it can be severely affected by bubble errors. There are several online and offline techniques to minimize these errors, such as bubble-proof encoding [29], bin realignment [30], and stepped-up tree encoder [31]. Wang and Liu [32] used both of the bin realignment and bin decimation techniques to minimize the nonlinearity. To improve the linearity performance, Chen and Li [33] integrated several techniques such as sub-tapped

delay line averaging, tap timing tests, a compensated histogram, and a mixed calibration method. All these techniques decrease the bubble problem at the cost of increased dead-time and/or higher resource utilization and/or LSB size degradation. Wang et al. [34] used a ones-counter encoder, which only counts the number of ones in the TDL and converts it to a binary number. A ones-counter encoder has the global ability to correct for bubbles because it does not depend on the tap sequence.

An important issue in FPGA-based TDCs is the non-uniformity of the delay elements from the carry chain, caused by process variation and mismatch. It is reflected in large Differential and Integral Nonlinearities (DNL and INL). Therefore, calibration becomes crucial for FPGA-based TDCs. The average delay method [35] and the bin-by-bin estimation approach [36] have been proposed for calibration. Although the former is a faster technique, the latter is better suited to FPGA-based TDL TDCs, because the sizes of the TDL delay elements have large differences. The bin-by-bin method is feasible by using a statistical estimation approach named code density test [37]. A table containing the measured bin widths of each delay cell can be stored in a Random-Access Memory (RAM) [38][39] to implement the online calibration. The bin widths are either fixed [38] during the time intervals measurements or updatable [39]. These online updatable calibration tables contemplate the ambient changes while measuring the time intervals. The bin widths are then updated at the cost of more logic resources and/or decreased conversion rates.

The input stage is another important block of the TDC. The input signal may be, on the one hand, noisy and, on the other hand, either longer or shorter than the required width. Hence, the pulse needs to be filtered and its width should be equalized before being injected into the TDL. Additionally, the input stage detects the input event and sends an enabling signal to the next blocks of the TDC to

inform them about receiving a new input signal. Different mechanisms have been proposed for the input stage [40]-[42]. Most of them consider the input signal as the clock of the flip-flop (FF). Tontini et al. [42] proposed an input stage that is highly synchronized and requires only one extra flip-flop.

The architecture reported in this paper shows the following strengths:

- Calibration technique

1. The selection of the best S-C combination improves linearity. By searching for the most uniform configuration of the delay line, the linearity improves without time resolution degradation and additional dead time and resource usage.

2. The online calibration resulted from a code density test, improving accuracy even further.

- Compactness

1. The synchronization module consists of only two FFs, efficiently shaping any input pulse;

2. The ones-zeros encoder requires low resource usage; it features a mere 8-ns propagation time. Moreover, it is robust against bubble errors, without requiring any additional correction logic.

3. Reference frequency optimization for short TDL. It is adapted to the FPGA speed grade. With this approach, we can implement 400 TDC channels at 125 Msamples/s.

- Full electrical characterization

1. We have provided full electrical characterization, including power consumption and resource usage estimation. These parameters are important in portable systems for distance ranging applications based on direct ToF, which requires multiple parallel channels.

The rest of the paper is organized as follows. The proposed low-resource FPGA-based tuned-TDL TDC, which uses a combinatory encoder of the time interpolator outputs, is described in Section 2.3. The evaluation procedure, the characterization of the TDC performance, and the comparison to the state-of-the-art works are provided in Section 2.4. Finally, Section 2.5 summarizes and concludes the article.

2.3 TDC Architecture

Figure 2.1 shows the architecture of the proposed FPGA-based TDC. It consists of an input stage, a coarse counter, a tuned-sampling-pattern TDL, a combinatory encoder of ones and zeros counters, and an online calibration block.

Modern FPGAs include Configurable Logic Blocks (CLB) which provide high-performance logic such as carry elements. A carry element is a dedicated high-speed component which is usually employed to implement fast arithmetic functions. The TDL in this paper employs cascade carry elements, each producing a short propagation delay. The implementation platform is Xilinx Artix-7 XC7A200T-1FBG484 (Xilinx Inc., San Jose, CA, USA), which is embedded in an Opal Kelly XEM7310 board (Opal Kelly Inc., Portland, OR, USA). The simplified structure of the CARRY4 block in these series-7 FPGAs is shown in Figure 2.2.

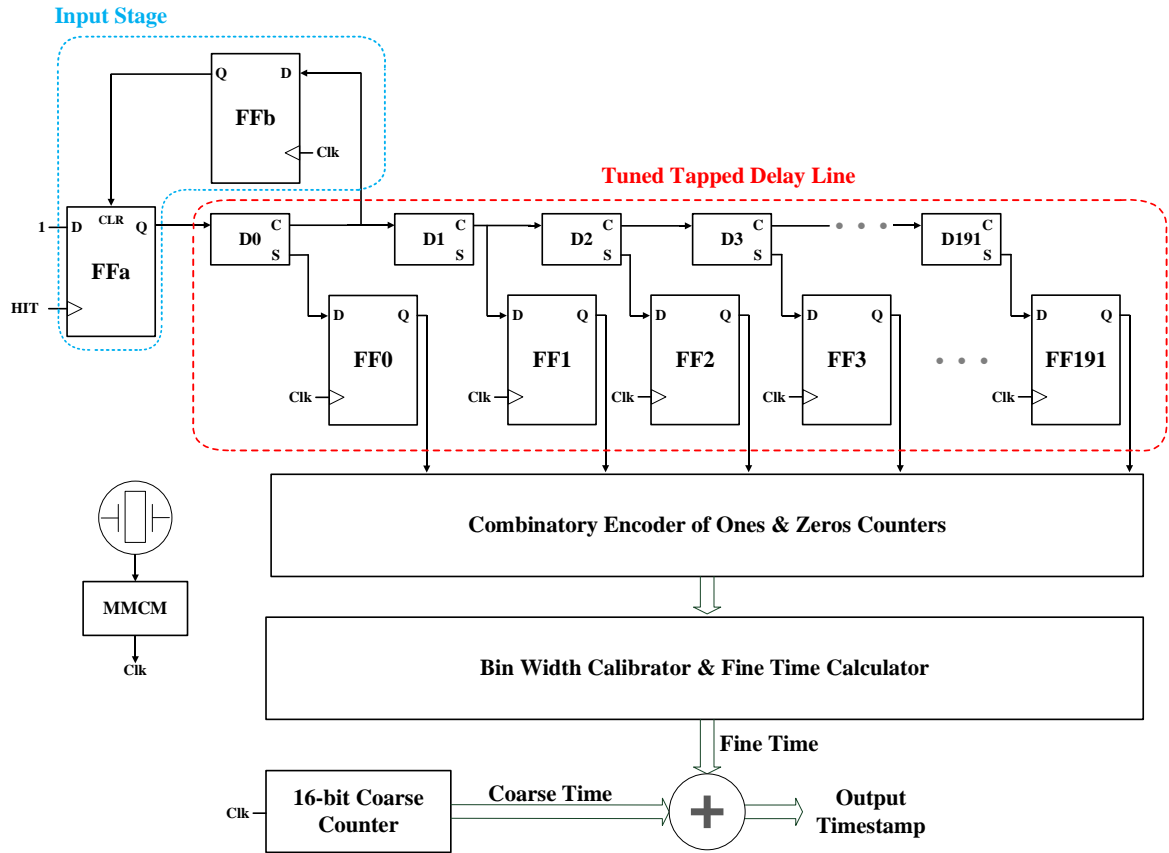


Figure 2.1. Architecture of the proposed FPGA-based TDC.

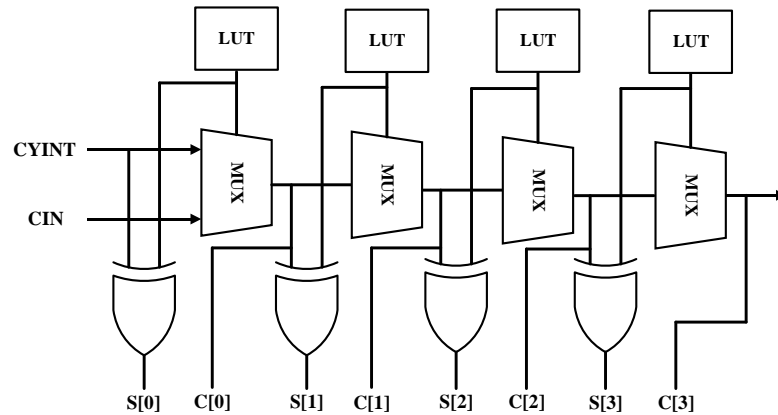


Figure 2.2. Simplified diagram of the structure of the CARRY4 block.

The input signal propagates through the multiplexers (MUX) and can be sampled at the carry out (C) or sum (S) nodes. The fine time resolution of TDL is determined by the propagation time through a delay unit. It depends on the FPGA fabrication technology, family, and speed grade. The linearity of the TDL is highly

dependent on the sampling pattern, meaning the exact sequence of C's and S's is selected for sampling the output bits. Hence, to find the sampling pattern yielding a more linear TDC output, all possible sequences of C's and S's should be tested. Their nonlinearity metrics are then compared. To avoid further contributions to mismatch, all the delay elements of the TDL have to be located in the same clock region of FPGA. In this way, the clock skew is minimized. Moreover, the total delay of the line should be only slightly longer than the system clock period. In this design, a 250 MHz clock frequency and a TDL with 192 delay cells have been considered. According to the clock frequency and the number of coarse counter bits, the longest time interval covered is equal to 262.14 μ s.

The input stage, shown in details in Figure 2.1, is used to properly shape the input signal. The first C of the delay line is employed to generate a clear signal of 'FFa.' When there is an input signal propagated through the delay line, the input stage resets the delay line in the next rising edge of the reference clock. Therefore, the width of the input signal equals the time interval between the input signal edge and the next rising edge of the clock. Furthermore, the input stage signals the next blocks about the incoming time sample injected into the TDL. Since the logic states of C and S are opposite, we cannot use the S instead of C in the input of the input stage. If the first element of the selected sampling pattern comes from C, the output of 'FF0' in Figure 2.1 can be used as the clear input of 'FFa' and 'FFb' can be removed from the circuit. However, as we will see later in section 2.4, the first bit of the selected sampling pattern comes from an S-type output.

The output of the TDL is a thermometer code that needs to be converted to a binary number. For that, we employ a thermometer-to-binary encoder (T2B) that needs to also take into account the TDL bubble errors. Ideally, the output of the TDL should be a clean thermometer code such as, for instance, 1111110000. However, because of uneven propagation delays within the TDL and the system

clock skew, in practice, bubbles appear distorting the thermometer code. For instance, instead of 1111110000, the sampled state of the TDL could be 1101010000. This can lead to serious errors in the output binary code. Therefore, it is essential to design an encoder that avoids these kinds of errors. Moreover, since the TDL is tuned to a particular sampling pattern, and the states of each C and S pairs are opposite, two separate encoders would be required to find the transitions in both of them. Consequently, more resources are needed. To minimize the resource usage while suppressing bubbles in the thermometer code, we have introduced a combinatory ones and zeros counters encoder. In this case, we are counting the ones for C codes and zeros for S codes. This does not depend on the transition stage in the TDL, and therefore, the output is not severely affected by bubbles.

The use of resources in this T2B encoder is equal to the case in which the same type of output is sampled for all of the delay elements. The architecture of the encoder is shown in Figure 2.3. In the first stage, the lookup tables (LUTs) connected to S nodes are configured to count the number of zeros, and the LUTs connected to C nodes are assigned to count the number of ones. In the next stages, the partial results have been combined to calculate the final binary number. Each set of LUTs consists of three 6-input LUTs and it converts 6-thermometer bins to a 3-bit binary code. A summary of the characteristics of the encoder is displayed in Table 2.1.

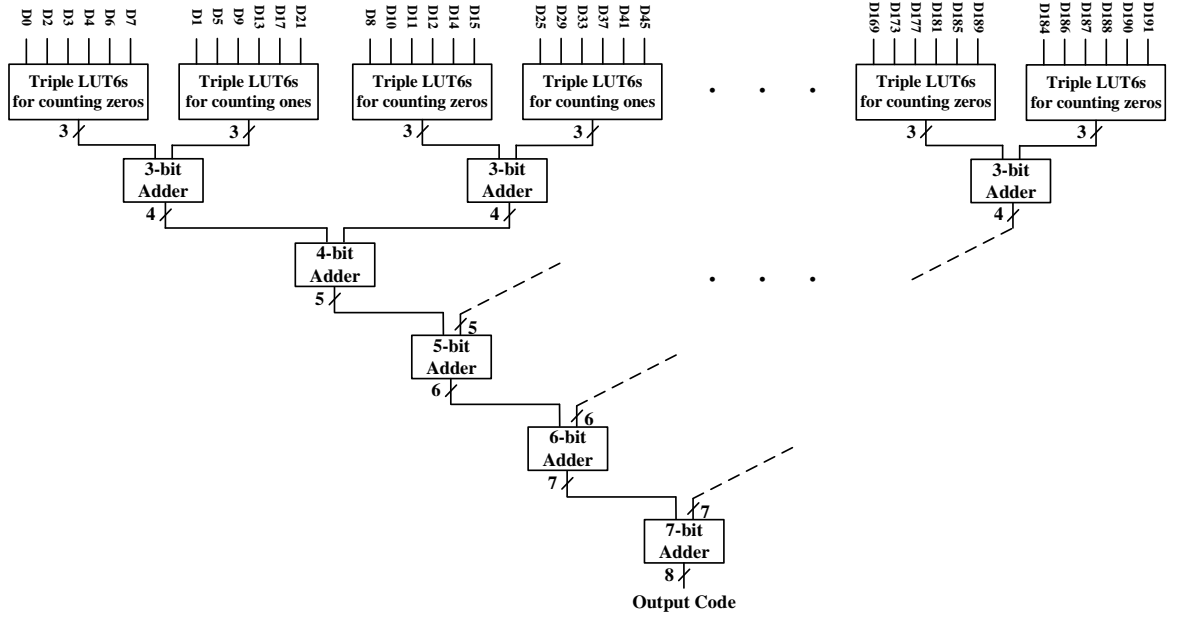


Figure 2.3. Combinatory encoder of ones and zeros counters.

Table 2.1. Summary of the ones and zeros combinatory counters encoder.

I/O		Resources		Processing Time
Input	Output	LUTs	FFs	
192 codes	8b	215	246	6 clocks

To implement real-time calibration, a table that maps each output binary code to a code representing the exact delay time has been built. The time related to each number has been obtained by a code density test which estimates the width of each bin. The time assignment procedure is as follows. Half of the first bin width corresponds to the delay mapped to the number '1.' Half of the second bin width and the entire width of the first bin are added to reach the delay (from the origin) equivalent to the number '2.' The procedure is the same for the other numbers and is summarized as follows:

$$t_k = \frac{w_k}{2} + \sum_{i=0}^{k-1} w_i \quad (2.1)$$

where t_k is the total delay (from the origin) related to number 'k' and w_k is the measured bin width of the k -th bin.

Each binary number obtained from T2B is mapped into a new time interval by using the above equation. The resulted calibration table extracted from this equation and experimental measurements is shown and compared with the ideal transfer function in section 2.4.

2.4 Experimental Results

2.4.1 Measurements

The proposed TDC has been implemented on the Artix-7 FPGA (XC7A200T-1FBG484) of an Opal Kelly XEM7310 board [43], as Figure 2.4 shows. To perform the code density test and send the results through the USB link, API components such as WireIn and PipeOut have been used [44]. The CFGMCLK output of the STARTUPE2 primitive has been used as the input source for the code density test [45]. Since this signal is generated by the internal ring oscillator of the ARTIX-7, there is no correlation between it and the system clock. We have collected 114,688 samples for the code density test. Then, we have measured the TDC bin widths with the following procedure. First, we have extracted the number of counts for each distinct sample. Then, this number is divided by the total number of samples. Finally, to estimate the bin width, the result is multiplied by the clock period of the TDC, i.e., 4000 ps in the proposed design. The bin widths of one of the TDCs are shown in Figure 2.5. This TDC employs a sampling pattern denoted by “SCSS,” where letters indicate the selected outputs of the CARRY4 delay cells, respectively. The reason for this choice will be explained later in this section.

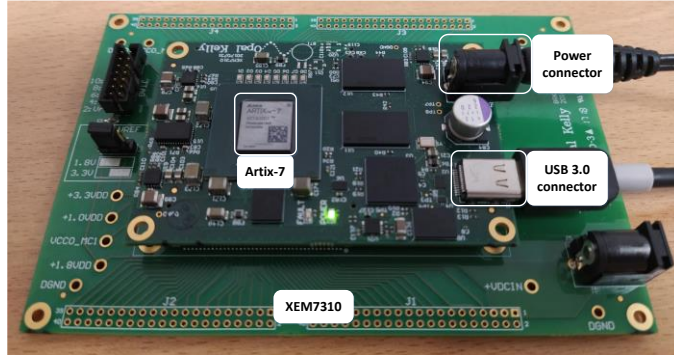


Figure 2.4. Evaluation board of the proposed TDC.

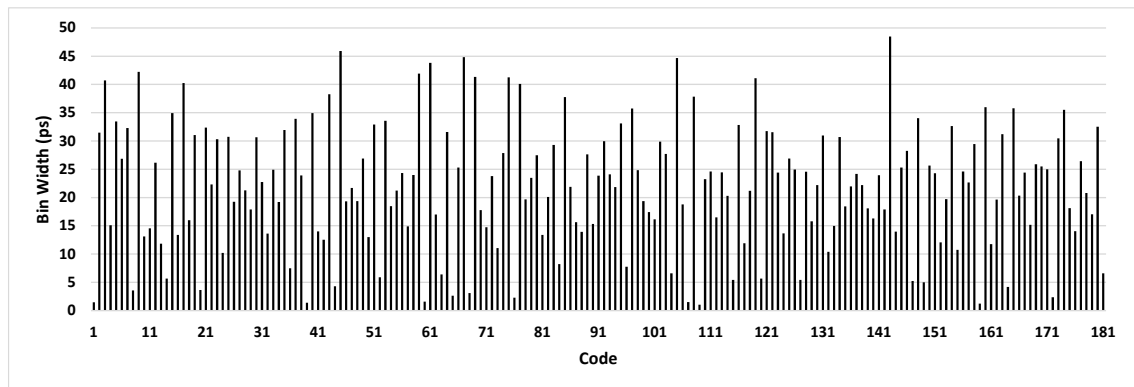


Figure 2.5. Measured bin widths of a TDC with “SCSS” sampling pattern resulted from code density test.

FPGA-based TDCs have different sources of nonlinearity, such as clock skew, target device structure, local deviations of transistor characteristics, and ambient conditions. The effects of most of these sources can be minimized without using any additional resources by accounting for them during design and implementation.

First of all, all the delay elements of TDL have been placed in the same region to avoid clock region crossings. The clock skew between the regions can be a few hundreds of picoseconds, which can deteriorate the linearity of the TDC. Moreover, because of process variations, changing the position of the TDL in the same region also has an effect on the linearity. Therefore, the TDL has been placed in different columns of the same clock region and the results have been compared to find the best place for the TDL. Additionally, the position of the input stage has

a direct relation with the linearity of the TDC. Thus, we have considered various positions for the input stage and compared their results.

Modern FPGAs, like Xilinx 7 series, contain a set of clocking resources, such as Mixed-Mode Clock Manager (MMCM), phase-locked loop, and different types of buffers [46]. To minimize the jitter of the system clock, an MMCM module is used. The TDC system is placed within a single clock region. We can use dedicated buffers, specifically designed for this kind of system. These buffers have access to high-speed, low skew local routing resources and can be driven by the MMCM.

To find out the most linear sampling pattern, the code density test has to be executed for all possible sampling patterns. To do this, we have designed and implemented different combinatory ones and zeros counters encoder for each of the sampling patterns. Then, we have tested all the patterns on the target device. Then, their DNL and INL values have been calculated as follows:

$$DNL_k = \frac{w_k - w_{LSB}}{w_{LSB}} \quad (2.2)$$

$$INL_k = \sum_{i=0}^{k-1} DNL_i \quad (2.3)$$

where w_{LSB} is the LSB size and according to the measurements, is equal to 22.2 ps. Additionally, the number of active bins is 181.

To illustrate the effect of the sampling pattern on the TDC performance, the DNLs and INLs of some combinations are shown in Table 2.2. Among all the sampling patterns, “SCSS” has reached the most linear results and has been selected to be used in the final TDC system. The DNL and INL values of the selected sampling pattern (“SCSS”) have been compared with the ordinary sampling pattern (“CCCC”) in Figure 2.6 and Figure 2.7, respectively. Figure 2.8

shows the bin width distributions of “SCSS” and “CCCC” sampling patterns. These plots demonstrate the notable linearity improvement of the selected sampling pattern accompanied by the proposed encoder in the presented architecture.

Table 2.2. Sampling patterns comparison.

Sampling Pattern	DNL (LSB)	INL (LSB)
CCCC	[-0.976, 1.779]	[-0.733, 6.660]
CSCS	[-0.987, 3.721]	[-0.733, 6.567]
SCSC	[-0.954, 1.425]	[-2.921, 1.274]
CCSC	[-0.978, 2.727]	[-0.119, 6.456]
CSCC	[-0.981, 3.698]	[-0.700, 6.700]
SCSS	[-0.953, 1.185]	[-2.750, 1.238]

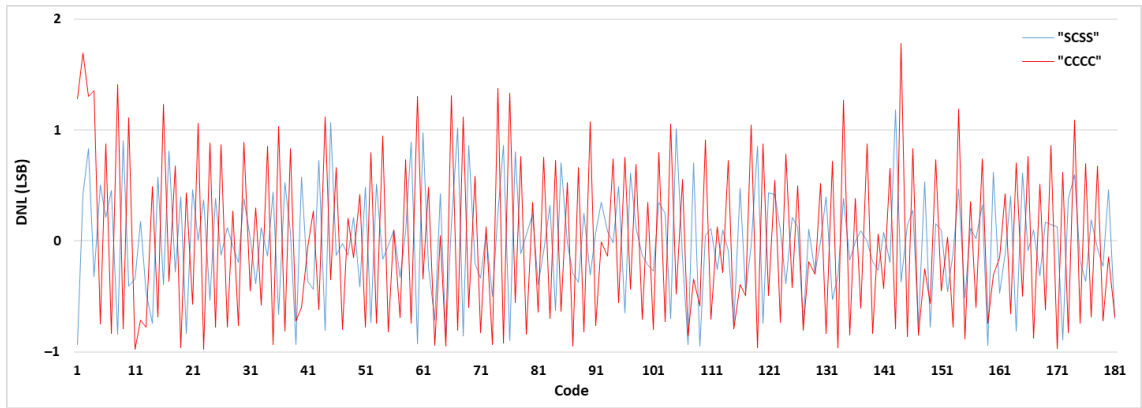


Figure 2.6. DNL results of “SCSS” and “CCCC” sampling patterns.

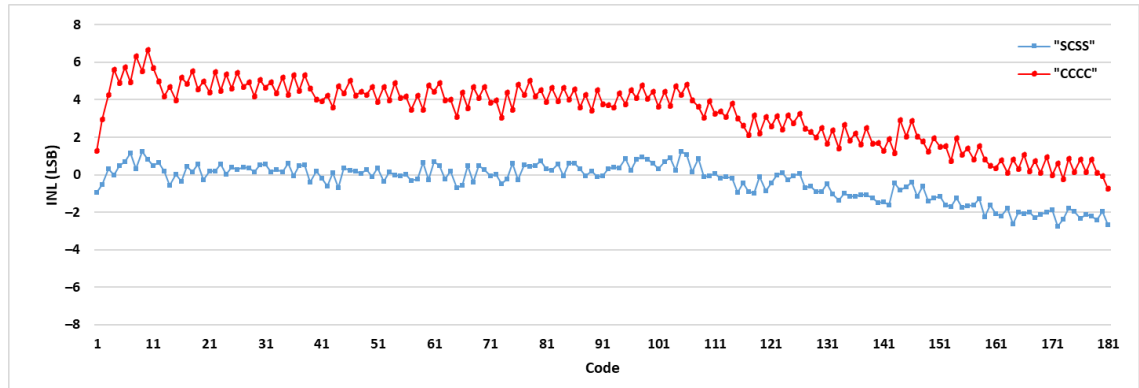


Figure 2.7. INL results of “SCSS” and “CCCC” sampling patterns.

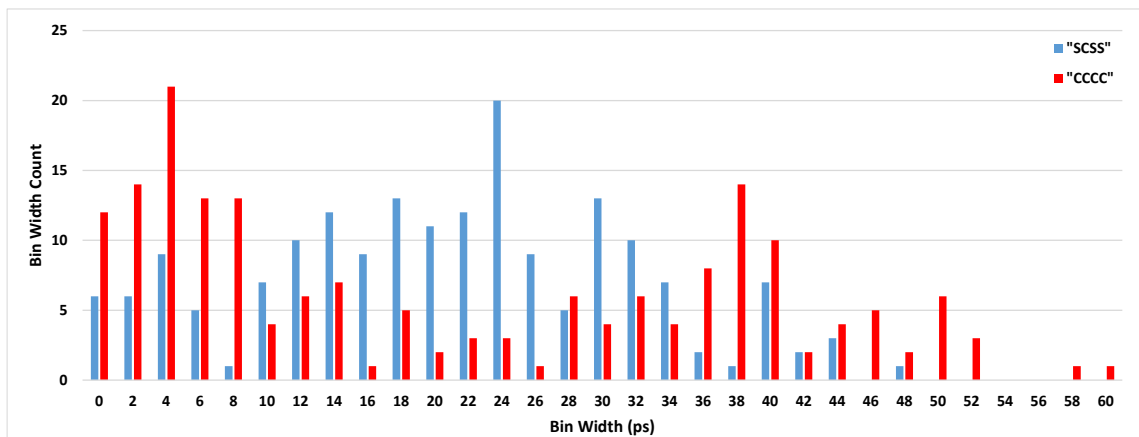


Figure 2.8. Bin width distributions of “SCSS” and “CCCC” sampling patterns.

Using the calibration method explained in Section 2.3, the calibration table of the proposed TDC has been calculated. In Figure 2.9, the content of the calibration table is compared with the ideal transfer function. We can compute the accuracy based on the measured and the ideal static characteristic. The absolute accuracy is the maximum deviation of the TDC calibration table from the ideal transfer function. The absolute accuracy of the proposed TDC is equal to 27.04 ps.

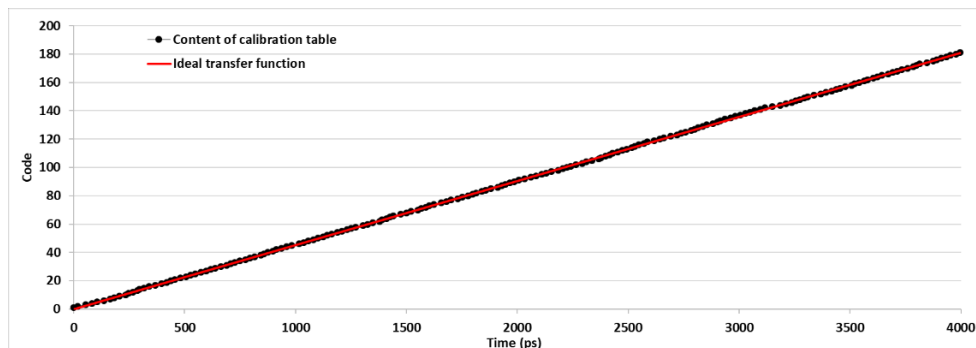
To evaluate the TDC measurement precision, a constant time interval has been measured by two TDC channels. One channel measures the start time of the interval and the other channel estimates its stop point. The time interval is calculated by subtracting the stop timestamp from the start one. Figure 2.10 shows the histogram of 114,688 samples measured by two TDC channels. The mean

value and the standard deviation (STD DEV) of the time interval are 127.81 ps and 26.04 ps, respectively. We generated the different time intervals by hiring the IDELAY2 primitive of the FPGA. Since the time intervals are generated in the FPGA, they have less jitter than the intervals generated outside the FPGA. Moreover, an IDELAYCTRL calibrates the IDELAY2 to realize an accurate time interval. The RMS precision of the different time intervals are shown in Figure 2.11. Since the time interval has been fixed during the tests, the achieved standard deviation is the single-shot precision of the TDC. The standard deviation is calculated as follows:

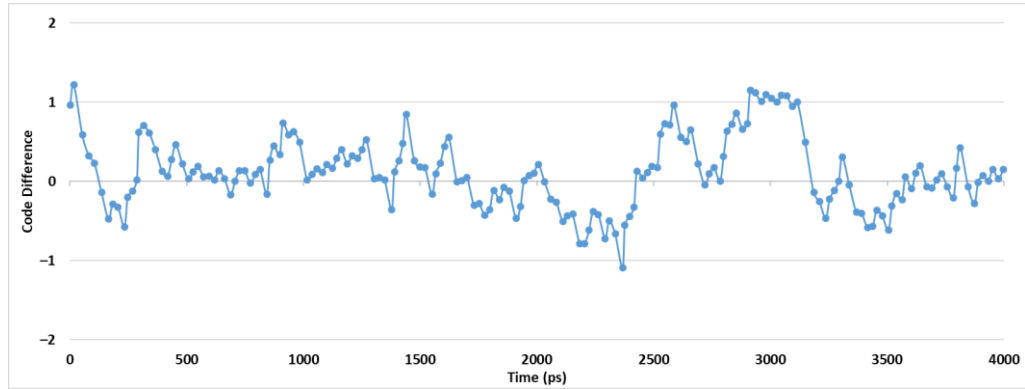
$$\sigma = \frac{1}{\sqrt{N-1}} \sqrt{\sum_{i=1}^N \left(t_i - \frac{\sum_{j=1}^N t_j}{N} \right)^2} \quad (2.4)$$

where σ is the standard deviation, t_i is the result of i -th measurement, and N is the number of measurements.

To estimate the RMS precision variations over temperature, the code density test is carried out in different temperatures from 30 °C to 75 °C and the corresponding RMS precision variations are shown in Figure 2.12. Figure 2.12 indicates that the RMS precision degraded slightly with increasing temperature.



(a)



(b)

Figure 2.9. (a) Comparison of the TDC calibration table content and the ideal transfer function. (b) The differences between the codes.

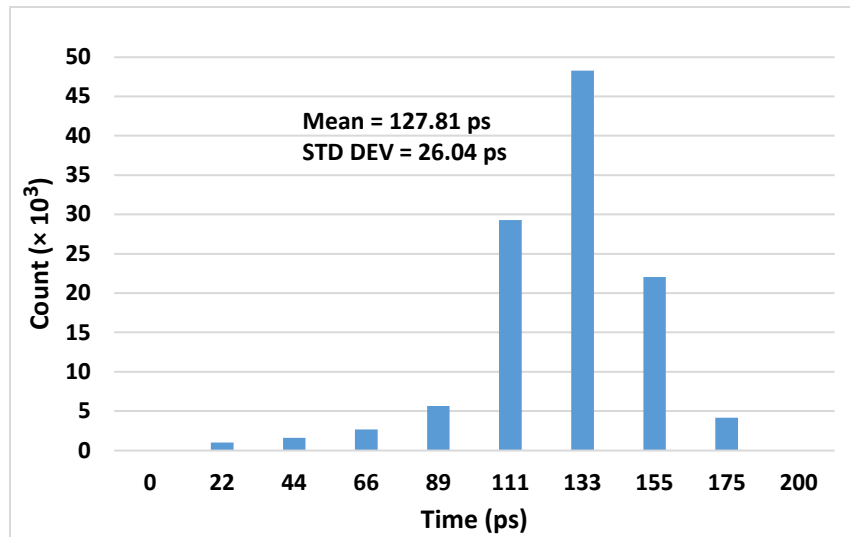


Figure 2.10. Measurement histogram of a constant time interval.

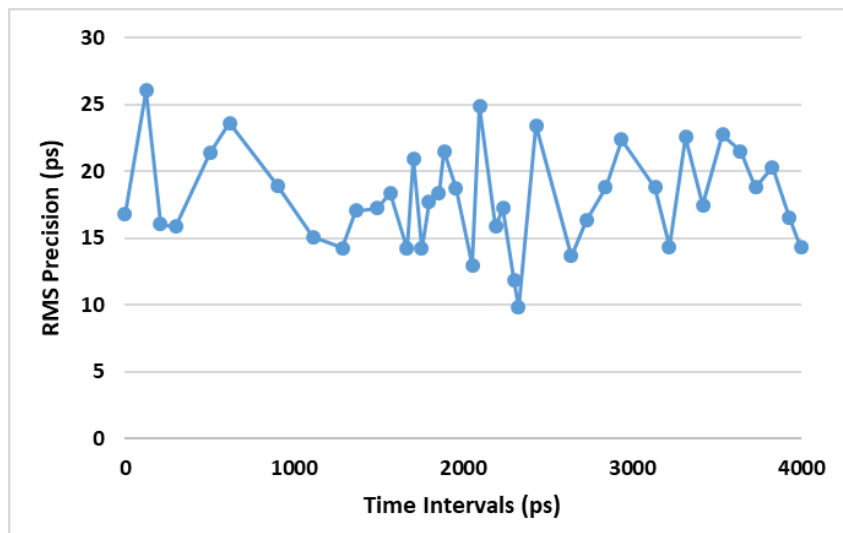


Figure 2.11. RMS precision of the different time intervals.

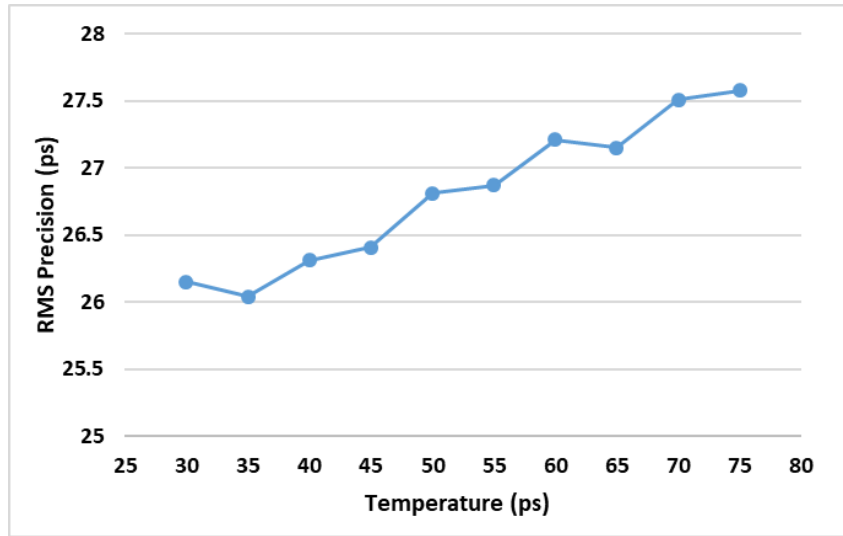


Figure 2.12. RMS precision variations with temperature.

Table 2.3 shows data regarding the usage of logic resources and the power consumption of one TDC channel. These data are extracted for the implementation report and demonstrates the low resource utilization and the low power consumption of the proposed TDC. The characteristics of the proposed TDC are summarized in Table 2.4.

Table 2.3. Resources usage and power consumption of one TDC channel.

Resource	Available	Utilization	Utilization (%)
LUT	133,800	216	0.16
FF	267,600	638	0.24
BRAM	365	2.50	0.68
Total Power Consumption		164 mW	
Dynamic Power		33 mW	

Table 2.4. Characteristics of the proposed TDC.

Parameter	Value/Range	Unit
Clock Frequency	250	MHz
Resolution	22.2	ps
Measurement Range	262.14	μ s
Dead-Time	8	ns
Readout Speed	125	MSample/s
INL	[-0.953, 1.185]	LSB
DNL	[-2.750, 1.238]	LSB
Single-Shot Precision	26.04	ps

2.4.2 Comparison

Table 2.5 provides a comparison with state-of-the-art FPGA-based TDCs. Note from the table that only some works report power consumption data, a few of them in detail and the others without indicating whether the reported amount is the total on-chip power consumption or not. Although the lack of data hinders comparison regarding power, the columns of the table highlight that the proposed TDC features low non-linearity and dead time while having a resource usage considerably lower than other works. Obviously, FPGA-based TDC performance depends on the FPGA fabrication technology. For example, the LSB width of TDL-based TDCs depends on the family, generation, and speed grade of the target device and newer technologies potentially lead to better performance. To elucidate the comparison with other works, we have used a simplified version of the Figure of Merit (FoM_TDC) presented in [47], which excludes power consumption because data regarding power are not reported in most of the FPGA-based TDCs:

$$\text{FoM_TDC} = 10 \times \log_{10}\left(\frac{1}{2^{\text{ENoB}} \times F_S}\right) \quad (2.5)$$

$$\text{ENoB} = N_{\text{bits}} - \log_2(\text{INL} + 1) \quad (2.6)$$

where F_S is the conversion rate of the TDC and ENoB is the effective number of bits. The FoM_TDC is plotted in Figure 2.13 for those references that report all the data required to calculate it.

Table 2.5. Comparison with the state-of-the-art FPGA-based TDCs.

Ref.	Used Method	FPGA	LSB [ps]	Precision [ps]	DNL [LSB]	INL [LSB]	Dead-Time [ns]	Resources Usage	Power [mW]	ENoB	FOM_TDC (dB)
Song [15]	TDL	Virtex-2	46.2	65.8	1.10	2	10	NS	NS	5.42	26.71
Wu [24]	Wave Union	Cyclone II	30	25	NS	NS	5	NS	NS	NA	NA
Amiri [19]	Matrix of Vernier Delays	Spartan-3	75	300	2.5	3	4.17	NS	92	5	24.95
Favi [16]	TDL	Virtex-5	17	24.2	3.55	3	50	1208 Slices	NS	5	31.94
Buchele [21]	Multi-phase Clock	Virtex-5	160	68	0.8	NS	NS	NS	NS	NA	NA
Fishburn [17]	TDL	Virtex-6	10	19.6	1.5	2.25	3.3	NS	NS	5.30	22.29
Zhang [22]	Delay Line Loops Shrinking	SmartFusion	63.3	61.7	0.55	0.72	1410	NS	NS	6.22	42.77
Liu [25]	Multi-Meas. TDL	Kintex-7	9.4	9.5	4.6	NS	1.47	400 Slices	NS	NA	NA
Wang [32]	TDL + Bin Realignment & Decimation	Kintex-7	17.6	15	1	0.8	NS	NS	NS	7.15	NA
Won [26]	Dual-phase TDL + Online Cal.	Virtex-6	10	12.83	1.91	3.93	NS	NS	NS	5.70	NA
Cao [39]	TDL + Bin Realignment	Cyclone-IV	45	18	0.5	0.48	13.3	NS	NS	6.43	21.88
Wang [34]	Mul-Ch. TDL + ones Counter Encoder	Kintex-7	2.45	3.9	NS	NS	3.61	6258 FFs + 2433 LUTs	821	NA	NA
Zhang [23]	Matrix of Counters	Virtex-5	7.4	6.8	0.74	1.57	80	1265 Slices	1113	8.64	23.03
Kuang [27]	Multi-Meas. RO-based TDL	Kintex-7	3	5.76	NS	9	22	NS	NS	6.68	23.27
Chen [33]	sub-TDL + tap timing + histogram + mixed cal.	Virtex-7	10.54	14.59	0.08	0.11	NS	1916 FFs + 1145 LUTs	NS	7.85	NA
Tontini [42]	Input Stage + Tuned TDL	Spartan-6	25.6	37	1.23	2.96	8.69	415 Slices	131	6.01	21.29
This work	Input Stage+ Tuned TDL + Combinatory Encoder	Artix-7	22.2	26.04	1.18	2.75	8	638 FFs + 216 LUTs	164 (Total)	6.10	20.68

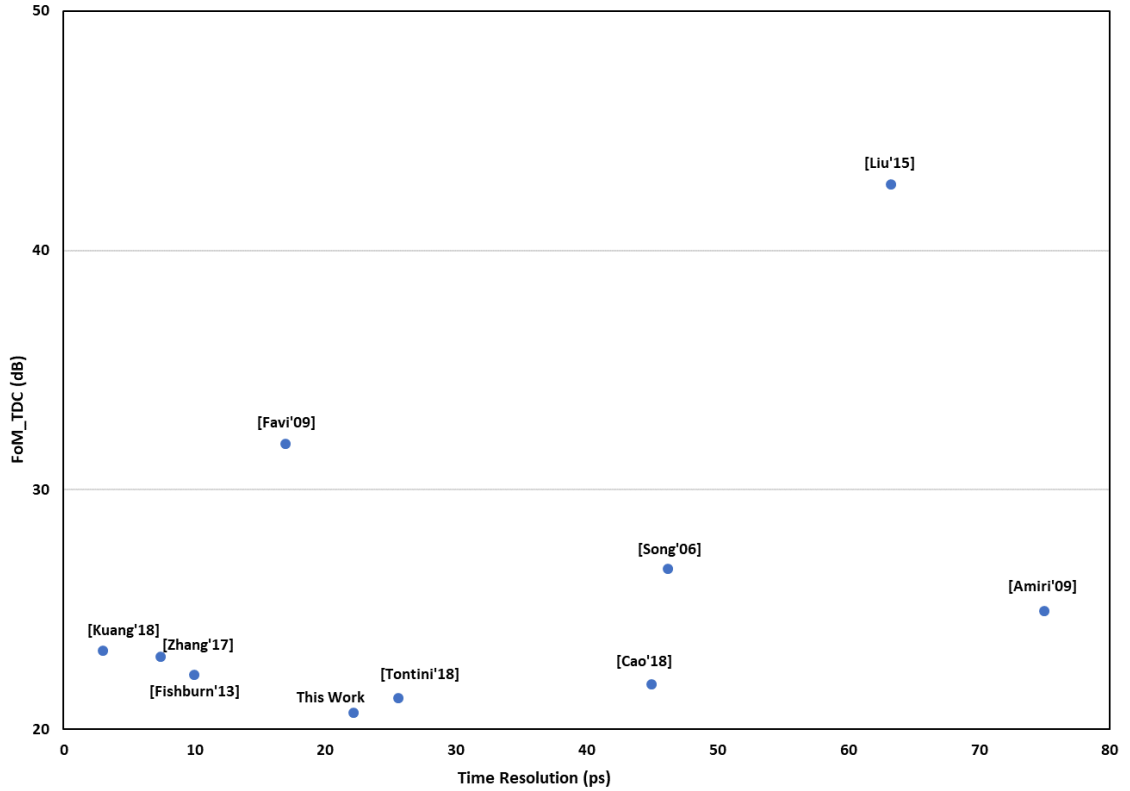


Figure 2.13. Comparison of FoM_TDC.

2.5 Conclusions

We have designed and tested a novel FPGA-based TDC architecture delivering high performance with low resource usage. It has been implemented in an Artix-7 with a 250 MHz clock frequency. We have tested all the sampling patterns to find out the one rendering the highest linearity. We have employed a combinatory ones and zeros counters encoder to achieve high immunity to bubbles in the TDL, composed of different sampling elements with opposite logic states. The obtained resolution and single-shot precision are 22.2 ps and 26.04 ps, respectively. The measurement throughput is 125 MSa/s. The presented architecture can measure input intervals beyond 260 μ s with 125 MSa/s conversion rate. The code density test results show $[-0.953, 1.185]$ LSB DNL and $[-2.750, 1.238]$ LSB INL. These characteristics make the proposed design suitable for a multi-channel direct ToF readout.

3. A Novel Approach for Measurement Throughput Maximization in FPGA-based TDCs

This work has been published in:

M. Parsakordasiabi, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “A Novel Approach for Measurement Throughput Maximization in FPGA-based TDCs,” 7th Int. Conf. on Event-based Control, Communication, and Signal Processing, Krakow, Poland, Jun. 2021.

Number of Citations (Google Scholar): 2 (2022, December 23)

Article Full-text Views: 127 (2022, December 23)

3.1 Abstract

This paper presents a new approach for dead-time minimization while preserving low resource usage and high resolution in FPGA-based time-to-digital (TDC) converters. The proposed TDC architecture can be employed in applications in which many events need to be detected in a short time, such as time-of-flight positron emission tomography (ToF-PET) applications. The presented architecture consists of a toggling input stage, a tapped delay line (TDL), a dual-mode counter-based encoder, a coarse counter, and a bin width calibration stage. The minimum dead-time of TDL TDCs is two clock cycles. The proposed architecture reduced

dead-time to one clock cycle. The measurement results of the proposed low-resources TDC in an Artix-7 FPGA show $[-0.80, 1.34]$ LSB differential non-linearity (DNL) and $[-0.73, 1.97]$ LSB integral non-linearity (INL). The measured LSB size and single-shot precision (SSP) are 22.1 ps and 28.43 ps, respectively.

3.2 Introduction

Time-to-digital converters (TDCs) play a key role in a wide range of applications requiring accurate estimation of time intervals, like time-of-flight (ToF) detection [7]. In positron emission tomography (PET), for instance, ToF measurement helps improving image resolution by establishing the most probable origin of the high-energy photons in the line of response determined by two coincident events [11]. This contributes to a higher signal-to-noise ratio (SNR). To this end, high-resolution TDCs are on demand. Furthermore, conversion rate and the number of channels are highly important to realize a count loss-free system. High measurement accuracy and high throughput—or, conversely, low dead-time—are often incompatible specifications in a TDC. It is even more challenging to achieve high accuracy and high throughput with limited hardware resources. However, meeting all of these requirements is necessary for systems such as ToF-PET scanners [48].

Field-Programmable Gate Arrays (FPGAs) represent an attractive implementation platform for fully-digital TDCs. They offer more flexibility, shorter development time, and lower implementation cost than Application-Specific Integrated Circuits (ASICs). Additionally, the FPGAs fabricated in the finest silicon technologies have the intrinsic delay of the carry elements of few tens of picoseconds. This makes a good fine-time interpolator, rendering FPGAs an interesting option for high-resolution TDCs [14]. Furthermore, considering the available resources on each FPGA, they are a suitable option also to implement multi-channel TDCs [1].

Several FPGA-based TDCs have been introduced in recent years to meet different application requirements [14]. The most common approach is the Nutt method, which employs a coarse counter and a fine time interpolator. This allows covering long time intervals while achieving high resolution. To implement the time interpolator, different techniques are reported [4], namely Tapped Delay Lines (TDLs), Vernier Delay Lines (VDLs), wave union TDCs, and stochastic TDCs. TDLs are built from several FPGA's carry elements as delay cells. The fine time measurement is obtained by sampling the state of the delay line at the next rising edge of the system clock after the input signal. The intrinsic propagation delay of the elementary delay cells determines the measurement resolution. TDLs are the best choice for applications requiring high-resolution, high throughput, and multichannel TDCs.

In TDCs based on the Nutt method, the throughput is improved by operating at the maximum allowable frequency and by minimizing dead time. The maximum reachable frequency is a limited value that depends on the integrating technology. On the other hand, the dead time is defined as the time required for the TDC to perform a measurement and to be ready to start a new measurement. Minimizing the number of logic functions to be realized surely contributes to reducing the dead time. Two-clock-cycle dead-time is usually reported in TDL-based TDCs [42],[28]: one cycle to reset the taps and another one to sample them. For example, the maximum frequency of the targeted Artix-7 whose speed grade is "-1", equals 464 MHz, and therefore, theoretically, its dead-time is 4.31 ns for TDL-based TDCs with dead-time of two clock cycles. Favi and Charbon [16] introduced a method to reduce the dead-time. Although, it could reach the dead-time of one clock cycle in Turbo mode architecture, their proposed TDC has some limitations such as asymmetry in propagation delay and metastability. Furthermore, since more than one hit signal can be injected into the delay line in the same clock cycle,

it is not cleared that how the encoder can measure multiple propagated signals in the next rising edge of the system clock. Some research works reached the dead-time of one clock cycle using wave union method [25],[49][49]. Wave union TDCs measure multiple transitions per hit signal generated by launchers. To decrease the dead-time, Liu and Wang [25] and Wang et. al [49][49] employed two-transition wave union TDC and multiple-TDL wave union TDC, respectively.

There is a trade-off between dead time reduction and accuracy. While the minimization of the dead time involves low resource usage, a higher accuracy requires the use of additional resources, especially for accurate thermometer-to-binary (T2B) encoding and online calibration. In this paper, we introduce a new architecture for an FPGA-based TDC, with a dead time equal to one system clock cycle, while preserving high resolution and low resource usage.

The paper is organized as follows. Section 3.3 describes the proposed TDC architecture. The elements introducing measurement throughput optimization are explained in Sections 3.4 and 3.5. Section 3.6 displays experimental results and measurements, and Section 3.7 summarizes the system features and provides a comparison with state-of-the-art FPGA-based TDCs.

3.3 TDC Architecture

FPGAs consist of an array of configurable logic blocks (CLB), which are the fundamental blocks to implement sequential and combinatorial circuits. CLBs contain lookup tables (LUTs), flip-flops (FFs), and carry elements. The tiny propagation delay of a carry element (CARRY4 in Xilinx terminology) can be employed to implement a time interpolator for a high-resolution TDC. The simplified schematic of a Xilinx CARRY4 structure is shown in Figure 3.1.

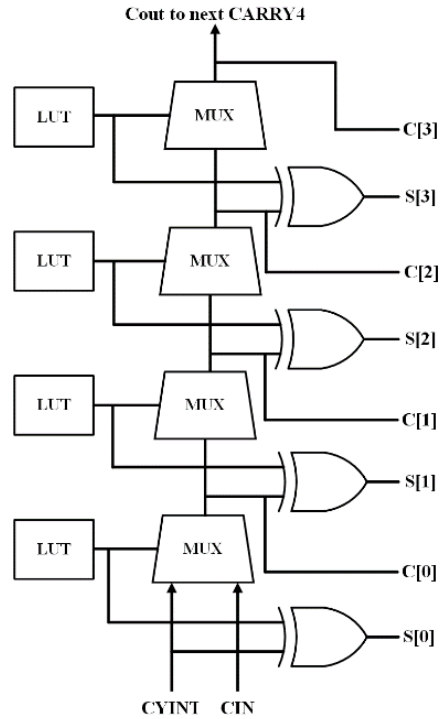


Figure 3.1. Simplified schematic of CARRY4 structure

The TDC we are presenting relies on a TDL, built by cascading multiple carry elements to construct a delay line, whose status is sampled by FFs (Figure 3.2). The target device is a Xilinx Artix-7 XC7A200T-1FBG484, embedded in an Opal Kelly XEM7310 board. We have employed 192 delay elements to build a TDL whose total propagation delay is slightly longer than the system clock period, i. e. 4ns in the proposed design.

FPGA-based TDL TDCs need one clock cycle to propagate the reset signal across the whole delay line and make it ready for a new measurement. Then, another clock cycle is necessary to sample the delay line. This makes two clock cycles the minimum achievable dead time. To reduce the minimum dead time to one single clock cycle, two different elements have been incorporated into the TDC architecture:

- a toggling mechanism at the input that eliminates the need of resetting the delay line, and

- a counter-based encoder, which is robust against bubble errors without requiring additional processing.

The other elements of the architecture of the proposed TDC are:

- a tapped delay line,
- a controller for the dual-mode counter-based encoder,
- a bin-width calibration block, and
- a coarse counter.

The simplified architecture of the proposed TDC is displayed in Figure 3.2.

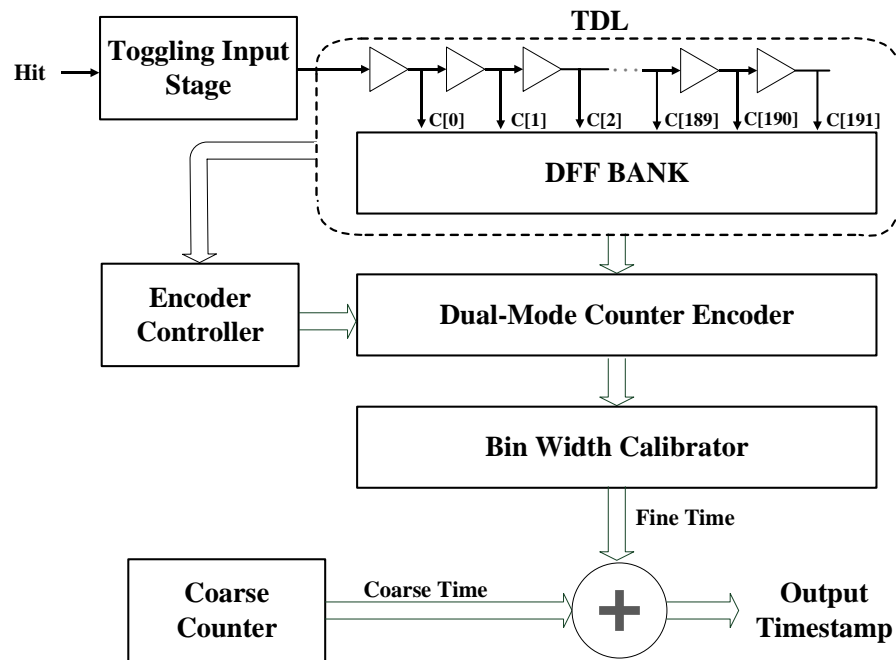


Figure 3.2. The simplified architecture of the proposed TDC

3.4 Toggling Input stage

The internal structure of the toggling input stage and its timing diagram are depicted in Figure 3.3. It provides alternative logic “1” and “0” for being injected into the delay line whenever a hit signal is detected. For example, as shown in Figure 3.3(b), if the logic state of ‘Trigger’ is “1” (‘Trigger’ is what is injected to the delay line), the input stage prepares logic “0” for the input of ‘FFa’ to be

propagated through the delay line at the arrival of the next hit signal. In this way, the TDC can perform one measurement in each clock cycle and regardless of whether the hit signals 'P1', 'P2', and 'P3' are injected into the input stage in consecutive or non-consecutive clock cycles, time intervals 'T1', 'T2', and 'T3' can be measured, and therefore the dead time is reduced to one single clock period. This input stage only toggles once per clock cycle. To avoid data corruption, the input stage is designed to inject at most one input signal into the delay line in each clock cycle. It also prevents the injection of spurious pulses due to metastability. Furthermore, only the rising edge of the hit signal is important for the input stage and it does not depend on its width.

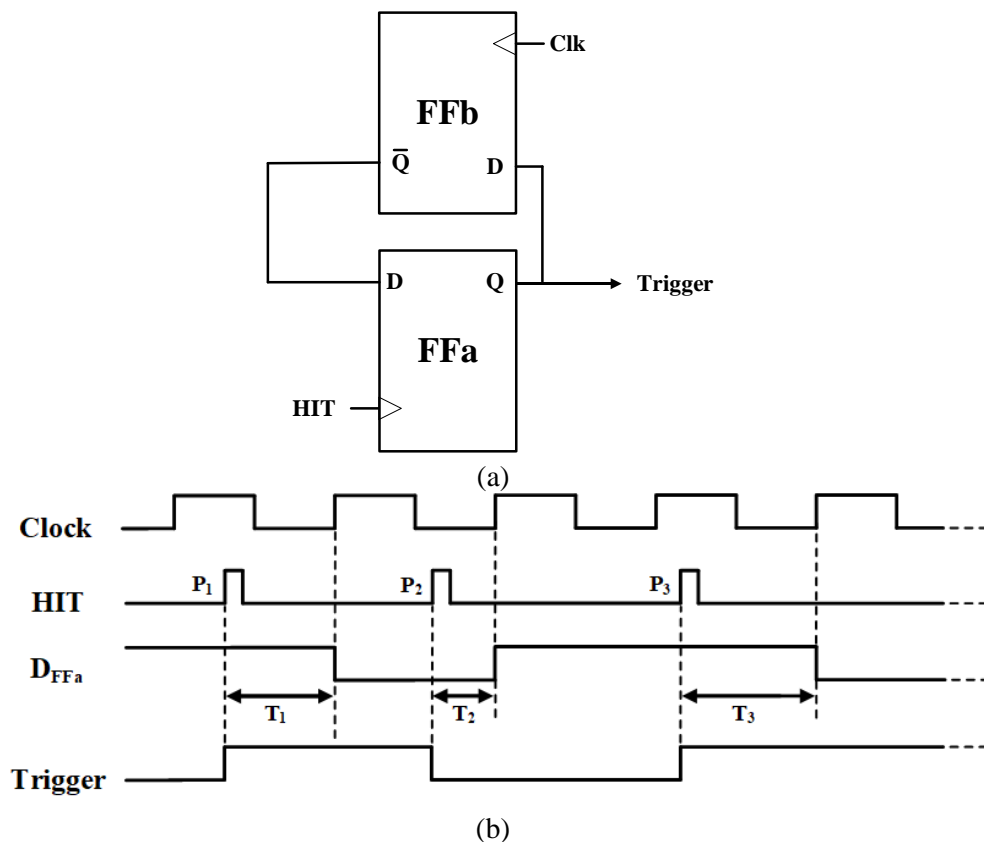


Figure 3.3. (a) Toggling input stage (b) Timing diagram

3.5 Dual-Mode Counter-Based Encoder

The output of the TDL is a thermometer code that needs to be converted to a binary number with the help of a T2B encoder. “One-hot” binary encoders, that find the transition stage from the TDL output, are not robust against bubble errors. Additional logic functions are required to eliminate the bubbles, which leads to utilizing more resources and/or degrading the LSB size and/or increasing the dead time. One possible solution is employing an ones-counter encoder [34], which counts the number of ones in the delay line output. Since the TDL transition point is not an issue for ones-counter encoders, bubbles do not have drastic effects on outputs.

Besides, when the input stage toggles, a chain of zeros is propagated through the delay line. To use the ones-counter encoder also in this case, we need to subtract the number of ones from the actual number of delay cells to obtain the actual number of zeros already in the TDL. In this way, we do not have to implement two counters. The ones-counter and a subtractor circuit will do the work, using fewer resources but still maintain robustness against bubble errors in both counting modes. This dual-mode counter encoder is implemented by using the basic functionalities on the FPGA. Its internal architecture is depicted in Figure 3.4. The encoder divides the TDL output into bundles of 6 signals. Next, each 6-signal bundle is converted to a 3-bit binary number that encodes the number of ones in the 6-signal bundle. For that, 6-input LUTs are employed. Then, the full adders in the next stages calculate the total sum of ones in the TDL output. If the encoder is in the zero-counting mode, this number is subtracted from the total number of delay cells to obtain the total sum of zeros.

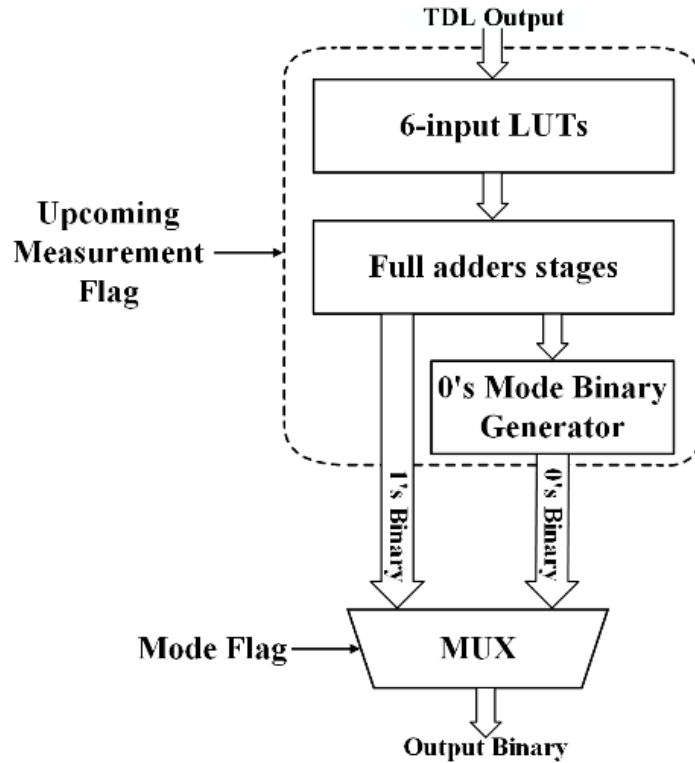


Figure 3.4. Dual-mode counter-based encoder

Usually, the T2B encoder can be simply informed about the arrival of a new measurement by the output of the first flip-flop of the DFF bank. After reset, its logic state changes to “1” whenever the next rising edge of the clock arrives after a hit signal is injected into the delay line. In this case, both logic states of the mentioned flip-flop, either “0” or “1”, can indicate the arrival of a new measurement. This needs to be taken into account.

Moreover, the encoder has to be aware of the counting mode, either if it will be counting ones or zeros in the current round. The controller shown in Figure 3.5, is designed to inform the encoder about these two aspects: the current counting mode and the availability of a new measurement. As shown in Figure 3.1, each delay element has two output nodes: carry out (C) and sum (S), whose logic states are complementary. To implement the controller, we use both the C and S nodes of the first delay cell of the delay line. When a logic “1” is propagated through the delay

line, the logic state of the mentioned C node certainly transits from “0” to “1”. Inversely, the logic state of the referred S node changes to “0” from “1” when a logic “0” is injected into the delay line. Consequently, there is a new measurement available whenever one of the mentioned transitions is detected. We employ these transitions to generate the upcoming measurement flag as shown in Figure 3.5. Besides, the encoder counting mode can be specified by checking the “C” or “S” nodes logic state. For example, if the upcoming measurement flag and C logic state are “1”, there is a new measurement available to be handled in ones-counting mode. If the upcoming measurement flag and S logic state are “1”, there is a new measurement available to be handled in zeros-counting mode.

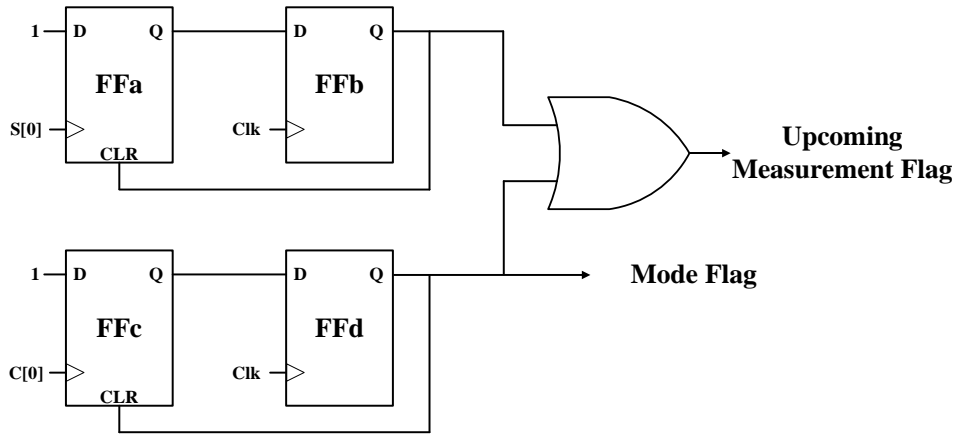


Figure 3.5. Encoder controller

3.6 Experimental Results

The presented TDC has been implemented on the Artix-7 FPGA (XC7A200T-1FBG484) of an Opal Kelly XEM7310 board. The performance of the TDC is characterized using a code-density test. API components such as WireIn and PipeOut are employed to transmit the results to an external computer through a USB link. The encoded TDC values are analyzed offline to estimate the average bin width, differential non-linearity (DNL), and integral non-linearity (INL).

To execute the code-density test, 114,688 randomly generated samples are collected. The input signal is generated by the internal ring oscillator of the

ARTIX-7 using the CFGMCLK output of the STARTUPE2 primitive, which is uncorrelated with the system clock. To estimate the bin widths, the counted occurrences at each different bin are divided by the total number of samples and then multiplied by the clock period. As all the possible values of the interval are equiprobable, the abovementioned operation results in an estimation of the width of the bins of the TDC. As all the possible values of the interval are equiprobable, the abovementioned operation results in an estimation of the width of the bins of the TDC, which are plotted in Figure 3.6. The widths of the bins follow a bell-shaped distribution, depicted in Figure 3.7. The average bin width, which corresponds to the LSB size, is equal to 22.1 ps.

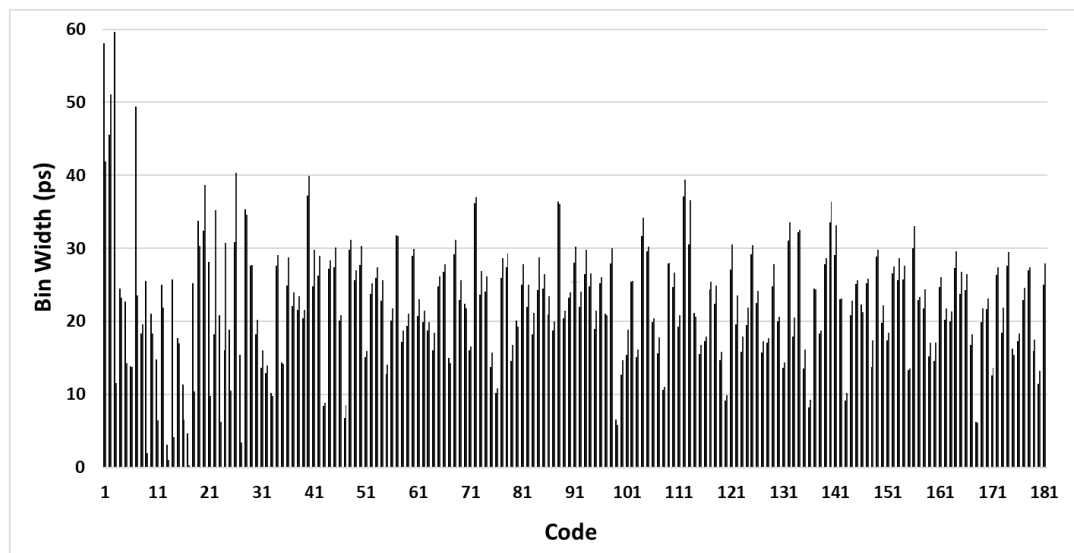


Figure 3.6. Bin widths of the proposed TDC

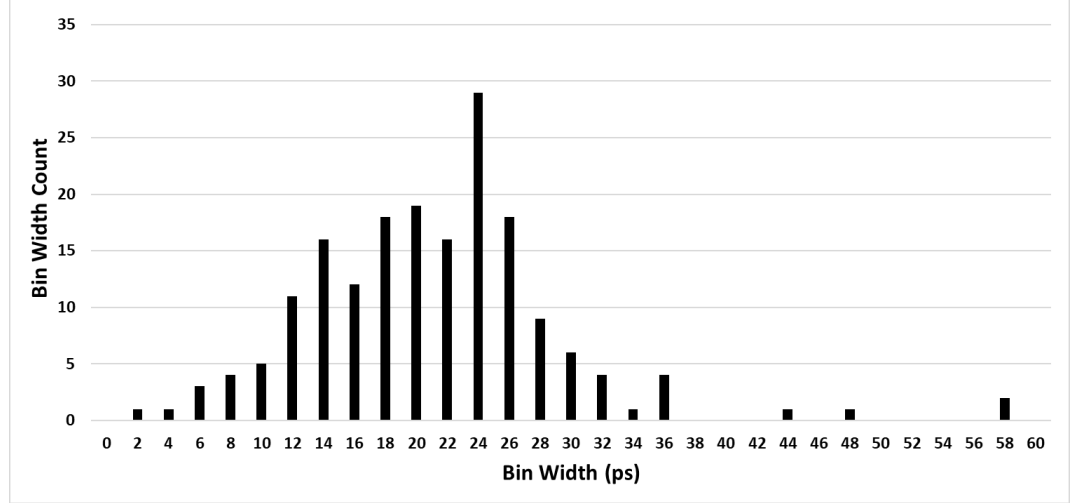


Figure 3.7. Bin widths distribution of the proposed TDC

The delay elements that compose the TDL are not perfectly identical, due to process variations and mismatch. This non-uniformity is reflected in a nonlinear characteristic that is evaluated in terms of its differential (DNL) and integral (INL) nonlinearities. Their values are computed as follows:

$$DNL_k = \frac{w_k - w_{LSB}}{w_{LSB}} \quad (3.1)$$

$$INL_k = \sum_{i=0}^{k-1} DNL_i \quad (3.2)$$

where w_k is the measured bin width of the k -th bin and w_{LSB} is the LSB size, which in this TDC equals 22.1 ps. The number of active bins is 181. Figure 3.8 displays the measured DNL that is within [-0.86 1.70] LSB. Figure 3.9 shows the INL that is in the range of [-0.13 5.38] LSB. These results prompt further linearity correction.

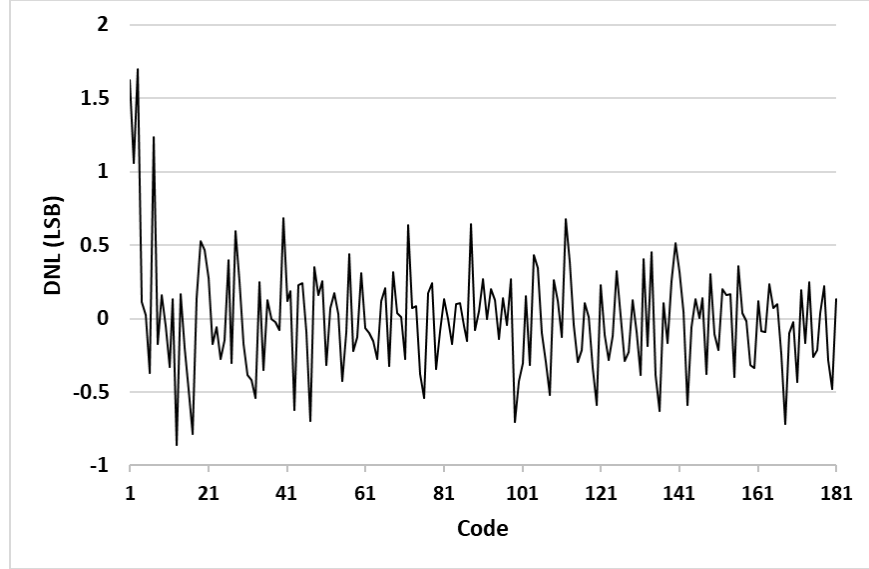


Figure 3.8. DNL of the uncalibrated TDC

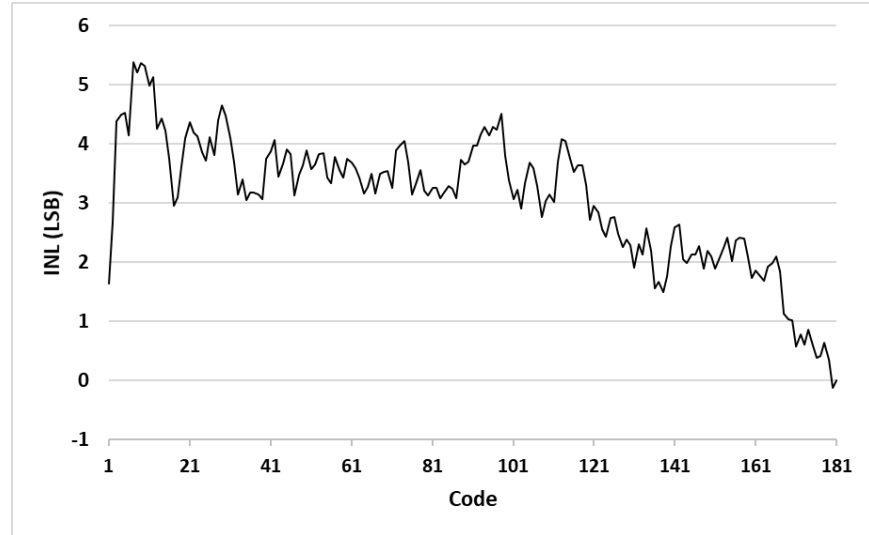


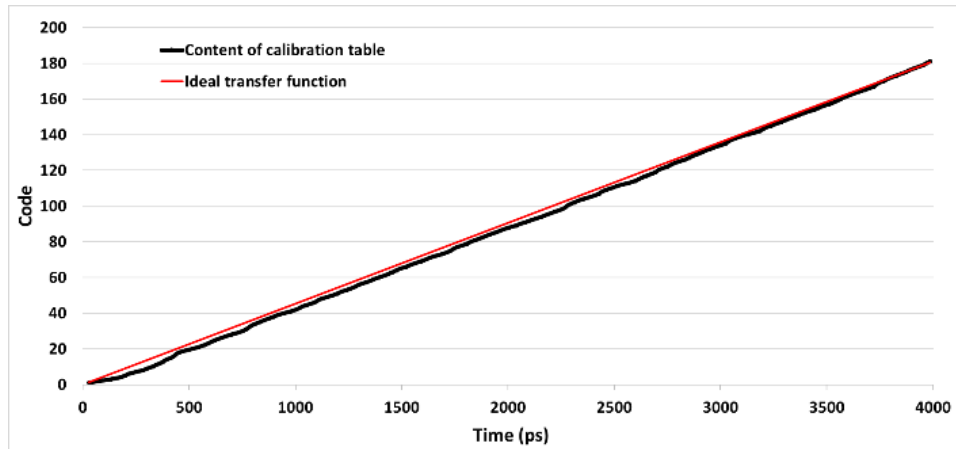
Figure 3.9. INL of the uncalibrated TDC

To overcome large nonlinearities, real-time bin-by-bin calibration is essential. This calibration is based on the calculated bin widths resulted from the code-density test. These values are employed to build a table. This table assigns a fine timestamp (t_k) to each binary number obtained from T2B encoder, that corresponds to the accumulated propagation delay from the time origin:

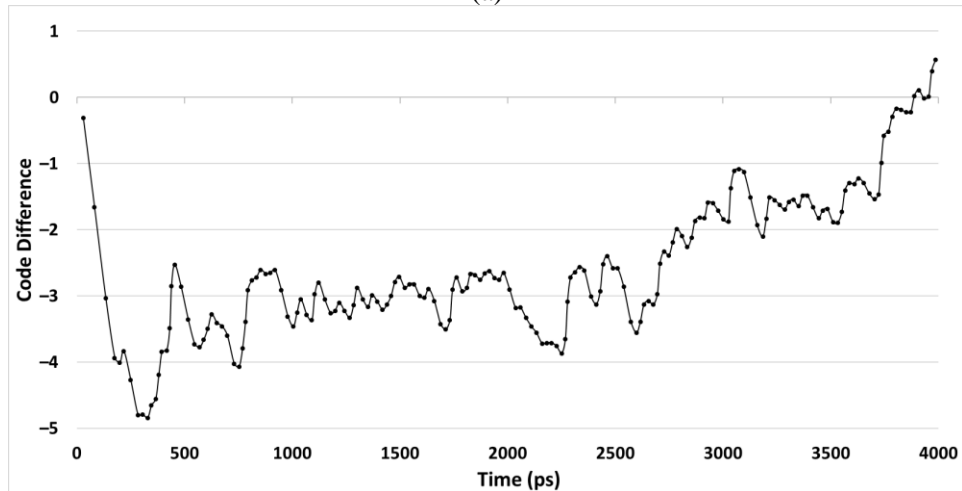
$$t_k = \frac{w_k}{2} + \sum_{i=0}^{k-1} w_i t_k = \frac{w_k}{2} + \sum_{i=0}^{k-1} w_i \quad (3.3)$$

where w_i is the estimated bin width of the i -th bin. The transfer function of the TDC built using the one-by-one calibration table, helps to reduce the RMS

measurement error [29]. Figure 3.10 shows the TDC transfer function and its difference with the ideal transfer function.



(a)



(b)

Figure 3.10. (a) TDC calibrated and ideal transfer functions (b) Their differences

The DNL and INL of the calibrated outputs are indicated in Figure 3.11 and Figure 3.12, respectively. The results demonstrate significant linearity improvement compared with uncalibrated TDC. The DNL is in the range of $[-0.80, 1.34]$ LSB and the INL is within $[-0.73, 1.97]$ LSB.

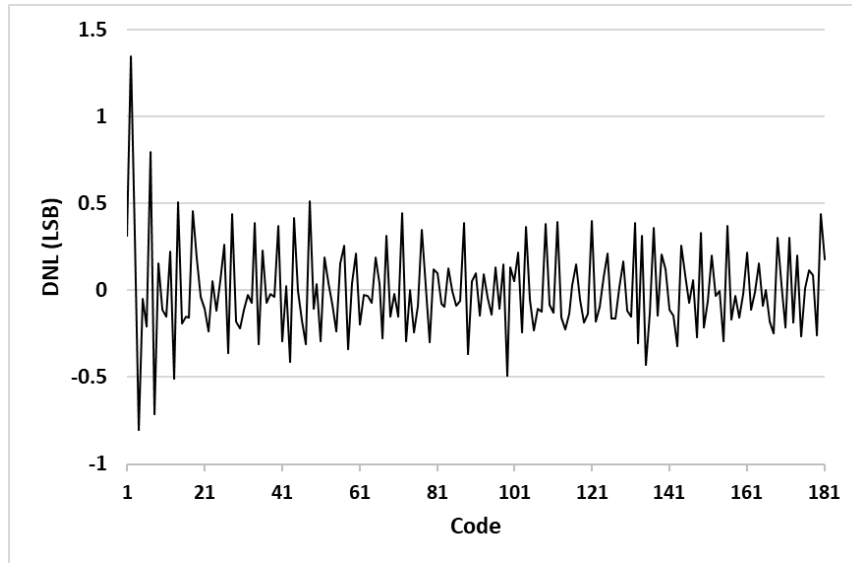


Figure 3.11. DNL of the calibrated TDC

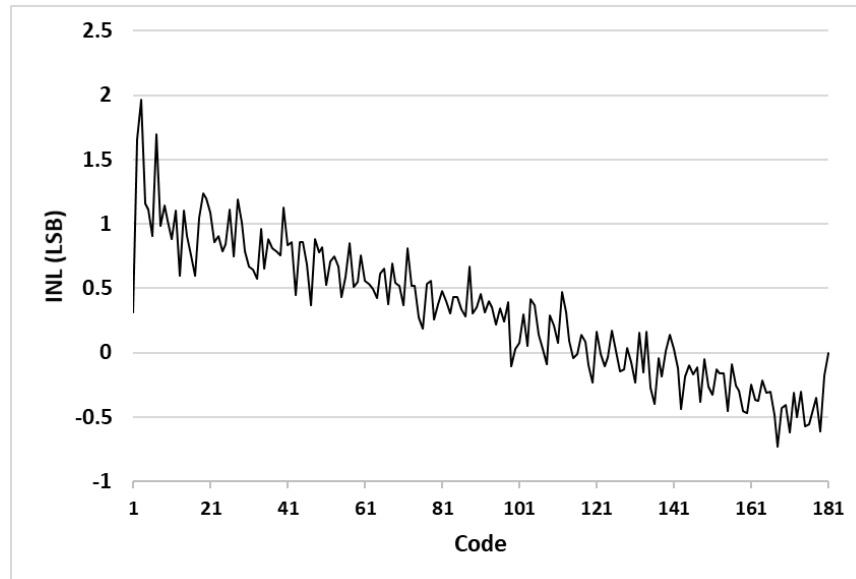


Figure 3.12. INL of the calibrated TDC

Since the propagation delay of ones and zeros could be different for the same delay cell, building a distinct bin-width calibration table for each counting mode could result in more accurate outputs.

Moreover, the single-shot precision (SSP) has been measured as well. Two TDC channels have been employed for this: one measures the start signal and the other measures the stop signal. Time intervals are generated by employing the IDELAY2 primitive of Artix-7. In this way, the time intervals have less jitter than the

intervals generated outside the device. The time interval is calculated by subtracting them. Each time interval is measured tens of thousands of times. The RMS precision for each interval is computed from the standard deviation:

$$\sigma = \frac{1}{\sqrt{N-1}} \sqrt{\sum_{i=1}^N \left(t_i - \frac{\sum_{j=1}^N t_j}{N} \right)^2} \quad (3.4)$$

where t_i is the i -th measurement value, and N is the total number of measurements. Figure 3.13 displays the RMS precision for 20 different intervals. The worst RMS precision of the TDC equals 28.43ps.

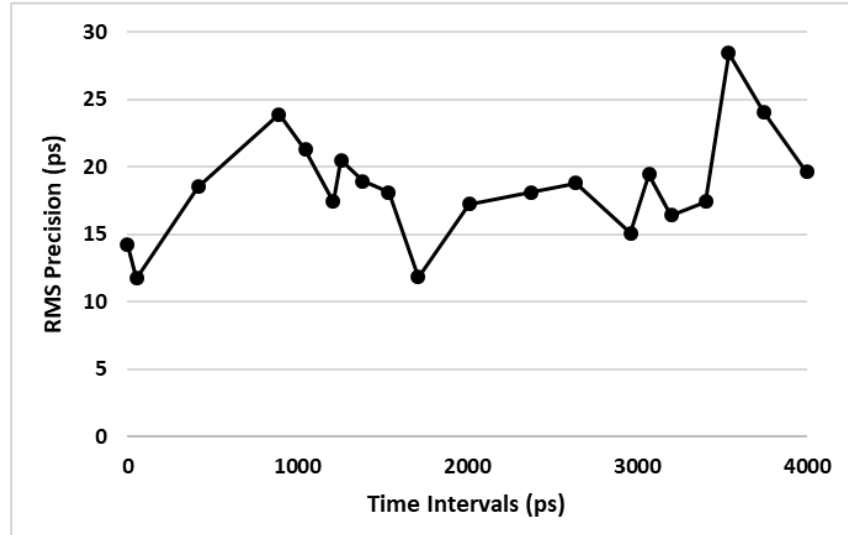


Figure 3.13. RMS precision for for different time intervals

3.7 System Features and Comparison

The proposed TDC allows for the implementation of multiple channels in a single FPGA. Table 3.1 summarizes the use of computational resources (LUTs, FFs, and BRAMs) and power consumption. Only a small fraction of the FPGA resources is employed in the implementation of one TDC channel. Besides, the characteristics of the proposed TDC are provided in Table 3.2.

Table 3.1. Resources and power consumption for one TDC channel

Resource	Available	Utilization (%)
----------	-----------	-----------------

LUTs	133800	228 (0.17)
FFs	267600	678 (0.25)
BRAMs	365	2.50 (0.68)
Total Power	171mW	
Dynamic Power	40mW	

Table 3.2. Characteristics of the proposed TDC

Parameter	Value/Range	Unit
Clock Frequency	250	MHz
Resolution	22.1	ps
Measurement Range	262.14	μ s
Dead-Time	4	ns
Readout Speed	250	MSample/s
DNL	[-0.80 1.34]	LSB
INL	[-0.73 1.97]	LSB
Single-Shot Precision	28.43	ps

Compared to other state-of-the-art FPGA-based TDCs (Table 3.3), the proposed TDC channel presents the lowest dead time in terms of system clock cycles, while maintaining acceptable values for resolution, precision, and linearity. The use of resources is rather small compared to contenders. This is mainly because we did not push the LSB below the gate delay.

Table 3.3. Comparison with the other FPGA-based TDCs

Parameter	[16]	[49]	[50]	[28]	This work
FPGA	Virtex 5	Zynq-7000	UltraScale	Kintex 7	Artix 7
LSB [ps]	17	NS	2.48	10.6	22.1
Precision [ps]	24.2	5.8	3.63	8.13	28.43
DNL _{pk-pk} [LSB]	4.55	NS	2.61	2.45	2.14
INL _{pk-pk} [LSB]	5.57	NS	4.45	5.53	2.70
Dead Time [Clk Cycles]	1	1	NS	2	1
LUT	1208	3310*	2460	577	228
FF	Slices	5497*	3463	1641	678
BRAM	NS	8.25*	7.5	NS	2.5
Power [mW]	NS	NS	1003	NS	171

* Values calculated using the reported resources usage of multi-channel TDC (4-block mode)

3.8 Conclusion

A novel approach has been introduced to minimize the dead-time of TDCs with low resource usage. The proposed TDC has been evaluated in an Artix-7. The dead time is one clock cycle, i.e., 4 ns for a 250-MHz system clock frequency. The LSB size and single-shot precision are 22.1 ps and 28.43 ps, respectively. These results show that the proposed architecture is suitable for applications requiring high-throughput in multiple channels, which is the case in ToF-PET scanners. The proposed TDC could be implemented in other types of Xilinx FPGAs by applying a few changes to the HDL code.

4. An Efficient TDC Using a Dual-Mode Resource-Saving Method Evaluated in a 28-nm FPGA

This work has been published in:

M. Parsakordasiabi, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “An Efficient TDC Using a Dual-Mode Resource-Saving Method Evaluated in a 28-nm FPGA,” IEEE Transactions on Instrumentation & Measurement, vol. 71, Dec. 2021.

Impact Factor: 4.016 (1st Flagship journal in the field of FPGA-based TDCs)

Quartile: Q1 (Instruments & Instrumentation)

Number of Citations (Google Scholar): 2 (2022, December 23)

Article Full-text Views: 384 (2022, December 23)

4.1 Abstract

FPGA-based time-to-digital converters (TDCs) are required to be accurate, linear and fast, while at the same time employing a reduced number of resources. Pushing these requirements to the limit is challenging, although it is constantly required by many applications. This paper presents a dual-mode tapped-delay-line (TDL) — propagating 1’s and 0’s in alternating measurement cycles— architecture for an FPGA-based TDC that complies with the mentioned specifications. The dead-time

of the proposed TDC is reduced to one system clock cycle by using a toggling input stage and a dual-mode counter-based encoder. To improve the TDC linearity, the TDL sampling sequence is tuned separately for each operating mode. The presented architecture employs a low-resources dual-mode combinatory encoder of one- and zero-counters to remove the bubbles and cover both operating modes. A dual-mode bin-width calibration has been carried out to improve the TDC performance in each mode. The proposed architecture has been implemented on a Xilinx Artix[®]-7 FPGA. Experimental results have shown a DNL within [-0.71 1.05] LSB and an INL within [-0.85 0.86] LSB for the propagation of 1's. DNL and INL are within [-0.73 1.06] LSB and [-1.17 0.04] LSB, respectively, for the propagation of 0's. The LSB size is 22.1 ps and the TDC precision is 22.35 ps. A comparison with recently published state-of-the-art FPGA-based TDCs is provided at the end of the paper.

4.2 Introduction

Time-to-digital converters (TDC) are critical components in delay-assessment systems, with applications, for instance, in light detection and ranging (LiDAR) and time-of-flight positron emission tomography (ToF-PET), and fluorescence lifetime imaging microscopy [48] [51] [52]. For example, to reach the desired signal-to-noise ratio in ToF-PET, a high-resolution measurement of the time-delays between coincident events is necessary [53]. Also, real-time operations require high throughput. Another important characteristic of these and other application scenarios is that simultaneous measurement of a huge number of parallel detector modules is required [17], which means that the required accuracy and speed should be achieved with a reduced number of resources.

Under these circumstances, timing resolution scales with the gate delay, therefore advances in CMOS fabrication technology help realizing higher-

resolution TDCs. However, compared to application-specific integrated circuits (ASICs), field-programmable gate arrays (FPGAs) offer more flexibility, shorter time-to-market, and lower development costs. For these reasons, FPGAs are suitable platforms for the implementation of fully-digital TDCs [4] [14] [54] [55].

Most FPGA-based TDCs are based on the Nutt interpolation method. They employ a coarse counter with a few nanoseconds resolution, running at the system clock frequency, and a fine time interpolator that allows reaching sub-clock-period resolution [56]-[58]. In this way, a long time interval of up to several hundreds of microseconds can be covered while achieving a fine resolution down to a few picoseconds.

Compared to a previous work [1] with the dead-time of two clock cycles, in this work, the dead-time is reduced to only one-clock cycle by employing a dual-mode configuration. A toggling input stage eliminates the need to reset the delay line by propagating 1's and 0's in alternating measurement cycles. The dual-mode tuned delay line, the dual-mode combinatory counter-based encoder, and the dual-mode bin width calibrator are designed to be adapted to the dual-mode propagation and different from the related sections in [1]. Furthermore, this work provides more linear results in terms of nonlinearity parameters in both operating modes and the measurement precision has been considerably improved. In another previous work [2], a 'CCCC' sampling sequence, a counter-based encoder, and a bin-width calibrator are used for both operating modes. In this work, the code density test is performed for all the possible sampling sequences and regarding that the propagation delays of 1's and 0's are different for the same delay element, the sampling sequences are separately evaluated for each operating mode. Consequently, the proposed work contains a toggling input stage, a tuned delay line with most linear results for both of the operating modes, a dual-mode combinatory counter-based encoder, and two distinct bin-width calibrators. The

measurement precision is enhanced 21% than [2] and moreover, the nonlinearity parameters are improved compared to [2] without any additional resources usage.

The rest of the article is organized as follows. A review of state-of-the-art techniques for the design of FPGA-based TDCs is realized in section 4.3. Then, the proposed minimal-dead-time resource-saving TDC is described in section 4.4. The tuned-TDL and the dual-mode combinatorial encoder are also explained in detail. Experimental measurements are provided in section 0, together with a comparison with state-of-the-art FPGA-based TDCs. Finally, section 4.6 summarizes the features of our TDC and concludes the paper.

4.3 Review of FPGA-based TDC Techniques

Most FPGA-based TDCs consist of an input stage, a coarse counter, a fine time interpolator, a thermometer-to-binary (T2B) encoder, and an online calibrator. All these blocks influence the system resolution, precision, nonlinearity, dead-time, and the use of the available resources. However, let us start reviewing the fine time interpolator, as its selection determines many of the characteristics of the complete TDC.

4.3.1 Fine time interpolation

Several techniques can be found in the literature for the implementation of a fine time interpolator, i. e. for the discrimination of extremely fine time intervals.

Phased clocks TDCs [59] [60] employ different phases of the same clock as the interpolation mechanism. Since the maximum number of different phases is finite, the largest achievable resolution with this technique is limited. Wang et. al. [59] reached 89.3 ps resolution using the phased clocks method. Szplet et. al. [61] achieved even higher resolution, i.e. 43 ps, at the cost of requiring calibration.

Tapped delay lines (TDLs) [33] [1] are the most widely employed technique to

realize high-resolution TDCs. TDLs make use of the available carry chains, that are predefined structures inside most of the FPGAs. The interpolator components are the FPGA's carry elements whose intrinsic propagation delays specify the size of the least significant bit (LSB), i. e. the resolution. To achieve even higher resolutions, multiple TDLs can be employed [62] [63]. In this method, the input signal is sampled by all the TDLs and the resolution is determined by averaging the results. The multi-TDL method reaches finer resolution than a single TDL at the cost of more resources and/or additional dead-time. Depending on the application scenario, they would not be the proper choice for a multi-channel high-throughput implementation. Kwiatkowski [64] used the FPGA's digital signal processing (DSP) slices to build the delay line. Qin et al. [65] employed a combination of the carry elements and DSP slices to develop the interpolator. DSP TDCs need to be improved in terms of nonlinearity.

Pulse shrinking method operation rely on the difference between the rising and falling times of delay elements [66] [67]. This difference is used to configure a delay line loop and shortens the pulse width at each loop cycle until the pulse disappears. The lowest resolution is given by the discrepancy between the rising and falling times of delay elements. The dead-time of this method depends on the length of the input time interval and can reach a few microseconds in case of long time intervals.

Vernier TDCs comprise the architectures that employ either two delay line loops with slightly discrepancy in propagation delays [22] [68] or two ring oscillators with slightly different frequencies [69] [70]. These architectures are referred as differential TDCs as well [14]. Contrary to the pulse shrinking method that uses both the rising and falling edges to shorten the pulse, the Vernier TDCs based on delay line loops only use the propagation delay of the rising edge to shrink the time interval. The architecture employing two ring oscillators is sometimes named as

ring-oscillator (RO)-based TDCs [14]. The resolution of Vernier TDCs is given by the difference in propagation delays and frequencies for delay line loops and RO-based TDCs, respectively. Although Vernier TDCs can achieve high resolution, they suffer from long dead-time.

In Wave Union (WU) methods [24] [25] [50] [71], a launcher generates multiple transitions per hit signal in the same TDL. Measuring these transitions improves the TDC resolution and reduces the nonlinearity. Lusardi et al. [72] achieved 0.3 ps resolution using a super WU method. The main disadvantage of WU method is its increased resource usage.

Stochastic techniques [23] [73], like the matrix of counters or large-scale parallel multi-phase matrix (LSPM)-TDC, use the routing resources for time interpolation. This type of TDC can reach resolutions below ten picoseconds. The routing resources of FPGAs are made up of metal tracks that are resistant to temperature variations. Zhang et al. [73] improved the resolution of an LSPM-TDC to 1.29 ps. The main drawbacks of this method are its long dead-time and huge resource utilization.

To provide a straightforward evaluation of these methods, they are compared in Table 4.1 in terms of time resolution, dead-time, the applicability of calibration techniques, and a qualitative indication of the use of resources. These parameters are extracted from the most recent publications. Stochastic methods relies on the Gaussian distribution of thresholds on a set of sampling elements. Nonlinearity of the bin widths is small. Besides, the referred distribution does not change with temperature and voltage. Because of this, calibration in this cases is restricted to the generation of a linear coarse estimation, like in [74] and [75]. Although TDCs consist of other challenging blocks that affect their performance, this table clarifies which of the fine interpolation techniques can better meet the requirements for a specific application. According to this table, TDL is the best choice for high-

resolution, high-throughput, low-resources applications.

Table 4.1. Fine Interpolators Comparison

Method	Resolution (ps)	Dead-time	Calibration	Resources Usage
Phased Clocks	89.3	Low	No	Low
TDL	5.02	Low	Yes	Medium
Pulse Shrinking	42	Very high	Yes	High
Vernier	8.5	Very high	Yes	High
WU	0.3	Medium	Yes	High
Stochastic	1.29	High	No	High

Won and Lee [28] changed the sampling pattern of the TDL for better linearity. The basic logic resources of most of the FPGAs and configurable logic blocks (CLBs) generally include flip-flops (FFs), lookup tables (LUTs), and carry elements (known as CARRY4 in Xilinx devices). A CARRY4 is a fast logic block employed to realize high-speed arithmetic functions. To build a TDL, CARRY4s are cascaded. Four FFs samples the outputs of the TDL. Each carry element provides two complementary outputs, i. e. one corresponding to the sum bit (S) and the other to the carry-out (C). Most published TDCs select directly ‘CCCC’ as the sampling pattern of the CARRY4, which causes large non-uniformity in the outputs. Won and Lee [28] tested all the possible sampling patterns to find the one providing the most linear results and showed the overall linearity of the TDC was considerably improved by employing the tuned TDL.

4.3.2 Input stage

Different circuits have been employed to implement an input stage for noise suppression and pulse length equalization [1],[40]-[42]. Furthermore, input stages generate flag signals to prepare the system for the upcoming measurement. Efficient use of resources is also important here, for instance, the input stages

presented in Tontini et al. [42] and in our previous work [1], only require one and two additional FFs, respectively. Since the delay line needs one clock cycle for reset, the input stage cannot accept another hit without waiting for at least one clock cycle after the last hit signal. Therefore, without considering any additional dead-time introduced by the T2B encoder and/or online calibration, the minimum achievable dead-time of a TDL-based TDC would be, in principle, equal to two clock cycles.

4.3.3 Thermometer-to-binary encoder

Ideally, the states of the TDL after the stop signal will follow a thermometric code. A T2B encoder is required to convert it into a binary number. This encoder finds the transition point in the TDL and calculates the binary number. However, in this method, the output binary number can be seriously corrupted by bubble errors. Hence, different techniques have been introduced to overcome the bubble problem [29]-[32]. The most used method is bin realignment, in which zero-width bins are detected and recovered [30][30]. In addition to bin realignment, to reach more precise results, Wang and Liu [32] employed bin decimation. Chen and Li [33] used sub-TDL averaging, tap timing tests, a direct compensation histogram, and a mixed calibration approach to improve the precision. All of these methods render more precise results at the expense of additional dead-time and/or extra resources and degradation of the resolution.

Before introducing the TDC based on ones-counter encoder [34], most of the published works use thermometric-to-one-hot encoder [28]. Wang et al. [34] employed a counter-based encoder that calculates the output binary by counting the number of ones in the thermometer code. In this case, this count determines the output binary code, not the transition point of the TDL, which results in a higher immunity to bubbles. Instead of a single-transition signal, Kong et al. [76] injected

a dual-transition signal into the delay line to shorten the delay line length and reduce the resource usage of both TDL and ones-counter encoder. One way of realizing a bubble-proof encoder for tuned TDLs without consuming additional logic resources is to combine the 1's and 0's counts into the encoder, as we did in a previous work [1].

4.3.4 Online calibration

Process variations and mismatch preclude propagation delays of the carry elements from being uniform, which is reflected in a degradation of the linearity of the TDC response. This is an important issue as it compromises precision. The parameters employed to characterize the TDC operation in terms of linearity are DNL (differential nonlinearity) and INL (integral nonlinearity). They are calculated as follows:

$$DNL_k = \frac{w_k - w_{LSB}}{w_{LSB}} DNL_k = \frac{w_k - w_{LSB}}{w_{LSB}} \quad (4.1)$$

$$INL_k = \sum_{i=0}^{k-1} DNL_i \quad INL_k = \sum_{i=0}^{k-1} DNL_i \quad (4.2)$$

where w_k is the bin width of the k -th bin, which can be estimated by performing a code density test (CDT) [77]. Besides, w_{LSB} is the width of the less significant bit.

The scattering of propagation delays together with clock skews make ultra-short or ultra-long bin widths to show up in the TDC characteristic. Therefore, calibration is necessary. Won et al. [26] and Chen et. al [78] employ a multiphase TDL to reduce the effect of the clock skew. Bin-by-bin calibration employs the estimations of the bin widths obtained from a CDT to compose a table that can be stored in a memory for real-time calibration purposes [29] [79]. Since there are large variations in the delay line elements, this method can effectively improve linearity by resizing and re-centering the bins. Furthermore, it requires fewer resources than calibration techniques in which the calibration can be updated while

being resilient to ambient changes.

4.3.5 Dead-time

The throughput of a TDC is the measurement rate which is inversely proportional to the dead-time. For FPGA-based TDCs, using the maximum allowable clock frequency—that depends on the target device technology node—maximizes readout speed. Using this maximum clock frequency, the dead-time is determined by the necessary cycles to complete the measurements and reset the TDC. Most TDL-based TDCs report two-clock-cycle dead-times [28] [42]. They require one clock cycle to sample the delay line and another clock cycle to reset it. Favi and Charbon [16] reduced the dead-time to one clock cycle in their proposed Turbo mode architecture. However, this approach suffers from metastability and asymmetry in the propagations of rising and falling edges, as mentioned in their article. In addition, if more than one hit signal is propagated through the delay line during one clock period, the encoder will provide wrong outputs. Wang et. al [80] and Liu and Wang [25] reported one-clock-cycle dead-time TDC based on the WU method. To reduce the dead-time, they used multiple-TDL WU TDC and two-transition WU method, respectively. This requires many additional resources.

4.4 Proposed TDC Architecture

Our proposed TDC architecture is going to be tested on a 28-nm Artix[®]-7 FPGA. It is intended for portable and multichannel applications, so it is important to concentrate on minimizing the use of resources and power consumption. The proposed TDC relies on the following design options:

- (i) The implementation of dual-mode—propagating 1's and 0's in alternating measurement cycles—tuned TDL with improved the linearity in both operating modes. This improvement is achieved by testing all the possible

sampling patterns and finding the configuration that provides the most linear output.

- (ii) Linearization needs to be achieved without incurring extra dead-time, time resolution degradation, and unreasonable additional resource utilization.
- (iii) The reduction of the dead-time to one single clock period by using a toggling input stage that eliminates the need to reset the TDL. A mechanism to monitor and control the operating mode of the TDL needs to be implemented.
- (iv) The minimization of bubble errors by using a pipelined encoding of the count of 1's and 0's present in the TDL.

Figure 4.1 displays the architecture of the proposed TDC, showing the toggling input stage, a coarse counter, a dual-mode tuned TDL, a dual-mode combinatory encoder of 1's and 0's counters and its controller, and a dual-mode bin width calibrator.

4.4.1 Target device technology

The choice of a target device is instrumental for the performance of the TDC. First of all, the technology node, together with the internal clock recovery and distribution support, and the internal architecture and components like BRAM and

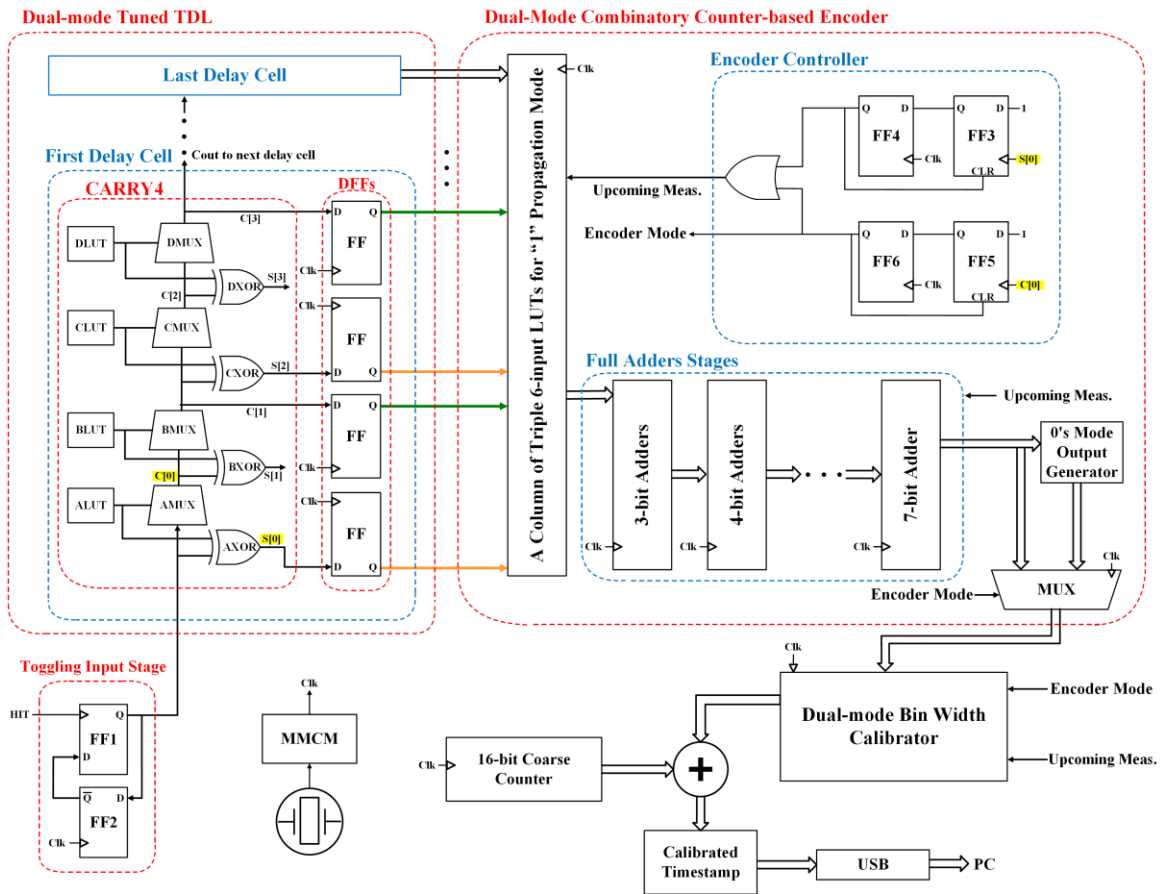


Figure 4.1. Architecture of the proposed TDC

DSPs, define the maximum operating frequency. Second, the particular device defines the available resources. In this work, our target device is an Artix[®]-7 FPGA. Table 4.2 displays the main characteristics of several FPGA platforms.

Table 4.2. Comparison of FPGA Platforms¹

Device	Tech. (nm)	LUTs	FFs	RAM (MB)	Fmax (MHz)
Artix [®] -7	28	134,600	269,200	2.9	464*
Kintex [®] -7	28	298,600	597,200	6.8	741
Virtex [®] -7	28	547,600	1,095,200	13.3	741
Kintex [®] Ultrascale [™]	20	842,400	1,684,800	11.6	850
Virtex [®] Ultrascale [™]	20	2,532,960	5,065,920	28.7	850

* This value is for speed grade -1 while the other values in this column indicate the highest frequencies of other platforms.

The Artix[®]-7 contains half of the resources of a Kintex-7 and less than a quarter of a Virtex-7. Its speed grade is ‘-1’. For this device, if a TDC operates at the maximum frequency, i. e. 464 MHz, the clock period is 2.16ns, then a two-clock-cycle dead-time means 4.31ns. Later we will see that, with the proposed architecture, we can implement 400 TDC channels at 250 Msamples/s in an Artix[®]-7.

4.4.2 Dual-mode tuned TDL

As shown in Figure 4.1, the TDL is built by cascading CARRY4s that are predefined elements of Xilinx FPGAs. Each CARRY4 block has four multiplexers. Each of the multiplexers plays as a delay element of the TDL. Their propagation delay determines the fine time resolution of the TDC. Since this propagation delay depends on the FPGA manufacturing technology, family, and speed grade, the

¹ <https://www.xilinx.com/>

resolution of the TDC is associated with the specific target device. In this work, the target device is a Xilinx Artix[®]-7 XC7A200T-1FBG484 embedded in an Opal Kelly XEM7310 board.

As already mentioned, the non-uniformity of the TDL is highly dependent on the sampling sequence. Although it is not necessarily the most linear pattern, ‘CCCC’ is the most used sampling sequence in literature. To reduce non-uniformity, all the possible sampling patterns of the delay line should be implemented, tested, and compared. In addition, since the propagation delays of 1’s and 0’s are different for the same delay element, the sampling sequences should be tested and compared separately for each of the operating modes. A particular sampling sequence that minimizes nonlinearity for one of the operating modes does not necessarily render the most linear response for the other one.

In order to reduce clock skew, all the delay elements have to be placed in the same clocking region of the FPGA. In addition, the delay line must have a total delay slightly longer than one clock period. Given that the clock frequency is 250 MHz, the delay line has 48 CARRY4, i. e. 192 delay elements. This number has been obtained by trial and error, and it situates the elementary delay around 20 ps.

As displayed in section 0, ‘SCSC’ is the most linear sampling sequence for both “1” and “0” propagation modes.

4.4.3 Toggling input stage

The most extended TDL-based TDC on FPGA operates as follows: when a hit signal is detected, a logic “1” is injected into the delay line, then the state of the TDL is sampled in the next rising edge of the clock. After that, another clock period is employed to propagate a logic “0” through the TDL to set it ready for a new measurement. In this configuration, the dead-time equals at least two clock cycles. That without taking into consideration further processing for improvement

of the encoder performance and/or online calibration, which most likely will increase the dead-time.

The dead-time can be reduced to one clock period by using a dual-mode TDL fed by a toggling input stage. A dual-mode encoder with its control mechanism is also needed. The structure of the input stage is depicted in Figure 4.1. It prepares alternate logic 1's and 0's to be propagated through the delay line. For instance, if the current measurement is being performed over the propagation of a logic "1", for the next measurement, the input stage will provide a logic "0" at the input of the delay line. In this way, TDL does not need one additional clock period to be reset, and new measurements can be carried out in consecutive clock cycles. The dead-time is then shortened to only one clock cycle. Propagating at most one edge through the delay line in each clock period guarantees the sampling of the states without interference between measurements. Since the input stage toggles once in each clock period, it prevents the propagation of spurious signals arising from metastability. However, since the synchronizer is a single flip-flop, metastability can still affect the TDC performance. This might be object of further design refinement.

In addition, the input stage shapes the hit signal generating a pulse whose width equals the time interval between two subsequent hits occurring in different clock periods. FFs sample the time interval between the detection moment and the next rising edge of the clock.

4.4.4 Dual-mode combinatory counter-based encoder

The state of the delay line encodes the interval between the hit signal and the next clock edge in a thermometric code, it has to be converted to a binary code using a T2B encoder. The sampling sequence is tuned to a particular pattern—that will depend on the propagation mode, i. e. 1's or 0's. The encoder needs to read 'C' and

‘S’ nodes and their complementary signals. Moreover, because of the bubble problem, the output of the TDL will not be a clean thermometer code. For example, an ideal thermometer code for a TDL with eight delay cells that samples ‘C’ nodes is 11100000. But, in practice, the sampled states could be for instance 11001000. Therefore, first, the encoder must suppress the bubbles in the thermometric code. In addition to this, since the sampled states have resulted from alternate injection of 1’s and 0’s, the encoder needs to be aware of the operating mode of the input stage. To satisfy these requirements while making a low use of resources, a dual-mode encoder that combines the counts of 1’s and 0’s is introduced.

Let us see how the dual-mode encoder, depicted in Figure 4.1, works in detail. When a logic “1” is injected into the delay line, the encoder counts the 1’s for C nodes and 0’s for S nodes. In the complementary operating mode, the encoder counts the 0’s for C nodes and 1’s for S nodes. In the first stage, the states of the delay line elements are divided into bundles of 6 signals. Then, the column of 6-input LUTs converts each bundle to a 3-bit binary number. For example, the LUTs connected to C nodes are configured to encode the number of 1’s and the LUTs connected to S nodes, to encode the number of 0’s. The full adders calculate the binary number using partial results of the LUTs. To reduce the use of resources, the same column and the same full adders are employed for 0’s propagation mode. In this case, only a subtractor circuit is needed to subtract the number generated for the 1’s propagation mode from the actual number of delay elements to calculate the binary code in 0’s propagation mode. A multiplexer selects the input to the dual-mode bin width calibrator by observing the operating mode. Since the encoder does not depend on the transition point in the TDL, the output is robust against bubble errors.

An encoder controller is employed to inform the encoder about the availability of

a new measurement as well as the operating mode, i. e. whether the hit signal injected into the delay line in the current measurement is a logic “1” or a logic “0”. As depicted in Figure 4.1, the encoder controller employs the C and S nodes of the first delay cell of the carry chain. When a logic “1” is injected into the delay line, the logic state of node C[0] changes to “1” from “0”. Conversely, when a logic “0” is injected, the state of node S[0] transits from “0” to “1”. These changes denote the availability of an upcoming measurement. Moreover, the controller specifies the operating mode of the encoder by observing the logic state of the either C or S nodes. For instance, if both signals, the upcoming measurement signal, and the C node state are “1”, then a logic “1” is propagated through the delay line. If both, the upcoming measurement signal and S node state, are “1” then a logic “0” is injected into the delay line. Table 4.3 summarizes the characteristics of the encoder.

Table 4.3. Characteristics of the encoder

I/O		Resources		Processing Time
Input	Output	LUTs	FFs	
192	8	226	286	7 clocks

4.4.5 Dual-mode bin width calibrator

A code density test with a sufficient number of events allowed us to estimate the size of each bin very precisely. Since the 0’s and the 1’s have different propagation delays, the bin widths in each mode are estimated separately. Then, two different tables containing these estimations are generated, one for each operating mode. These tables assign a fine timestamp to every binary code delivered by the T2B. This assignment mechanism maps each binary output of the encoder to a delay time (t_k) from the origin. In the uncalibrated TDC, t_k is equal to $k \times w_{LSB}$ in both

operating modes. With the calibration mechanism these t_k are calculated as follows:

$$t_{k,i} = \frac{w_{k,i}}{2} + \sum_{j=0}^{k-1} w_{j,i} t_{k,i} = \frac{w_{k,i}}{2} + \sum_{j=0}^{k-1} w_{j,i} \quad (4.3)$$

where $t_{k,i}$ is the accumulated delay (from the origin) and $w_{k,i}$ is the measured bin size, both corresponding to the k -th bin in mode i . Figure 4.2 summarizes the calibration procedure. Section 0 displays the effective reduction of DNL and INL.

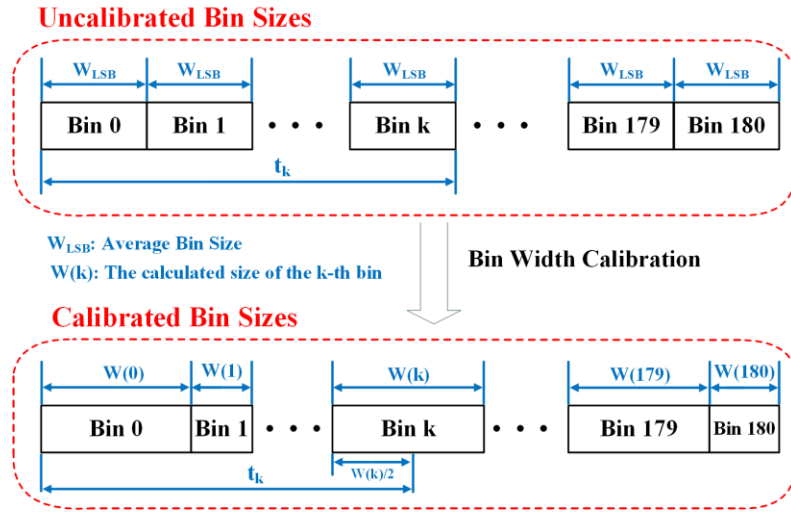


Figure 4.2. Calibration procedure in each operating mode

4.5 Performance Evaluation

4.5.1 Experimental results

An Artix[®]-7 FPGA (XC7A200T-1FBG484), embedded in an Opal Kelly XEM7310 board, has been used to evaluate the performance of the proposed TDC. The application programming interface (API) components of the Opal Kelly such as WireIn and PipeOut are employed to send the measurement results to a host PC through a USB connection. To generate an uncorrelated signal for the code density test, the CFGMCLK port of the STARTUPE2 primitive is employed. This signal is generated by the internal oscillator of the Artix[®]-7 and therefore, it does not have any correlation with the system clock. The code density test is performed by

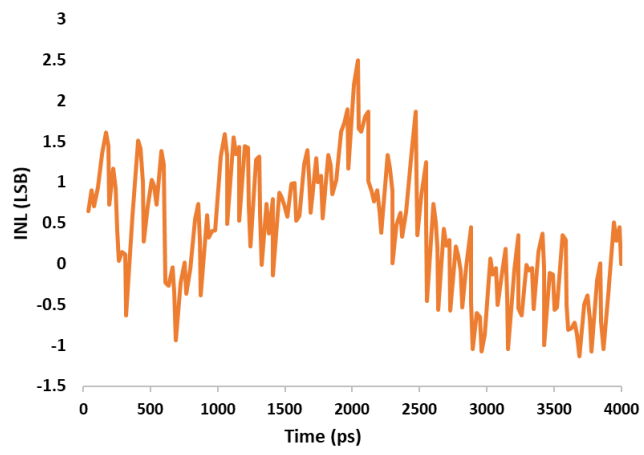
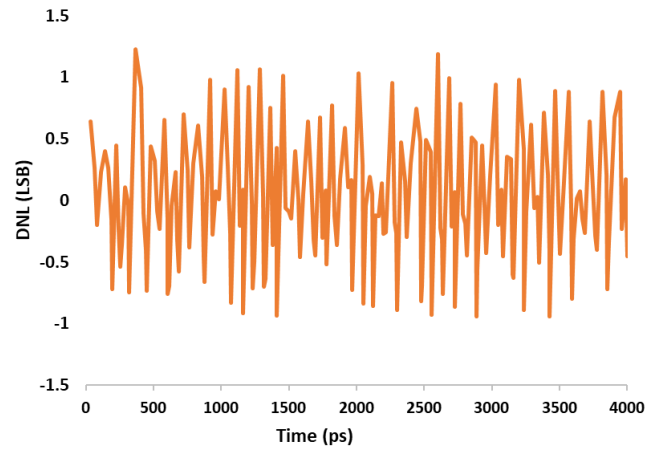
measuring more than one hundred thousand events. The LSB size is obtained by averaging the bin widths and equals 22.1 ps.

To find the sampling sequence that renders the most linear response, a code density test is carried out for all the possible sampling patterns in each operating mode. Among all of them, ‘SCSC’ provides the most linear results for both propagation modes. Table 4.4 illustrates the effectiveness of TDL tuning in each mode. It compares the optimum DNL and INL with that obtained by the default ‘CCCC’ sequence and ‘SCSS’ sequence which provides the second most linear results. Also, DNL and INL of the ‘SCSC’ in “1” and “0” propagation modes, are displayed in Figure 4.3.

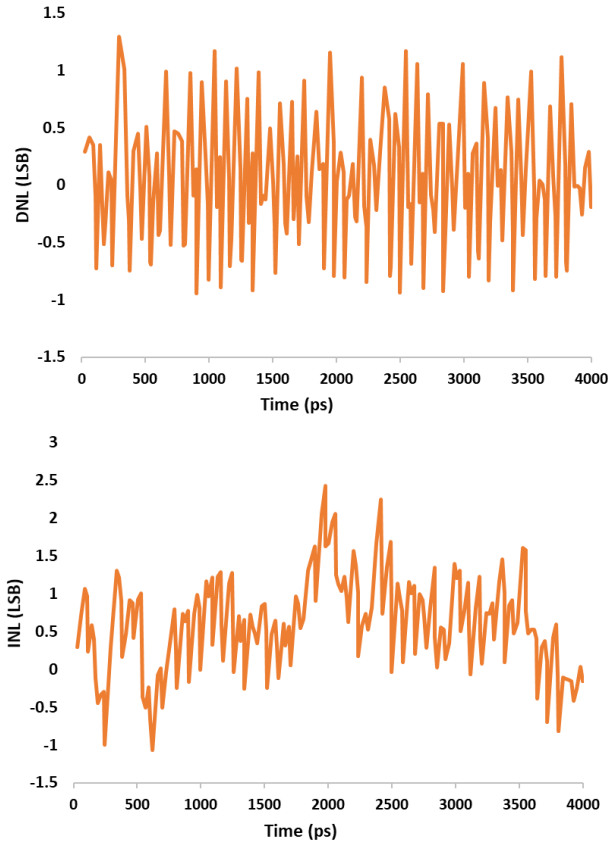
Table 4.5 contains the DNL and INL of calibrated TDC for ‘CCCC’, ‘SCSS’, and ‘SCSC’ sampling sequences in both operating modes. As this table indicates, DNL of the calibrated TDC includes bins with negative widths in at least one of the operating modes of ‘CCCC’ and ‘SCSS’ sequences.

Table 4.4. DNL and INL of uncalibrated TDC for ‘CCCC’, ‘SCSS’, and ‘SCSC’ in both operating modes

Sampling Sequence	Operating Mode	DNL (LSB)	INL (LSB)
‘CCCC’ (default)	“1”	[-0.97 1.88]	[0.00 6.60]
	“0”	[-0.98 1.68]	[-0.94 4.82]
‘SCSS’	“1”	[-0.97 0.89]	[-1.89 2.09]
‘SCSS’	“0”	[-0.95 1.82]	[0.00 5.97]
‘SCSC’	“1”	[-0.94 1.23]	[-1.14 2.49]
‘SCSC’	“0”	[-0.94 1.30]	[-1.06 2.42]



(a)



(b)

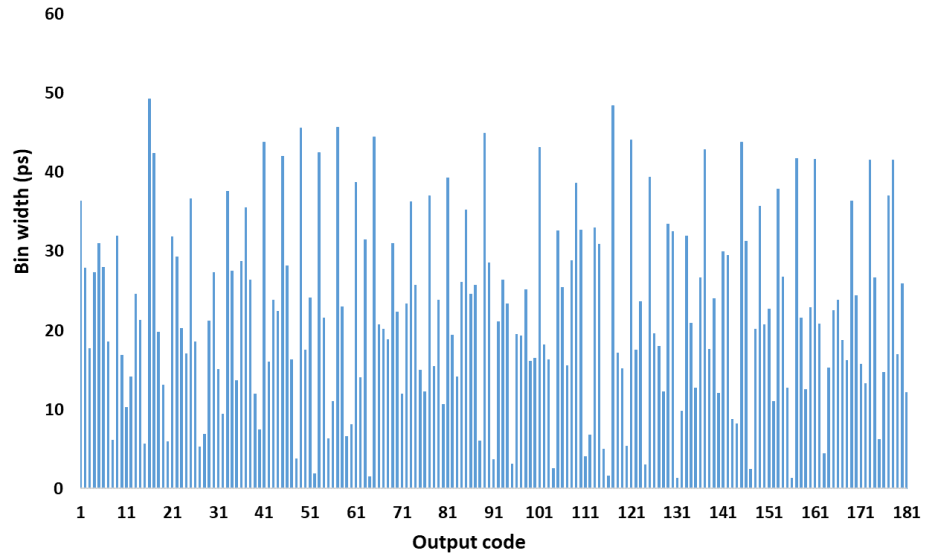
Figure 4.3. DNL and INL of “SCSC” in (a) “1” (b) “0” propagation modes

Table 4.5. DNL and INL of calibrated TDC for ‘CCCC’, ‘SCSS’, and ‘SCSC’ in both operating modes

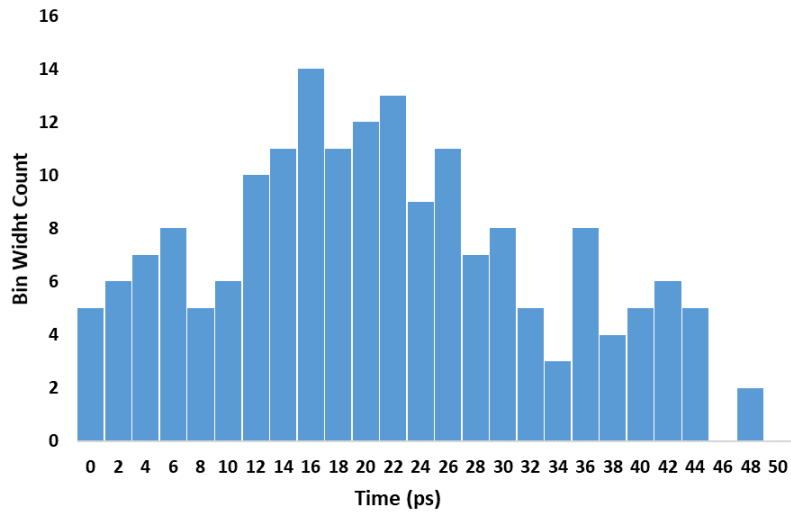
Sampling Sequence	Operating Mode	DNL (LSB)	INL (LSB)
‘CCCC’ (default)	“1”	[-1.37 1.44]	[-0.89 1.58]
	“0”	[-1.18 1.21]	[-0.95 1.71]
‘SCSS’	“1”	[-0.86 1.22]	[-1.15 0.60]
‘SCSS’	“0”	[-1.28 1.26]	[-0.90 1.21]
‘SCSC’	“1”	[-0.71 1.05]	[-0.85 0.86]
‘SCSC’	“0”	[-0.73 1.06]	[-1.17 0.04]

The width of the delay elements and the bin width histogram for ‘SCSC’ in “1” and “0” propagation modes are shown in Figure 4.4 and Figure 4.5, respectively. These measured widths are employed to build the calibration tables. Figure 4.6 shows the calculated content of the calibration tables. The DNL and INL of the calibrated TDC for both operating modes are depicted in Figure 4.7. For the “1”

propagation mode, the DNL and INL are in the range of $[-0.71 \ 1.05]$ LSB and $[-0.85 \ 0.86]$ LSB, respectively. For the “0” propagation mode, the DNL and INL are within $[-0.73 \ 1.06]$ LSB and $[-1.17 \ 0.04]$ LSB, respectively.

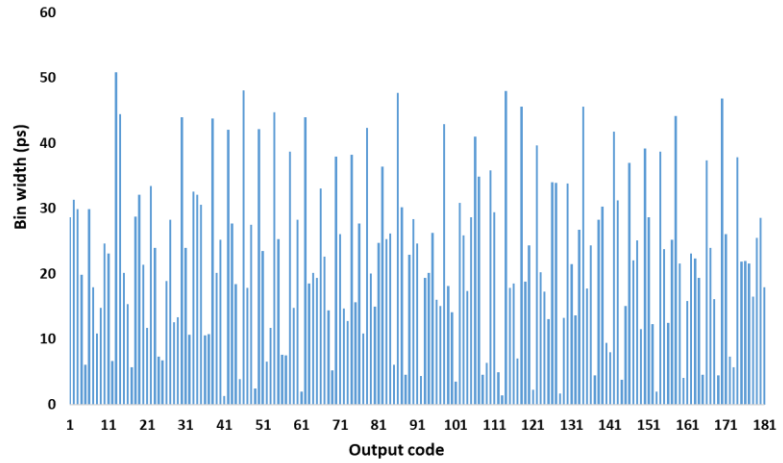


(a)

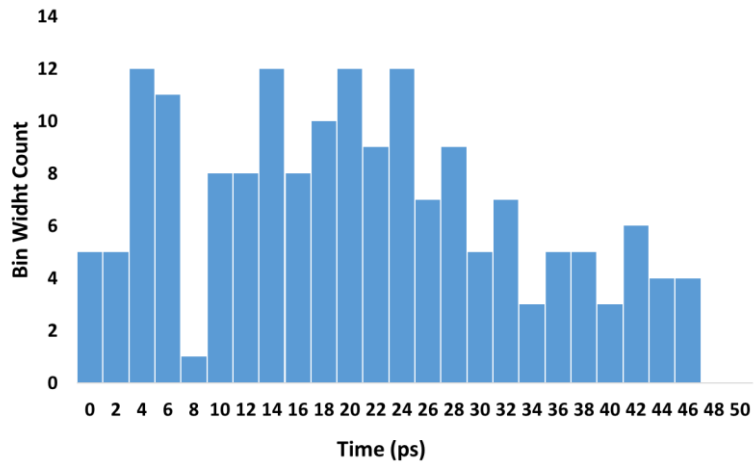


(b)

Figure 4.4. (a) Estimated bin widths and (b) bin width histogram of ‘SCSC’ sequence in “1” propagation mode

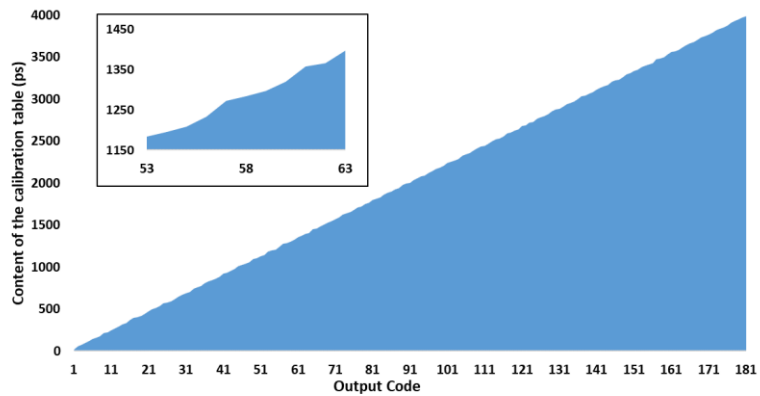


(a)

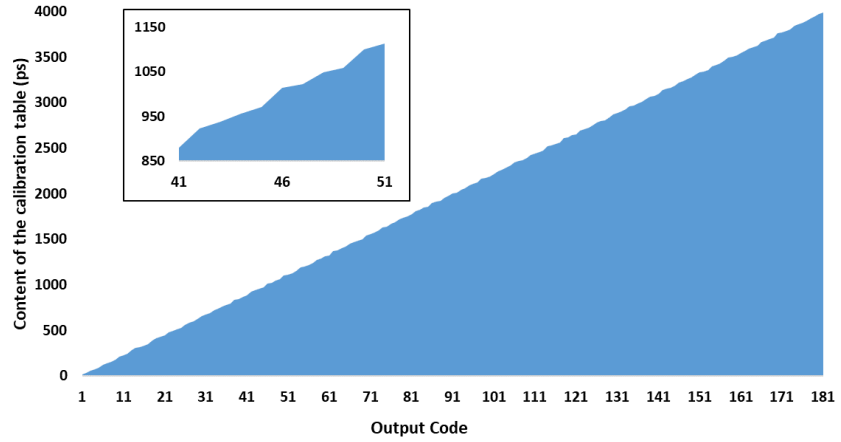


(b)

Figure 4.5. (a) Estimated bin widths and (b) bin width histogram of ‘SCSC’ sequence in “0” propagation mode

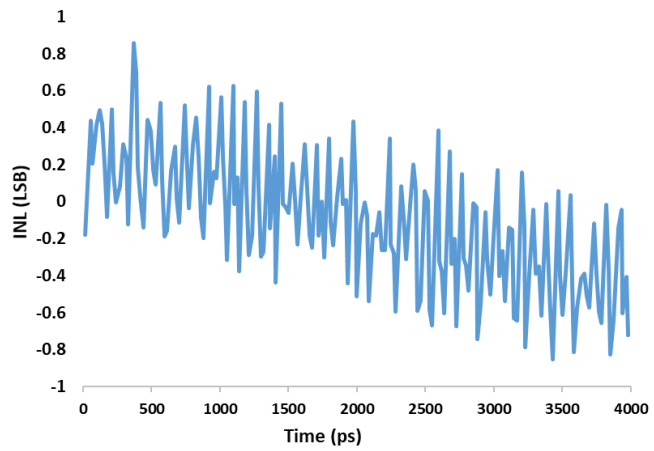
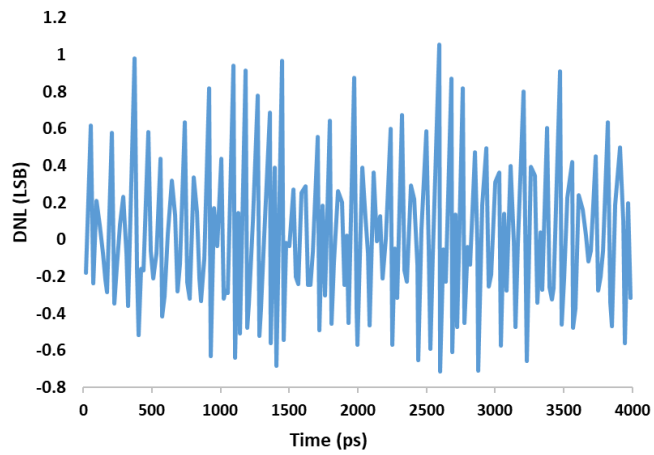


(a)

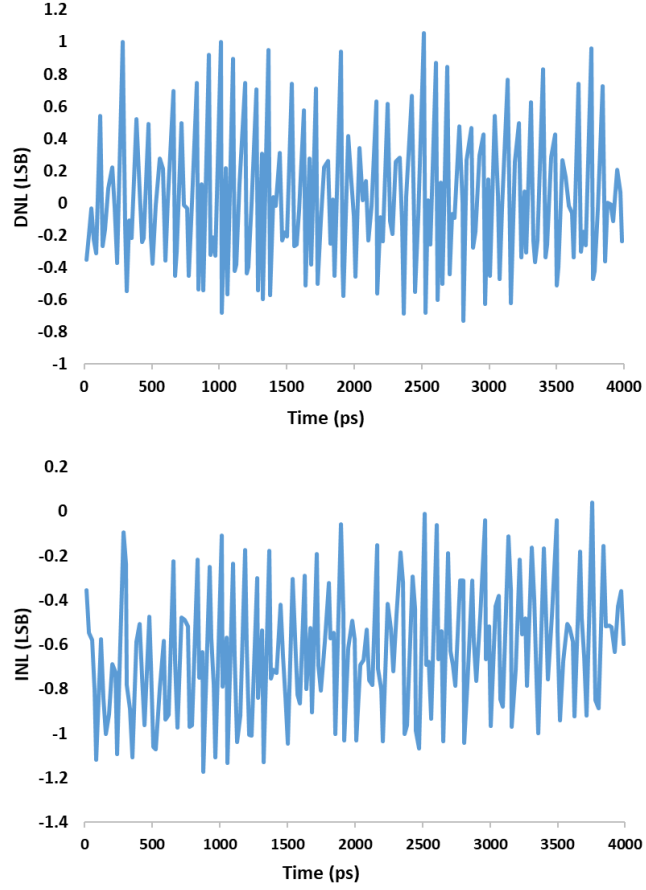


(b)

Figure 4.6. The content of the calibration tables (a) in “1” propagation mode (b) in “0” propagation mode



(a)



(b)

Figure 4.7. DNL and INL of the calibrated TDC for (a) in “1” propagation mode (b) in “0” propagation mode

To calculate the precision of the measurements, several constant time intervals are evaluated by two TDC channels that measure the beginning and end of the interval. The IDELAY2 primitive of the FPGA is employed to generate the time intervals. Regarding that the time intervals are generated inside the FPGA, the jitter is less than the case that the intervals are produced outside of the FPGA. Each interval has been evaluated 10^4 times. The TDC precision (σ) for each time interval is calculated as follows:

$$\sigma = \frac{1}{\sqrt{N-1}} \sqrt{\sum_{k=1}^N \left(t_k - \frac{\sum_{j=1}^N t_j}{N} \right)^2} \quad (4.4)$$

where t_k is the k -th measured value and N is the number of iterations of the

measurement. Since the time interval is constant during the measurements, the calculated TDC precision is the single-shot precision (SSP) of the TDC. Figure 4.8 indicates the TDC precision of the different time intervals. Regarding the values from Figure 4.8, the precision of the TDC equals 22.35 ps in the worst case. This value includes the effect of clock jitters, jitters of input signals, electronic noise, process variations, and temperature and voltage drifts [20].

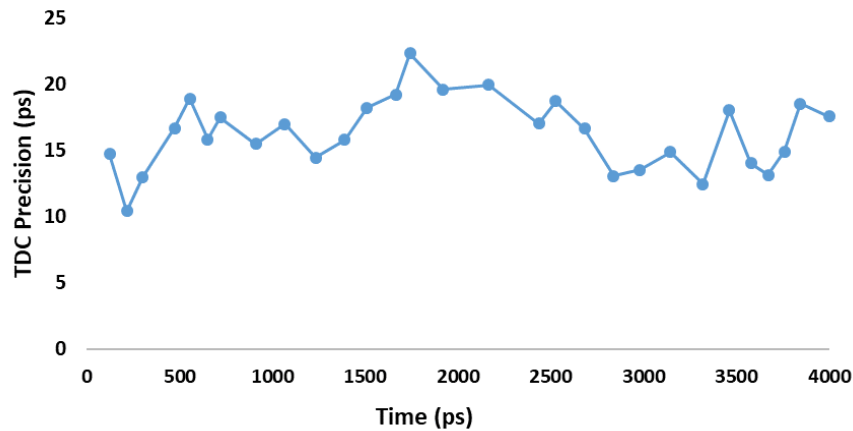


Figure 4.8. TDC precision

Time resolution is the RMS error in the set of time interval measurements. Its theoretical value is $LSB/\sqrt{6}$. In practice, the time resolution of the TDC is estimated by calculating the RMS of the single-shot precision [81].

According to the target board specifications [43], the oscillator RMS period jitter is 2.5 ps. The peak-to-peak jitters of the reference clock generated by PLL and MMCM are available in the summary tab of the Clocking Wizard IP. The peak-to-peak jitter can be converted to RMS jitter using the following equation [82]:

$$Jitter_{p-p} = \alpha \times Jitter_{RMS} \quad (4.5)$$

where α is the crest factor and its typical value is 14.069. The peak-to-peak jitters of the reference clock generated by PLL and MMCM are 79.92 ps and 89.12 ps, respectively. According to the equation (4.5), the clock RMS jitters are 5.68 ps and 6.33 ps in cases of using PLL and MMCM, respectively.

Furthermore, to verify the TDC performance at different temperatures, the TDC precision is evaluated in a wide range of temperatures. As shown in Figure 4.9, the results prove that the TDC has low-temperature sensitivity.

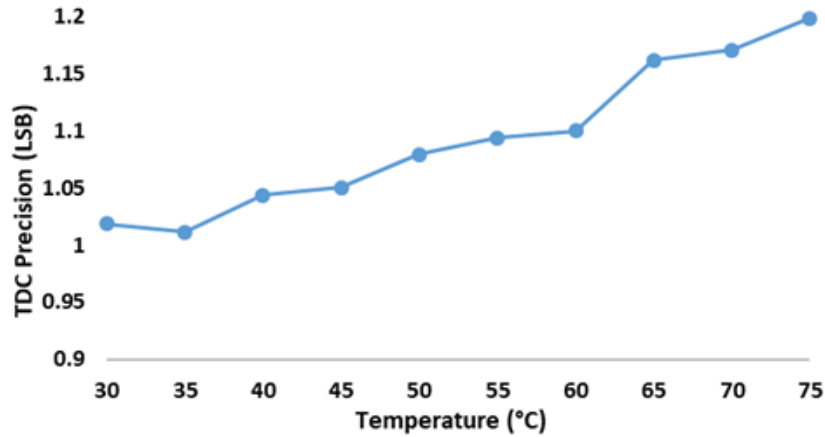


Figure 4.9. TDC precision variations over temperature

Finally, Table 4.6 and Table 4.7 display the measured characteristics and the use of resources and power consumption of one single TDC channel —extracted from the post place & route report— of the proposed architecture. This is evidence of the validity of the approach to deliver high performance while maintaining a low use of resources and low power consumption.

Table 4.6. Characteristics of the proposed TDC

Parameter	Value/Range	Unit
Clock Freq.	250	MHz
LSB	22.1	ps
Meas. Range	262.14	μs
DNL (“1” mode)	[-0.71 1.05]	LSB
INL (“1” mode)	[-0.85 0.86]	LSB
DNL (“0” mode)	[-0.73 1.06]	LSB
INL (“0” mode)	[-1.17 0.04]	LSB
Dead-Time	1	Clock Cycle
Readout Speed	250	MS/s

SSP 22.35 ps

Table 4.7. Resources and power consumption of one channel

Resource	Available	Utilization	Utilization (%)
LUT	133,800	228	0.17
FF	267,600	678	0.25
BRAM	365	2.50	0.68
Total Power		164 mW	
Dynamic Power		33 mW	

4.5.2 Comparison with state-of-the-art FPGA-based TDCs

Table 4.8 compares the proposed architecture with previously reported FPGA-based TDCs. Parameters like the maximum achievable frequency and the LSB size of the TDC are highly dependent on the fabrication technology, FPGA family, and speed grade of the target device. In comparison with our previous work [1], the dead time is halved at the cost of slightly more resource utilization. As shown in Table 4.8, [25] reported a TDC based on the WU method with one clock cycle dead-time. However, according to the reported data on 150 TDC channels, our TDC uses much fewer resources than the average resources for one channel. In addition, since it has not reported the LSB size, INL, and power consumption, a further comparison is not possible. Compared with the other mentioned works, the presented architecture needs fewer clock cycles (i.e., only one clock cycle) to be ready for the next measurement while utilizing much fewer resources and preserving the measurement accuracy. Working with almost equal LSB size and system clock frequency, although [42] consumes slightly less power, it yields worse precision and INL and longer dead-time. Another work with a similar LSB size [69] achieved better DNL and INL at the expense of a much longer dead-time and more power consumption.

4.6 Conclusion

This work presented an FPGA-based TDC based on a dead-time-minimizing and resource-saving approach. The proposed architecture has been evaluated and characterized on a 28-nm Xilinx Artix[®]-7 FPGA. The dead-time is reduced to one clock cycle by using a toggling input stage and a dual-mode combinatory encoder of 1's and 0's counters. The presented encoder is robust against bubble errors while using low resources. To improve the linearity, the most linear sampling sequence is exploited and the encoder outputs are calibrated using bin-width calibration. The measured LSB resolution and TDC precision are 22.1 ps and 22.35 ps, respectively. The proposed TDC features a high-throughput, high-precision, and low-resources therefore is well-suited for high-speed high-accuracy multi-channel applications such as LiDAR and ToF-PET systems.

Table 4.8. Comparison with the state-of-the-art FPGA-based TDCs

Ref.	Year	Method	Device	LSB (ps)	Precision (ps)	DNL (LSB)	INL (LSB)	Dead-time (Tclk)	LUT	FF	BRAM	Power (mW)
[16]	2009	TDL in Turbo mode	Virtex [®] -5	17	24.2	[-1,3.55]	[-2.99,2.58]	1	1208 Slices	NS	NS	NS
[32]	2015	TDL+Bin Realign. & Dec.	Kintex [®] -7	17.6	15	[-1,0.8]	[-0.8,0.8]	NS	NS	NS	NS	NS
[22]	2015	DL loop shrinking	SmartFusion [®]	63.3	61.7	[-0.55,0.28]	[-0.72,0.63]	79	NS	NS	NS	NS
[25]	2015	Two-transition WU	Kintex [®] -7	NS	<10	NS	NS	1	289 Slices ¹	4.75 ¹	NS	NS
[26]	2016	2-phase TDL + Online Cal.	Virtex [®] -6	10	12.83	[-1,1.91]	[-2.2,3.93]	NS	NS	NS	NS	NS
			Kintex [®] -7	10.6	8.13	[-1,1.45]	[-1.23,4.3]	2	577	1641	NS	NS
[28]	2016	Tuned TDL	Virtex [®] -6	10.1	9.82	[-1,1.18]	[-3.03,2.46]	2	577	1641	NS	NS
			Spartan [®] -6	16.7	12.75	[-1,1.22]	[-0.7,2.54]	2	261	787	NS	NS
[30]	2017	Multich. TDL + 1's Counter	Kintex [®] -7	2.45	3.9	NS	NS	2	2433	6258	4	821
[23]	2017	Matrix of counters	Virtex [®] -5	7.4	6.8	[-0.74,0.74]	[-1.52,1.57]	11	666	1410	2	1113
[78]	2017	Tuned TDL + direct hist.	Virtex [®] -7	10.5	5.11	[-0.38,0.87]	[-1.23,1.02]	NS	NS	NS	NS	NS
[42]	2018	Input Stage + Tuned TDL	Spartan [®] -6	25.6	37	[-0.9,1.23]	[-0.43,2.96]	2	415 Slices	NS	NS	131
[27]	2018	Multi-meas. RO-based TDL	Kintex [®] -7	3	5.76	NS	[-8.0,9.3] ²	11	NS	NS	NS	NS
			Virtex [®] -7	10.5	14.59	[-0.05,0.08]	[-0.09,0.11]	NS	1145	1916	1.5	NS
[33]	2019	Sub-TDL+avg. hist.+bin cal.	UltraScale [®]	5.02	7.80	[-0.12,0.11]	[-0.18,0.46]	NS	703	1195	1.5	NS
[71]	2019	8-edge WU	Kintex [®] -7	1.77	3.00	[-1,4.4] ²	[-38,14] ²	2	4010	7503	14	1027
[76]	2020	Dual-transition input+TDL	Kintex [®] -7	10.2	9.7	NS	NS	2	836	1704	3	390
[69]	2020	RO-based Vernier TDC	Stratix [®] III	24.5	28	[-0.20,0.25]	[0.03,0.82]	361	172	986	NS	534
[58]	2020	Multi-time coding line	Kintex [®] -7	1.01	4.5	[-0.98,2.73]	[-17.8,5.1]	NS	2000	2000	NS	NS
			Virtex [®] -6	5.5	6.69	[-0.84,1.67]	[-3.48,3.33]	NS	414	1090	2	464
[73]	2020	Large scale parallel routing	Kintex [®] -7	1.29	3.54	[-1.2,1.4]	[-3.28,3.78]	NS	1002	3900	2	453
			UltraScale TM	3.95	5.55	[-2.75,3.0]	[-5.75,6.0]	NS	334	1100	0.5	634
[1]	2021	Tuned TDL+ combinatory encoder	Artix [®] -7	22.2	26.04	[-0.95,1.18]	[-2.75,1.23]	2	216	638	2.5	164
			UltraScale TM	1.23	3.67	[-0.84,7.93]	[-6.4,24.7]	NS	2460	3463	7.5	1003
[50]	2021	Sub-TDL WU + dual sampl.	UltraScale TM	2.48	3.63	[-0.93,1.68]	[-1.78,2.67]	NS	2460	3463	7.5	1003
[2]	2021	Toggle input + dual-mode encoder	Artix [®] -7	22.1	28.43	[-0.80,1.34]	[-0.73,1.97]	1	228	678	2.5	171
This work		Dual-mode TDC	Artix [®] -7	22.1	22.35	[-0.71 1.05] [-0.73 1.06]	[-0.85 0.86] [-1.17 0.04]	1	228	678	2.5	164

¹Values calculated by averaging the reported resources usage of 150 TDC channels²Values estimated from the related plots

Bibliography

- [1] M. Parsakordasiabi, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “A Low-Resources TDC for Multi-Channel Direct ToF Readout Based on a 28-nm FPGA,” *Sensors*, vol. 21, no. 1, p.308, 2021.
- [2] M. Parsakordasiabi, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “A Novel Approach for Measurement Throughput Maximization in FPGA-based TDCs,” in *IEEE 7th International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP'21)*, Krakow, Poland, Jun. 2021.
- [3] M. Parsakordasiabi, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “An Efficient TDC Using a Dual-Mode Resource-Saving Method Evaluated in a 28-nm FPGA,” *IEEE Transactions on Instrumentation & Measurement*, vol. 71, Dec. 2021.
- [4] M. Parsakordasiabi, I. Vornicu, R. Carmona-Galán and Á. Rodríguez-Vázquez, "A survey on FPGA-based high-resolution TDCs," *In Proc. of the 13th Int. Conf. on Distributed Smart Cameras*, Trento, Italy, Sept. 2019.
- [5] M. Parsakordasiabi, Á. Rodríguez-Vázquez and R. Carmona-Galán, “Design of Readout Channels for Direct-ToF LiDAR,” in *IEEE 36th Int. Conf. on Design of Circuits and Integrated Systems (DCIS)*, Vila do Conde, Portugal, Nov. 2021.
- [6] S. Burri, C. Bruschini, and E. Charbon, “Linospad: A compact linear spad camera system with 64 fpga-based tdc modules for versatile 50 ps resolution time-resolved imaging,” *Instruments*, vol. 1, no. 1, p. 6, 2017.
- [7] D. Stoppa, L. Pancheri, M. Scandiuozzo, L. Gonzo, G.-F. D. Betta, and A. Simoni, “A CMOS 3-D imager based on single photon avalanche diode,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 1, pp. 4–12, Jan. 2007.
- [8] I. Vornicu, R. Carmona-Galán, and Á. Rodríguez-Vázquez, “Arrayable voltage-controlled ring-oscillator for direct time-of-flight image sensors,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 11, pp. 2821–2834, Nov. 2017.
- [9] M. Gersbach, R. Trimananda, Y. Maruyama, M. Fishburn, D. Stoppa, J. Richardson, R. Walker, R. K. Henderson, and E. Charbon, “High frame-rate TCSPC-FLIM using a novel SPAD-based image sensor,” *Detectors and Imaging Devices: Infrared, Focal Plane, Single Photon*, vol. 7780, p. 77801H, 2010.
- [10] C. Veerappan, J. Richardson, R. Walker, D. Li, M. W. Fishburn, Y. Maruyama, D. Stoppa, F. Borghetti, M. Gersbach, R. K. Henderson, and E. Charbon, “Characterization of large-scale non-uniformities in a 20 k TDC/SPAD array integrated in a 130 nm CMOS process,” in *Proc. IEEE Eur. Solid-State Device Res. Conf.*, San Francisco, CA, USA, 20–24 February 2011; pp. 312–314.
- [11] V. C. Spanoudaki and C. S. Levin, “Photo-detectors for time of flight positron emission tomography (ToF-PET),” *Sensors*, vol. 10, no. 11, pp. 10484–10505, 2010.

- [12] L.H.C. Braga, L. Gasparini, L. Grant, R.K. Henderson, N. Massari, M. Perenzoni, D. Stoppa, and R. Walker, "A Fully Digital 8×16 SiPM Array for PET Applications With Per-Pixel TDCs and Real-Time Energy Output," *IEEE J. Solid-State Circuits* 2013, 49, 301–314, doi:10.1109/jssc.2013.2284351.
- [13] M. Zieliński and M. Kowalski, "Review of single-stage time-interval measurement modules implemented in FPGA devices," *Metrology Meas. Syst.*, vol. 16, no. 4, pp. 641–647, 2009.
- [14] R. Machado, J. Cabral, and F. Alves, "Recent Developments and Challenges in FPGA-Based Time-to-Digital Converters," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 11, pp. 4205–4221, Nov. 2019.
- [15] J. Song, Q. An, and S. Liu, "A high-resolution time-to-digital converter implemented in field-programmable-gate-arrays," *IEEE Trans. Nucl. Sci.*, vol. 53, no. 1, pp. 236–241, Feb. 2006.
- [16] C. Favi and E. Charbon, "A 17 ps time-to-digital converter implemented in 65 nm FPGA technology," in *Proc. ACM/SIGDA Int. Symp. Field Program. Gate Arrays*, Monterey, CA, USA, 2009, pp. 113–120.
- [17] M. W. Fishburn, L. H. Menninga, C. Favi, and E. Charbon, "A 19.6 ps, FPGA-based TDC with multiple channels for open source applications," *IEEE Trans. Nucl. Sci.*, vol. 60, no. 3, pp. 2203–2208, Jun. 2013.
- [18] R. Szplet, J. Kalisz, and R. Szymanowski, "Interpolating time counter with 100 ps resolution on a single FPGA device," *IEEE Trans. Instrum. Meas.*, vol. 49, no. 4, pp. 879–883, Aug. 2000.
- [19] A. M. Amiri, M. Boukadoum, and A. Khouas, "A multihit time-to-digital converter architecture on FPGA," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 3, pp. 530–540, Mar. 2009.
- [20] A. Balla, M.M. Beretta, P. Ciambrone, M. Gatta, F. Gonnella, L. Iafolla, M. Mascolo, R. Messi, D. Moricciani, and D. Riondino, "The characterization and application of a low resource FPGA-based time to digital converter," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 739, no. 2, pp. 75–82, 2014.
- [21] M. Büchele, H. Fischer, M. Gorzellik, F. Herrmann, K. Königsmann, C. Schill, and S. Schopferer, "A 128-channel time-to-digital converter (TDC) inside a Virtex-5 FPGA on the GANDALF module," *J. Instrum.*, vol. 7, p. C03008, Mar. 2012.
- [22] J. Zhang and D. Zhou, "A new delay line loops shrinking time-to-digital converter in low-cost FPGA," *Nucl. Instrum. Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip.*, vol. 771, pp. 10–16, 2015.
- [23] M. Zhang, H. Wang, and Y. Liu, "A 7.4 ps FPGA-based TDC with a 1024-unit measurement matrix," *Sensors*, vol. 17, no. 4, Apr. 2017.
- [24] J. Wu and Z. Shi, "The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay," in *2008 IEEE Nuc. Sci. Symp. Conf. Record*, Oct. 2008, pp. 3440–3446.
- [25] C. Liu and Y. Wang, "A 128-Channel, 710 M Samples/Second, and Less Than 10 ps RMS Resolution Time-to-Digital Converter Implemented in a Kintex-7 FPGA," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 3, pp. 773–783, 2015.
- [26] Won, J.Y.; Kwon, S.I.; Yoon, H.S.; Ko, G.B.; Son, J.-W.; Lee, J.S. Dual-Phase Tapped-Delay-Line Time-to-Digital Converter With On-the-Fly Calibration Implemented in 40 nm FPGA. *IEEE Trans. Biomed. Circuits Syst.* 2016, 10, 231–242, doi:10.1109/tbcas.2015.2389227.

- [27] J. Kuang, Y. Wang, Q. Cao, and C. Liu, "Implementation of a high precision multi-measurement time-to-digital convertor on a Kintex-7 FPGA," *Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip.*, vol. 891, pp. 37–41, May 2018.
- [28] J. Y. Won and J. S. Lee, "Time-to-Digital Converter Using a Tuned-Delay Line Evaluated in 28-, 40-, and 45-nm FPGAs," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 7, pp. 1678–1689, Jul. 2016.
- [29] J. Wu, "Several Key Issues on Implementing Delay Line Based TDCs Using FPGAs," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 3, pp. 1543–1548, Jun. 2010.
- [30] Y. Wang and C. Liu, "A 3.9 ps Time-Interval RMS Precision Time-to-Digital Converter Using a Dual-Sampling Method in an UltraScale FPGA," *IEEE Trans. Nucl. Sci.*, vol. 63, no. 5, pp. 2617–2621, Oct. 2016.
- [31] X. Hu, L. Zhao, S. Liu, J. Wang, and Q. An, "A stepped-up tree encoder for the 10-ps Wave Union TDC," *IEEE Trans. Nucl. Sci.*, vol. 60, no. 5, pp. 3544–3549, Oct. 2013.
- [32] Y. Wang and C. Liu, "A nonlinearity minimization-oriented resource-saving time-to-digital converter implemented in a 28 nm Xilinx FPGA," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 5, pp. 2003–2009, Oct. 2015.
- [33] H. Chen and D. D. Li, "Multichannel, Low Nonlinearity Time-to-Digital Converters Based on 20 and 28 nm FPGAs," *IEEE Trans. Ind. Electron.*, vol. 66, no. 4, pp. 3265–3274, Apr. 2019.
- [34] Y. Wang, J. Kuang, C. Liu, and Q. Cao, "A 3.9-ps RMS Precision Time-to-Digital Converter Using Ones-Counter Encoding Scheme in a Kintex-7 FPGA," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 10, pp. 2713–2718, Oct. 2017.
- [35] J. Wu, Z. Shi, and I. Y. Wang, "Firmware-only implementation of timeto-digital converter (TDC) in field-programmable gate array (FPGA)," in *Proc. IEEE Nucl. Sci. Symp. Conf. Rec.*, Oct. 2003, pp. 177–181.
- [36] J. Wang, S. Liu, Q. Shen, H. Li, and Q. An, "A fully fledged TDC implemented in field-programmable gate arrays," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 2, pp. 446–450, Apr. 2010.
- [37] S. Cova and M. Bertolaccini, "Differential linearity testing and precision calibration of multichannel time sorters," *Nucl. Instrum. Methods*, vol. 77, no. 2, pp. 269–276, Jan. 1970.
- [38] E. Arabul, A. Girach, J. Rarity, and N. Dahnoun, "Precise multi-channel timing analysis system for multi-stop LIDAR correlation," in *Proc. IEEE Int. Conf. Imag. Syst. Techn. (IST)*, Beijing, China, Oct. 2017, pp. 1–6.
- [39] G. Cao, H. Xia, and N. Dong, "An 18-ps TDC using timing adjustment and bin realignment methods in a cyclone-IV FPGA," *Rev. Sci. Instrum.*, vol. 89, no. 5, 2018, Art. no. 054707.
- [40] T. Townsend, Y. Tang, and J. Chen, "Highly-linear FPGA-based Data Acquisition System for Multi-channel SiPM Readout," In *Proceedings of the Topical Workshop on Electronics for Particle Physics—PoS(TWEPP2019)*, Santiago De Compostela, Spain, 2–6 September 2019.
- [41] H. Homulle and E. Charbon, "FPGA designs for reconfigurable converters," TUDelft, Netherlands, 2015, [Online], Available: http://cas.tudelft.nl/fpga_tdc/TDC_basic.html
- [42] A. Tontini, L. Gasparini, L. Pancheri, and R. Passerone, "Design and Characterization of a Low-Cost FPGA-Based TDC," *IEEE Trans. Nucl. Sci.*, vol. 65, no. 2, pp. 680–690, Feb. 2018.
- [43] Opal Kelly, "XEM7310 User's Manual," 3 March 2018. Available online: <https://docs.opalkelly.com/display/XEM7310>

- [44] Opal Kelly, “Front Panel User’s Manual,” Available online:
<http://assets00.opalkelly.com/library/FrontPanel-UM.pdf>
- [45] Xilinx, “Artix-7 FPGAs Data Sheet: DC and AC Switching Characteristics (DS181),” 18 June 2018, Available online:
https://www.xilinx.com/support/documentation/data_sheets/ds181_Artix_7_Data_Sheet.pdf
- [46] Xilinx, “7 Series FPGAs Clocking Resources (UG472)” 30 July 2018, Available online:
https://www.xilinx.com/support/documentation/user_guides/ug472_7Series_Clocking.pdf
- [47] K.S. Kim, Y.H. Kim, W. Yu, and S.H. Cho, “A 7 bit, 3.75 ps resolution two-step time-to-digital converter in 65 nm CMOS using pulsetrain time amplifier,” *IEEE J. Solid-State Circuits*, vol. 48, no. 4, pp. 1009–1017, Apr. 2013.
- [48] J. Y. Won and J. S. Lee, “Highly integrated FPGA-only signal digitization method using single-ended memory interface input receivers for time-of-flight PET detectors”, *IEEE Trans. Biomed. Circuits Syst.*, vol. 12, no. 6, pp. 1401-1409, Dec. 2018.
- [49] Y. Wang, J. Kuang, C. Liu, Q. Cao, and D. Li, “A flexible 32-channel time-to-digital converter implemented in a Xilinx Zynq-7000 field programmable gate array,” *Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip.*, vol. 847, no. September, pp. 61–66, 2017.
- [50] W. Xie, H. Chen, and D.D.U. Li, “Efficient time-to-digital converters in 20 nm FPGAs with wave union methods,” *IEEE Trans. Ind. Electron.*, vol. 69, no. 1, pp. 1021-1031, Jan. 2022.
- [51] D. Li, M. Liu, R. Ma, and Z. Zhu, “An 8-ch LIDAR receiver based on TDC with multi-interval detection and real-time in-situ calibration,” *IEEE Trans. Instrum. Meas.*, vol. 69, no. 7, pp. 5081-5090, Jul. 2020.
- [52] H. Seo, H. Yoon, D. Kim, J. Kim, S.J. Kim, J.H. Chun, and J. Choi, “Direct TOF Scanning LiDAR Sensor With Two-Step Multievent Histogramming TDC and Embedded Interference Filter,” *IEEE J. of Solid-State Circuits*, vol. 56, no. 4, pp. 1022 – 1035, Jan. 2021.
- [53] W. Jiang, Y. Chalich, and M.J. Deen, “Sensors for positron emission tomography applications,” *Sensors*, vol. 19, no. 22, p. 5019, Nov. 2019
- [54] E. Venialgo et al., “Toward a full-flexible and fast-prototyping TOF-PET block detector based on TDC-on-FPGA,” *IEEE Trans. Radiat Plasma Med. Sci.*, vol. 3, pp. 538–548, Sept. 2019.
- [55] M.A. Daigneault and J. P. David, “A high-resolution time-to-digital converter on FPGA using dynamic reconfiguration,” *IEEE Trans. Instrum. Meas.*, vol. 60, no. 6, pp. 2070–2079, Jun. 2011.
- [56] Y. Wang, Q. Cao, and C. Liu, “A Multi-Chain Merged Tapped Delay Line for High Precision Time-to-Digital Converters in FPGAs,” *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 65, no. 1, pp. 96–100, Jan. 2018.
- [57] T. Sui, Z. Zhao, S. Xie, Y. Xie, Y. Zhao, Q. Huang, J. Xu, and Q. Peng, “A 2.3-ps RMS Resolution Time-to-Digital Converter Implemented in a Low-Cost Cyclone V FPGA,” *IEEE Trans. Instrum. Meas.*, vol. 68, no. 10, pp. 3647–3660, Oct. 2019.
- [58] P. Kwiatkowski and R. Szplet, “Efficient implementation of multiple time coding lines-based TDC in an FPGA device,” *IEEE Trans. Instrum. Meas.*, vol. 69, no. 10, pp. 7353–7364, Oct. 2020.

- [59] Y. Wang, P. Kuang, and C. Liu, "A 256-channel multi-phase clock sampling-based time-to-digital converter implemented in a Kintex-7 FPGA," in *Proc. Of IEEE Instrum. and Meas. Technology Conf.*, Taipei, Taiwan, May 2016.
- [60] T. Xiang, L. Zhao, X. Jin, T. Wang, S. Chu, C. Ma, S. Liu, and Q. An, "A 56-ps multi-phase clock time-to-digital convertor based on Artix-7 FPGA," in *19th IEEE-NPSS Real Time Conf.*, Nara, Japan, May 2014, pp. 1–4.
- [61] R. Szplet, P. Kwiatkowski, and J. Tyburski. "Precise Time Digitizer Based on Counting Method and Multiphase In-Period Interpolation," In *Joint Conf. of the IEEE Int. Frequency Control Symp. and European Frequency and Time Forum (EFTF/IFC)*, Orlando, USA, Apr. 2019.
- [62] X. Qin, L. Wang, D. Liu, Y. Zhao, X. Rong, and J. Du, "A 1.15 ps bin size and 3.5 ps single-shot precision time-to-digital-converter with on-board offset correction in an FPGA," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 12, pp. 2951–2957, Dec. 2017.
- [63] Q. Shen, S. Liu, B. Qi, Q. An, S. Liao, P. Shang, C. Peng, W. Liu, "A 1.7 ps equivalent bin size and 4.2 ps rms FPGA TDC based on multichain measurements averaging method," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 3, pp. 947-954, Jun. 2015.
- [64] P. Kwiatkowski, "Employing FPGA DSP blocks for time-to-digital conversion," *Metrol. Meas. Syst.*, vol. 26, no. 4, pp. 631–643, Dec. 2019.
- [65] X. Qin, M.-D. Zhu, W.-Z. Zhang, Y.-H. Lin, Y. Rui, Z. Rong, and J. Du, "A high resolution time-to-digital-converter based on a carry-chain and DSP48E1 adders in a 28-nm field-programmable-gate-array," *Rev. Sci. Instrum.*, vol. 91, no. 2, p. 024708, Feb. 2020.
- [66] R. Szplet and K. Klepacki, "An FPGA-Integrated Time-to-Digital Converter Based on Two-Stage Pulse Shrinking," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 6, pp. 1663–1670, Jun. 2010.
- [67] T. E. Rahkonen and J. T. Kostamovaara, "The use of stabilized CMOS delay lines for the digitization of short time intervals," *IEEE J. SolidState Circuits*, vol. 28, no. 8, pp. 887–894, Aug. 1993.
- [68] J. Zhang and D. Zhou, "An 8.5-ps Two-Stage Vernier Delay-Line Loop Shrinking Time-to-Digital Converter in 130-nm Flash FPGA," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 2, pp. 406–414, Feb. 2018.
- [69] K. Cui and X. Li, "A high-linearity Vernier time-to-digital converter on FPGAs with improved resolution using bidirectional-operating Vernier delay lines," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 8, pp. 5941-5949, Aug. 2020.
- [70] S. Berrima, Y. Blaquièrre, and Y. Savaria, "A Multi-Measurements RO-TDC implemented in a Xilinx Field Programmable Gate Array", in *2017 IEEE Int. Symp. on Circuits & Syst. (ISCAS)*, May 2017, pp. 1-4.
- [71] Y. Wang, X. Zhou, Z. Song, J. Kuang, and Q. Cao, "A 3.0-ps rms Precision 277-MSamples/s Throughput Time-to-Digital Converter Using Multi-Edge Encoding Scheme in a Kintex-7 FPGA," *IEEE Trans. Nucl. Sci.*, vol. 66, no. 10, pp. 2275–2281, Oct. 2019.
- [72] N. Lusardi, F. Garzetti, N. Corna, R. D. Marco, and A. Geraci, "Very high-performance 24-channels time-to-digital converter in Xilinx 20-nm kintex UltraScale FPGA," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, Manchester, UK, Oct. 2019.

- [73] M. Zhang, K. Yang, Z. Chai, H. Wang, Z. Ding, and W. Bao, "High-Resolution Time-to-Digital Converters Implemented on 40-, 28-, and 20-nm FPGAs," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1-10, Nov. 2020.
- [74] A. Samarah and A. C. Carusone, "A digital phase-locked loop with calibrated coarse and stochastic fine TDC," *IEEE J. Solid-State Circuits*, vol. 48, no. 8, pp. 1829–1841, Aug. 2013.
- [75] S. Ito, S. Nishimura, H. Kobayashi, S. Uemori, Y. Tan, N. Takai, T.J. Yamaguchi, and K. Niitsu, "Stochastic TDC architecture with self-calibration," *IEEE Asia Pacific Conf. on Circuits and Syst.*, Kuala Lumpur, Malaysia, Dec. 2010.
- [76] X. Kong, Y. Wang, Z. Song, X. Zhou, J. Lin, and J. Kuang, "A Resource-saving Method for Implementation of High-Performance Time-to-Digital Converters in FPGA," In *2020 IEEE Int. Instrum. and Meas. Technology Conf. (I2MTC)*, Dubrovnik, Croatia, May 2020, pp. 1-5.
- [77] J. Kalisz, "Review of methods for time interval measurements with picosecond resolution," *Metrologia*, vol. 41, no. 1, pp. 17–32, Dec. 2003.
- [78] H. Chen, Y. Zhang, and D. D.-U. Li, "A Low Nonlinearity, Missing-Code Free Time-to-Digital Converter Based on 28-nm FPGAs With Embedded Bin-Width Calibrations," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 7, pp. 1912–1921, Jul. 2017.
- [79] J. Zheng, P. Cao, D. Jiang, and Q. An, "Low-Cost FPGA TDC With High Resolution and Density," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 6, pp. 1401–1408, Jun. 2017.
- [80] Y. Wang, J. Kuang, C. Liu, Q. Cao, and D. Li, "A flexible 32-channel time-to-digital converter implemented in a Xilinx Zynq-7000 field programmable gate array," *Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip.*, vol. 847, no. September, pp. 61–66, 2017.
- [81] F. Baronti, L. Fanucci, D. Lunardini, R. Roncella, and R. Saletti, "On the differential nonlinearity of time-to-digital converters based on delay-locked-loop delay lines," *IEEE Trans. Nucl. Sci.*, vol. 48, no. 6, pp. 2424–2431, Dec. 2001.
- [82] Silicon Labs, "Timing Jitter Tutorial & Measurement Guide", Available online: <https://www.silabs.com/documents/public/white-papers/timing-jitter-tutorial-and-measurement-guide-ebook.pdf>

Scientific Publications

The contributions to the scientific bibliography made in this dissertation are listed below.

Journal Papers

- [1] **M. Parsakordasiabi**, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “An Efficient TDC Using a Dual-Mode Resource-Saving Method Evaluated in a 28-nm FPGA,” *IEEE Transactions on Instrumentation & Measurement*, vol. 71, Dec. 2021.
- [2] **M. Parsakordasiabi**, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “A Low-Resources TDC for Multi-Channel Direct ToF Readout Based on a 28-nm FPGA,” *Sensors, MDPI*, vol. 21, no. 1, p.308, Jan. 2021.

Conference Papers

- [1] **M. Parsakordasiabi**, Á. Rodríguez-Vázquez and R. Carmona-Galán, “Design of Readout Channels for Direct-ToF LiDAR,” in *IEEE 36th Int. Conf. on Design of Circuits and Integrated Systems (DCIS)*, Vila do Conde, Portugal, Nov. 2021.
- [2] **M. Parsakordasiabi**, I. Vornicu, Á. Rodríguez-Vázquez and R. Carmona-Galán, “A Novel Approach for Measurement Throughput Maximization in FPGA-based TDCs,” *IEEE 7th Int. Conf. on Event-based Control, Communication, and Signal Processing (EBCCSP)*, Krakow, Poland, Jun. 2021.
- [3] **M. Parsakordasiabi**, I. Vornicu, R. Carmona-Galán and Á. Rodríguez-Vázquez, “A survey on FPGA-based high-resolution TDCs,” In *Proceedings of the 13th Int. Conf. on Distributed Smart Cameras (ICDCS)*, Trento, Italy, Sep. 2019.

Workshops

- [1] **M. Parsakordasiabi**, I. Vornicu, R. Carmona-Galán and Á. Rodríguez-Vázquez, “A 26ps RMS Precision Time to Digital Converter Using an Improved Delay Line

Implemented in an Artix 7 FPGA”, in *8th Workshop on Architecture of Smart Camera (WASC’20)*, Dec. 14-16, 2020, Ghent, Belgium (Online edition).

[2] **M. Parsakordasiabi**, I. Vornicu, R. Carmona-Galán and Á. Rodríguez-Vázquez, “Evaluation of Architectures for FPGA-Implementation of High-Resolution TDCs”, in *9th Workshop on Architecture of Smart Camera (WASC’19)*, Jul. 1-2, 2019, Rennes, France.

Media Appearances

[1] Proyecto Achieve-ITN, La Noche Europea de Los Investigadores, Nov. 2020
<https://www.youtube.com/watch?v=EtQbv4L0LSw>

[2] Mojtaba Parsakordasiabi (ESR2), ACHIEVE-ITN Network Activities, Oct. 2020
<https://www.youtube.com/watch?v=mx3W37KYeBg>