A Review of New Time-to-Digital Conversion Techniques

Scott Tancock, Ekin Arabul, and Naim Dahnoun

Abstract— **Time-to-digital converters (TDCs) are vital components in time and distance measurement and frequency-locking applications. There are many architectures for implementing TDCs, from simple counter TDCs to hybrid multi-level TDCs, which use many techniques in tandem. This article completes the review literature of TDCs by describing new architectures along with their benefits and tradeoffs, as well as the terminology and performance metrics that must be considered when choosing a TDC. It describes their implementation from the gate level upward and how it is affected by the fabric of the device [field-programmable gate array (FPGA) or application-specific integrated circuit (ASIC)] and suggests suitable use cases for the various techniques. Based on the results achieved in the current literature, we make recommendations on the appropriate architecture for a given task based on the number of channels and precision required, as well as the target fabric.**

Index Terms— **Application-specific integrated circuits (ASIC), field-programmable gate arrays (FPGAs), measurement techniques, review, time measurement, time-to-digital conversion.**

I. Introduction

TIME-TO-DIGITAL converters (TDCs) play a vital role in almost all computational systems in existence. From their appearance in phase-locked loops (PLLs), where they measure the difference between the loop and the reference clock to avoid clock drift, to time-of-flight (ToF) applications where the time between an emission and reception is measured to discover the information about an object from which the signal was reflected or the environment through which the signal passed. In addition, there are also quantum versions of these applications, where the signal is a single quantum, and the PLL or ToF measurement must perform well despite some quanta being lost in-flight. They also make an appearance in medical imaging, as some systems such as positron emission tomography (PET) and fluorescence lifetime imaging (FLIM) use the ToF or absorption time of tissues or substances to form an internal image of a complex structure such as a human body.

TDCs can be also utilized as time taggers in time-correlation systems such as coincidence counters. These systems play an essential role in quantum physics experiments for gating the events of interest from the background noise and measuring the gamma-ray correlation in PET systems. Coincidence counters are correlator tools which are tailored to measure the occurrences of simultaneous signal events over multiple channels. This is done by checking whether the events are happening within the same time window called the coincidence window. In a PET scanner setup, a positron-emitting radiotracer substance is introduced into the subject's body and the subject is surrounded by detectors which observe the gamma rays. While the positron-emitting substance decays inside the patient's body, the positrons meet with electrons and this results in the annihilation of both the positron and electron. An annihilation of a positron and electron pair generates two gamma rays that travel in opposite directions toward two-photon detectors placed in the surroundings. Counting the coincidences caused by the two gamma rays emitted allows analysing the radiotracer's distribution in the body, which is then used for image formation. TDCs can be utilized to digitize gamma-ray pairs' ToFs in such a setup. An example of such a scheme can be found in [1].

Another area where TDCs are commonly used is spectrometry. Spectrometry can be defined as distinguishing mixed substances based on an interaction between light and their matter. Common spectrometry examples where TDCs can be used are ToF mass spectrometry (TOFMS) and fluorescence spectrometry. In TOFMS, the ToFs of ions are used to measure the ions' mass to charge ratio. This process starts with ionizing the atoms and molecules to be measured causing the required number of electrons to be knocked off to form a positive ion. Then, the ions are accelerated to the same kinetic energy and projected for a known distance. Since heavier and lighter ions have different velocities due to their different masses, their ToF will vary and this reveals information about their charge to mass ratio [2]. ToF tomography is an example of mass spectrometry (MS). Fluorescence spectroscopy, which is commonly used in chemistry, biomedicine and medicine to analyze the organic compounds [3], is used to determine the fluorescence content of the substance by measuring the decay time after the substance has been excited by a light beam. TDCs are employed as a part of the time-correlated single-photon counting (TCSPC) tools used to measure the decay time [4].

Range finding ToF systems such as light detection and rangings (LiDARs) are another common application of TDCs. In a typical LiDAR, a start signal corresponds to the time when the laser transmitter starts to illuminate the target with photons and the detection of reflected photons from the target by the receiver is denoted as the stop signal. The photons' ToF can be used to measure the distance. The differences between these values are used to determine the photons' ToF between transmission and detection. A TDC is employed to quantize and digitize the events of start and stop signals in such applications.

Manuscript received January 28, 2019; revised July 12, 2019; accepted July 31, 2019. Date of publication August 22, 2019; date of current version September 13, 2019. This work was supported by the EPSRC. The Associate Editor coordinating the review process was Fabricio Baptista. *(Corresponding author: Scott Tancock.)*

The authors are with the Department of Electrical and Electronic Engineering, University of Bristol, Bristol BS8 1TH, U.K.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2019.2936717

As the time resolution of single-photon detectors is in the order of 100 ps [5], the desired resolution and precision of a TDC is approximately 10 ps. On the other hand, readily available avalanche photodiodes (APDs) can easily reach $<$ 10-ps rise time error [6], so a TDC resolution and precision of $<$1 ps is desirable to obtain the highest accuracies. The differential nonlinearity (DNL) must also be low enough to avoid significant mismeasurements if the largest code is hit by coincidence. For PET detectors, the full-width at half-maximum (FWHM) is in the order of 10 ns [7], so a detector resolution of 1 ns, achievable with a counter, is acceptable.

Many TDC products are available commercially for different purposes. An example of a low-cost two-channel LiDAR TDC is the Texas Instruments TDC7201 chip which provides 55-ps resolution [8]. For applications such as coincidence correlation, a TDC such as Swabian Instrument's Time Tagger, which provides 18 channels with 10-ps second resolution, can be used [9]. PicoQuant's Picoharp and Hydraharp series, which can provide down to 1-ps resolution in up to eight channels of operation, are popular TDC products in TCSPC [10]. In addition, IdQuantique's ID900 Time Controller is another example of a commercial time tagging box which provides 20-ps resolution with up to 64-channel operation [11].

Previously, Porat [12] wrote the first review on sub-nanosecond time interval measurements, examining time-to-amplitude converter (TAC), counter, and Vernier TDCs. Then, Kalisz [13] wrote a review on time interval measurement methods, which analyzed the architecture of TAC, dual-slope time amplifier, counter, delay line, Vernier and voltage-controlled oscillator (VCO) TDCs. Zielinski [14] wrote a review of time interval measurement techniques, which covered counter, multi-phase clock, delay line, and Vernier TDCs implemented on field-programmable gate arrays (FPGAs).

Napolitano *et al.* [15] wrote a "survey" on time interval measurement techniques, which discussed counter, dual-slope time amplifier, TAC, Vernier, delay line, and stochastic TDCs, while Henzler [16] wrote a book describing all the aforementioned as well as local passive interpolation (LPI), gated ring oscillator (GRO), pulse shrinking and metastable time amplifier TDCs in detail, with a particular focus on their application-specific integrated circuit (ASIC) implementation and proof of concept. Then, Wang *et al.* [17] also performed a review focused on the CMOS implementations of the delay line, pulse-shrinking, Vernier, GRO, metastable time amplifier, and stochastic TDCs. Most recently, Chaberski *et al.* [18] published a comparison covering two forms of delay line TDC (delayed start and delayed stop) and a Vernier TDC. They also mention the use of multiple delay lines in an equivalent coding line (ECL) topology.

This article aims to complete the current review literature by describing the methods that have not yet been mentioned, along with their benefits and drawbacks, to aid the designer in choosing an appropriate TDC to satisfy their design requirements. It will look at TDCs implemented both on FPGAs) and ASICs, which differ vastly in terms of the resources and flexibility available, both in production and in the field and hence benefit more or less from various techniques.

The rest of this article will be structured as follows. The article will start with Section II where calibration and linearization techniques for TDCs will be described. We will then explore new TDC architectures in Section III. Section III-A will be an introduction to the section. Section III-B will describe successive approximation TDCs. Section III-C will describe algorithmic TDCs. Section III-D will describe wave union launchers. Section III-E will describe SERDES TDCs. Section III-F will describe DSP delay lines. Section IV will then compare the results achieved from the various designs in the literature while considering the difference in technology platforms. This will lead on to Section V, where the various advantages and disadvantages of the architectures will be compared, with Section VI concluding with recommendations on which architectures provide the best tradeoffs.

II. Calibration and Linearization Techniques

In an ideal fine TDC, each delay element that is used for time quantization should have an equal bin width. However, due to the internal routing of the TDC and temperature and power fluctuations, the TDC suffers from non-constant and inconsistent bin widths. DNL and integral nonlinearity (INL) are expressions used to define the measurement errors affecting the TDCs' linearity. To overcome these converter errors, various calibration (II-A and II-B) and linearization (II-C and II-D) methods are utilized.

A. Statistical Code Density Testing Method

The statistical code density testing method is one of the common methods used for TDC calibration and is described in [19] and [20].

The measurement of each bin width determines the likelihood of each bin which gets hit within the delay line, so the probability density function (PDF) can be applied to characterize the bin width in comparison to the other bins. The width of the ith bin can be seen in (1), where τ_i is the width of the ith bin, T_{clk} is the clock period, N_i is the number of hits in the ith bin and N_{total} is the total number of counts.

The transfer function of the delay line can then be formed by determining the cumulative distribution function (CDF) of the calculated bin widths [see (2)]. The transfer function is used as a lookup table (LUT) to correct the code generated by the converter.

To be able to statistically analyze the bin width distribution of the delay line, there needs to be a sufficient number of random triggers to provide a high confidence level. As was described in [21], the required number of hits can be formulated as in (3), where B is the number of bits needed to represent the code, $z_{\alpha/2}$ is the area under the normal distribution, and β is the tolerance level.

For instance, if the tolerance level required is 10% ($\beta = 0.10$), the confidence level is 97% ($\alpha = (1 - 0.97) = 0.03$) and there are 10 bits of resolution, then 481 760 histogram hits are required

$$\tau_i = \frac{T_{\text{clk}} \times N_i}{N_{total}} \tag{1}$$

$$H(n) = \frac{\tau_n}{2} + \sum_{i=1}^{n-1} \tau_i \tag{2}$$

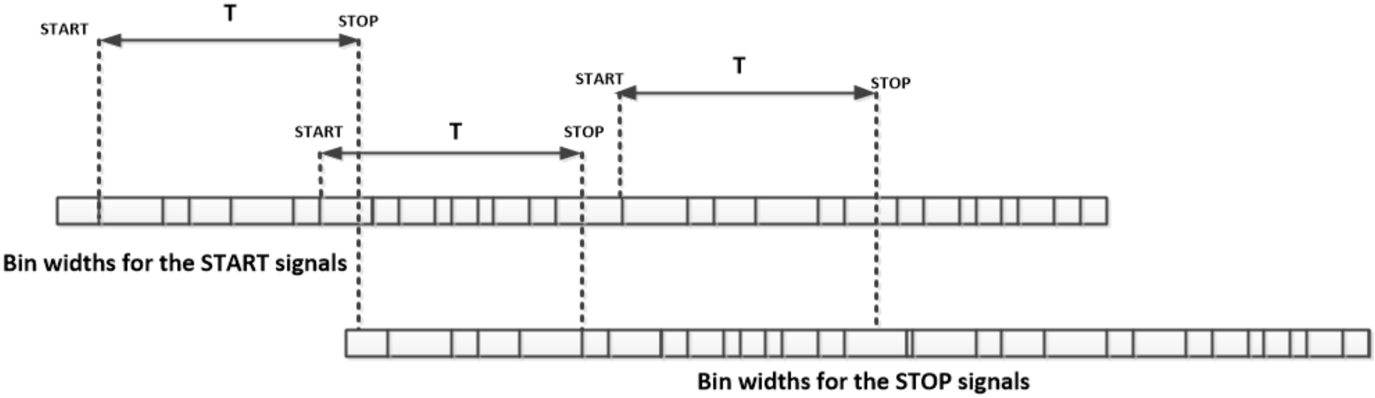

START
T
STOP
START
T
STOP
START
T
STOP
Bin widths for the START signals
Bin widths for the STOP signals

Fig. 1. Illustration of the sliding-scale method.

$$M = \left(\frac{z_{\alpha/2}}{\beta}\right)^2 \times (2^B - 1). \quad (3)$$

B. Direct Calibration

Alternatively, a TDC can also be calibrated directly using an adjustable delay. With this method, the width of each bin can be characterized manually by adjusting the delay on the input.

The direct calibration process starts by setting the input delay to a value which will result in the generation of the least significant bit (LSB) code of the converter. The input delay is gradually increased and by observing the change in each digit of the code, each bin's width in the delay line can be characterized. It should be noted that this method is only usable if the delay generator allows the user to adjust delays smaller than the time value represented by the LSB of the code. This method can be very exhaustive, especially for large carry chains. Thus, it cannot be considered an ideal calibration method for a typical picosecond resolution TDC. The variable clock generation methods described in [20] and [22] are an example of direct calibration.

C. Double Registration

An alternative approach to compensate for the effects of the nonlinearity is the double-registration approach, which is a multi-hit technique used for reducing the DNL error affecting the bins [23] by averaging after registering the codes for the same input trigger twice. In this method, the length of the delay line is chosen to be longer than the clock period. Thus, a logic transition for the trigger can be recorded twice, once on each of the two consecutive clock edges. The average of these codes is utilized as the final fine code which has improved linearity. If the registered logic transitions for the input are K_1, K_2 and the clock period is T_{clk}, the LSB bin width for the bin can be formulated as $T_{\text{clk}}/(K_1 - K_2)$. This method provides fast runtime but does not provide bin-by-bin calibration.

D. Sliding-Scale Technique

Another TDC linearization method used is called the sliding-scale technique. The sliding-scale technique is a simple method to average the bin width through the delay line to improve the linearity. In order to find the average bin widths, signals that are asynchronous to the reference signal are used as stop signals to generate codes [24].

This method aims to generate codes for the same asynchronous pulse in different regions of the delay line and then the average of the generated codes is taken in order to find the average bin widths to improve the linearity.

A few random delay periods are added to the pulses, and different codes are generated for the same signal. Once different codes are generated, the delays added to the signals are subtracted from the codes. Thus, the same signal is represented by multiple different codes. It should be noted that the time difference between the start and stop signals is kept unchanged and separate delay lines are employed for the start and stop signals. After the added delays are subtracted from the code, the same pulse becomes represented by different codes. Finally, the codes are averaged and an average bin width value is calculated. The diagram for this implementation can be seen in Fig. 1.

The main advantage of this method is an improvement in linearity without measuring each bin width inside the delay line, which is significantly faster in terms of the runtime. However, the implementation requires two signal interpolations, start and stop, and it will introduce a large quantization error. Even though this method improves the linearity, it does not achieve perfect linearity since every bin is not characterized and the linearity depends on a randomly chosen bin range and randomly delayed signals. With this method, 0.04-LSB DNL was achieved at 17-ps rms precision in [24].

Depending on the implementation of the differing delays, architectural changes may or may not be required. If the start signal is synchronous to the TDC system clock, then the start time is known precisely and time-scrambling circuitry is required on the stop signal. If the start signal is asynchronous to the TDC system clock, then merely measuring multiple times will introduce the required random delay. If the start signal is of a higher frequency than the system clock or the scrambler logic can produce two pulses less than a clock period apart, additional decoder circuitry is required to read multiple starts/stops from the delay lines. Markovic *et al.* [24] ops for an asynchronous start signal (explicitly stated) slower

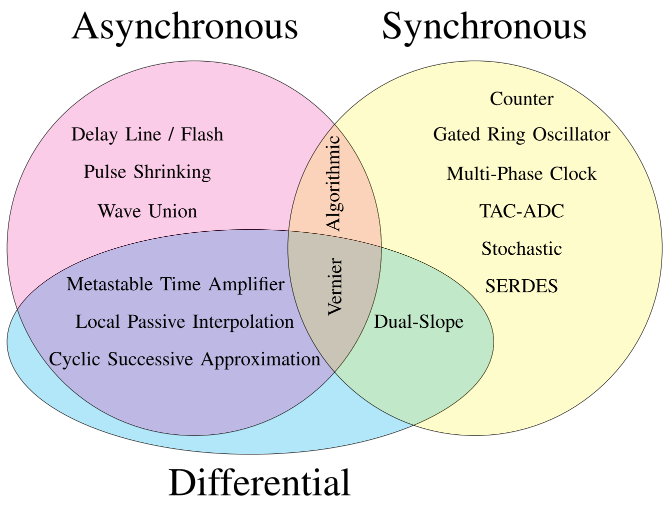


Fig. 2. Venn diagram of some of the classifications of TDCs.

than the system clock (implied by the use of a looped Vernier delay line) and so does not need extra circuitry.

III. TDC Architectures

A. Introduction

In this section, we review new TDC architectures that have not appeared in previous literature reviews. For completeness, a Venn diagram showing the TDC architectures described in this article and other review literature, classified by three categories (synchronous, asynchronous, and differential), is shown in Fig. 2. For the architectures not mentioned in this article, the reader is advised to read previous review literature [16].

As a brief overview, the asynchronous methods use asynchronous logic to generate small delays based on the width of a logic element, whereas the synchronous methods operate relative to an oscillator (clock). The differential methods take the difference between two measurements to produce a finer measurement. Some examples would be the delay line, which sequences the smallest delay elements and counts the number that transition asynchronously, the multi-phase clock, which takes multiple different phases of an oscillator and compares them all to sub-divide the oscillator, and the Vernier method, which uses the difference between two logic elements (asynchronous) or two oscillators (synchronous) to obtain finer time resolution.

B. Successive Approximation TDC

Successive Approximation TDCs (SA-TDCs), named due to their derivation from SA analog-to-digital converters (ADCs), operate by delaying the start and stop signals using a variable delay line and comparing the signal that exits first, then routing the signal to propagate through a shorter delay line.

Simple linear SA TDCs are relatively straightforward to implement, as each stage of the approximation is a fixed delay line with the propagation delay being half that of the previous stage. However, they subsequently suffer in terms of large area utilization and poor matching over the course of the TDC due to local process variations. The delay can be stabilized by applying bias voltages derived from a delay-locked loop (DLL) that divides the previous stage's time by two, but this adds extra complexity, and therefore cost, to the circuit.

Therefore, cyclic SA-TDCs (CSA-TDCs) have been proposed where the signal is routed repeatedly through the same delay element—a digital-to-time converter (DTC), which has its delay repeatedly halved as the start and stop signals converge. As the same delay elements are being used each time, the effect of local process variation is much smaller, and for the same range and precision, only half the delay elements are needed, according to (4)–(6). This means that the full-scale range of the CSA TDC (without the addition of another level of TDC) is twice the length of the largest generatable delay $(\sum_{i=0}^{\infty}(1/2^i) = 2)$

$$N = \frac{T}{T_{\min}} \tag{4}$$

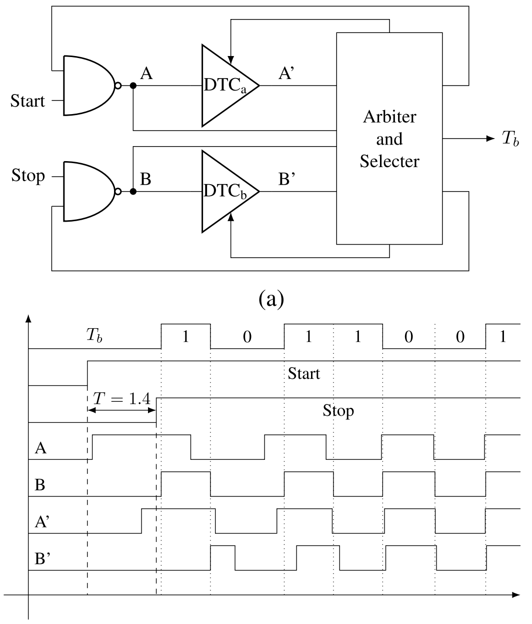
$$\begin{aligned} A_{SA} &= A_{\min}\left(\frac{N}{2} + \frac{N}{4} + \cdots + \frac{N}{N} + \frac{N}{N}\right) \\ &\quad + B\left(2A_{\text{mux}} + A_{\text{arb}}\right) \\ &= A_{\min}\left(\frac{N}{N = 2^B} + \sum_{i=1}^{B}\left(\frac{N}{2^i}\right)\right) \\ &\quad + B(2A_{\text{mux}} + A_{\text{arb}}) \\ &= N * A_{\min} + B(2A_{\text{mux}} + A_{\text{arb}}) \end{aligned} \tag{5}$$

$$\begin{aligned} A_{CSA} &= A_{\text{DTC}} + A_{\text{arb}} + A_{\text{sel}} \\ &= \left(\frac{N * A_{\min}}{2} + (B-1)A_{\text{mux}}\right) \\ &\quad + A_{\text{arb}} + (2A_{\text{mux}} + 2A_{\text{OR}}). \end{aligned} \tag{6}$$

In particular, (4) describes the number of elements N in the DTC as a function of the full-scale range T and the minimum time-resolution $T_{\min}$. For simplicity, only a homogeneous DTC (only one type of delay element) is considered in these equations. Equation (5) describes the area of a linear SA-TDC in terms of the area of a minimum-sized delay element $A_{\min}$, N as defined in (4), the number of bits of resolution B, the area of a multiplexer A_{mux}, and the area of an arbitrator (flip–flop, set-reset (SR) latch, current-sense amplifier, etc.) A_{arb}. It can be seen that the result includes roughly the same number of elements as a simple delay-line TDC, so the linear SA-TDC only benefits when nonhomogeneous elements are used (e.g., by adjusting load through capacitors).

Equation (6) shows the area of a CSA-TDC in terms of the area of a DTC A_{DTC}, the area of an arbitrator A_{arb} and the area of a selector A_{sel}. The DTC only needs to cover half the full-scale range T of the TDC so only needs half the delay elements $(N/2)$, and then must use $B - 1$ multiplexers to enable or disable sections of the DTC. The selector can be composed of two multiplexers and two OR gates.

Fig. 3 shows the operation of a CSA TDC. The "Arbiter and Selector" block shown in Fig. 3(a) chooses to forward either A' and B or A' and B' to its outputs depending on whether A' or B arrives first. If A' arrives before B, this implies that the delay between A and B is longer than the period of DTC$_a$ (DTC "a"), therefore it forward A' and B to reduce this delay and then



(a)

(b)

Fig. 3. (a) Block diagram and (b) waveform of a CSA TDC.

halves the delay for the next cycle. If B arrives before A', then the delay between A and B does not exceed DTC$_a$, therefore it forward A' and B', which have the same delay difference as A and B, so that the delay is maintained for the next round where the delay is once again halved. If A' arrives before B, a "1" is output on T_b when B arrives, otherwise a "0" is output on T_b when B arrives. The A signal can be used to allow double-ended operation (stop before start) or to clear the state of the arbiter for the next round. The TDC here outputs the code "1011001," which is a fractional number with an MSB of 1, resulting in $1 + 0.25 + 0.125 + 0.015625 = 1.390625$, which is the closest number below 1.4.

It can be seen that the hierarchical TDC [25] bears a remarkable resemblance to the SA TDC, and in fact can be considered a less efficient form of the SA TDC, as the next stage is triggered twice, whereas the SA TDC only triggers the next stage once. Also, as the hierarchical TDC does not choose between the delayed and nondelayed versions of the signals, some conditional bit flipping is needed on the outputs which are not needed in the SA TDC.

The asynchronous pipelined TDC demonstrated in [26] is also a form of SA-TDC. However, since the aim is to quantify pulse time (rising edge to falling edge), the residue is formed by finding the dead time or overlap between the signal and the delayed version of the same signal. If the signal overlaps its delayed version, then the period of the signal is longer than the delay; otherwise, it is shorter. The residue is the quantity by which it overlaps or misses the delayed version, and is then quantized by an exponentially shorter delay. Akgun [26] achieved a resolution of 200 ps due to the dead zone of the residue generator, which was shown to be 189.7 ps.

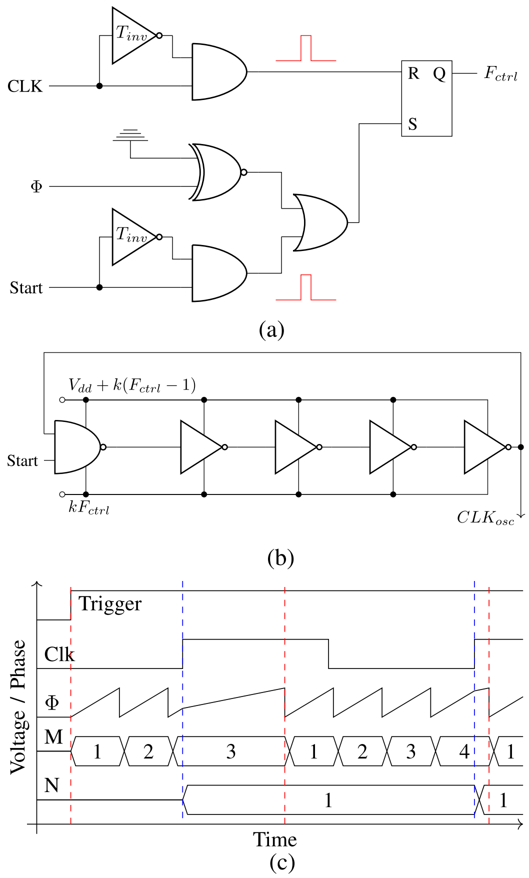
Similar to LPI TDCs (see [16]), SA TDCs are relatively new in the current literature, with more than half of the articles published by Mantyniemi *et al.* [27]–[29]. These articles show a TDC operating at up to 610-fs resolution with a 5-ns range (or 1.21-ps resolution with a 328-us range) using a switched capacitor array (and a VCO) for the 328-us range) for the DTC, which is explained in more detail in [30]. Building on this, Chung *et al.* [31] propose unrolling the SA loop in order to increase the sample rates. Using 65-nm CMOS (compared with 350 nm in [29]), 80 MS/s was achieved, compared to 5 MS/s in [29]. However, due to the inferior switched-capacitor implementation, Chung *et al.* [31] only managed a 9.77-ps resolution. Jiang *et al.* [32] present a technique one might call a linear SA TDC, which linearly increases the delay until the two signals align. However, as it has no signal recovery, duplication, or residue, it requires multiple samples of the input signal before it can detect the correct time difference, which is not tolerable in many applications. Also, this system exhibits a 474-ps resolution, which is well behind even delay line implementations available in 2016 (17 ps in [32]). References [33] and [34] also present a linear (unrolled) implementation of the SA-TDC, achieving 25 and 12.5 ps, respectively, on 180 nm.

C. Algorithmic TDC

Algorithmic TDCs, first proposed by Keranen and Kostamovaara [35], [36], are functionally similar to CSA TDCs (see Section III-B). However, at each stage of the SA, instead of looping the residue back round and reducing the delay exponentially, it amplifies the residue exponentially and quantizes it with the same delays.

Keranen's article [35] uses a scheme similar to a dual-slope TAC, but instead of increasing and decreasing the amplitude, it increases the phase of a ring oscillator at two different speeds. The number of "fast" oscillations between the start and stop (system clock edge) are counted, and then the oscillator is switched to its "slow" mode. The time taken for the oscillator to reach a full oscillation (phase is zero) will be dependent on the quantization residue of the counter process, and will be amplified by a ratio of $(F_{\text{fast}}/F_{\text{slow}})$, where F_{fast} is the frequency in 'fast' mode, and F_{slow} is the frequency in "slow" mode.

Fig. 4(c) shows the wave trace for an algorithmic TDC. The ring oscillator starts oscillating at a high frequency (e.g., $F_{\text{fast}} = 6 \times F_{\text{clk}}$) when the trigger signal transitions from 0 to 1 (first dashed line, red) and oscillates until the next clock edge (second dashed line, blue). At this point, the value of the M counter (which counts ring oscillator periods) is sampled to produce the first residue. Then, the ring oscillator is switched to a low frequency (e.g., $F_{\text{slow}} = 2 \times F_{\text{clk}}$) and the N counter (which counts whole clock periods) is started. This runs until the ring oscillator wraps around to $\Phi = 0$ at which point the N counter is sampled to produce the next residue and the counter is set back to the high frequency. At each stage, the algorithmic TDC is amplifying and quantizing the residue from the previous stage through the change of the ring oscillator's frequency. If F_{slow} were to be slower than F_{clk}, it would be possible to reach $N > 1$ and also increase the amplification at each stage, at the expense of longer conversion times.



(a)

(b)

(c)

Outputs (M, N, M, N, ...) are 3, 1, 4, 1, ...

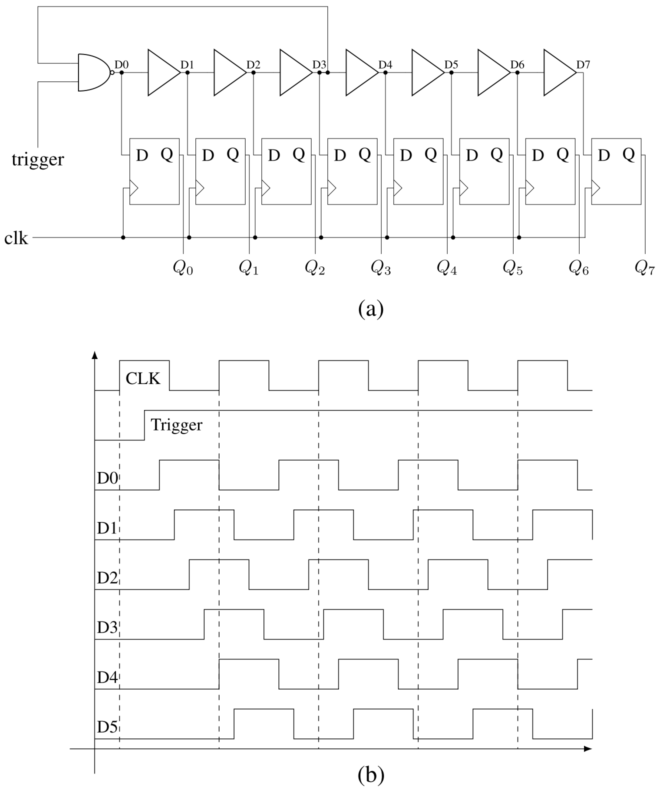
Fig. 4. Algorithmic TDC. (a) Frequency control. (b) Ring oscillator. (c) Wave trace for a sample conversion based on [35] (residue = 38% of a clock period).

In [36], a second oscillator is started in fast mode while the first is in slow mode to quantize the amplified residue, and then the first oscillator is used to quantify the second amplified residue. On the other hand, in [35], the system clock is used to quantify the amplified residue, and then a further residue is generated from the time between the oscillator reaching zero phase and the next system clock edge, which is quantified using the same method as the original pulse (using the oscillator in fast mode to quantify, then switching to slow mode to amplify).

D. Wave Union Launchers

Wu and Shi [23] propose a method of improving precision past the gate delay: the wave union TDC. Rather than dispatching one edge per trigger and quantizing this edge, they suggest dispatching multiple edges and quantizing all of them by a method similar to the GRO, but without the need for more than one input sample.

The authors suggest two methods for doing this. The first (type A) is to store a wavelet inside a delay line and release it on the incidence of a trigger. When the stop signal occurs, the wavelet is held in place and quantized. Each edge in the wavelet is individually quantized and the edges are then combined to give a more accurate measurement of the original trigger position. This is referred to as a finite-step response (FSR) wave union launcher.



(a)

(b)

Fig. 6. (a) Gate-level implementation and (b) waveform of a Type-B wave union launcher.

Fig. 5 shows the design of an FSR wave union TDC. The first M bins are used to store the FSR pulse and for quantization, with the remaining N bins being used solely for quantization. In [23], M was 16 bins (with the distance between edges in the FSR being 13 bins), while N was 48 (i.e., the delay line was 3 times the length of the FSR storage).

The second method (type B) is to attach a startable ring oscillator to the front of the delay line. The trigger signal starts the ring oscillator, which then oscillates for a number of cycles before stopping. This is referred to as an infinite-step response (ISR). The oscillations occur over multiple system clock (stop) cycles.

In the type B wave union TDC, as shown in Fig. 6(a) and (b), the period of the oscillator must not be too similar to the period of the system clock so that the sampling process does not repeatedly hit the same large bin (as this would result in a large DNL). However, this means that there will be cases where ring oscillator edges will not be seen once. Wu and Shi [23] identify three possible cases, namely, U, V, and W patterns, corresponding to jumps of 0, 1, and 2 ring oscillator periods. The jump type is determined from the output values of the priority encoder.

1) For a value in the range $3N/4 \rightarrow N$ followed by a value in the range $N/4 \rightarrow N/2$, this implies both signals were the same ring oscillator edge (based on the operation of the priority encoder) and hence is a case of the U pattern.
2) For a value in the range $N/4 \rightarrow N/2$ followed by a value in the range $3N/4 \rightarrow N$, this implies that a ring oscillator edge has been missed, and hence is a case of

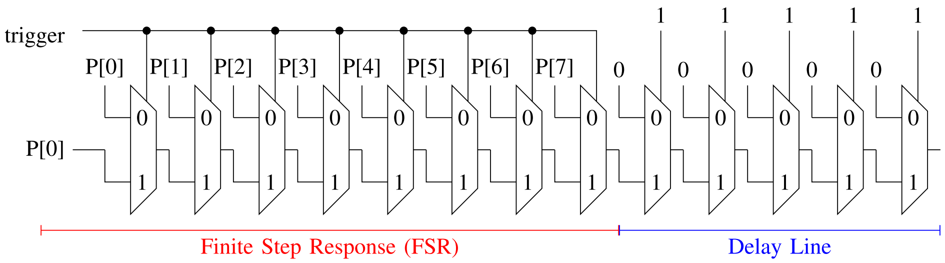


Fig. 5. Initial section of a wave union launcher using multiplexers (the delay line continues further).

TABLE I
PERFORMANCE OF THE WAVE UNION TDCs IN [23]

Method	Mean Bin	Worst Bin	RMS Error	Dead Time
Delay line	60 ps	165 ps	40 ps	2.5 ns
Type A (FSR)	30 ps	65 ps	25 ps	5 ns
Type B (ISR)	–	–	10 ps	45 ns

the W pattern. This will only happen if the ring oscillator is faster than the clock period (meaning an edge can pass between two samples).

3) All other jumps are classified as V patterns, and are indicative of the standard operation of the TDC.

The first method is able to increase the accuracy quite significantly, from 165 ps per bin worst case and 60 ps per bin average case (in the original TDC), to 65 ps per bin worst case and 30 ps per bin best case. It does this without significantly increasing the dead time (2.5–5 ns), but does increase the decoding complexity due to increasing the number of edges to be decoded per output (although this was performed on a computer in the original article). The second method was measured through the rms error of measuring a fixed time difference, and resulted in an improvement from 40 to 10 ps for 16 measurements (in comparison to 25-ps rms for the FSR method), albeit at 18 times dead time increase (2.5–45 ns). This is summarized in Table I.

Subsequently, Bayer and Traxler [37] used wave union TDC and managed a 1.8 times improvement on the bins inside their Virtex 4 FPGA from 16- to 9-ps rms.

Hu *et al.* [38] suggest a stepped-up tree encoder (SUTE) to efficiently encode the edges on a Virtex 4 FPGA in the presence of bubbles and the nonthermometer code presented by the type A wave union TDC. The encoder uses a preprocessing stage capable of removing single-bit bubbles (e.g., 0000 1011 1111) which encodes the position of the 0 → 1 edge in subgroups of 4 bits, plus a flag to determine if the transition occurs in that subgroup and a flag to determine if a transition happens on the border of subgroups.

The four-wide grouping suppresses the single-bit bubbles and hence allows the resultant outputs to be sent to an array of standard priority encoders for encoding via some switching multiplexers which distribute the edges to the encoders. This ensures that multiple edges can be encoded in a single-clock cycle, and the maximum number of edges is determined by the number of terms in the FSR (wavelet generator).

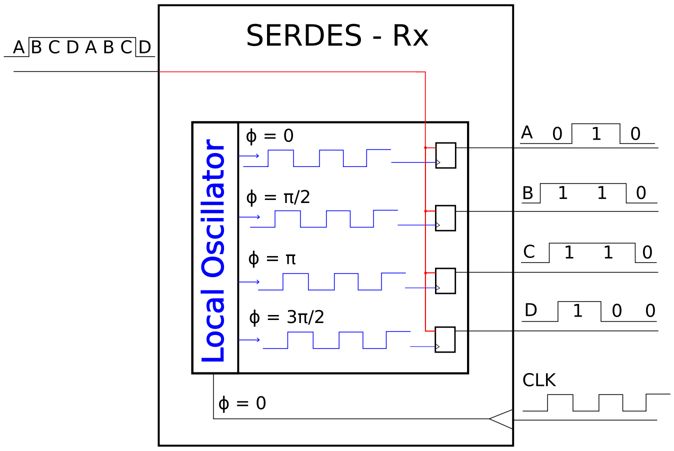


Fig. 7. SERDES TDC with 4x interpolation.

E. SERDES TDC

When operating on an FPGA, serializer–deserializer (SERDES) blocks can be used to create uniform delay elements to form high-resolution fine-time interpolation TDCs. SERDES are generally used in high-speed communication applications where the input–output (I/O) channels are limited. The transmitter's parallel input is serialized using a high-speed clock. At the receiver, the data are deserialized to the original parallel format. In other words, SERDES blocks have lower data rates at the input, they conduct the transmission at a faster clock frequency and have the lower data rate again at the output. As a result, the I/O required for the transmission is minimized and no data is lost during transmission due to the faster data rate. This can be seen in Fig. 7, where 4x interpolation is provided by the SERDES TDC with the four phases appearing on the output lines A through to D.

Modern FPGAs offer SERDES blocks which can provide 10 times clock multiplication and so 10 times faster serialization. Since SERDES blocks are uniform chains of shift registers which are synchronized with a high-speed clock throughout the transmission, they provide high-resolution fine-time quantization. As described in [39], a SERDES-based 96-channel TDC was implemented on two Altera Stratix EP1S30F780C6 FPGAs which achieved a 1.2-ns resolution.

F. DSP Delay Line

Tancock *et al.* [40] published an article on implementing the delay lines in DSP blocks on an FPGA. Conventional TDCs on FPGAs utilize LUTs or carry chains built into the

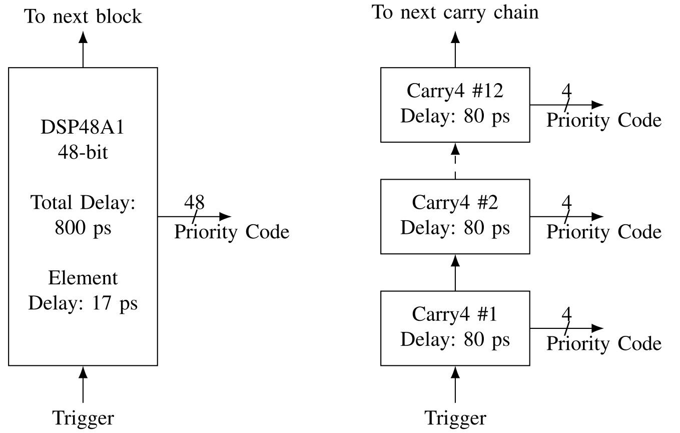


Fig. 8. DSP delay line compared to a carry chain.

FPGA fabric to generate small delays. However, as this logic is configurable, the fan-out and ON-resistance of the elements is much higher than dedicated logic, resulting in suboptimal results. The on-chip DSP blocks (DSP48A1) contain dedicated carry logic for the 48-bit adder, so it was suggested that this logic could be used to generate more optimal delays. While the total delay of a DSP block was less than the equivalent number of carry elements in the general-purpose fabric (Fig. 8), the elements were severely out-of-order and the majority of the delay fell into a single large bin at the boundary of DSP blocks.

Subsequently, Tancock and Dahnoun [41] published a follow-up article detailing the use of a population counter and starting offsets to compensate for the nature of the DSP blocks. The population counter relies on the monotonic increase of the number of bins in the "1" state (from the "0" state) to effectively reorder the bins into a linearly increasing sequence. Then, the initial offsets on four parallel delay lines allow each DSP block to cover for a third of the large bin present in each of the others, with the codes being summed to produce the final output code. On the newer Artix-7 architecture (DSP48E1 blocks), the authors achieved 5.25-ps resolution, compared to approximately 20 ps that would be achieved with carry chains or 10 ps for four parallel carry chains.

IV. Performance

Discussion of various architectures' merits and demerits is, unfortunately, insufficient to make a decision on which architecture is best for a particular use case. Hence, this section analyses a portion of TDCs available in current literature, showing the achieved performance in each case as well as the architecture and process technology used. Table II displays key performance metrics from a number of recent (past 20 years) articles on TDC. The table is sorted by resolution, as this is arguably the most important metric for a TDC, but it also includes information on process technology, INL and DNL, single-shot precision (SSP), number of channels, and the architecture used.

For many implementations, such as the wave union method, the only appropriate measure of resolution is the SSP, so for these articles, the resolution and SSP are equivalent (SSP is 1 LSB). For other articles, there is a distinct difference between the resolution (which defines the quantization noise floor, and is the minimal discernible difference between two values) and SSP (which is affected by other aspects such as clock jitter, voltage variation and temperature variation, and is the standard deviation of measurements of a single time pulse).

In many cases, the SSP is not measured or stated, whereas some measure of resolution is always available, hence the choice of resolution as the main figure of merit. The resolution will always be a lower bound for precision, but the precision may be higher for the aforementioned reasons.

While a high internal DNL and INL are beneficial to multiple hit or multiple registration methods (such as the wave union method), a high DNL or INL at the output (the value which is stated in Table II) implies a loss of precision even after applying these methods.

Not all the information can be expressed in such a table, and so exceptions worthy of note are as follows: [51], [61], and [69] are the only articles in the table that do not use CMOS, using an unspecified FPGA, a 0.8 μm BiCMOS process and an unspecified ECL process respectively. Richardson *et al.* [65], Veerappan *et al.* [67], and Niclass *et al.* [71] are unique in that they also integrate arrays of single-photon APD (SPAD) pixels on the same chip, with either a one-to-one or one-to-many matching between TDC and SPAD, thereby incorporating an entire depth-mapping system onto a single chip.

References [53], [58], and [66] are also worth mentioning, as they use multi-dimensional schemes to reduce the area requirements. The former arranges its pulse-shrinking elements into a 2-D grid, with row and column decoders being used to ascertain the position at which the pulse was extinguished, and the latter two employ a scheme of splitting their delays into a sequential set (e.g., $1, 2, \ldots, 8$) which operates on each column, and a sparse set (e.g., $1, 9, 17, \ldots$) which operates on the rows, with the arbiters comparing the two delayed signals to each other, as opposed to a delayed signal against an undelayed signal. Finally, the use of switched capacitor arrays in [27] and [29], and resistive dividers in [46] and [47] show how analog components can be used to great effect while still exhibiting technology scaling improvements.

V. Architecture Comparison

Table II demonstrates that most TDCs make a tradeoff between conversion rate, resolution, and range. Stochastic and metastable time amplifier systems are both similar in that they sacrifice their range to increase their resolution and conversion rate. This makes them excellent for systems that have events happening very quickly (such as short-range ToF) or close to a known reference (such as frequency synthesis), but makes them bad for applications that require a large dynamic range, such as long-range ToF systems. However, due to their high count rate and resolution, they are also useful as the lowest level in a multi-level system, where they provide the LSBs while leaving the more significant bits to other methods.

Vernier, pulse-shrinking, dual-slope time amplifier and, to a lesser extent, SA and algorithmic systems instead decide to sacrifice conversion rates and area for better range and resolution. This is excellent for systems with a low repetition

TABLE II

COMPARISON OF TDCS IN THE REVIEWED LITERATURE. N/A = NOT AVAILABLE. * CMOS ASICS IN NANOMETERS AND FPGAS BY SERIES NUMBER

Method	Technology*	Resolution (ps)	DNL (LSB)	INL (LSB)	SSP (LSB)	Channels	Ref
Algo TDC	350	0.61	±0.4	±4.5	±1.2	1	[36]
CSA-TDC + Counter	350	0.61	N/A	N/A	N/A	1	[29]
STDC	130	0.7	±1.4	±2.4	N/A	1	[42]
Branching CRO	65	0.85	±0.27	±2.94	1	1	[43]
TAC-ADC	N/A	1	N/A	±10	1	1	[44]
CSA-TDC + Counter	350	1.2	N/A	±6.67	3	1	[27]
DL + TA + DL	90	1.25	±0.8	±3	1	1	[45]
Algo TDC	350	2	N/A	±1.25	±0.15	1	[35]
Looped LPI-TDC	90	4.7	±0.6	±1.2	0.7	1	[46]
Looped LPI-TDC	90	4.7	±0.5	±1.0	N/A	1	[47]
Wave Union A	FPGA	6	N/A	N/A	1	48	[37]
GRO	130	6	N/A	N/A	N/A	1	[48]
VDL + DL + Counter	180	10	N/A	N/A	N/A	1	[49]
Wave Union B	FPGA	10	N/A	N/A	1	8	[23]
TAC-ADC + Counter	ECL	10	N/A	±2	1.5	1	[50]
TA + DL	180	11.25	N/A	N/A	1.33	1	[51]
Flash (2D) + CRO + Counter	350	12.2	N/A	±0.41	0.66	1	[52]
Vernier SA-TDC	180	12.5	±0.4	±0.4	N/A	1	[34]
Looped ps	800	20	±0.5	N/A	1	1	[53]
TA + SA-TDC	90	20	N/A	N/A	N/A	8	[54]
DDL + Counter	90	21	±0.7	±0.7	N/A	1	[55]
VDL + CRO + Counter	350	24	±0.55	±1.5	N/A	1	[56]
Flash (2D) + CRO + Counter	600	30	N/A	±1.33	32	1	[57]
VDL	700	30	N/A	±1.0	0.2	1	[58]
Hierarchical TDC	90	31.25	N/A	N/A	N/A	1	[59]
Dual-Slope TAC + Counter	800	32	N/A	±0.16	0.94	1	[60]
DL + Counter	130	40	N/A	N/A	N/A	1	[61]
CSA-TDC + CRO + Counter	350	42	N/A	N/A	N/A	1	[62]
GRO	130	45	N/A	N/A	N/A	1	[63]
CRO + Counter	130	50	±0.5	±2.4	N/A	1024	[64]
PS (2D Array)	800	50	N/A	±3.5	N/A	1	[65]
CRO + Counter	130	55	±0.3	±2.5	N/A	20480	[66]
Wave Union A	FPGA	60	+1.17	±1.08	0.42	1	[23]
CRO + Counter	180	61	±0.23	±0.3	1	24	[67]
VDL + DL + Counter	FPGA	75	N/A	N/A	0.53	1	[68]
CRO	65	80	N/A	N/A	N/A	1	[69]
DDL + CRO + Counter	350	97	±0.09	±1.89	N/A	32	[70]
TAC-ADC + Counter	500	312.5	±0.2	±0.3	0.32	1	[71]
CRO	800	530	±0.36	±0.36	N/A	1	[72]
SERDES, 2 chips x 48 channels	FPGA	1200	±0.167	N/A	±0.5	48x2	[39]

rate or where the repetition rate can be controlled (such as long-range ToF), but suffers when a high repetition rate (such as short-range ToF or quantum key distribution) is required, as often the only option is to employ an interpolation or pipelining scheme which can massively hurt area efficiency.

TABLE III

MERITS AND DEMERITS OF THE TDCS REVIEWED HERE

Method	Merits	Demerits
Successive Approximation	• Area-efficient for an asynchronous design. • Variable delay architecture.	• Asynchronous path is difficult to design. • ASIC only.
Algorithmic	• Very small area. • High resolution.	• Long dead time (multiple clock cycles).
Wave Union	• Low area penalty for up to 1 order of magnitude improvement.	• Design of decoder is difficult and increases area utilisation.
SERDES	• Order of magnitude increase without using general FPGA fabric. • Easy to use.	• Uses high-speed communications resources, which may be needed elsewhere.
DSP Delay Line	• Higher resolution than carry chains with few changes. • Can be used alongside carry chains for higher channel numbers.	• Not as many DSP resources as carry chains. • Need multiple cascaded to account for DNL. • Requires population counter.

Delay line, controlled ring oscillator, and GRO methods avoid low count rates and low range, but suffer in terms of resolution, meaning that they are often a good choice either as a mid or upper level of a multi-level TDC, or in the case where the application does not require high precision, such as for low-rate frequency synthesis, long-distance low-resolution ToF, and quantum key distribution with low dead-count rates. The GRO method can also suffer from some signal integrity issues in low count rate systems but excels when measuring the same time period multiple times over due to its first-order noise shaping.

TAC-ADCs and LPI methods perform very well in all three areas but suffer from a lack of technology scaling in the TAC-ADC's case as well as signal integrity and area efficiency problems. If an LPI TDC were extended to its logical extreme with a large number of resistors in its potential divider, it could probably achieve much higher resolutions than shown in Table II, although the area utilization would increase exponentially due to the number of resistors needed.

Flash and Wave Union TDCs are highly configurable, with high resolutions and ranges available at the expense of area and conversion time, making them excellent as a mid or low-level section of a system, although they do not achieve the same resolutions as stochastic, time amplification, Vernier, or LPI methods.

Table III shows a summary of the merits and demerits of the new architectures described in this article.

VI. CONCLUSION

In conclusion, time-to-digital conversion has developed many approaches over the past two decades, each of which has its own unique characteristics. When measuring the same signal many times over, such as with frequency locking applications or high-resolution low-speed surface mapping, it is worth employing a GRO for its first-order (and higher order when multiple are used together) noise shaping. If the conversion rate of the target system is of little concern, then, depending on area constraints, utilizing looped Vernier or pulse-shrinking methods is advised since the analog components in dual-slope systems do not scale well with technology.

Beyond this, the systems that have been shown to perform best are multi-level systems that exploit the benefits of multiple architectures while covering the weaknesses with the others. Most notably, SA-TDCs are delay-element agnostic, which means that they can incorporate a large variety of delay generation methods to assist in obtaining the correct range and resolution, and, at each stage, also output the residue—the difference between the measured and actual value of the time difference, which can then be passed directly to a higher resolution TDC. If a small-enough residue can be obtained at the final output of the SA-TDC, we suggest utilizing either a stochastic or metastable time-amplification method be used to obtain optimal accuracy. Similarly, if a CSA-TDC were employed, a time amplifier could be switched in as the system approaches the lowest bits to amplify the time difference and hence reduce the requirements for small delays.

Methods that use traditional analog components in a way that allows them to scale, such as LPI and switched capacitor arrays (as seen in the CSA-TDC articles) allow for very small time differences typically not seen in digital methods. However, these systems require a careful choice of components and layout to minimize nonlinearity and interference from other components and often require large areas of the layout (although not as much as other analog methods). Hence, they are worth considering if the expertise and design constraints allow for such an approach.

When constrained to FPGAs for the underlying technology, the only options often available are delay line, stochastic, and wave union TDCs. For resolutions of as high as 10 ps, this can be achieved with delay lines (sometimes requiring averaging between many) on their own. If the required range is less than 100 ps, stochastic TDCs may be an option depending on the device size and metastability window of the discriminators (normally D-type flip-flops or latches). Otherwise, wave union TDCs are the only option, and care must be taken to optimize the design of the encoder to reduce its area requirements. If a low count rate is acceptable (relative to the system clock), a type B wave union TDC will be optimal; otherwise, a type A will be required.

REFERENCES

[1] E. Arabul, J. Rarity, and N. Dahnoun, "FPGA based fast integrated real-time multi coincidence counter using a time-to-digital converter," in *Proc. 7th Medit. Conf. Embedded Comput. (MECO)*, Jun. 2018, pp. 1–4.

[2] (Nov. 2013). *Time-of-Flight Fundamentals*. [Online]. Available: https://msr.dom.wustl.edu/time-of-flight-fundamentals/

[3] K. Naresh, "Applications of fluorescence spectroscopy," *J. Chem. Pharmaceutical Sci.*, vol. 2014, no. 5, pp. 18–21, 2014. [Online]. Available: https://www.jchps.com/specialissues/Special%20issue5/05%20jchps%20si5%20k%20naresh%2018-21.pdf

[4] M. Wahl. (2014). *Time-Correlated Single Photon Counting*. [Online]. Available: https://www.picoquant.com/images/uploads/page/files/7253/technote tcspc.pdf

[5] P. Vines *et al.*, "High performance planar germanium-on-silicon single-photon avalanche diode detectors," *Nature Commun.*, vol. 10, no. 1, p. 1086, 2019.

[6] OSI Optoelectronics. (2013). *1.25 Gbps Silicon Photodiodes*. [Online]. Available: http://osioptoelectronics.com/standard-products/silicon-photodiodes/high-speed-silicon-photodiodes/1-25gbps-photodiodes.aspx

[7] D. Parker, R. Forster, P. Fowles, and P. Takhar, "Positron emission particle tracking using the new Birmingham positron camera," *Nucl. Instrum. Methods Phys. Res. Sect. A, Accel., Spectrometers, Detectors Associated Equip.*, vol. 477, nos. 1–3, pp. 540–545, 2002. [Online]. Available: http://www.sciencedirect. com/science/article/pii/S0168900201019192

[8] *TDC7201 (ACTIVE)*. Accessed: Apr. 26, 2019. [Online]. Available: http://www.ti.com/product/TDC7201

[9] J. Smith. *Time Tagger Series*. Accessed: Apr. 26, 2019. [Online]. Available: https://www.swabianinstruments.com/time-tagger/

[10] Picoquant. *Home*. Accessed: Apr. 26, 2019. https://www.picoquant.com/products/category/tcspc-and-time-tagging-modules

[11] *ID900 Time Controller*. Accessed: Apr. 26, 2019. [Online]. Available: https://www.idquantique.com/single-photon-systems/products/id900-time-controller/

[12] D. I. Porat, "Review of sub-nanosecond time-interval measurements," *IEEE Trans. Nucl. Sci.*, vol. 20, no. 5, pp. 36–51, Oct. 1973.

[13] J. Kalisz, "Review of methods for time interval measurements with picosecond resolution," *Metrologia*, vol. 41, no. 1, p. 17, 2004.

[14] 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.

[15] P. Napolitano, A. Moschitta, and P. Carbone, "A survey on time interval measurement techniques and testing methods," in *Proc. IEEE Instrum. Meas. Technol. Conf. (I2MTC)*, May 2010, pp. 181–186.

[16] S. Henzler, *Time-to-Digital Converters*, vol. 29. Dordrecht, The Netherlands: Springer, 2010.

[17] Z. Wang, C. Huang, and J. Wu, "A review of CMOS time-to-digital converter," *J. Circuits, Syst. Comput.*, vol. 23, no. 07, 2014, Art. no. 1430001. doi: 10.1142/S0218126614300013.

[18] D. Chaberski, R. Frankowski, M. Gurski, and M. Zieliński, "Comparison of interpolators used for time-interval measurement systems based on multiple-tapped delay line," *Metrol. Meas. Syst.*, vol. 24, no. 2, pp. 401–412, 2017.

[19] M. W. Fishburn and E. Charbon, "Time-to-digital converters for PET: An examination of metrology aspects," in *Proc. IEEE Nuclear Sci. Symp. Med. Imag. Conf. Rec. (NSS/MIC)*, Anaheim, CA, USA, Oct./Nov. 2012, pp. 839–840. doi: 10.1109/NSSMIC.2012.6551222.

[20] 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.

[21] M. Mota and J. Christiansen, "A high-resolution time interpolator based on a delay locked loop and an RC delay line," *IEEE J. Solid-State Circuits*, vol. 34, no. 10, pp. 1360–1366, Oct. 1999.

[22] K. Katoh and K. Namba, "A low area calibration technique of TDC using variable clock generator for accurate on-line delay measurement," in *Proc. 16th Int. Symp. Qual. Electron. Design*, Mar. 2015, pp. 430–434.

[23] J. Wu and Z. Shi, "The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay," in *Proc. IEEE Nucl. Sci. Symp. Conf. Rec. (NSS)*, Oct. 2008, pp. 3440–3446.

[24] B. Markovic, S. Tisa, F. A. Villa, A. Tosi, and F. Zappa, "A high-linearity, 17 ps precision time-to-digital converter based on a single-stage Vernier delay loop fine interpolation," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 3, pp. 557–569, Mar. 2013.

[25] A. S. Yousif and J. W. Haslett, "A fine resolution TDC architecture for next generation PET imaging," *IEEE Trans. Nucl. Sci.*, vol. 54, no. 5, pp. 1574–1582, Oct. 2007.

[26] O. C. Akgun, "An asynchronous pipelined time-to-digital converter using time-domain subtraction," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2018, pp. 1–5.

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

[28] S. Al-Ahdab, A. Mäntyniemi, and J. Kostamovaara, "Cyclic time domain successive approximation time-to-digital converter (TDC) with sub-ps-level resolution," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf.*, May 2011, pp. 1–4.

[29] Salim Alahdab, A. Mäntyniemi, and J. Kostamovaara, "A time-to-digital converter (TDC) with a 13-bit cyclic time domain successive approximation interpolator with sub-ps-level resolution using current DAC and differential switch," in *Proc. IEEE 56th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2013, pp. 828–831.

[30] S. Alahdab, A. Mäntyniemi, and J. Kostamovaara, "A 12-bit digital-to-time converter (DTC) with sub-ps-level resolution using current DAC and differential switch for time-to-digital converter (TDC)," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf.*, May 2012, pp. 2668–2671.

[31] H. Chung, H. Ishikuro, and T. Kuroda, "A 10-bit 80-MS/s decision-select successive approximation TDC in 65-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 47, no. 5, pp. 1232–1241, May 2012.

[32] R. Jiang *et al.*, "Successive approximation time-to-digital converter with vernier-level resolution," in *Proc. IEEE 21st Int. Mixed-Signal Test. Workshop (IMSTW)*, Jul. 2016, pp. 1–6.

[33] D. Kościelnik, M. Miśkowicz, J. Szyduczyński, and D. Rzepka, "Optimizing time-to-digital converter architecture for successive approximation time measurements," in *Proc. IEEE Nordic-Medit. Workshop Time-Digit. Converters*, Oct. 2013, pp. 1–8.

[34] D. Kościelnik, J. Szyduczyński, D. Rzepka, W. Andrysiewicz, and M. Miśkowicz, "Optimized design of successive approximation time-to-digital converter with single set of delay lines," in *Proc. 2nd Int. Conf. Event-Based Control, Commun., Signal Process.*, Jun. 2016, pp. 1–8.

[35] P. Keränen and J. Kostamovaara, "Algorithmic time-to-digital converter," in *Proc. NORCHIP*, Nov. 2013, pp. 1–4.

[36] P. Keränen and J. Kostamovaara, "A wide range, 4.2 ps(rms) precision CMOS TDC with cyclic interpolators based on switched-frequency ring oscillators," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 12, pp. 2795–2805, Dec. 2015.

[37] E. Bayer and M. Traxler, "A high-resolution ($<$ 10 ps RMS) 48-channel time-to-digital converter (TDC) implemented in a field programmable gate array (FPGA)," *IEEE Trans. Nucl. Sci.*, vol. 58, no. 4, pp. 1547–1552, Aug. 2011. doi: 10.1109/TNS.2011.2141684.

[38] 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.

[39] M. Bogdan *et al.*, "A 96-channel FPGA-based time-to-digital converter (TDC) and fast trigger processor module with multi-hit capability and pipeline," *Nucl. Instrum. Methods Phys. Res. Sect. A, Accel., Spectrometers, Detectors Associated Equip.*, vol. 554, nos. 1–3, pp. 444–457, 2005.

[40] S. Tancock, E. Arabul, N. Dahnoun, and S. Mehmood, "Can DSP48A1 adders be used for high-resolution delay generation?" in *Proc. 7th Medit. Conf. Embedded Comput. (MECO)*, Jun. 2018, pp. 1–6.

[41] S. Tancock and N. Dahnoun, "A 5.25 ps-resolution TDC on FPGA using DSP blocks," in *Proc. Digit. Image Signal Process.*, 2018, pp. 1–4.

[42] V. Kratyuk, P. K. Hanumolu, K. Ok, U.-K. Moon, and K. Mayaram, "A digital PLL with a stochastic time-to-digital converter," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 8, pp. 1612–1621, Aug. 2009.

[43] J. S. Teh, L. Siek, A. M. Alonso, A. Firdauzi, and A. Matsuzawa, "A 14-b, 850fs fully synthesizable stochastic-based branching time-to-digital converter in 65nm CMOS," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2018, pp. 1–5.

[44] P. Keranen, K. Maatta, and J. Kostamovaara, "Wide-range time-to-digital converter with 1-ps single-shot precision," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 9, pp. 3162–3172, Sep. 2011. doi: 10.1109/TIM.2011.2122510.

[45] M. Lee and A. A. Abidi, "A 9b, 1.25ps resolution coarse-fine time-to-digital converter in 90nm CMOS that amplifies a time residue," in *Proc. IEEE Symp. VLSI Circuits*, Jun. 2007, pp. 168–169.

[46] S. Henzler, S. Koeppe, W. Kamp, H. Mulatz, and D. Schmitt-Landsiedel, "90nm 4.7ps-resolution 0.7-LSB single-shot precision and 19pJ-per-shot local passive interpolation time-to-digital converter with on-chip characterization," in *IEEE Int. Solid-State Circuits Conf.-Dig. Tech. Papers*, Feb. 2008, pp. 548–635.

[47] S. Henzler, S. Koeppe, D. Lorenz, W. Kamp, R. Kuenemund, and D. Schmitt-Landsiedel, "Variation tolerant high resolution and low latency time-to-digital converter," in *Proc. 33rd Eur. Solid-State Circuits Conf.*, Sep. 2007, pp. 194–197.

[48] C.-M. Hsu, M. Z. Straayer, and M. H. Perrott, "A low-noise wide-BW 3.6-GHz digital $\Delta\Sigma$ fractional-N frequency synthesizer with a noise-shaping time-to-digital converter and quantization noise cancellation," *IEEE J. Solid-State Circuits*, vol. 43, no. 12, pp. 2776–2786, Dec. 2008.

[49] V. Ramakrishnan and P. T. Balsara, "A wide-range, high-resolution, compact, CMOS time to digital converter," in *Proc. 19th Int. Conf. VLSI Design Held Jointly 5th Int. Conf. Embedded Syst. Design*, Jan. 2006, p. 6.

[50] K. Maatta and J. Kostamovaara, "A high-precision time-to-digital converter for pulsed time-of-flight laser radar applications," *IEEE Trans. Instrum. Meas.*, vol. 47, no. 2, pp. 521–536, Apr. 1998.

[51] M. Safi-Harb and G. W. Roberts, "Embedded narrow pulse measurement in digital CMOS," in *Proc. IEEE Instrum. Meas. Technol. Conf.*, Apr. 2006, pp. 1195–1200.

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

[53] S. Tisa, A. Lotito, A. Giudice, and F. Zappa, "Monolithic time-to-digital converter with 20ps resolution," in *Proc. 29th Eur. Solid-State Circuits Conf.*, Sep. 2003, pp. 465–468.

[54] M. Takayama, S. Dosho, N. Takeda, M. Miyahara, and A. Matsuzawa, "A time-domain architecture and design method of high speed A-to-D converters with standard cells," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Nov. 2011, pp. 353–356.

[55] R. B. Staszewski, S. Vemulapalli, P. Vallur, J. Wallberg, and P. T. Balsara, "1.3 V 20 ps time-to-digital converter for frequency synthesis in 90-nm CMOS," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 3, pp. 220–224, Mar. 2006.

[56] C.-S. Hwang, P. Chen, and H.-W. Tsao, "A high-precision time-to-digital converter using a two-level conversion scheme," *IEEE Trans. Nucl. Sci.*, vol. 51, no. 4, pp. 1349–1352, Aug. 2004.

[57] A. Mantyniemi, T. Rahkonen, and J. Kostamovaara, "An integrated 9-channel time digitizer with 30 ps resolution," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, vol. 1, Feb. 2002, pp. 266–465.

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

[59] M. Rashdan, A. Yousif, J. Haslett, and B. Maundy, "A new time-based architecture for serial communication links," in *Proc. 16th IEEE Int. Conf. Electron., Circuits Syst.*, Dec. 2009, pp. 531–534.

[60] E. Räisänen-Ruotsalainen, T. E. Rahkonen, and J. Kostamovaara, "An integrated time-to-digital converter with 30-ps single-shot precision," *IEEE J. Solid-State Circuits*, vol. 35, no. 10, pp. 1507–1510, Oct. 2000.

[61] R. B. Staszewski *et al.*, "All-digital TX frequency synthesizer and discrete-time receiver for Bluetooth radio in 130-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 39, no. 12, pp. 2278–2291, Dec. 2004.

[62] A. Mäntyniemi and J. Kostamovaara, "Time-to-digital converter (TDC) based on startable ring oscillators and successive approximation," in *Proc. NORCHIP*, Oct. 2014, pp. 1–4.

[63] B. M. Helal, M. Z. Straayer, G.-Y. Wei, and M. H. Perrott, "A highly digital MDLL-based clock multiplier that leverages a self-scrambling time-to-digital converter to achieve subpicosecond jitter performance," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 855–863, Apr. 2008.

[64] J. Richardson *et al.*, "A 32 × 32 50ps resolution 10 bit time to digital converter array in 130nm CMOS for time correlated imaging," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2009, pp. 77–80.

[65] K. Karadamoglou, N. P. Paschalidis, E. Sarris, N. Stamatopoulos, G. Kottaras, and V. Paschalidis, "An 11-bit high-resolution and adjustable-range CMOS time-to-digital converter for space science instruments," *IEEE J. Solid-State Circuits*, vol. 39, no. 1, pp. 214–222, Jan. 2004.

[66] C. Veerappan *et al.*, "A 160 × 128 single-photon image sensor with on-pixel 55ps 10b time-to-digital converter," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2011, pp. 312–314.

[67] I. Nissinen and J. Kostamovaara, "Time-to-digital converter based on an on-chip voltage reference locked ring oscillator," in *Proc. IEEE Instrum. Meas. Technol. Conf.*, Apr. 2006, pp. 250–254.

[68] D. K. Xie, Q. C. Zhang, G. S. Qi, and D. Y. Xu, "Cascading delay line time-to-digital converter with 75 ps resolution and a reduced number of delay cells," *Rev. Sci. Instrum.*, vol. 76, no. 1, p. 014701, Jan. 2005.

[69] V. Dhanasekaran *et al.*, "A 20MHz BW 68dB DR CT $\Delta\Sigma$ ADC based on a multi-bit time-domain quantizer and feedback element," in *IEEE Int. Solid-State Circuits Conf.-Dig. Tech. Papers*, Feb. 2009, pp. 174–175.

[70] C. Niclass, C. Favi, T. Kluter, M. Gersbach, and E. Charbon, "A 128 × 128 single-photon image sensor with column-level 10-bit time-to-digital converter array," *IEEE J. Solid-State Circuits*, vol. 43, no. 12, pp. 2977–2989, Dec. 2008.

[71] B. K. Swann *et al.*, "A 100-ps time-resolution CMOS time-to-digital converter for positron emission tomography imaging applications," *IEEE J. Solid-State Circuits*, vol. 39, no. 11, pp. 1839–1852, Nov. 2004.

[72] D. M. Santos, S. F. Dow, J. M. Flasck, and M. E. Levi, "A CMOS delay locked loop and sub-nanosecond time-to-digital converter chip," *IEEE Trans. Nucl. Sci.*, vol. 43, no. 3, pp. 1717–1719, Jun. 1996.

Scott Tancock received the master's degree in computer science and electronics from the University of Bristol, Bristol, U.K., in 2016, where he is currently pursuing the Ph.D. degree in high-resolution time-to-digital converters (TDCs) on field-programmable gate arrays (FPGAs) with the Department of Electrical and Electronic Engineering.

His current research interests include embedded systems, microcontrollers, very large scale integration (VLSI), digital signal processing (DSP), image processing, and computer vision.

Ekin Arabul was born in Ankara, Turkey, in 1992. He received the M.Eng. degree in computer science and electronics from the University of Bristol, Bristol, U.K., in 2015, where he is currently pursuing the Ph.D. degree in electrical and electronic engineering.

He is currently a member of the Photonic Quantum Information Research Group, University of Bristol. His current research interests include precise time measurement tools in time-of-flight applications, coincidence counters, time-to-digital converters (TDCs), and field-programmable gate arrays (FPGAs).

Naim Dahnoun received the Ph.D. degree in biomedical engineering from the University of Leicester, Leicester, U.K., in 1990.

He was a Researcher on bloodflow measurements for femoral bypass grafts with the Leicester Royal Infirmary, Leicester. He was a Lecturer in digital signal processing (DSP) with the University of Leicester. He was with the Department of Electrical and Electronic Engineering, University of Bristol, Bristol, U.K., where he became a Reader in learning and teaching in 1994. In 1993, he was with the Institute of Science and Technology, University of Manchester, Manchester, U.K., where he was involved in the research on optical communication, especially wideband optical communication links. His current research interests include real-time digital signal processing applied to biomedical engineering, video surveillance, automotive, and optics.

Dr. Dahnoun was a recipient of the first Texas Instruments DSP Educator Award from Texas Instruments (NYSE:TXN) for his outstanding contributions to furthering education in DSP technology in 2003, in recognition of the important role played by universities in educating engineers in new technologies such as real-time DSP.