An Ultra-Low-Power Polarity-Coincidence Feedback Time-Delay-to-Digital Converter for Sound-Source Localization

Daniel de Godoy, *Member, IEEE*, and Peter R. Kinget, *Fellow, IEEE*

***Abstract*— The time-delay between audio signals in a microphone array is the most important feature for sound-source localization. This article presents a polarity-coincidence, adaptive time-delay estimation (PCC-ATDE), a mixed-signal technique that uses 1-bit quantized signals and a negative-feedback architecture to directly determine the time-delay between signals in the analog inputs and convert it to a digital number. This direct conversion, without a multi-bit analog-to-digital converter and further digital-signal processing, allows for ultra-low power consumption. A prototype chip in 0.18-um CMOS with four analog inputs consumes 78 nW with a three-channel 8-bit digital time-delay output while sampling at 50 kS/s with a 20-us resolution and 6.06 ENOB. We present a theoretical analysis for the non-linear, signal-dependent feedback loop of the PCC-ATDE. A delay-domain model of the system is developed to estimate the power-bandwidth of the converter and predict its dynamic response. Results are validated with experiments using real-life stimuli, captured with a microphone array, that demonstrates the ability of the technique to localize a sound source. The chip is further integrated into an embedded platform and deployed as an audio-based vehicle-bearing IoT system.**

***Index Terms*— Time-delay estimation (TDE), time-difference of arrival, ultra-low-power sound-source localization.**

I. Introduction

In A cyber-physical system (CPS), the data converters are key blocks connecting the digital signal processing or control blocks to the real world by encoding the response of sensors into a format that can be easily manipulated by the digital blocks. In a CPS with a machine-learning classifier backend, the digital backend is only interested in the features present in the sensor signals. A traditional analog-to-digital converter (ADC), however, converts the complete, raw sensor signal into a digital signal, which is then processed by digital feature extraction blocks. In the *analog-to-feature* approach, the sensor interface is specifically optimized to extract the features in the analog domain and only digitizes the features. In this article, we demonstrate the analog-to-feature approach in the context of a sound-source localization CPS to detect cars.

Manuscript received June 4, 2019; revised September 21, 2019; accepted October 21, 2019. Date of publication November 13, 2019; date of current version May 27, 2020. This article was approved by Associate Editor Piero Malcovati. This work was supported by NSF under Grant CNS-1704899. *(Corresponding author: Daniel de Godoy.)*

The authors are with the Department of Electrical Engineering, Columbia University, New York, NY 10027 USA (e-mail: dd2697@columbia.edu).

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/JSSC.2019.2950322

Vehicle awareness systems are of significant interest to the smart-city community [1], [2]. The ability to localize vehicles in an urban area can be the first step to reduce the number of traffic accidents involving pedestrians. Large-scale systems, integrated in traffic lights or in smart vehicles, use the various techniques to detect cars from light detection and ranging (LIDAR) [3], [4] to stereo-vision [5] to radio frequency networks [6]. Such systems, however, need to have access to a large power source, either the vehicle's battery or the power grid. Designing a wearable vehicle-aware system powered only by small batteries and with reduced footprint [7] still faces significant technological challenges. Audio-based integrated systems have demonstrated encouraging progress toward reaching ultra-low power consumption [8], [9], since the low frequency of the acoustic signal allows the integrated circuits to operating with a reduced clock frequency and supply voltage.

In this article, we focus on an ultra-low-power wearable application, where an array with four microphones separated by 25–35 cm is used to localize vehicles in urban areas [7]. For that, we need to find three time-difference of arrivals (TDoAs) between three microphones in the array and the reference microphone that can range from -1 to 1 ms. The noise from approaching automobiles or other vehicles [7], [10] has dominant spectral components below 250 Hz. We aim to find the angle-of-arrival of approaching cars that are at least 15-m away from the microphone array, with a speed limit of 30 mph, leading to a maximum angular velocity of 1 rad/s in the TDoA signal. The TDoAs are then used as features for a machine-learning classifier which aims to place the vehicle in the field-of-view of the user, leading to a final required accuracy of $\pm 60°$.

Time-delay estimations (TDE) are used to extract TDoA between microphones in sound-source localization systems [**?**], [11]–[13]. The standard approach uses the direct cross correlation (DCC) function. For each TDE, time frames of the input signals are stored, and all the points of the DCC function are calculated. The argument of the maximum of the DCC corresponds to the intersignal time delay between the signal inputs. With a sampling frequency above 50 kS/s, the storage of the frames and the arithmetic operations to calculate the DCC values are a roadblock toward achieving a sub-μW implementation [12] as is needed in mobile, wearable, cyber-physical, or Internet-of-Things (IoT) applications.

The low complexity of adaptive TDE (ATDE) techniques, such as the least mean square (LMS)-TDE [14], makes them

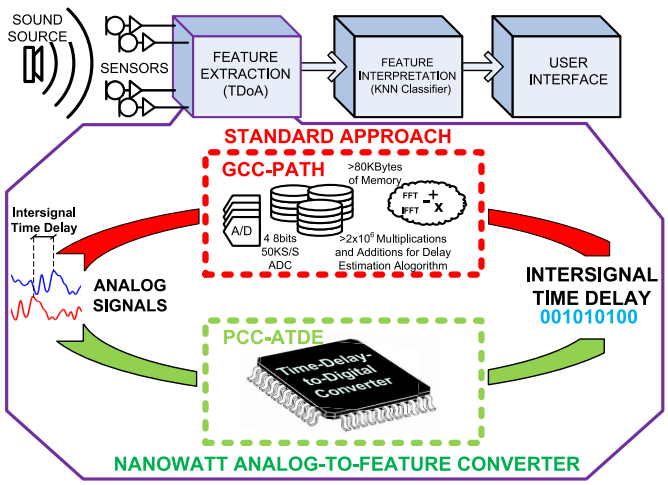

Fig. 1. Example of how PCC-ATDE can be used to reduce components in a sound-source localization IoT system. PCC-ATDE is used to directly extract the time-difference of arrival between the analog signals of the microphones to digital domain. It uses significantly less resources than traditional approaches, such as the GCC-PHAT.

an attractive approach, but they still require a high-resolution ADC after the sensors. The bio-inspired silicon cochlea in [15] estimates the time difference by translating the audio stimulus into asynchronous events, but its power consumption is still in the μW range.

We present a 78.2-nW 50-kS/s time-delay-to-digital converter with four audio input channels and three 8-bit delay outputs [16]. The presented architecture, illustrated in Fig. 1, does not require a multi-bit ADC, does not use memory blocks to store frames or intermediate results, and does not execute any computationally expensive algorithm.

This article is organized as follows. Section II presents the negative-feedback tracking loop architecture of the proposed TDE. Section III analyses the discrete-time implementation of the method. Section IV introduces a delay-domain model of the loop used in behavioral simulations to analyze and validate the proposed method. Section V describes the silicon implementation of the ultra-low-power TDE prototype and its characterization is presented in Section VI. Section VII compares our performance with previous work. The prototype is used to build a sound-source localization system, tested in Section VIII-A in a controlled indoor environment, and used in Section VIII-B to detect the bearing of approaching cars in the streets of New York, NY, USA.

II. Proposed Time-Delay Estimation Approach Using a Feedback Loop

When designing a TDE block, the considerations for selecting the sampling frequency are different from other feature-extraction blocks where it is typically set by the signal bandwidth, which sets the Nyquist rate. In a TDE, the sampling frequency of the data converters, F_S, defines the resolution. In this article, we support a $-$ 1–1-ms delay range with an 8-bit resolution for noise-like audio sources with dominant spectral components below 250 Hz; for this case, the audio signal needs to be sampled at $>$ 50 kS/s, i.e., 100$\times$ the Nyquist rate.

A. TDE With Direct Cross Correlation

Consider the outputs of two microphones, $M_1(t)$ and $M_2(t)$ at different positions in space, which capture the signal of a single-source $x(t)$

$$M_1(t) = x(t) + n_1(t) \tag{1}$$

$$M_2(t) = (1+\epsilon)\cdot x(t-D) + n_2(t) \tag{2}$$

where D is the time delay that the algorithm needs to determine, $n_1(t)$ and $n_2(t)$ are random noise, and ϵ is the gain (or attenuation) difference in the microphones. The estimation of D requires computing the DCC

$$DCC_{M_1,M2}(\tau) = \frac{1}{T}\int_0^T M_1(t).M_2(t-\tau)\mathrm{d}t \tag{3}$$

for many different τ and then determining the argument of the peak

$$D = argmax(DCC_{M_1,M2}(\tau)). \tag{4}$$

There are multiple ways to compute $\mathrm{DCC}_{M_1,M2}(\tau)$: in time domain, frequency domain, or you can apply weighting functions to emphasize the peak as done in the generalized cross correlation phase transform method (GCC-PHAT) [14].

B. TDE with Polarity-Coincidence Correlation

An alternative to reduce the complexity of the $\mathrm{DCC}_{M_1,M2}(\tau)$ calculations is to use the polarity-coincidence correlation function [17]

$$\mathrm{PCC}_{M_1,M_2}(\tau) = \frac{1}{T}\int_0^T sgn(M_1(t)).sgn(M_2(t-\tau))\mathrm{d}t. \tag{5}$$

In [18], it is proven that

$$\mathrm{PCC}_{M_1,M2}(\tau) = \frac{2}{\pi} sin^{-1}\left(\frac{DCC_{M_1,M2}(\tau)}{\max(DCC_{M_1,M2}(\tau))}\right) \tag{6}$$

Hence, $\mathrm{argmax}(\mathrm{PCC}_{M_1,M2}) = \mathrm{argmax}(\mathrm{DCC}_{M_1,M2})$. Note that the computation of PCC only needs 1-bit signals, in contrast to DCC which requires multi-bit signals. The 1-bit quantization of the PCC also makes it less sensitive to the gain difference of the microphones ϵ.

Regardless of how you obtain $\mathrm{DCC}_{M_1,M2}(\tau)$ or $\mathrm{PCC}_{M_1,M2}(\tau)$, their computation involves storing a large frame of both $M_1(t)$ and $M_2(t)$; finding D requires calculating and storing the cross correlation for the various τ within the TDE range, and finally searching for the argument of the peak.

C. Polarity-Coincidence, Adaptive Time-Delay Estimation

We present the polarity-coincidence correlation, adaptive, TDE (PCC-ATDE) approach. The PCC-ATDE uses only two values of the PCC function to close a negative feedback loop that continuously tracks the intersignal delay D. Fig. 2 illustrates the principle of the PCC-ATDE. Two points of $\mathrm{PCC}_{M_1,M_2}(\tau)$ are calculated, one with argument Δ, marked with a red square, and the other at $\Delta + \tau_{\mathrm{fix}}$, marked with a blue circle, both controlled by the output V_Δ. To search for the $D = \mathrm{argmax}(\mathrm{PCC}_{M_1,M_2})$, the loop takes the difference between the two PCC_{M_1,M_2} values. In Fig. 3, three

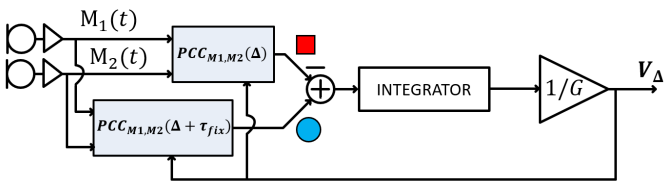

Fig. 2. Simplified block diagram of the negative-feedback tracking loop used by the PCC-ATDE loop to estimate the intersignal delay D.

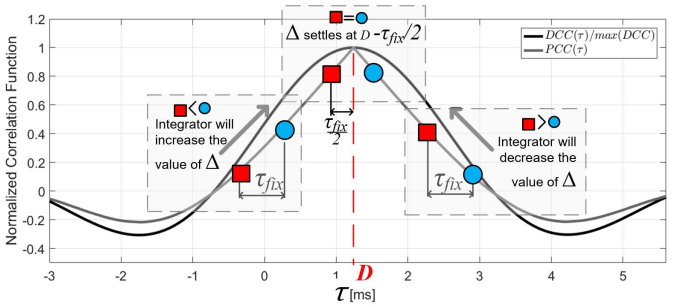

Fig. 3. Polarity-coincidence correlation function, PCC(τ), defined in (3), and the normalized DCC function, DCC(τ)/max(DCC(τ)), defined in (5), of band-limited noise delayed by $D = 1.2$ ms. The x-axis is the time shift τ applied to the input $M_2(t)$ to calculate each point of the functions. Marked as red triangles and blue circles are three possible pairs of values of $\text{PCC}_{M_1,M_2}(\Delta)$ and $\text{PCC}_{M_1,M_2}(\Delta + \tau_{\text{fix}})$ in the PCC-ATDE loop in Fig. 2.

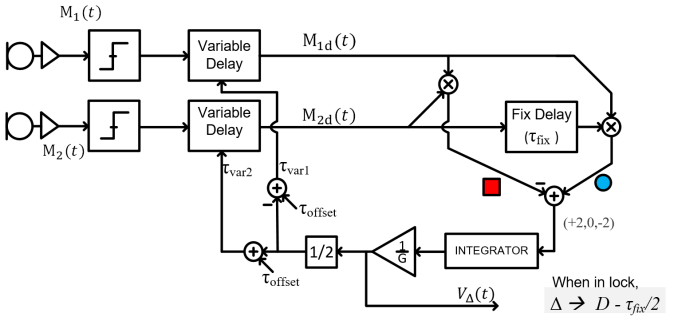

Fig. 4. Block diagram of the proposed PCC-ATDE loop.

possible cases are shown. If the current Δ is sufficiently close to the argument of the peak, the difference between $\text{PCC}_{M1,M2}(\Delta)$ and $\text{PCC}_{M1,M2}(\Delta+\tau_{\text{fix}})$ indicates whether Δ is smaller or larger than the argument of the peak. The integrator will continuously increase or decrease the value of V_Δ until $\text{PCC}_{M1,M2}(\Delta)$ and $\text{PCC}_{M1,M2}(\Delta + \tau_{\text{fix}})$ have equal values, locking the loop at $\Delta = D-\tau_{\text{fix}}/2$, which gives a measurement of the desired intersignal delay D. The attenuator $1/G$ sets the speed and bandwidth of the loop. Its effects on the TDE will be investigated in Section IV. A practical problem for the architecture in Fig. 2 is that to calculate the PCC_{M_1,M_2} value pair a large frame of each input signal has to be stored.

The block diagram in Fig. 4 shows how $\text{PCC}_{M_1,M_2}(\Delta)$ and $\text{PCC}_{M_1,M_2}(\Delta + \tau_{\text{fix}})$ can be extracted in the PCC-ATDE without storing signal frames. The analog microphone signals are directly connected to a comparator acting as a 1-bit ADC. Then, each signal goes through a variable-delay cell controlled

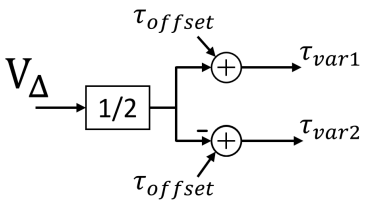

Fig. 5. Block digram of how τ_{var1} and τ_{var2} are obtained from V_Δ. τ_{offset} guarantees that neither assume negative values. τ_{var1} and τ_{var2} are the control voltage for the variable delay cells with 1-s/V conversion gain as expressed in (7) and (8).

by τ_{var1} and τ_{var2}

$$M_{1d}(t) = sgn(x(t - \tau_{var1}) + n_1(t - \tau_{var1})) \tag{7}$$

$$M_{2d}(t) = sgn((1 + \epsilon) \cdot x(t - \tau_{var2} - D) + n_2(t - \tau_{var2})) \tag{8}$$

As shown in Fig. 5, the variable delay τ_{var1} and τ_{var2} are defined such that $V_\Delta = \tau_{\text{var2}} - \tau_{\text{var1}}$. Assuming a 1-s/V conversion gain in the variable delay cells, the corresponding difference in delay Δ introduced by the variable delays cells is $\Delta = V_\Delta$. And, since a variable delay line can only introduce positive delay values, an offset τ_{offset} is added such that τ_{var1} and τ_{var2} are always positive. When the loop settles, Δ corresponds to the TDE between the inputs, and can assume both positive and negative values depending on which input is ahead of the other.

Next, $M_{1d}(t)$ and $M_{2d}(t)$ are multiplied to create $V_{\text{MIXER1}}(t)$. The average of $V_{\text{MIXER1}}(t)$ is the same as $\text{PCC}_{M_1,M_2}(\Delta)$

$$\begin{aligned} &\frac{1}{T}\int_{t-T}^{t} V_{\text{MIXER1}}(t)\text{d}t \\ &\quad = \frac{1}{T}\int_{t-T}^{t} sgn(M_1(t)).sgn(M_2(t-\Delta))\text{d}t \\ &\quad := PCC_{M_1,M2}(\Delta). \end{aligned} \tag{9}$$

$M_{2d}(t)$ is further delayed by a fixed value, τ_{fix}, and then multiplied with the upper branch to create V_{MIXER2}. The average V_{MIXER2} is $\text{PCC}_{M1,M2}(\Delta + \tau_{\text{fix}})$.

Note that the averages of the multiplier's output, V_{MIXER1} and V_{MIXER2}, do not need to be explicitly calculated. The loop integrator used to obtain a low-pass feedback loop performs the averaging and attenuates the higher frequency components of these signals. $1/G$ controls the loop bandwidth, guaranteeing that the average value is being properly calculated.

III. Discrete-Time Implementation of the PCC-ATDE Loop

So far, we presented a continuous-time PCC-ATDE loop; however, realizing programmable variable delays lines for audio signals is very difficult; there are substantial design simplifications when implementing a discrete-time realization since it only requires conventional building blocks typically available in a signal processing library.

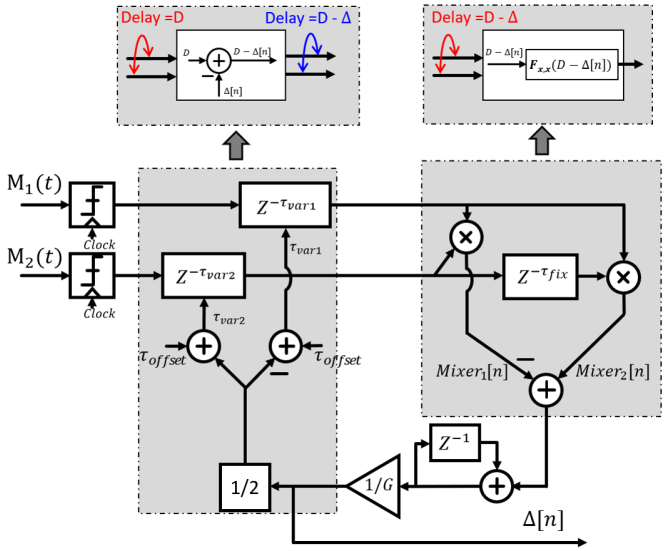

Fig. 6. Block diagram of the time-discrete implementation using latched comparators, digital delay cells, XORs, adders, and dividers. The two shaded areas have different functionalities: the section on the left is used to add a delay of $\Delta[n]$ to the microphone signals; on the right is part of the loop responsible $F_{x,x}$, which is a function of the intersignal time-delay of its input.

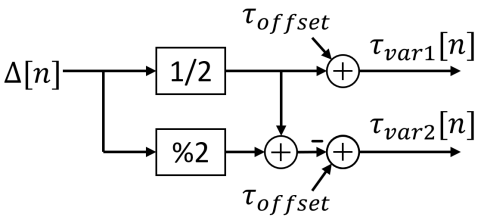

Fig. 7. Discrete-time implementation of $\tau_{\text{var1}}[n]$ and $\tau_{\text{var2}}[n]$. $\Delta[n]$ is a signed integer, the division by 2 is done by an arithmetical shift right, and the modulus operator % will add 1 to lower branch in case of an odd value of $\Delta[n]$. τ_{var1} and τ_{var2} are unsigned integers used to control the number of elements in two chains of flip-flops.

Fig. 6 shows a discrete-time realization of the PCC-ATDE loop in Fig. 4. The output $\Delta[n]$ at timestep n is given by:

$$\Delta[n] = \Delta[n-1] + \frac{1}{G} \cdot (\text{Mixer}_2[n] - \text{Mixer}_1[n]). \quad (10)$$

For the time-discrete implementation, the control of the variable delay cells $\Delta[n]$ is a digital value. It will control the number of elements in two chains flip-flops, hence introducing one sampling interval per quantization step. That defines T_{LSB} of the PCC-ATDE as $1/F_s$. To keep the values of τ_{var1} and τ_{var2} positive integers, the scheme of Fig. 7 is used. As in (9), since $\tau_{\text{var1}}[n] - \tau_{\text{var2}}[n] = \Delta[n]$, we can show that the average value of Mixer_1 is the same as $PCC_{M_1,M_2}(\Delta)$:

$$\begin{aligned} &\frac{1}{T}\sum_{k=n-T}^{n} \text{Mixer}_1[n] \\ &\quad = \frac{1}{T}\sum_{k=n-T}^{n} sgn(M_1[k-\tau_{var1}[k]]).sgn(M_2[k-\tau_{var2}[k]]) \\ &\quad := PCC_{M_1,M_2}(\Delta[n]) \end{aligned} \quad (11)$$

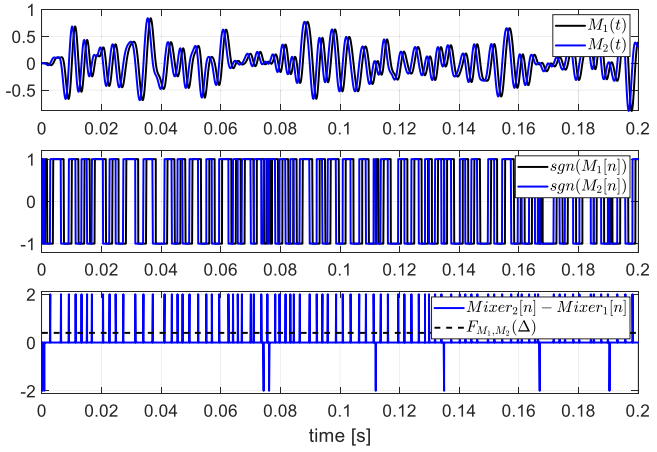

Fig. 8. Simulated results of microphone inputs, $M_1(t)$ and $M_2(t)$; the resulting $\text{Mixer}_2[n] - \text{Mixer}_1[n]$ is a fast-switching signal with much higher frequencies than the PCC-ATDE loop can track; its average, $F_{M_1,M_2}(\Delta)$, is a function of the intersignal delay between the signals at the microphones.

and, consequently, the average value of Mixer_2 is the same as $\text{PCC}_{M_1,M_2}(\Delta + \tau_{\text{fix}})$. Since both the average of Mixer_1 and Mixer_2 are function of Δ and if we assume that the PCC_{M_1,M_2} and τ_{fix} are static, the average value of the subtraction $\text{Mixer}_2[n] - \text{Mixer}_1[n]$ can be defined as $F_{M_1,M_2}(\Delta[n])$

$$\begin{aligned} &\frac{1}{T}\sum_{k=n-T}^{n} [\text{Mixer}_2[n] - Mixer_1[n]] \\ &\quad = PCC_{M_1,M_2}(\Delta[n] + \tau_{fix}) - PCC_{M_1,M_2}(\Delta[n]) \\ &\quad = F_{M_1,M_2}(\Delta[n]). \end{aligned} \quad (12)$$

Fig. 8 shows a simulated example of $\text{Mixer}_2[n] - \text{Mixer}_1[n]$. It is a fast-switching digital signal that can only assume values of $\{-2,0,2\}$. The ac components of $\text{Mixer}_2[n] - \text{Mixer}_1[n]$ can be expressed as $e[n]$, since they will be attenuated by the PCC-ATDE loop that has a much lower cutoff frequency. Rewriting $\text{Mixer}_2[n] - \text{Mixer}_1[n]$ as its dc and ac components, we have

$$\text{Mixer}_2[n] - \text{Mixer}_1[n] = F_{M_1,M_2}(\Delta[n]) + e[n]. \quad (13)$$

We can now substitute $F_{M_1,M_2}(\Delta[n])$ and $e[n]$ into (10)

$$\Delta[n] = \Delta[n-1] + \frac{1}{G} \cdot F_{M_1,M_2}(\Delta[n]) + \frac{1}{G} \cdot e[n]. \quad (14)$$

$e[n]$ does not introduce a dc error in $\Delta[n]$ but only contributes some noise at the output. If we neglect $e[n]$ and focus on the low-frequency output of the loop, we obtain a non-linear feedback loop that continuously increases or decreases $\Delta[n]$ to keep $F_{M_1,M_2}(\Delta[n]) = 0$.

IV. Analysis of the Discrete-Time PCC-ATDE Loop

We now analyze the behavior of the PCC-ATDE loop. We will focus specifically on the case where $M_1(t)$ and $M_2(t)$ are signals coming from the same source $x(t)$, but are just delayed due the different signal path to the microphones as expressed in (1) and (2).

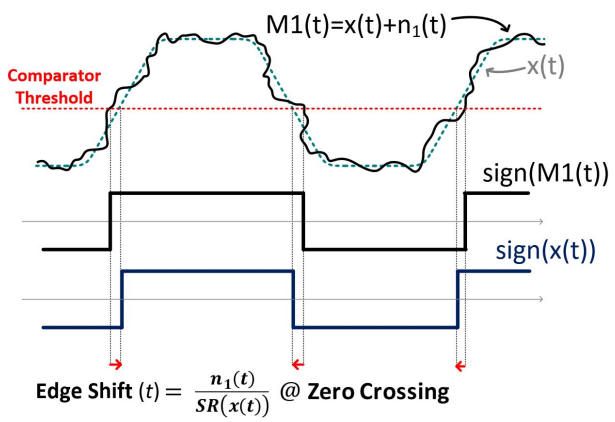

Fig. 9. Effect of uncorrelated voltage noise on the comparator's output. The noise causes shifts in the edges that are proportional to the amplitude of $n_1(t)$ at the crossing point and the slew rate of the $x(t)$. However, the shifts in the edges are random and average out to zero.

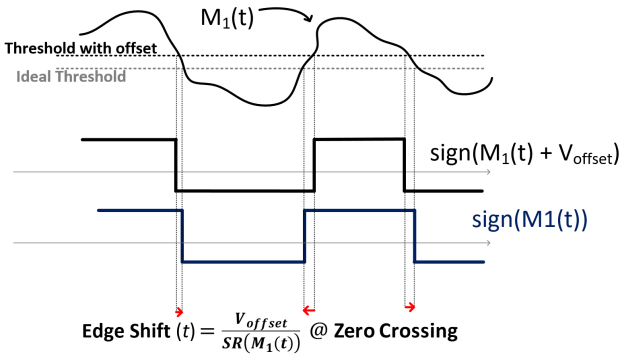

Fig. 10. Degradation of the comparator's output caused by a small offset in the threshold voltage. The shifts in the edges are proportional to the offset voltage and the slew rate of the $M_1(t)$. With a random input, and random slew rate, the shifts in the edges are also random and with average zero.

A. Impact of Noise and Comparator Offset

The effect of noise can be analyzed by adding the uncorrelated, input-referred noise terms $n_1(t)$ and $n_2(t)$ into (1) and (2). As shown in Fig. 9, the noise will move the edge of the comparator's output, and the amount by which the edge is shifted depends on the noise amplitude and the slew rate of the source signal $x(t)$. However, given that the low bandwidth of the loop, and since the shifts in the edges are uncorrelated and random, with a zero average, their impact on the noise at the output of the time-delay estimator in minimal.

Similarly, we can analyze the effect of a small offset in the comparator threshold voltage. Fig. 10 shows how the edge of the comparator's output will shift depending on the offset voltage and the slew rate of $M_1(t)$ at the crossing point. For a random signal,[1] as shown in Fig. 10, the random slew rate will cause random shifts in the edges that look like as noise with a zero mean at the output of the delay estimator.

This noise, however, will also be significantly attenuated by the loop transfer function.

B. Delay-Domain Model

Based on (14), we now propose a delay-domain model to predict the behavior of the PCC-ATDE loop, which operates across multiple signal domains. The analysis of a phase-locked loop (PLL) also faces this multi-domain analysis challenge[2] and their phase-domain models are often used to analyze and design PLLs [19]. Here, a delay-domain model will assist in the PCC-ATDE design. In a PLL, the conversion from the time domain to the phase domain is done by the (linear) phase detector (PD) that takes as inputs the reference and VCO signals and outputs a value corresponding to their phase difference. In the PCC-ATDE, the (non-linear) function $F_{M_1,M_2}(\Delta)$ is responsible for that domain conversion; the value of $F_{M_1,M_2}(\Delta)$ is directly dependent on the difference between estimated time delay Δ and the intersignal time delay from the microphones D.

Now, we are going to rewrite F_{M_1,M_2} to make the dependence on D explicit. Since the only correlation between $M_1(t)$ and $M_2(t)$ comes from the source $x(t)$, the polarity-coincidence correlation function between M_1 and M_2 can be approximated by the auto-polarity-coincidence correlation function of $x(t)$ shifted by the intersignal delay D

$$PCC_{M_1,M_2}(\Delta) \approx PCC_{x,x}(D-\Delta). \tag{15}$$

The contribution of the microphone delay D to F_{M_1,M_2} can now be derived, resulting in the function $F_{x,x}$

$$\begin{aligned} F_{M_1,M_2}(\Delta) &\approx PCC_{x,x}(D-\Delta-\tau_{fix}) - PCC_{x,x}(D-\Delta) \\ &= F_{x,x}(D-\Delta) \end{aligned} \tag{16}$$

The approximation in (15) and $F_{x,x}$ are shown in Fig. 11. Examples of PCC_{M_1,M_2} and $\text{PCC}_{x,x}$ are shown for a band-limited noise $x(t)$ and a sinusoidal $x(t)$, and the resulting $F_{x,x}(D-\Delta)$. Notice that $F_{x,x}$ only depends on the source signal $x(t)$[3] and the fixed delay τ_{fix}; and, since it is defined by the difference of two $\text{PCC}_{x,x}$ values spaced by τ_{fix}, it can be understood as the derivative of $\text{PCC}_{x,x}$.

Substituting $F_{x,x}$ into (14), we obtain a direct relation between the output of the loop $\Delta[n]$ and the intersignal delay D

$$\Delta[n] = \Delta[n-1] + \frac{1}{G}\cdot F_{x,x}(D-\Delta[n]) + \frac{1}{G}\cdot e[n] \tag{17}$$

resulting in the delay-domain model in Fig. 12.

This model is a powerful tool to extract the system's transient and steady-state responses and to determine the boundaries for the correct operation of the loop. The low number of elements in the model and the first-order negative-feedback architecture might a misleading impression that the

[1]For deterministic periodic signals, such as sinusoidal inputs, the threshold offset will cause deterministic shifts in the edges. If the slew rate for the rising and falling edge are equal, the shift will have an average of zero and a high-frequency component at the frequency of the periodic $M_1(t)$ that will be filtered by the loop. For unequal slew rates, a non-zero average can occur, however, the audio signals of interest here, such as engine noise, can be modeled like random noise-like signals.

[2]Note that the analogy to a PLL only relates to the modeling approach but not to the operation of the loops.

[3]For this application we focus on a bandwidth-limited noise $x(t)$, but the same analysis can be followed for different types of signal.

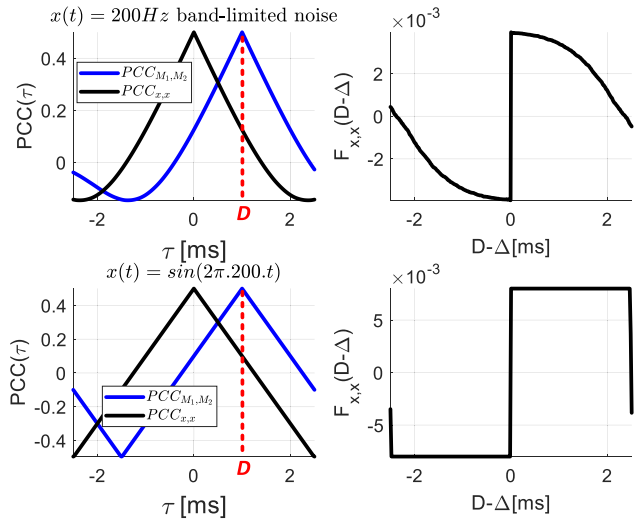

Fig. 11. Simulated PCC_{M_1,M_2}, $\mathrm{PCC}_{x,x}$, and $F_{x,x}$ for band-limited noise and sinusoidal source signal $x(t)$. PCC_{M_1,M_2} peaks at the intersignal delay D, while the auto-polarity-coincidence correlation $\mathrm{PCC}_{x,x}$ always peaks at 0, both with similar shape since M_1 and M_2 only correlated factor is $x(t)$. $F_{x,x}$ is the derivative of $\mathrm{PCC}_{x,x}$.

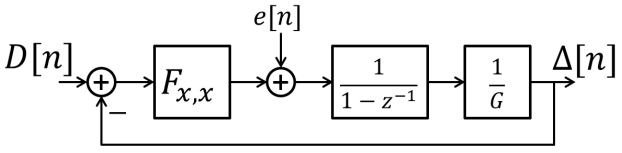

Fig. 12. Delay-domain model of the PCC-ATDE. The input for this model is the intersignal time delay between the microphones $D[n]$. Since $F_{x,x}$ is the dc component of operation, the remaining undesired high-frequency components are expressed as an error $e[n]$.

PCC-ATDE will follow a conventional analysis. The non-linear, $x(t)$-dependent function $F_{x,x}$ is a complex mathematical element that affects all the parameters of the system, from the range of converter to the settling time and bandwidth.

C. Audio Input Bandwidth and Converter Range

In order to maintain the negative feedback and guarantee the convergence to the correct TDE, $F_{x,x}[D-\Delta]$ has to have positive values for $\Delta < D$ and negative values for $\Delta > D$. Since $F_{x,x}(D-\Delta)$ is defined as the difference of two consecutive values of $\mathrm{PCC}_{x,x}(\tau)$ (16), the equivalent condition is that the derivative of $\mathrm{PCC}_{x,x}(\tau)$ is positive for positive τ and negative for negative τ.

Fig. 13 shows the $\mathrm{PCC}_{x,x}(\tau)$ of band-limited noise signals. The local minima that limit the convergence condition for the PCC-ATDE, are located τ_{MAX} away from the peak of the $PCC_{x,x}(\tau)$ at $\tau = 0$. Higher bandwidth analog input signals have their local minimum closer to the origin, and hence a have smaller τ_{MAX}.

To define a time-delay range for the system, we need to ensure that any possible time delay between the input signals D differs from any current output values $\Delta[n]$ by less than τ_{MAX}

$$|\Delta[n] - D| < \tau_{MAX}. \tag{18}$$

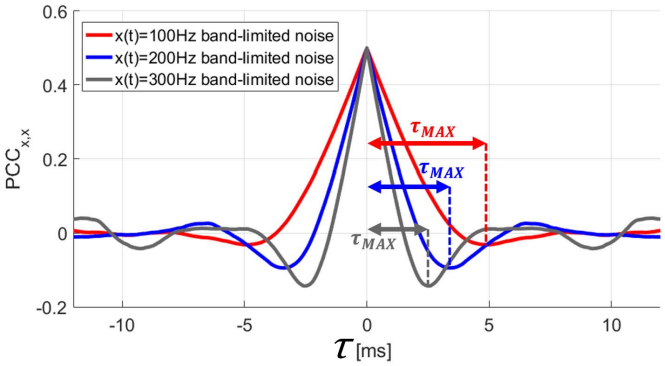

Fig. 13. Example of the resulting $\mathrm{PCC}_{x,x}(\tau)$ for inputs $x(t)$ signals with different bandwidth. τ_{MAX} indicates, for each bandwidth value, the maximum difference between $\Delta[n]$ and $D[n]$ that the PCC-ATDE can tolerate and still correctly track the delay.

A $\Delta[n] - D$ that exceeds τ_{MAX} will cause the PCC-ATDE to lock to the nearest adjacent local maximum of $\mathrm{PCC}_{x,x}$, shifting the TDE to a wrong value.

This is an important design parameter for sound-source localization systems, where the maximum time delay between the input signals is limited by the spacing of the microphones. For example, in this application, if the microphones are separated by 35 cm, $|D| < 1$ ms, for a speed of sound of 343 m/s. Limiting the output so that $|\Delta| < 1.5$ ms will guarantee that the loop stays within the covered range for $x(t)$ sources with a bandwidth up to 200 Hz that have a $\tau_{\mathrm{MAX}} > 2.5$ ms. Low-pass filters can be used before the PCC-ATDE to limit the bandwidth of $x(t)$ to guarantee this condition. Dynamic variations in the spectrum of $x(t)$ will cause changes in the shape of $\mathrm{PCC}_{x,x}$; as long as (18) is respected at all times, the loop will still be able to correctly track the intersignal delay.

D. Response to a Step in the Intersignal Time Delay

As highlighted in the delay-domain model, the input to the PCC-ATDE feedback loop is the intersignal time delay D of the analog signal $M_1(t)$ and $M_2(t)$. Hence, to analyze the step response of the system, we vary the delay between two identical 200-Hz band-limited noise applied to the microphone inputs. Fig. 14 shows the step responses simulated with the proposed delay-domain model when steps of 0.3, 0.6, 0.9, 1.2, and 1.5 ms are applied in the time delay 2 s after the beginning of the simulation. The experimental data is also shown and will be discussed in Section VI.

The settling times of the responses vary from 0.36 s for a 0.3-ms step to 1.72 s for a 1.5-ms step and depend on the step amplitude. Looking back at the delay-domain model in Fig. 12, we see that this behavior comes from the slope limitation caused by non-linear element $F_{x,x}$

$$\Delta[n] - \Delta[n-1] = \frac{1}{G} \cdot F_{x,x}(D - \Delta). \tag{19}$$

With the same $x(t)$ used to obtain $F_{x,x}$ shown in the top right corner of Fig. 11, for small amplitudes of $D - \Delta$, we can approximate $F_{x,x}$ as a step function with limits $\pm 0.004\ T_{\mathrm{LSB}}$.

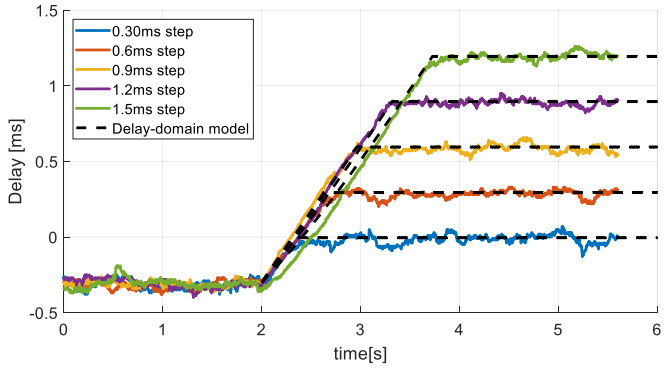

Fig. 14. Step response of the time-delay-to-digital converter with different step amplitudes. Colored continuous lines represent experimental data, and dashed lines are simulated using the delay-domain model. The close match of the results validates the delay-domain model transient response.

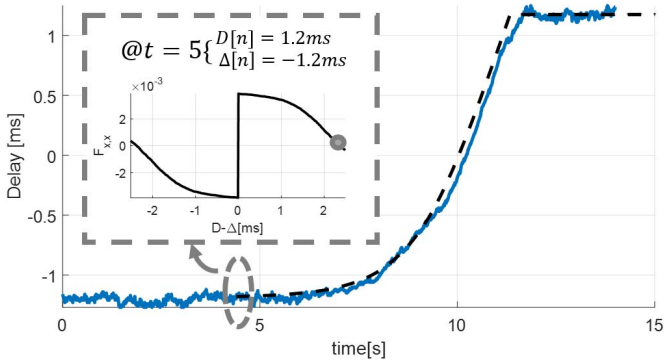

Fig. 15. Isolated 2.4-ms step. The colored continuous lines represent experimental data and the dashed line is simulated using the delay-domain model. The large step is used to highlight the effect of $F_{x,x}$, shown inside the dashed box, on the transient response. At $t = 5$ s, $D - \Delta = 2.4$ ms making $F_{x,x}$ very small, as shown by the gray dot in the plot, hence reducing the slope of the output.

Now, using (20) with $G = 4$, $|F_{x,x}| = 0.004\ T_{\text{LSB}}$ and an input step $A_{\text{step}} = 0.3$ ms, we calculate a settling time of $T_{\text{set}} = 0.3$ s with

$$T_{\text{set}} = \frac{|A_{\text{step}}| \cdot G}{|F_{x,x}(0)|} \tag{20}$$

which fairly close to the measured result. As the amplitude of the input step starts to increase, the shape of $F_{x,x}$ will affect the settling time of the system. To show that, a similar plot with an isolated 2.4-ms step is presented in Fig. 15. In a large step the initial value of $F_{x,x}(D - \Delta)$ is small, and therefore, the slope of the curve is also small. And, as Δ gets closer to D, $F_{x,x}(D - \Delta)$ approaches its maximum, and the slope of the TDE increases.

Both properties of the step response, namely, the delay-amplitude-dependent response time and its overall shape, are captured by the delay-domain model.

E. Response to a Sinusoidally Varying Intersignal Time Delay

Next, sinusoidal variations on the intersignal time delay D of the analog signals are applied to verify the steady-state response of the PCC-ATDE. As in bang–bang

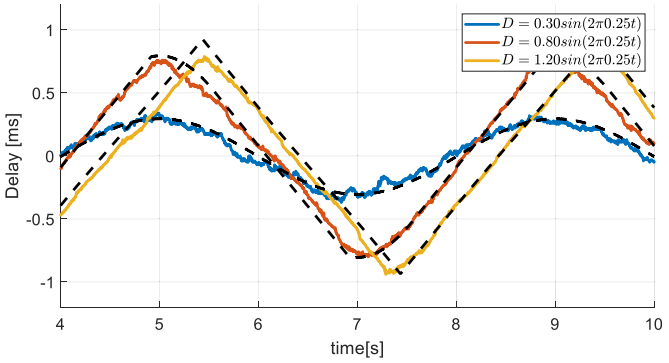

Fig. 16. Steady-state response of the PCC-ATDE for sinusoidal delay inputs with three different amplitude and same frequency. Colored continuous lines represent experimental data, and dashed lines are simulated using the delay-domain model. The response to high-amplitude signals are distorted due to the slope limitation.

PLLs [19], or slew-limited amplifiers [20], the slope limitation cause by the non-linear element is expected to affect the steady-state response of the PCC-ATDE as well. In the steady state, assuming that PCC-ATDE is able to track the input, the loop operates around $\Delta[n] - D[n] = 0$, allowing us to predict the maximum increment the loop is able to track

$$max(|\Delta[n] - \Delta[n-1]|) = \frac{1}{G} \cdot |F_{x,x}(0)|. \tag{21}$$

The loop can handle a high-frequency signal with low amplitude, but will distort large, low-frequency signals. If we assume a sinusoidal input for the intersignal delay, we calculate maximum amplitude-frequency product before we reach slope saturation with

$$max(2\pi \cdot f_{delay} \cdot A_{delay}) = \frac{1}{G} \cdot |F_{x,x}(0)|. \tag{22}$$

In Fig. 16, the outputs of three 0.25-Hz sinusoidal intersignal time-delay inputs with different amplitudes are shown. For this simulation, $G = 4$ and $|F_{x,x}(0)| = 0.004\ T_{\text{LSB}}$. The continuous lines are measurement data, and the dashed lines are the results simulated with the delay-domain model. Only the signal with $A_{\text{delay}} = 0.3$ ms is correctly tracked by the PCC-ATDE; the slope overload clearly distorts the other two responses where the delay-amplitude-frequency product exceeds 0.001 s-rad. The 0.001-s-rad/s limit suffices for the vehicle tracking application requirements of 1-ms maximum delay and 1-rad/s maximum angular speed. The delay-domain model is also able to faithfully capture the steady-state response of the loop.

Using (22), we can find the power-bandwidth of the PCC-ATDE, PBW, for a maximum allowable, rail-to-rail amplitude A_{MAX}

$$PBW = \frac{1}{G} \cdot |F_{x,x}(0)| \cdot \frac{1}{2\pi \cdot A_{MAX}}. \tag{23}$$

V. CMOS Prototype Implementation

The block diagram of the CMOS prototype of the time-discrete implementation is presented in Fig. 17. The chip has four analog inputs that are connected to a microphone

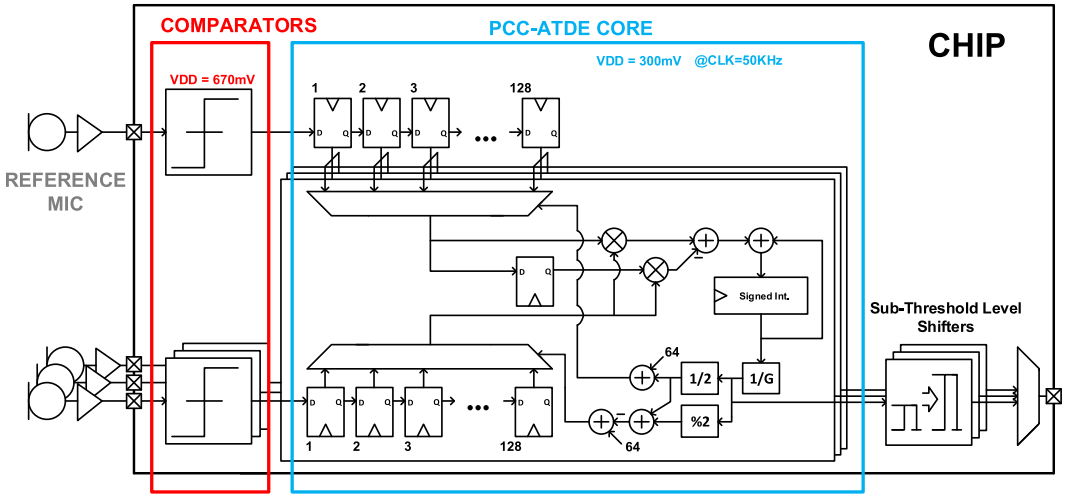

Fig. 17. Block diagram of the CMOS prototype. The system has two main blocks: four latched comparators that receive the input from the microphones; and the ultra-low-power core that outputs three digital time-delay values. Sub-threshold level shifters interface the low-voltage signals with the digital I/O pads.

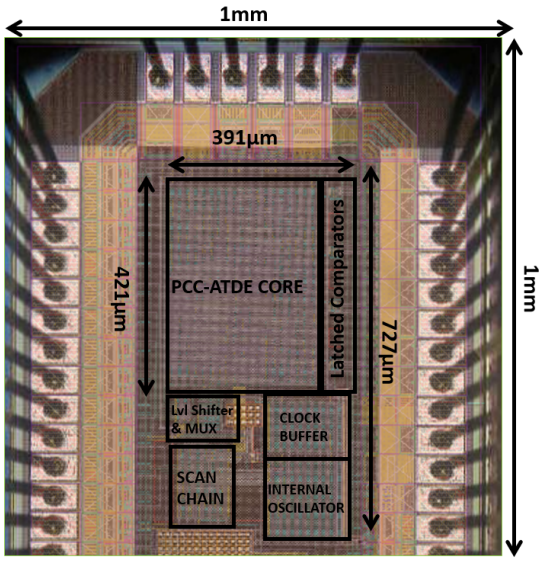

Fig. 18. Microphotograph of the PCC-ATDE CMOS prototype.

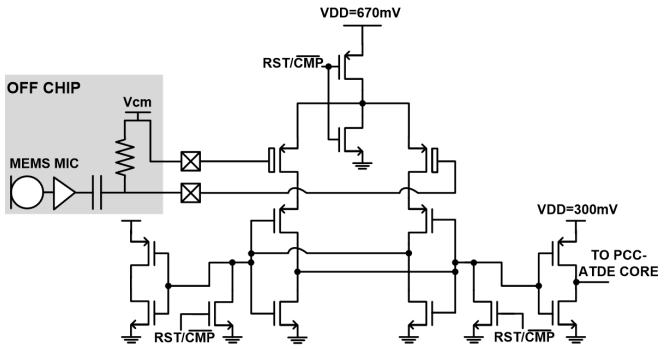

Fig. 19. Schematic of the latched comparator. In the shaded, the microphone connection to the circuit is shown.

array. One of the microphones provides the reference for the TDE; the chip outputs the time-delay of the other three analog signals with relation to this reference microphone.

Fig. 18 shows the die photograph of the PCC-ATDE prototype in a standard 0.18-μm CMOS technology with a total area of 1 mm^2. Digital blocks were synthesized from sub-threshold CMOS logic cells, and the overall power consumption of the three-channel PCC-ATDE time-delay estimator is 78.2 nW.

A. Front-End Comparator

The latched comparator shown in Fig. 19 is based on [22] and [23]. It does not consume power when in reset mode, leading to the total power consumption of 3.1 nW per comparator from 670 mV when operated at 50 kHz. The front end is designed with thick-oxide transistors for better electrostatic discharge (ESD) robustness. It supports differential inputs, but, to simplify the integration with the off-the-shelf microphones and pre-amplifiers, it was used in a single-ended fashion in the final system.

B. Ultra-Low-Power Processing Core

A 0.18-μm CMOS technology was selected for its low-leakage current, while easily meeting the speed and density requirements. The core of the PCC-ATDE was synthesized with sub-threshold CMOS logic [23], [24]. Reducing the power supply voltage helps to decrease both dynamic and static power consumption.

The variable delay cells are implemented with chains of 128 flip-flops and multiplexers. This number of flip-flops sets the maximum delay range to $\pm 128\ T_{\mathrm{LSB}}$. More flip-flops would increase the delay range, but it would also require an output with more than 8 bits to control the multiplexers. The delay is chosen by selecting the output of one of the stages of the flip-flop chain. After the 1-bit quantization, the multiplications are easily computed with XOR logic gates. An extra flip-flop, placed after the upper multiplexer, provides the fixed delay τ_{fix}. A 10-bit register and adder are used as an accumulator, and the 1/G attenuation is realized with arithmetical shifts of 0, 1, or 2 bits. The output is divided

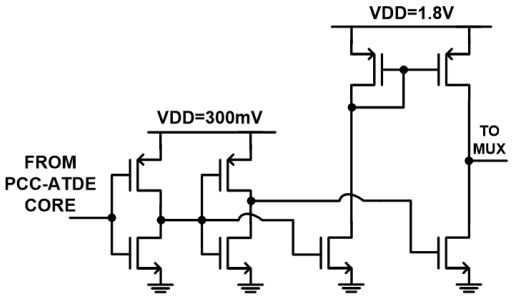

Fig. 20. Schematic of the sub-threshold level shifter used to convert the 300-mV digital signal from the ultra-low-power core to the 1.8-V level of the I/O pads.

by two to evenly distribute $\Delta[n]$ to both variable delay cells. In case of an odd $\Delta[n]$, one is added to the value of the lower variable delay cell.

C. Sub-Threshold Level Shifter

For experimental purposes, the 300-mV signals from the PCC-ATDE core are converted to 1.8-V I/O levels with the sub-threshold level shifters shown in Fig. 20. The current-mirror level shifters [25] guarantee the conversion of the sub-threshold logic signals. In a fully integrated system implementation, these level shifters are not required, and hence, their power consumption has not been included in the functional power consumption.

VI. Experimental Characterization of the PCC-ATDE Operation

For the experimental performance characterization, arbitrary waveform generators (AWGs) are used to provide the analog inputs. The AWGs output a 600-mVpp 60–200-Hz band-limited noise signal to simulate the sound of approaching vehicles. All AWGs are synchronized and under software control so that the delay between all channels can be precisely set for accurate measurements. Similar measurements were made to obtain the plots in Figs. 14–16, but for the characterization plots, the system is operating at the optimal figure of merit (FOM) sampling frequency of $F_S = 50$ kS/s, as shown in Section VII-A.

A. Step Response

The first measurement, as shown in Fig. 21, is the step response of the PCC-ATDE with different attenuator G values. As it is shown inside the dashed-line box, the step function is in the intersignal time delay D between the analog inputs. Before the 5-s mark, input 2 is $D = -1$ ms delayed compared with the input 1; after the 5-s mark, the delay changes to $D = 1$ ms. The -1–1 ms step response varies from 514 ms when $G = 1$ to 2.05 s when $G = 4$. As detailed in Section IV-D, the step response is amplitude dependent, and a $\pm$1-ms step was chosen since it fits a reasonable microphone spacing of 35 cm for sound-source localization devices.

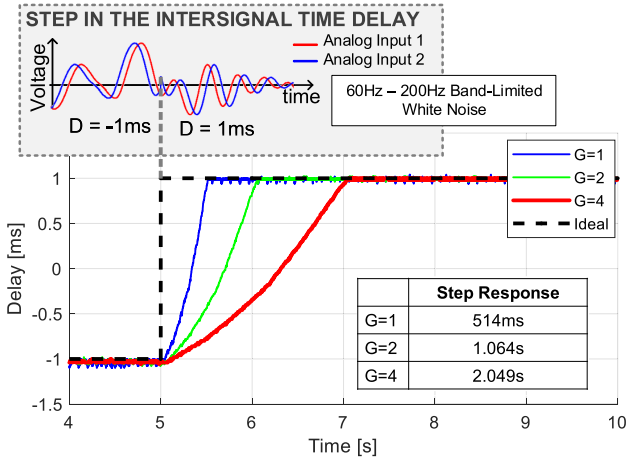

Fig. 21. Measured $\pm$1-ms step responses of the PCC-ATDE. As shown in the shaded area, the step is in the intersignal time delay, not in amplitude. The dashed lines show the delay switching from $D = -1$ ms to $D = 1$ ms at the 5-s mark. The step response for the same input changes with the attenuation value G from 514 ms to 2.049 s.

B. Steady-State Response

The non-linear, slew-limited behavior does not allow us to provide a straightforward number for the bandwidth of the PCC-ATDE. Instead, we are going to use the *power-bandwidth*, see (23), to characterize the system. The calculated power-bandwidth of the PCC-ATDE with a $\pm$2.52-ms range is 0.252, 0.126, and 0.063 Hz for $G = 1$, $G = 2$, $G = 4$, respectively.

C. Linearity

The rail-to-rail pure tone response is plotted in Fig. 22. Fig. 23 shows the spectrum of the measured signal to calculate the signal-to-noise-and-distortion ratio (SNDR) and extract the ENOB, which varies from 5.41 to 6.06 bits.

Static linearity tests have also been conducted. The y-axis of Fig. 24 are the measured digital codes obtained when the analog inputs are delayed by the corresponding value on the x-axis. The clocked nature of the algorithm helps it achieve a linear operation with peak integral nonlinearity (INL) of $-1.57/1.33$ least significant bit (LSB) and peak differential nonlinearity (DNL) of $-0.85/0.97$ LSB with $T_{\mathrm{LSB}} = 20$ μs without the need for calibration.

VII. Performance Comparison

A. Power Consumption FOM

To establish a metric that captures most of the design aspects of the PCC-ATDE, we use an FOM similar to the one used to compare ADCs

$$FOM = \frac{Power}{N \cdot 2^{ENOB} \cdot F_s} \tag{24}$$

where N is the number of channels in the time-delay-to-digital converter. The plot in Fig. 25 shows the contribution of the comparator and the ultra-low-power core to the FOM at different frequencies from 10 to 800 KHz. In each of the measurements, the power supply was adjusted to its minimum value to sustain normal operation at the given F_s. When the

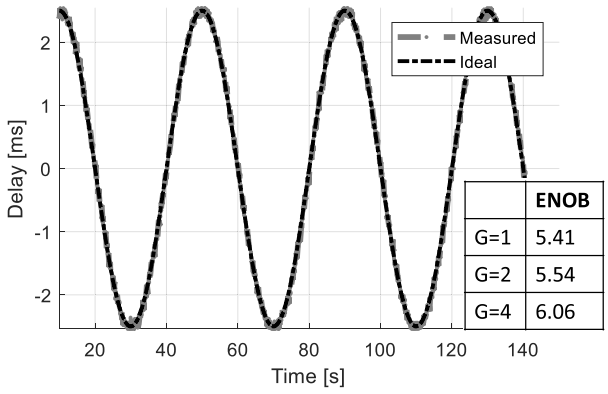

Fig. 22. Steady state measurement used for ENOB calculations. A low-frequency rail-to-rail input was used to avoid slope saturation as detailed in Section IV-E.

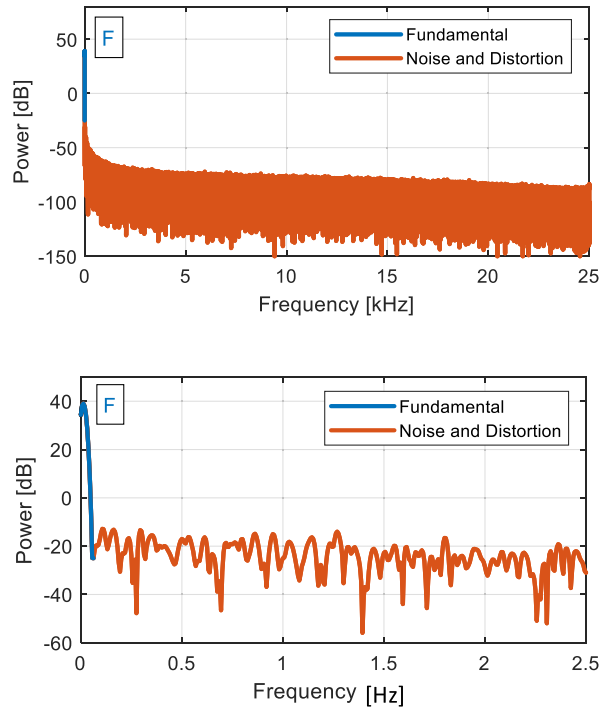

Fig. 23. Spectrum of the low-frequency rail-to-rail delay input used to calculate the ENOB. The top image is the full spectrum range from 0 Hz to 25 kHz. On the bottom, the same plot is zoomed in from 0 to 2.5 Hz. The measured SNDR is 38.22 dB when $G = 4$.

system is running at $F_s = 50$ kHz, the FOM reaches an optimal value of 7.84 fJ/Conv.-Step. Operating with higher clock frequencies reduces T_{LSB} and can be used to enable TDoA calculation for applications that need to cover smaller delay ranges and have higher analog inputs frequencies, such as ultrasound.

B. Comparison to the State-of-the Art

The LMS-TDE is the most commonly implemented adaptive TDE, yet, to the best of our knowledge, no silicon implementations are available in the open literature. Details on the performance of the LMS-TDE were extracted from simulations presented in [14]. Similar to the PCC-ATDE, the LMS-TDE uses feedback to perform the estimation of the time delay, but the algorithm still requires a front-end ADC

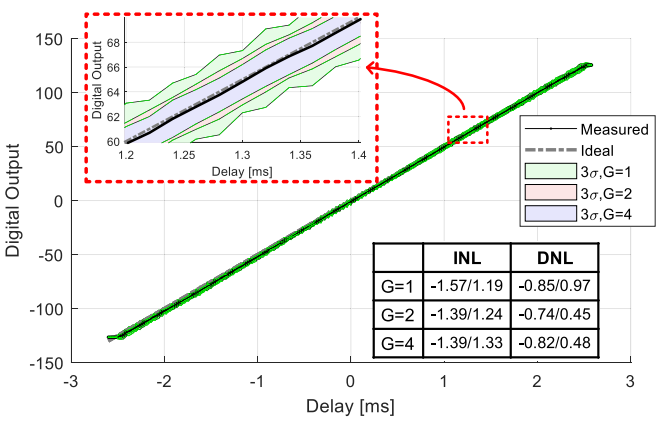

Fig. 24. Linearity plot of the PCC-ATDE. The continuous black line is the ideal linear response, the dashed gray lines are measurement data, and the colored areas around the plot are 3σ regions for different attenuation values G.

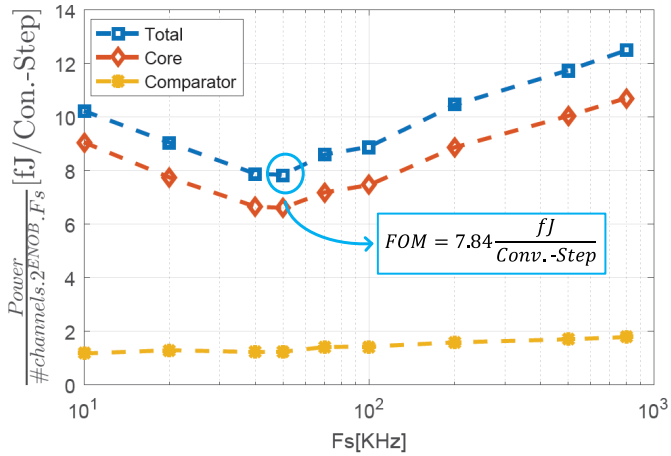

Fig. 25. FOM plot oversampling frequency. The selected FOM is plotted as the clock frequency of the system is changed. For each FOM measurement, the supply voltage of the blocks is adjusted to the minimum possible value to sustain correct operation.

to capture the audio from the microphones and significantly more arithmetical operations.

Cross correlation-based TDE is the conventional approach. In [12], in 0.18-μm CMOS with subsample ($T_{\text{LSB}} < 1/F_S$), TDE is presented. To store the audio frames and intermediate results from the algorithm, this solution is using 20 kBytes of memory, in contrast with the 257 D-type flip flop (DFF) required by the PCC-ATDE. The audio frame size in GCC solutions can be related to the step response of the system; since only when all the values of the frame are taken after the step in the delay, the output will have the correct value. The calculations also involve taking the fast Fourier transform (FFT) and inverse fast Fourier transform (iFFT) of 1024-point vectors that are much more complex than the basic XORs and adders present in the PCC-ATDE, as highlighted on Table I. Even without accounting for the power and area of the ADCs required to interface the microphones with the ASIC, the result is a normalized area more than 6× larger and a power FOM $10^5\times$ higher than the PCC-ATDE, as shown in Fig. 26.

TABLE I

COMPARISON TABLE OF SOUND-SOURCE LOCALIZATION LOW-POWER SOLUTIONS. HIGHLIGHTED ARE THE PARAMETERS IN WHICH THIS WORKS EXCEL: LOW ARITHMETICAL COMPLEXITY; LOW POWER PER CONVERSION STEP; AND SMALL NORMALIZED AREA

SOUND SOURCE LOCALIZATION SOLUTIONS				
Method	LMS-ATDE	GCC-PHAT	Silicon Cochlea	PCC-ATDE
Reference	[14]	[12]	[27]	[This Work]
Technology	**Simulation**	180 nm CMOS	350 nm CMOS	180 nm CMOS
Sampling Frequency	1 kS/s	20 kS/s	~20 k eps	50 kS/s
Interface with Analog Inputs	8-bit ADC	8-bit ADC	BPF bank, Half-wave Rectifier, Pulse-frequency Modulator	Comparators
Memory Required per TDE	31Bytes	20 kBytes	-	257 Flip-Flops, 1 10bit register
Arithmetical Complexity per ATDE	362 Additions, 373 Multiplications	2 1024-point FFT, Maximum Likelihood Search	Peak Seach on Events Matching Histogram (~150 kOperations/s)	2 XORs, 4 Adders, 1 Arithm. Shifter
Range	-	[-1.5 ms,1.5 ms]	[-700 µs,700 µs]	[-2.52 ms,2.52 ms]
T_{LSB}	1 ms	5 µs	10 µs	20 µs
Step Response	2 s to 20 s	12.8 ms to 51.2 ms	-	514 ms to 2.05 s
Number of TDE Channels (N)	1	1	1	3
TDE ENOB	-	-	-	6.06 bits
Total Power	-	28.98 mW	357 µW	78.2 nW
TDE Energy per Conversion Step per Channel [1]	-	~2.4 nJ/Conv.-Step	~127 pJ/Conv.-Step	7.84 fJ/Conv.-Step
Area		6.25 mm²	16.2mm²	1mm²
Normalized Area [2]	-	193 µ(mm/nm)²	132 µ(mm/nm)²	31 µ(mm/nm)²
Portable to Off-the-Shelft Solutions	Yes	Yes	No	Yes

1. $FOM = \frac{POWER_{TOTAL}}{N.2^{ENOB}.F_S}$ 2. $A_{norm} = \frac{A_{TOTAL}}{\lambda^2}$

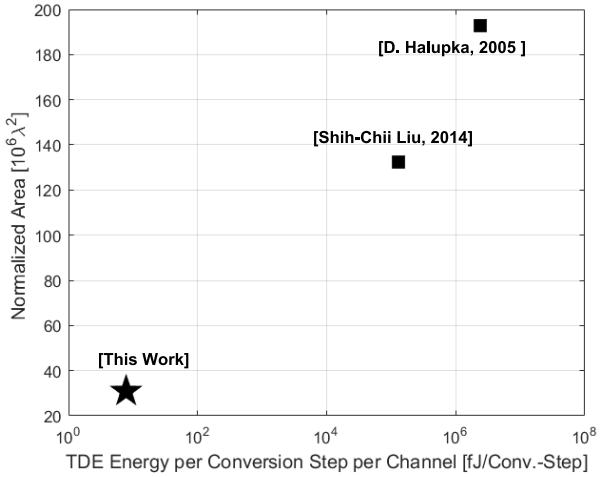

Fig. 26. Comparison plot of previous sound-source localization solutions. The power FOM of the solutions is plotted versus the normalized area of the prototypes.

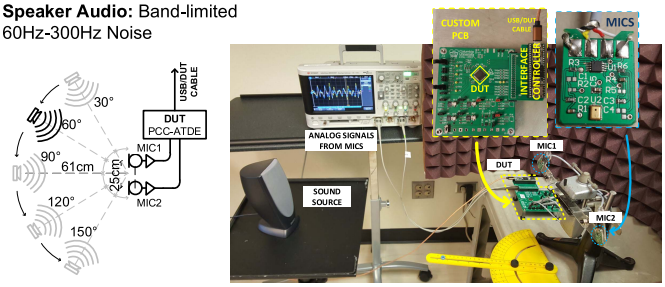

Fig. 27. Setup used to compare the performance of the PCC-ATDE prototype and traditional TDE techniques. On the left, the diagram shows how the sound-source rotates the microphone pair. And on the right is a photograph of the corresponding setup.

In [26], a Binaural silicon cochlea that uses an address-event representation (AER) to estimate the time delay between microphones is presented. The AER also provides information on spectral content, and a more recent article on AER silicon cochlea has been published in [15] and [27], but [26] has more details on its sound-source localization performance. The silicon cochlea needs a significant amount of operations to convert the AER into a TDE. Combined with the AER circuitry the solution is still more than 4× lager than the PCC-ATDE and has a power FOM $10^4\times$ worse.

VIII. SOUND-SOURCE LOCALIZATION EXPERIMENTS

A. Sound-Source Localization in a Controlled Environment

All previous experiments prove that the PCC-ATDE is able to extract the intersignal time delay from two analog inputs. But, in order to deploy this technique in a sound-source localization system, we needed to verify if the second-order effects, such as reverberation or mismatches in microphone responses, would affect the operation of the system. Experiments with microphones and speakers were conducted to the PCC-ATDE with real-life audio inputs.

*1) **TDE Performance Comparison**:* In this experiment, we compared the results of TDEs using the PCC-ATDE prototype and a standard GCC-PHAT approach. In this experiment (Fig. 27), a single-sound source playing a band-limited white-noise recording is placed near a microphone pair. The sound source is rotated around the microphone pair, and, for each angular position, the delay estimations from the PCC-ATDE device under test (DUT) and 400-ms frames from both analog signals are collected. The experiment was conducted in a 4 m × 6 m closed room without acoustic isolation, but, when the microphones were close to the walls, a piece of acoustic foam was used to reduce the reverberation.

The result of the experiment is shown in Fig. 28. The delays acquired with the PCC-ATDE DUT are plotted for each incidence angle and compared with the delays obtained using a GCC-PHAT algorithm with the collected frames; the results match closely, with an RMS error of 37.2 μs, or 2.3%.

*2) **Two Audio Sources** :* Next, we investigate the effect of a second sound source on the TDE. We now place two speakers next to the microphone pair, as shown in Fig. 29. The speakers are at different positions with a distinct time difference of arrivals. The ratio of power of the speakers is swept from −20 dB to 20 dB, and the resulting TDE is plotted in Fig. 30. If D_A is the expected TDE if only source A was present, and D_B the delay if only B is present, the final result leads to an empirical observation that the resulting TDE (D_{out}) can be expressed by the average of each individual delay weighted by their relative power

$$D_{out} = \frac{D_A.P_A + D_B.P_B}{P_A + P_B}. \tag{25}$$

B. Estimating the Bearing of an Approaching Vehicle on a City Street

Finally, the PCC-ATDE DUT is integrated into an IoT system for a vehicle bearing estimation. The four microphones are

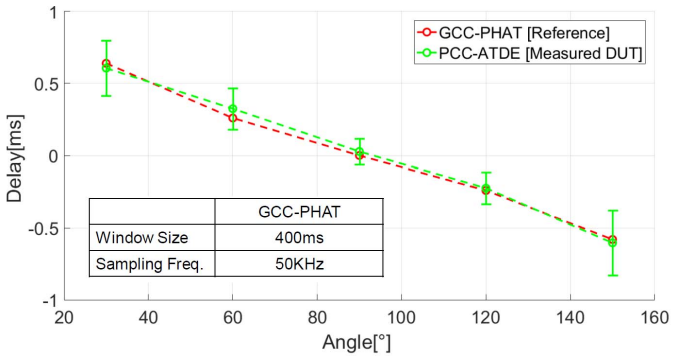

Fig. 28. TDE results from the PCC-ATDE, in green, and from GCC-PHAT, in red. The GCC-PHAT TDE was done using 400 ms sampled at 50 kS/s by the oscilloscope.

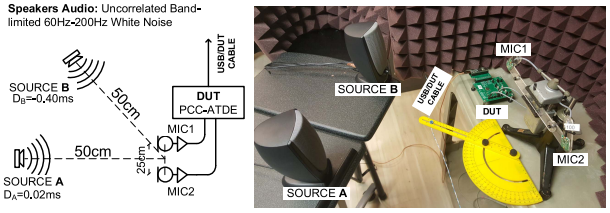

Fig. 29. Setup used to measure the effectiveness of an interfering sound source in the PCC-ATDE estimation. The diagram on the left shows two sound sources with unique TDoA playing uncorrelated recordings with different power.

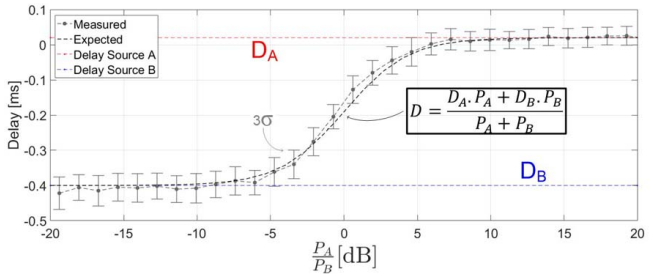

Fig. 30. Measured TDE of the PCC-ATDE prototype as a function of the relative power of two interfering sound sources. Inside the box is an empirical expression for the resulting delay.

placed in a pyramid structure and connected to the DUT, and outputs of the time-delay-to-digital converter are connected to a host computer. Fig. 31 shows a photograph of the setup placed in a street in New York, where the experiment was conducted. The system was placed in a one-way street that connects two busy avenues and measurements were conducted during business hours on a weekday. The time difference of arrival of the vehicle's noise to each microphone varies as the car moves from the right to the left of the array. This is captured by the three extracted intersignal TDE of the PCC-ATDE plotted in Fig. 32.

On the host computer, a k-nearest neighbor (KNN) [28] machine-learning classifier is used to convert the time delays into incidence angles in real-time. Even though the classifier was trained indoors with car sound recordings playing at different incidence angles from the array, it was able to track the vehicle accurately in the outdoors setting. Fig. 33 shows

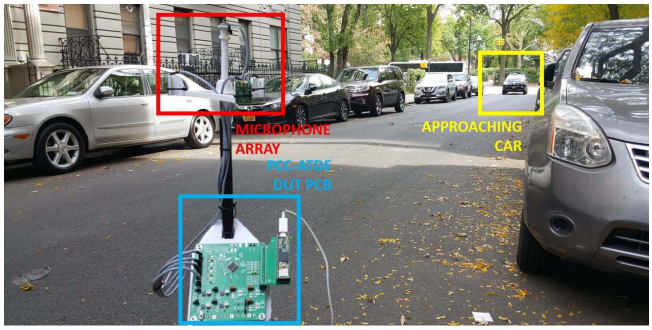

Fig. 31. Vehicle-bearing experiment. Inside the yellow box is the approaching car that the system is detecting; the red box marks the pyramid microphone array, and the blue box shows the PCC-ATDE DUT PCB.

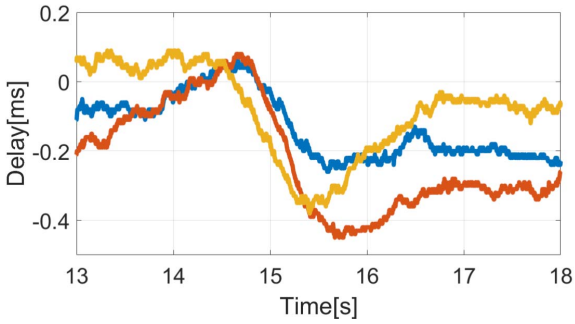

Fig. 32. Measured three channels TDE using the PCC-ATDE embedded setup. The measured delays are caused by an approaching car noise in the experiment shown in Fig. 31. The reference microphone is the one closest to the street. When the car crosses the setup between 15 and 16 s, the TDoA from the reference to the other microphones reaches their minimum.

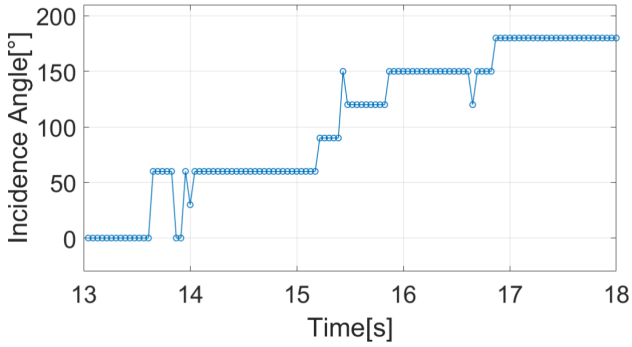

Fig. 33. Output of the KNN classifier for the vehicle bearing estimation. With the time delay captured in Fig. 32 the classifier predicts the incidence angle of the sound waves, 0° for a wave hitting from the right and 180 ° from the left.

the recorded real-time output of the KNN classifier changing from 0° when the car is on the right side of the microphone array to 180° after it crosses to the left in with a 30° resolution.

IX. Conclusion

By replacing the traditional ADC and DSP processing chain for feature extraction with an analog-to-feature converter, we were able to reduce the power consumption by four orders of magnitude less than conventional techniques to less than 100 nW for the presented sound-source localization application. Our analog-to-feature converter, prototyped

in 0.18-μm CMOS, successfully estimates the time-delay under all tested conditions. We carefully modeled and analyzed the behavior of the proposed PCC-ATDE loop, and introduced a delay-domain model that accurately predicts the behavior of the time-delay estimator. This mixed-signal analog-to-feature conversion approach is promising for resource-constrained solutions in CPSs or IoT applications, particularly for always-on battery-powered systems. Even though the PCC-ATDE was demonstrated in a sound-source localization system, the technique can be applied to other systems that rely on TDE, and it is particularly effective when the required T_{LSB} is very small, and the frequency of the analog input signals is relatively low.

ACKNOWLEDGMENT

The authors would like to thank Prof. Fred Jiang and Prof. S. Xia from ICSL, Columbia University, New York, NY, USA, for the valuable discussions on the embedded implementation of sound-source localization solutions. They would also like to thank Prof. M. Seok and Prof. J. P. Cerqueira from VLSI Laboratory, Columbia University, for their assistance with the ASIC integration.

REFERENCES

[1] K. Shaver. (Sep. 2014). *Safety Experts to Pedestrians: Put the Smartphones Down and Pay Attention*. [Online]. Available: https://www.washingtonpost.com/local/trafficandcommuting/safety-experts-to-pedestrians-put-the-smartphones-down-and-pay-attention/

[2] G. P. Hancke, B. de Carvalho e Silva, and G. P. Hancke, "The role of advanced sensing in smart cities," *Sensors*, vol. 13, no. 1, pp. 393–425, 2012.

[3] S.-L. Jeng, W.-H. Chieng, and H.-P. Lu, "Estimating speed using a side-looking single-radar vehicle detector," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 607–614, Apr. 2014.

[4] Z. J. Chong, B. Qin, T. Bandyopadhyay, M. H. Ang, E. Frazzoli, and D. Rus, "Synthetic 2D LIDAR for precise vehicle localization in 3D urban environment," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2013, pp. 1554–1559.

[5] M. Bertozzi, A. Broggi, A. Fascioli, and S. Nichele, "Stereo vision-based vehicle detection," in *Proc. IEEE Intell. Vehicles Symp.*, Oct. 2000, pp. 39–44.

[6] C. T. Barba, M. Á. Mateos, P. R. Soto, A. M. Mezher, and M. A. Igartua, "Smart city for VANETs using warning messages, traffic statistics and intelligent traffic lights," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 902–907.

[7] D. de Godoy *et al.*, "PAWS: A wearable acoustic system for pedestrian safety," in *Proc. IEEE/ACM 3rd Int. Conf. Internet–Things Design Implement.*, Apr. 2018, pp. 237–248.

[8] S. Jeong *et al.*, "A 12nW always-on acoustic sensing and object recognition microsystem using frequency-domain feature extraction and SVM classification," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2017, pp. 362–363.

[9] R. Sarpeshkar *et al.*, "An ultra-low-power programmable analog bionic ear processor," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 4, pp. 711–727, Apr. 2005.

[10] N. Bhave and P. Rao, "Vehicle engine sound analysis applied to traffic congestion estimation," in *Proc. Int. Symp. Comput. Music Multidisciplinary Res. (CMMR)*, 2011, pp. 1–5.

[11] A. A. Handzel, S. B. Andersson, M. Gebremichael, and P. Krishnaprasad, "A biomimetic apparatus for sound-source localization," in *Proc. 42nd IEEE Int. Conf. Decis. Control*, vol. 6, Dec. 2003, pp. 5879–5884.

[12] D. Halupka, N. J. Mathai, P. Aarabi, and A. Sheikholeslami, "Robust sound localization in 0.18 μm CMOS," *IEEE Trans. Signal Process.*, vol. 53, no. 6, pp. 2243–2250, Jun. 2005.

[13] J. DiBiase, H. Silverman, and M. Brandstein, "Robust localization in reverberant rooms," in *Microphone Arrays*, M. Brandstein and D. Ward, Eds. Springer, 2001, ch. 8, pp. 157–180.

[14] H. C. So and P. C. Ching, "Comparative study of five LMS-based adaptive time delay estimators," *IEE Proc.-Radar, Sonar Navigat.*, vol. 148, no. 1, pp. 9–15, Feb. 2001.

[15] M. Yang, C.-H. Chien, T. Delbruck, and S.-C. Liu, "A 0.5 V 55 μW 64 $\times$ 2 channel binaural silicon cochlea for event-driven stereo-audio sensing," *IEEE J. Solid-State Circuits*, vol. 51, no. 11, pp. 2554–2569, Nov. 2016.

[16] D. de Godoy, X. Jiang, and P. R. Kinget, "A 78.2nW 3-channel time-delay-to-digital converter using polarity coincidence for audio-based object localization," in *Proc. IEEE Custom Integr. Circuits Conf.*, Apr. 2018, pp. 1–5.

[17] A. Fertner and A. Sjolund, "Comparison of various time delay estimation methods by computer simulation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 34, no. 5, pp. 1329–1330, Oct. 1986.

[18] J. H. V. Vleck and D. Middleton, "The spectrum of clipped noise," *Proc. IEEE*, vol. 54, no. 1, pp. 2–19, Jan. 1966.

[19] S. Tertinek, J. P. Gleeson, and O. Feely, "Statistical analysis of first-order bang-bang phase-locked loops using sign-dependent random-walk theory," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 9, pp. 2367–2380, Sep. 2010.

[20] P. Allen, "A model for slew-induced distortion in single-amplifier active filters," *IEEE Trans. Circuits Syst.*, vol. 25, no. 8, pp. 565–572, Aug. 1978.

[21] J. Craninckx and G. Van der Plas, "A 65fJ/conversion-step 0-to-50MS/s 0-to-0.7mW 9b charge-sharing SAR ADC in 90nm digital CMOS," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2007, pp. 246–600.

[22] D. Zhang, A. Bhide, and A. Alvandpour, "A 53-nW 9.1-ENOB 1-kS/s SAR ADC in 0.13-μm CMOS for medical implant devices," *IEEE J. Solid-State Circuits*, vol. 47, no. 7, pp. 1585–1593, Jul. 2012.

[23] M. Seok, D. Sylvester, and D. Blaauw, "Optimal technology selection for minimizing energy and variability in low voltage applications," in *Proc. Int. Symp. Low Power Electron. Design*, 2008, pp. 9–14.

[24] M. Alioto, "Ultra-low power VLSI circuit design demystified and explained: A tutorial," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 1, pp. 3–29, Jan. 2012.

[25] S. N. Wooters, B. H. Calhoun, and T. N. Blalock, "An energy-efficient subthreshold level converter in 130-nm CMOS," *IEEE Trans. Circuits Syst. II, Express Briefs*, vol. 57, no. 4, pp. 290–294, Apr. 2010.

[26] S.-C. Liu, A. van Schaik, B. A. Minch, and T. Delbruck, "Asynchronous Binaural Spatial Audition Sensor With 2$\times$64$\times$4 channel output," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 4, pp. 453–464, Aug. 2014.

[27] J. Anumula, E. Ceolini, Z. He, A. Huber, and S.-C. Liu, "An event-driven probabilistic model of sound source localization using cochlea spikes," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2018, pp. 1–5.

[28] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerg. Artif. Intell. Appl. Comput. Eng.*, vol. 160, pp. 3–24, Jun. 2007.

Daniel de Godoy (M'15) was born in Recife, Brazil, in 1988. He received the B.Eng. degree in electrical engineering from the Federal University of Pernambuco, Recife, in 2012, and the M.S. degree in electrical engineering and the Ph.D. degree in electrical engineering with a focus on ultra-low-power analog feature extraction front-end integrated circuits for machine-learning audio-based systems from Columbia University, New York, NY, USA, in 2015 and 2019, respectively.

In 2018, he was with Silicon Laboratories Inc., Austin, TX, USA, where he is currently with the Timing division. His research interest includes ultra-low-power analog and mixed-signal sensor interfaces and their roles in embedded IoT systems.

Dr. de Godoy was a recipient of a Columbia Electrical Engineering Department Research Award from Columbia University. He was also a recipient of the Science Without Borders Fellowship from CAPES, Brasilia, Brazil, and the Lemann Foundation Fellowship.

Peter R. Kinget (M'90–SM'02–F'11) received the Engineering degree *(summa cum laude)* in electrical and mechanical engineering and the Ph.D. degree in electrical engineering *(summa cum laude)* (with Congratulations of the Jury) from the Katholieke Universiteit Leuven, Leuven, Belgium, in 1990 and 1996, respectively.

From 1991 to 1995, he received a Graduate Fellowship from the Belgian National Fund for Scientific Research. From 1991 to 1995, he was a Research Assistant with the ESAT-MICAS Laboratory, Katholieke Universiteit Leuven. From 1996 to 1999, he was a Member of Technical Staff with the Design Principles Department, Bell Laboratories, Lucent Technologies, Murray Hill, NJ, USA. From 1999 to 2002, he held various technical and management positions in IC design and development with Broadcom, Irvine, CA, USA, CeLight, Iselin, NJ, USA, and MultiLink, Somerset, NJ, USA. In 2002, he joined Columbia University, New York, NY, USA, where he is currently the Bernard J. Lechner Professor and the Chair of the Department of Electrical Engineering. From 2010 to 2011, he was on sabbatical leave with the Université catholique de Leuven, Leuven. He also serves as an expert on patent litigation and a technical consultant to industry. His research interests are in analog, RF, and power integrated circuits and the applications they enable in communications, sensing, and power management. He has widely published in circuits and systems journals and conferences and coauthored three books. He holds 32 U.S. patents with several applications under review. His research group has received funding from the National Science Foundation, Semiconductor Research Corporation, the Department of Energy, the Department of Defense, and an IBM Faculty Award. The group has further received in-kind and grant support from several of the major semiconductor companies.

Dr. Kinget was an IEEE Distinguished Lecturer of the Solid-State Circuits Society from 2009 to 2010 and from 2015 to 2017. He was an Elected Member of the IEEE Solid-State Circuits Adcom from 2011 to 2013 and from 2014 to 2016. He was a co-recipient of several awards including the Best Student Paper Award First Place at the 2008 IEEE Radio Frequency Integrated Circuits (RFIC) Symposium, the First Prize at the 2009 Vodafone Americas Foundation Wireless Innovation Challenge, the Best Student Demo Award at the 2011 ACM Conference on Embedded Networked Sensor Systems, the 2011 IEEE Communications Society Award for Advances in Communications for an outstanding article in any IEEE Communications Society publication in the past 15 years, the First Prize ($100K) at the 2012 Interdigital Wireless Innovation Challenge, the Best Student Paper Award Second Place at the 2015 IEEE RFIC Symposium, the Best Poster Award at the 2015 IEEE Custom Integrated Circuits Conference, and the Best Student Paper Award Third Place at the 2018 IEEE RFIC Symposium. He has served as a member of the Technical Program Committee for the IEEE Custom Integrated Circuits Conference from 2000 to 2005 and from 2016 to 2018, the Symposium on VLSI Circuits from 2003 to 2006, the European Solid-State Circuits Conference from 2005 to 2010, and the International Solid-State Circuits Conference from 2005 to 2012. He was an Associate Editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS from 2003 to 2007 and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II from 2008 to 2009.