# Ultra Low Power Event-Driven Sensor Interfaces

# Ultra Low Power Event-Driven Sensor Interfaces

Pieter Harpe, Yuting Shen, Hanyue Li, Kevin Pelzers, Haoming Xin and Eugenio Cantatore
Integrated Circuits Group, Eindhoven University of Technology, Eindhoven, The Netherlands
Email: p.j.a.harpe@tue.nl

***Abstract*— This paper reviews several examples of ultra low power sensor interfaces for IoT applications. In such applications, the sensing operation is often performed at a relatively low frequency, and sometimes it is heavily duty-cycled, or it should be triggered by particular events or thresholds. The paper reviews why dynamic sensor interface architectures are a good choice in this context, and gives several design examples that can operate dynamically and that can be triggered by a single clock pulse. Suitable ADC design strategies are explained, and two exemplary sensor interfaces are described: a capacitive sensor interface, and a resistor-based temperature sensor interface including analog correction techniques. Both designs are reviewed and the main features in terms of efficiency and performance are discussed.**

## I. Introduction

Sensor interfaces appear in all kinds of forms and for a large variety of applications in electronic systems: from high-precision industrial or medical sensors, to high-bandwidth ultrasound or RF sensors, to high data-rate large-array image sensors, down to low-speed environmental sensors. The focus of this paper is on environmental sensors for Internet-of-Things (IoT) applications, where power consumption is often critical due to battery constraints. Instead of giving a broad literature overview, this work summarizes some of the basic challenges and design approaches to minimize power consumption, and it recapitulates a few sensor interface examples following the same basic architectural principles.

While there are many different types of environmental sensors that can be of interest for IoT, such as pressure sensors, humidity sensors, and light sensors, Fig. 1 shows a literature survey of temperature sensors [1] as an example. As can be expected, temperature sensor interfaces with a finer (i.e.: better) resolution in degrees Kelvin tend to consume more energy per conversion. In fact, based on the resolution Figure-of-Merit (FoM) [1], a tenfold improvement in resolution will cost 100× as much energy, due to the inherent scaling of a circuit's energy consumption as function of its signal-to-noise ratio (SNR). As a result, it can be observed in the figure that high-precision sensors (resolution better than 1 mK) may consume an energy in the order of $\mu$J's per measurement, while sensors with relaxed resolution can have a consumption below 1 pJ. Even though the figure only considers temperature sensors, a similar trend also applies to other sensor applications. Considering the scope of this work is for energy-constrained IoT applications, the main challenge that will be discussed is how to minimize the energy consumption of the sensor interface down to the pJ level, while maintaining an acceptable resolution.

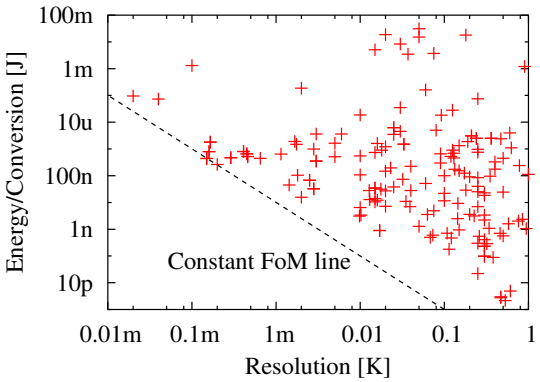


Fig. 1. Temperature sensor survey. Data from [1].

The organization of this paper is as follows: Section II describes the overall system considerations for a low-power sensor readout interface, and proposes to use dynamic readout circuitry. Next, Section III studies which ADC architectures are most suitable in such a scenario, and it describes the key challenges to minimize the power consumption at the system level. After that, having discussed the generic components, Sections IV and V will recapitulate a few concrete design examples for capacitive and resistive sensor interfaces, respectively. Lastly, conclusions are drawn in Section VI.

## II. Sensor Readout Considerations

A simplified (abstract) view of a sensor readout interface is shown in Fig. 2. The transducer element converts the physical parameter of interest (such as temperature or humidity) to an electrical quantity (such as current, resistance, or capacitance). At the end of the system, an Analog-to-Digital Converter (ADC) is used to convert the sensed parameter to the digital domain. Usually, some form of interface is required between these blocks. Firstly, because most sensors are passive and thus some form of biasing or excitation is required to obtain a response that can be digitized. Secondly, the interface may help to amplify the sensed signal to ease the digitization process. In practice, some implementations use the structure as shown in the figure, but it is also possible that the functions are highly intertwined.

In terms of sensor challenges, a first aspect is that many transducers have a relatively high nominal value, while they have a relatively small variation as function of the sensed

Yuting Shen is now with NXP Semiconductors, Eindhoven, The Netherlands. Hanyue Li is now with imec, Leuven, Belgium. Haoming Xin is now with imec, Eindhoven, The Netherlands.

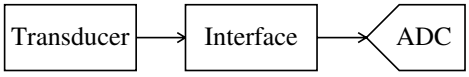


Fig. 2. Sensor, interface and ADC.

parameter. This implies that only a fraction of the "full range" is used. For instance, a temperature-sensitive resistance as can be integrated in modern CMOS technologies may have a variation of only 10% to 20% over a temperature range of 100 °C. Besides that, the transducer elements also have spread due to manufacturing tolerances. For instance, for on-chip resistors, the random mismatch and process corners can also account for a variation in the order of 10% to 20%. Due to these effects, the sensor interface often needs to accommodate a much larger range than the useful range, and calibration is necessary to counteract manufacturing variations.

In the context of environmental monitoring for IoT, the sampling rate can often be relatively low. Dependent on what is being measured, a sensing rate of 1 kHz (or even far below that) might be more than sufficient. Considering the energy per measurement may be as low as a few pJ, this implies that the expected *average* power consumption should be in the order of nW's. With classical analog design, using static biasing techniques, such a power consumption level is not easily reachable due to leakage and high variability in deep sub-threshold operation. For that reason, it becomes more attractive to use circuits with dynamic operation, or to duty-cycle the parts of the circuitry that would otherwise use static biasing. In that way, the power consumption becomes fully dynamic, enabling further down-scaling at low frequencies.

A last consideration is the required clocking for the sensor interface. Many interfaces use oversampling to achieve good resolution with relatively relaxed hardware. While that is an effective and efficient solution, it is not always practical for IoT due to the required clock. For instance, if parameters are to be monitored while a large part of the system is in sleep, or if a measurement is performed only once in a while, it is much more convenient at the system level if this can be done with a single clock pulse. For that reason, interfaces that can operate from a single (or a limited) set of clock pulses are preferred over highly oversampled interfaces.

## III. Power efficient ADCs for dynamic sensor interfaces

To select an appropriate ADC architecture, a literature survey of ADCs is given in Fig. 3 [2], which shows the ADC energy consumption per conversion as function of the resolution, expressed in terms of Signal-to-Noise-and-Distortion Ratio (SNDR). The Schreier FoM trend line indicates ADCs with a similar efficiency. As such, it can be deducted that for medium resolutions (in the order of 50 to 80 dB, equivalent to approximately 8 to 13 bit of performance), Successive Approximation Register (SAR) ADCs or hybrid ADCs that are partially based on a SAR converter tend to be the most efficient solution. Also in an absolute sense, the SAR ADC achieves the lowest energy per conversion. On the contrary, for very high resolutions, other ADC architectures (most notably Sigma-Delta ADCs) tend to be a better alternative.

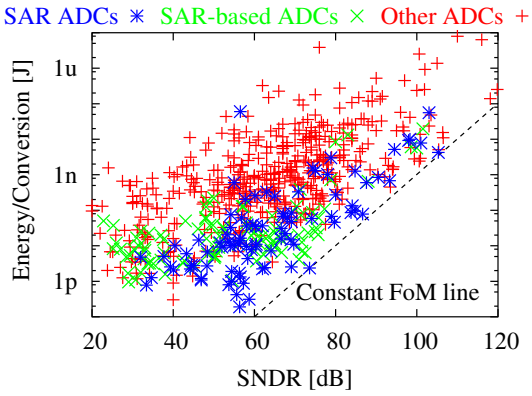


Fig. 3. ADC survey. Data from [2].

Apart from the high efficiency at medium resolutions, the SAR ADC has a few further advantages compared to alternative architectures. First, it is a Nyquist-rate converter and thus does not need oversampling. If the sequential conversion process is done by means of self-synchronization [3], [4], [5], a single external clock pulse is sufficient to perform the entire conversion, which is convenient for event-based IoT systems. Furthermore, commonly used SAR ADC implementations only use dynamic circuits and thus have no static power consumption (except for leakage), which enables a very low absolute power when operating at low sampling rates.

An example of a basic SAR ADC is shown in Fig. 4, where an N-bit differential charge-redistribution DAC, a single comparator, and digital logic compose the overall system. After sampling the input signal, the logic controls the SAR operation based on sequential comparisons to find the N-bit output code in N steps. To perform a binary search, the capacitors are binary-scaled multiples of a unit element ($C_u$).

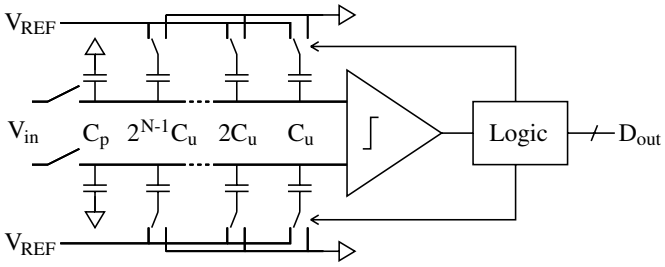


Fig. 4. Basic charge-redistribution SAR ADC topology.

While not strictly necessary, the reference voltage for the DAC ($V_{REF}$) is often set to the supply level ($V_{DD}$) for simplicity. Ideally that would result in an ADC with a rail-to-rail input range, but this range can be reduced by intentionally adding parasitic capacitance ($C_p$). In that way, the ADC range could be reduced to match the expected sensor output range.

Alternatively, an amplifier could be used to match the sensor range to the ADC range, but unless a dynamic-biased amplifier is used [6], that would obstruct the dynamic consumption of the system. Another challenge of an intermediate amplifier is that it could suffer from inaccurate gain and PVT variations, which may result in more complex calibration or compensation.

From system perspective, the ADC is best designed with the highest possible input impedance (i.e.: with the smallest possible value of the total DAC capacitance), since it minimizes loading and thus minimizes the consumption in the sensor frontend. However, the minimum capacitance is limited by the noise requirements ($kT/C$ noise due to sampling) and by capacitor mismatch, which induces non-linearity.

In order to make small-sized binary-scaled capacitors with decent matching, a unit-length capacitor layout technique was proposed in [7]. As sketched in Fig. 5, each capacitor $C_i$ is in fact composed of two elements, $C_i$ and $C_i$', that are nearly identical except for a length difference $2^i\Delta$. At schematic level, these two elements are always switched with an opposite signal polarity, which effectively creates a capacitance value that is defined by the difference $C_i - C_i$' of the two elements. The first advantage of this approach is that this can create very small effective capacitors, since $\Delta$ can be made much smaller compared to the total size of each metal strip. Secondly, one can now make accurate binary-scaled capacitors by scaling $\Delta$ rather than by connecting multiple capacitors in parallel. This results in a more area-efficient solution and reduced interconnect overhead. A drawback of the unit-length method for generic ADCs is that the total capacitance is much larger than the effective capacitance (since large elements are subtracted from each other). Effectively this results in a high value for $C_p$ and thus limits the full-scale range. However, in the context of a sensor interface with a limited expected signal swing, that is actually not an issue.

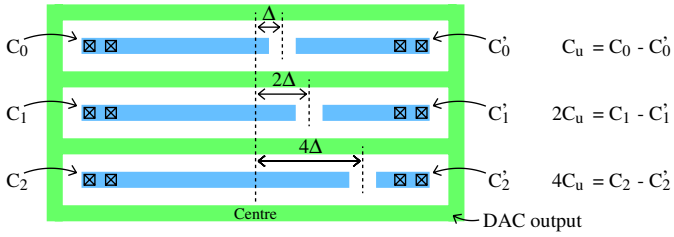


Fig. 5. Capacitive DAC implementation with unit-length elements [7].

## IV. A DYNAMIC CAPACITIVE SENSOR INTERFACE

According to the general principles explained in the previous sections, this section discusses the implementation of a fully dynamic capacitive sensor interface, also called Capacitance-to-Digital Converter (CDC). Some capacitive sensors, like MEMS-based pressure sensors, have a single capacitor element, of which the value changes as function of the applied pressure. On the other hand, for instance MEMS-based accelerometers, often have two capacitive elements, where one element increases in value while the other one decreases in value as function of the acceleration. To enable operation with both single-ended and differential sensors, the interface as shown in Fig. 6 was developed [8]. Either a differential sensor (indicated by $C_{d+}$, $C_{d-}$) is connected, or a single-ended sensor (indicated by $C_s$) is connected. In the second case, an on-chip reference element ($C_r$) is used, while $C_r$ is disconnected for differential sensors. The output of the capacitive divider (or half-bridge structure) is directly sampled by one of the two DAC capacitors in the ADC (since it has a differential implementation).

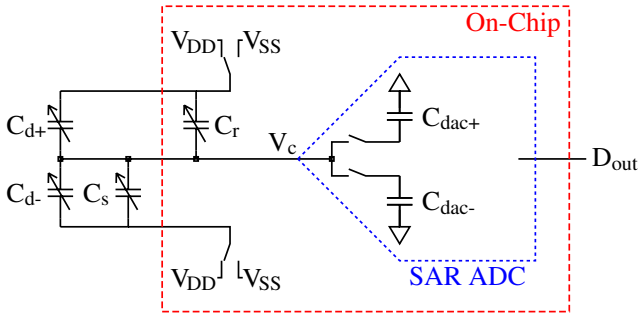


Fig. 6. Dynamic Capacitance-to-Digital-Converter from [8].

In terms of operation, all capacitors are first discharged by means of reset switches (not shown in the figure). Next, the supply voltage is applied to the sensor bridge, while the first DAC capacitance is connected. As a result, the voltage $V_c$ that is established and sampled on the DAC is a function of the value of the sensing capacitor, and it is also proportional to the applied supply voltage $V_{DD}$. In a second phase, the same procedure is repeated, but now the voltage polarity to the sensor bridge is reversed, while the second DAC capacitance is now connected. As a result, the second sample will have an inverted polarity, which (together with the first sample) results in a differential signal. After those two phases, the AD conversion is performed asynchronously and a digital code as function of the sensed capacitance is produced. Optionally, one could repeat such a two-step measurement but then with reversed DAC connections. If those two results are combined, a system-level correlated double-sampling (CDS) is achieved, which is able to cancel out ADC offset and 1/f noise. To improve resolution beyond the nominal value of a single measurement, the system can be oversampled and results can be averaged, which would result in constant-FoM scaling, at the cost of requiring multiple clock cycles, taking more time, and requiring post-processing to accumulate multiple results.

Since the reference voltage of the DAC is identical to the voltage applied to the sensor bridge, a *ratiometric* measurement is obtained. This implies that, to the first order, supply variations are automatically canceled out at system level. Since $C_{dac}$ loads the sensor bridge, effectively attenuating the voltage level $V_c$, it should be relatively small compared to the sensing elements. In the implemented 10 bit design, $C_{dac}$ is programmable from 300 fF up to 1.2 pF to enable programmability of the sensing range. Moreover, $C_r$ is also

programmable to enable a larger range of nominal sensor capacitances $C_s$.

Since all circuits have only dynamic power consumption, the interface can scale down to a very low absolute power consumption of 0.1 nW at low sampling rates. The energy per conversion is mostly dominated by the $CV^2$ term caused by the sensor itself since most sensors have a relatively large capacitance, and is in the order of 4 to 20 pJ/conversion. Based on this observation, a second implementation was made with as aim to re-use the charge in the sensing capacitance as much as possible. To do so, the capacitors in the system are not anymore reset at every measurement, but instead the charge is maintained on the sensor and ADC capacitors from one measurement to the next. If there is no leakage, it can be shown that this will result in exactly the same voltage $V_c$ as previously, thanks to the conservation of charges [9]. Since the large sensing capacitors are not anymore discharged and charged for every conversion, this results in major power savings. Unfortunately, due to capacitor leakage (mostly due to the connected switches), one cannot re-use charge indefinitely. Therefore, once every few measurements a reset is necessary to re-establish the charges. By doing so, the energy per conversion in [9] could be reduced down to between 1 and 4 pJ/conversion, while the leakage consumption was further reduced to 44 pW.

## V. A Dynamic Resistive Temperature Sensor Interface

As a second example, this section discusses the implementation of a fully dynamic resistive temperature sensor interface, using a similar design philosophy as before. While different on-chip components (such as bipolar transistors, diodes, and resistors) are temperature sensitive and can thus be used in principle as a temperature sensing element, resistor-based sensors are known to be relatively power efficient [1]. The overall temperature sensor interface is shown in Fig. 7, where the on-chip resistive bridge is composed of two resistors with a positive temperature coefficient ($R_p$) and two resistors with a constant or negative temperature coefficient ($R_n$) [10]. Compared to the CDC in Section IV, this bridge is differential, and thus a single clock pulse is sufficient to sample a differential output voltage $V_{out}$, which is now a function of temperature, on the SAR ADC for digitization. However, since the resistive bridge has static power consumption, it is essential to duty-cycle it to achieve overall dynamic behavior. Assuming (for simplicity) that the nominal values of $R_p$ and $R_n$ are equal to $R_0$, the power consumption in the on-state is proportional to $V_{DD}^2/R_0$ and the time-constant of the system is proportional to $R_0(C_p+C_{dac})$. For a finite settling error, the bridge should be enabled for a number of $RC$ constants, and thus the energy for a single measurement is proportional to the product of on-state power consumption and the $RC$ time, which is hence proportional to $(C_p + C_{dac})V_{DD}^2$. Importantly, this equation for energy consumption is independent from $R_0$. Therefore, its value can be chosen relatively small (to save chip area), while still achieving low power consumption. In fact, a small $R_0$ also reduces its parasitic capacitance $C_p$ which effectively lowers the energy consumption. Ultimately, the consumption is only limited by $C_{dac}$, which is bounded by the required resolution due to $kT/C$ noise. As in case of the CDC, this system is also ratiometric, CDS can be applied as well to compensate ADC offset and 1/f noise, and oversampling could be used to achieve constant-FoM scaling of energy consumption versus resolution.

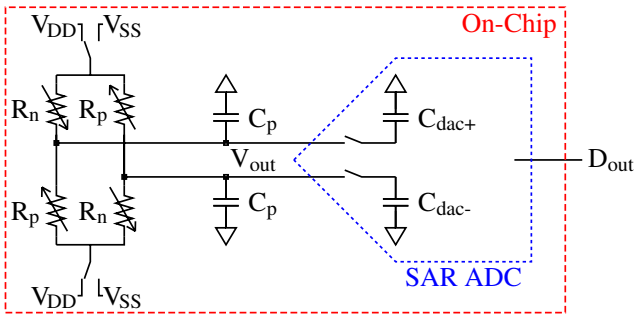


Fig. 7. Dynamic Temperature-to-Digital-Converter from [10].

The initial design, published in [10], achieves an energy consumption of approximately 5 pJ per conversion (including CDS), which scales down with the sampling rate down to a leakage level below 0.2nW, while achieving a resolution of 0.61 °C. In an improved prototype [11], the resistors in the bridge were optimized for better sensitivity and the ADC was changed to the unit-length topology to reduce $C_{dac}$ and to save area. Furthermore, power gating was added to reduce the leakage consumption further. As a result, the energy consumption was reduced to about 2.2 pJ per conversion (including CDS) and leakage was reduced to 52 pW while the resolution was simultaneously improved to 0.53 °C and the chip area was reduced to only 36 $\mu$m $\times$ 46 $\mu$m.

A disadvantage of the above sensor interfaces (which also applies to most other interfaces that can be found in literature) is that they do require calibration to compensate for e.g. offset, gain errors and non linearities. It should be noted (in the context of Fig. 1), that for high-resolution sensor interfaces (consuming in the order of $\mu$W's per conversion), there is a reasonable energy budget available for such corrections without causing too much overhead. On the other hand, for the low-power interfaces discussed in this paper (with a consumption in the order of pJ's), the acceptable budget for calibration is very small. Therefore, [12] and [13] investigated the feasibility of analog compensation of the various sensor imperfections. Ultimately, offset, gain and distortion could be compensated by re-using the capacitors in the SAR ADC's DAC and by adding a few additional capacitors and logic gates. This solution is attractive because the re-use minimizes area overhead. Further, the DAC capacitors are very small (with a unit element of 125aF) and the DAC can create many steps, thus enabling low power consumption and precise tuning. Lastly, the capacitor values are stable, and thus the compensation remains accurate over varying conditions.

The compensation by means of the capacitive DAC is essentially performed as follows [13]: the sensor bridge offset is compensated by pre-setting the DAC in the tracking phase to an equivalent offset value. After sampling, the DAC is reset to the nominal value, which effectively subtracts the offset from the sampled value before the AD conversion is performed. The sensor bridge gain error is simply compensated by adding a programmable $C_p$ (as shown in Fig. 4), which effectively adjusts the gain of the ADC to counteract the gain error of the sensor bridge. The non-linearity of the bridge is predominantly second order distortion, which is inherent to the resistive bridge topology [13]. This error is compensated by adding an opposite piece-wise linear function to the DAC by means of a few logic gates and capacitors. Overall, this design with integrated compensation of the sensor bridge errors increases the energy consumption to almost 3 pJ per conversion (including CDS) but the resolution was improved to 0.47 °C such that the FoM is degraded by less than 10% and no external processing is required anymore.

## VI. Conclusion

In this paper, architectural choices to develop low-power sensor interfaces were discussed, and various prototypes for capacitive and resistive sensor interfaces were reviewed. Fully dynamic operation is highlighted as a main feature to achieve a low absolute power consumption, with an inherent adaptability as function of the required sampling rate, the option to use correlated double sampling and the option to use constant-FoM oversampling to improve resolution if required. Since the operation of dynamic circuits depends on $CV^2$ terms, minimizing the DAC capacitance in the applied SAR ADC is imperative to reach the best possible efficiency. All presented designs benefit from ratiometric operation to minimize the impact of supply variations on the measurement result. Lastly, it was also shown that compensation of sensor variability (in terms of offsets, gain errors, and systematic distortion terms) can be done efficiently by mostly re-using the already available components in the ADC.


## Acknowledgements

This work was partially financed by the Netherlands Organisation for Scientific Research (NWO), with project number 16594. Further, Europractice is acknowledged for its MPW and design tool support.


## References


[1] K. A. A. Makinwa. (2023, Apr.) Smart temperature sensor survey. [Online]. Available: http://ei.ewi.tudelft.nl/docs/TSensor_survey.xls

[2] B. Murmann. (2022, July) ADC performance survey 1997 - 2022. [Online]. Available: http://www.stanford.edu/~murmann/adcsurvey.html

[3] S.-W. M. Chen and R. W. Brodersen, "A 6-bit 600-MS/s 5.3-mW asynchronous ADC in 0.13-$\mu$m CMOS," *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2669 – 2680, Dec. 2006.

[4] M. van Elzakker, et al., "A 1.9$\mu$W 4.4fJ/conversion-step 10b 1MS/s charge-redistribution ADC," *ISSCC Dig. Tech. Papers*, pp. 244 – 245, Feb. 2008.

[5] P. Harpe, C. Zhou, X. Wang, G. Dolmans, and H. de Groot, "A 30fJ/Conversion-Step 8b 0-to-10MS/s asynchronous SAR ADC in 90nm CMOS," *ISSCC Dig. Tech. Papers*, Feb. 2010.

[6] K. Bult, "High-efficiency residue amplifiers," in *Low-Power Analog Techniques, Sensors for Mobile Devices, and Energy Efficient Amplifiers*. Springer, 2019.

[7] P. Harpe, "A compact 10-b SAR ADC with unit-length capacitors and a passive FIR filter," *IEEE J. Solid-State Circuits*, vol. 54, no. 3, pp. 636 – 645, Mar. 2019.

[8] H. Xin, M. Andraud, P. Baltus, E. Cantatore, and P. Harpe, "A 0.1nW$-$1$\mu$W energy-efficient all-dynamic versatile capacitance-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 54, no. 7, pp. 1841 – 1851, July 2019.

[9] H. Xin, K. Pelzers, P. Baltus, E. Cantatore, and P. Harpe, "A compact fully dynamic capacitance-to-digital converter with energy-efficient charge reuse," *Solid-State Circuits Letters*, vol. 3, no. 1, pp. 514 – 517, Jan. 2020.

[10] H. Xin, M. Andraud, P. Baltus, E. Cantatore, and P. Harpe, "A 174pW$-$488.3nW 1S/s$-$100kS/s all-dynamic resistive temperature sensor with speed/resolution/resistance adaptability," *Solid-State Circuits Letters*, vol. 1, no. 3, pp. 70 – 73, Mar. 2018.

[11] K. Pelzers, H. Xin, E. Cantatore, and P. Harpe, "A 2.18-pJ/conversion, 1656-$\mu$m$^2$ temperature sensor with a 0.61-pJ·K$^2$ FoM and 52-pW standby power," *Solid-State Circuits Letters*, vol. 3, pp. 82 – 85, 2020.

[12] Y. Shen, M. van der Struijk, K. Pelzers, H. Li, E. Cantatore, and P. Harpe, "A 2.74pJ/conversion 0.0018mm$^2$ temperature sensor with on-chip gain and offset correction," in *proc. ESSCIRC*, Sept. 2022.

[13] Y. Shen, H. Li, E. Cantatore, and P. Harpe, "A 2.98pJ/conversion 0.0023mm$^2$ dynamic temperature sensor with fully on-chip corrections," in *ISSCC Dig. Tech. Papers*, Feb. 2023.