

FREYA

A 0.023-mm²/Channel, 20.8-μW/Channel, Event-Driven 8-Channel SoC for Spiking End-to-End Sensing of Time-Sparse Biosignals

Van Assche, Jonah; Frenkel, Charlotte; Safa, Ali; Gielen, Georges

DOI

[10.1109/TCSI.2024.3504264](https://doi.org/10.1109/TCSI.2024.3504264)

Publication date

2024

Document Version

Final published version

Published in

IEEE Transactions on Circuits and Systems I: Regular Papers

Citation (APA)

Van Assche, J., Frenkel, C., Safa, A., & Gielen, G. (2024). FREYA: A 0.023-mm²/Channel, 20.8-μW/Channel, Event-Driven 8-Channel SoC for Spiking End-to-End Sensing of Time-Sparse Biosignals. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 72(3), 1093-1104. <https://doi.org/10.1109/TCSI.2024.3504264>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

<https://www.openaccess.nl/en/you-share-we-take-care>

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

FREYA: A 0.023-mm²/Channel, 20.8-μW/Channel, Event-Driven 8-Channel SoC for Spiking End-to-End Sensing of Time-Sparse Biosignals

Jonah Van Assche¹, Member, IEEE, Charlotte Frenkel², Member, IEEE, Ali Safa³, Member, IEEE, and Georges Gielen¹, Fellow, IEEE

Abstract—Biomedical systems-on-chip (SoCs) for real-time monitoring of vital signs need to read out multiple recording channels in parallel and process them locally with low latency, at a low per-channel area and power consumption. To achieve this, event-driven SoCs that exploit the time-sparse nature of biosignals such as the electrocardiogram (ECG) have been proposed; they only process the signal when it shows activity. Such SoCs convert time-sparse biosignals into spike trains, on which spiking neural networks (SNNs) can perform event-driven signal classification. State-of-the-art event-driven SoCs, however, still suffer from poor area and power efficiency and use inflexible, hard-coded spike-encoding schemes. To improve on these challenges, this paper presents FREYA, an 8-channel event-driven SoC for end-to-end sensing of time-sparse biosignals. The proposed SoC consists of the following key contributions: 1) an 8-channel time-division-multiplexed level-crossing sampling (LCS) analog-to-spike converter (ASC) that encodes analog input signals into input spikes for an on-chip SNN; 2) an ASC spike-encoding algorithm that is fully programmable in resolution (4 to 8 bits) and conversion algorithm (offset and decay parameters); 3) an on-chip integrated, flexible SNN processor based on a programmable crossbar architecture, that allows for efficient event-driven processing, and that can be reconfigured towards multiple sensing applications; 4) a custom offline end-to-end training framework for the fast retraining of the spike-encoding algorithm and SNN architecture towards new applications or patient-dependent signal variations. A prototype IC has been fabricated in a 40nm CMOS technology. It has a per-channel active area of 0.023 mm² (0.184 mm² in total), a 7× improvement over the state of the art. For the use case of ECG-based QRS-labeling, a detection accuracy of 98.67% is achieved, while the system consumes 20.8 μW per channel and achieves a latency of

only 80 ms, thus paving the way for multi-channel, high-fidelity, event-driven SoCs in biomedical applications.

Index Terms—Biomedical SoC, event-driven sensing, spiking neural networks, level-crossing sampling.

I. INTRODUCTION

REAL-TIME monitoring applications for time-sparse biosignals such as electrocardiogram (ECG), require power- and area-efficient sensing systems that can read out multiple recording channels in parallel and process them locally with low latency. To facilitate sensor fusion, which enables a higher-fidelity classification of the signals, a high channel count is required, from 3 channels for wearable ECG [1] to 16 channels or more for an electroencephalogram (EEG) array [2].

Conventional biomedical systems-on-chip (SoCs) multiplex multiple readout channels and perform classification in a frame-based manner (see Fig. 1a) [2], [3], [4]. This results in an area-efficient analog-to-digital converter (ADC), since the same hardware can be reused for multiple channels. However, frame-based approaches suffer from a large processing latency (> 1 s) [2], [4]. Moreover, the digital classifiers used in the frame-based approach, such as support vector machines (SVMs) [2], decision trees [3] or convolutional neural networks (CNNs) [4], [9], occupy a large chip area due to the required on-chip memory. Besides the large chip area, large computational models also consume a significant amount of energy while continuously processing the incoming sensor data. Such systems do not exploit the time-sparse nature of signals like the ECG, and likely waste power in processing redundant data.

Event-based processing SoCs with on-chip spiking neural networks (SNNs) that perform inference on a spike train, on the other hand, promise a low latency and a power consumption proportional to the signal activity, and are therefore an attractive alternative to traditional systems [6], [7], [8], [16]. Moreover, area-efficient SNN implementations with dense synapses and neurons have been proposed recently [16]; they can offer a smaller on-chip area for the classifier. Event-based SoCs can therefore be an attractive alternative to traditional systems. Several possible architectures for event-based SoCs exist; each approach, however, still faces several limitations.

Received 1 June 2024; revised 29 August 2024 and 24 October 2024; accepted 13 November 2024. Date of publication 28 November 2024; date of current version 27 February 2025. This work was supported in part by the Fonds Wetenschappelijk Onderzoek (FWO) Stroke CAvities Treatment Mechanism with Active Neural interfaces (SCATMAN) Project S000221N. This article was recommended by Associate Editor M. Ballini. (Corresponding author: Jonah Van Assche.)

Jonah Van Assche was with the MICAS Research Group, KU Leuven, 3001 Leuven, Belgium. He is now with the Department of Computer Science, Electrical Engineering and Information Technology, University of Stuttgart, 70550 Stuttgart, Germany (e-mail: jonah-van.assche@iis.uni-stuttgart.de).

Charlotte Frenkel is with the Department of Micro-Electronics, Delft University of Technology, 2628 CD Delft, The Netherlands.

Ali Safa was with IMEC, 3001 Leuven, Belgium, and also with the MICAS Research Group, KU Leuven, 3001 Leuven, Belgium. He is now with the College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar.

Georges Gielen is with the MICAS Research Group, KU Leuven, 3001 Leuven, Belgium.

Digital Object Identifier 10.1109/TCSI.2024.3504264

1549-8328 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See <https://www.ieee.org/publications/rights/index.html> for more information.

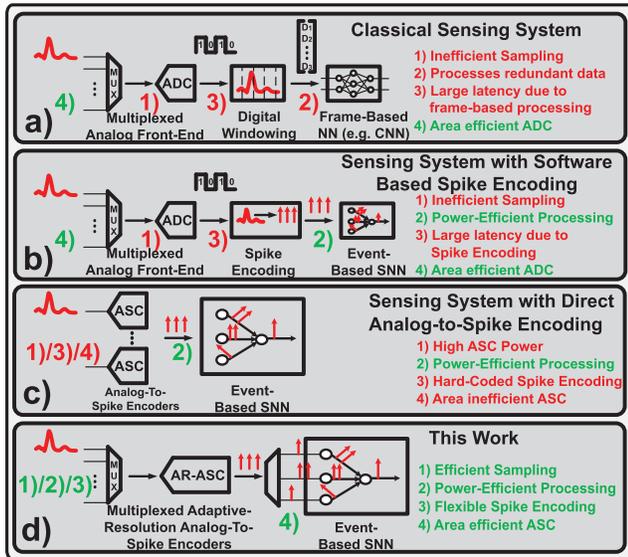


Fig. 1. Overview of the alternative architectures for biomedical SoCs and their challenges: a) classical SoC with fixed-rate multi-channel ADC; b) event-driven SoC with fixed-rate multi-channel ADC and software based spike encoding; c) event-driven SoC with asynchronous, fixed-threshold ASCs; and d) the proposed event-driven SoC with a multi-channel, synchronous ASC with adaptive thresholds. Note that the analog front-end (instrumentation amplifier and filters) is omitted from this overview.

One approach is to first digitize the signal, followed by software-based spike encoding, as shown in Fig. 1b [5]. In [5], the effect of several encoding techniques such as rate coding, time-to-first-spike coding, etc. on the accuracy of the SNN classification task was examined for two datasets. This approach, however, still wastes power by first sampling all sensor data before converting them to spikes. The digital spike conversion also introduces a latency overhead. Directly converting the analog signals to spikes by means of a level-crossing sampling (LCS) analog-to-spike converter (ASC) therefore is a promising alternative [7], [8], [9], [10], [14], [33], [34]. A conventional ADC converts analog signals into digital codes at a fixed sample rate (hence, not exploiting the signal sparsity). In contrast, an LCS ASC compares analog signals to two thresholds, and only takes a sample (hence, generates a spike) whenever the analog signal crosses one of those thresholds. This automatically results in an event-driven spike train that scales with the signal sparsity [14]. To generate an even sparser spike train, instead of using fixed thresholds, alternating thresholds can be used, in which the threshold switches over time from initial coarse to more fine afterwards [14], [33], [38].

Several systems with on-chip ASC and SNNs have been presented in recent years. [7] uses two ASCs with a fixed threshold to convert action potential signals from the nerves into a spike train as input for an on-chip SNN. In [8] an 8-channel EEG recording systems is presented, consisting of instrumentation amplifiers, filters, an LCS ASC and on-chip SNN. The analog signals from the instrumentation amplifiers are filtered by bandpass filters, and these filtered signals are then converted into spike trains by the ASC. The output spike trains are then fed to four on-chip SNN cores. A similar system was presented in [34], which introduces a full 16-channel sensor system that can encode analog signals either with an

LCS ASC, or by using an on-chip pulse-frequency modulation sampling scheme that converts the analog signal amplitude into a spike rate.

While systems with LCS ASCs and on-chip SNNs show great promise, the solutions presented till now still have several limitations. Most LCS ASC systems cannot be multiplexed, and hence, a full ASC is required per channel, which limits the channel count [7], [8], [34]. Moreover, conventional ASCs have power efficiencies that are up to $10\times$ worse than a conventional ADC [7], [8]. Finally, the works that combined LCS ASC with SNN [6], [7], [8] use hard-coded, fixed-threshold spike-encoding schemes that are not flexible (see Fig. 1c). This leads to an undesirable trade-off between the amount of threshold levels (which determine the information that is generated) and the sparsity of the spike train (which determines the system power consumption).

To address these issues, this paper introduces FREYA, an area- and power-efficient 8-channel event-driven SoC for spiking end-to-end sensing of time-sparse biosignals (see Fig. 1d). FREYA offers the following key contributions:

- 1) An 8-channel time-division-multiplexed LCS ASC that encodes analog input signals into spikes as input for an on-chip SNN, which results in an area of $0.023 \text{ mm}^2/\text{channel}$ and a power consumption of $20.8 \mu\text{W}/\text{channel}$.
- 2) The ASC spike-encoding algorithm is fully programmable in resolution (4 to 8 bits) and conversion algorithm (offset and decay parameters), and can flexibly be adjusted to various signals, to achieve an optimal classification accuracy.
- 3) An on-chip integrated, flexible SNN processor based on a programmable all-to-all crossbar architecture allows for efficient event processing, and can be reconfigured towards multiple sensing applications. For a QRS-labeling task from ECG, a detection accuracy of 98.67% is achieved.
- 4) Based on our previous work, in which the relation between SNN accuracy and the spike encoding scheme of LCS ASCs was explored [18], we introduce a custom offline end-to-end training framework for the SoC. We have shown before [18] that information-theoretic criteria such as the Corrected Akaike information criterion can predict for which setting of an LCS ASC, an SNN will likely reach the maximum detection accuracy. This technique is used in the proposed framework to only train an SNN model for a specific subset of ASC settings. This allows for a fast retraining (69% less training epochs are required compared to a conventional grid-search approach) of the spike-encoding algorithm and SNN architecture towards new applications or patient-dependent signal variations.

The paper is organized as follows. Section II gives an overview of the proposed SoC, consisting of a multi-channel adaptive-resolution ASC and an SNN processor. Section III discusses in detail the design of the ASC. Section IV introduces the end-to-end training framework for the SoC, Section V gives the measurement results of the event-driven SoC. Section VI concludes the paper.

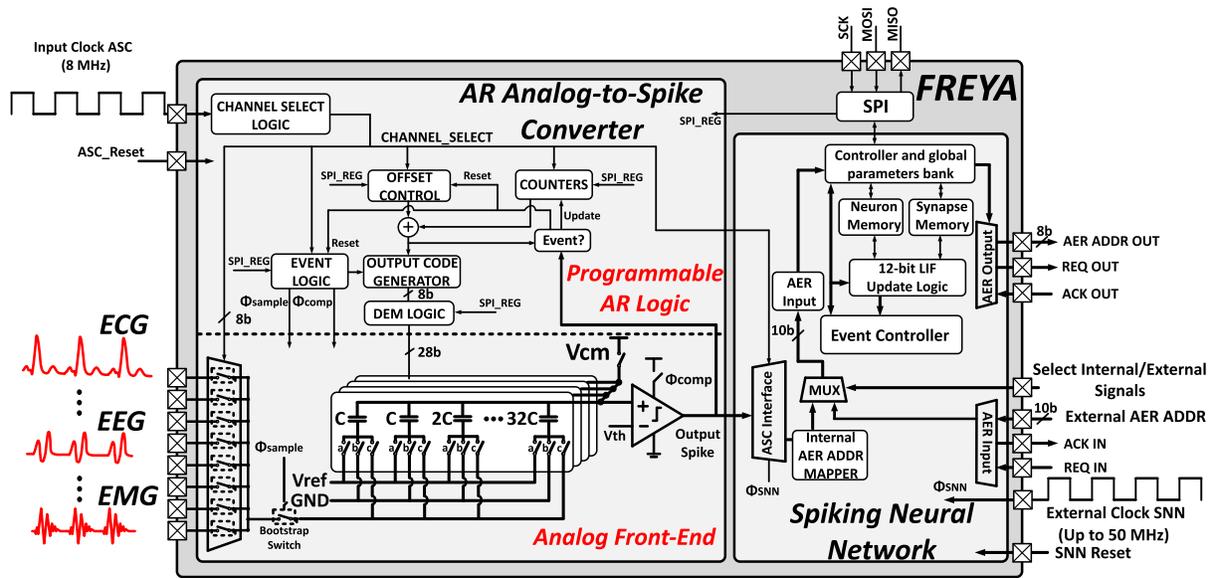


Fig. 2. Overview of the proposed event-driven SoC with adaptive-resolution ASC (drawn single-ended for simplicity) and integrated SNN processor.

II. SYSTEM OVERVIEW

The block diagram of the proposed event-driven SoC is depicted in Fig. 2. It consists of two main parts: an adaptive-resolution differential LCS ASC, that converts an incoming analog signal into a spike train, and an SNN that performs inference on this spike train. Each channel has a bandwidth of 1000 Hz, targeting biomedical applications such as ECG readout, and a full-scale input of 0.7 V_{pp}. The analog part of the ASC consists of multiplexer switches that select the input channels in a round-robin fashion, sampling switches that sample the analog input of the selected channel, a capacitive DAC (CDAC) array and a single dynamic comparator. The CDAC is split in 4 sub-arrays to enable dynamic element matching (DEM), which will be explained in Section III. The digital part of the ASC includes the programmable logic that controls the multiplexer (*Channel Select Logic*), the adaptive-resolution logic (*Offset Control*, *Output Code Generator*), the logic that slows down the sampling frequency for signals with low input frequency (*Event Logic*), the counters that keep track of the current channel value (*Counters*), and the *DEM Logic* that implements the DEM algorithm of the CDAC. The back-end SNN is a lightweight version of the SNN processor¹ first described in [16], ideally suited for low-power edge systems. The SNN consists of a time-division-multiplexed fully programmable crossbar with 256 neurons and 64k 4-bit-weight synapses. The accelerator supports integrate-and-fire neurons with optional leakage, and can interface both with the ASC via an internal ASC interface, as well as directly with external sensors via an address-event representation (AER) interface [16]. This makes the SoC ideal to fuse data from different sensor signals.

The digital circuits of both the ASC and the SNN consist of synthesizable logic and are programmable via a shared SPI interface. The digital core of the SNN runs at a clock frequency up to 50 MHz, while the digital circuits of the ASC work at a clock frequency of 8 MHz, both generated externally.

¹This processor is available at: <https://github.com/ChFrenkel/tinyODIN>

To enable clock domain crossing and not miss any events from the ASC to the SNN, the ASC interface oversamples the output of the ASC. Using a mixed-mode simulation in which spikes from the ASC were fed to the SNN, it was found that the SNN clock frequency must be $> 7\times$ faster than the ASC clock. Note that the SNN clock frequency does not influence the spike rate, which is fully determined by the ASC.

III. MULTI-CHANNEL ANALOG-TO-SPIKE CONVERTER

We now describe in more detail the ASC design, the ASC timing, and the detailed circuit block implementations.

A. Level-Crossing Sampling ASC

To exploit the time-sparse nature of biosignals such as ECG with event-driven hardware, LCS ASCs have been proposed [7], [8], [9], [10], [11], [34]. Such event-driven ASCs only take samples whenever the input signal changes beyond a threshold (see Fig. 3), which then generates a spike event. These spike events can then be processed directly by SNNs [6], [7], [8], [34].

While interfacing an LCS ASC directly with an SNN allows reducing the latency and the power overhead thanks to the direct analog-to-spike conversion, it still faces circuit challenges compared to traditional fixed-rate sampling systems or software-based spike conversion (see Fig. 1a and Fig. 1b). Indeed, since conventional LCS ASCs are asynchronous data converters that are always on, it is difficult to reuse the same hardware in a time-multiplexed fashion, which results in a poor area efficiency for the ASC, as a full ASC would be required per readout channel [7], [8], [34]. Moreover, most LCS ASCs have a high DC power consumption due to the continuous-time comparators used to detect a level crossing [7], [8], [14], [34]. Finally, for a given classification task, traditional ASCs face a trade-off between task accuracy and power consumption. If a coarse crossing threshold is used, the ASC generates few spikes, which leads to a low SNN dynamic power consumption (Fig. 3), but also to a low quantization

resolution, and hence a low dynamic range (DR), thereby reducing the SNN accuracy [6]. On the other hand, using a finer crossing threshold yields a higher SNN accuracy, but results in higher spike rates and SNN power consumption [6]. This makes ASCs with a fixed threshold a rather inflexible spike-encoding scheme, with only a single degree of freedom based on the threshold selection.

The ASC used in the proposed FREYA SoC solves these challenges. Firstly, a time-multiplexing technique is used where the ASC circuitry is shared among all readout channels (see Fig. 2), thereby significantly reducing the area per channel. This is made possible by introducing a system clock for the ASC, which allows to time-division multiplex the readout channels. Since the input signal of each channel is converted to spikes in different clock cycles, the same hardware can be reused for each channel. The proposed SoC consists of 8 differential input channels - single-ended operation is also possible - across which the ASC cycles at a programmable rate T_{scan} . Secondly, to fully exploit the time-sparse nature of the biosignals, and hence to reduce the ASC power consumption, the ASC uses event-driven adaptive-rate clocking (see the *Event Logic* in Fig. 2) with dynamic comparators, which results in a signal-activity-dependent, dynamic power consumption [11]. If no spike is generated in a conversion cycle, independently for each channel, the event logic gradually slows down the sampling speed, adjusting the ASC power consumption to the activity of the input signal. Moreover, the adaptive clock allows for a synchronized ASC output, which would be easy to integrate with any subsequent processing/transmitter circuit. Thirdly, each channel of the ASC encodes the input signal by using an adaptive-resolution level-crossing algorithm [14], [33], in which the threshold initially is coarse (e.g., multiple LSBs), before gradually switching to a finer resolution after a programmable amount of cycles T_{Decay} (see Fig. 3). If a threshold is crossed, the output of the ASC updates to the value of this threshold and resets the thresholds to the maximum programmable offset value. By employing this adaptive-resolution algorithm for the spike encoding, the usual trade-off between the DR and the ASC spiking activity is improved. By switching from coarse to fine adaptive thresholds, a fine resolution can be achieved that can capture detailed signal features, while at the same time the average spike rate of the ASC, and therefore the SNN dynamic power, is reduced significantly. The adaptive-resolution algorithm and the ASC's programmable resolution thus provide three degrees of freedom (decay time T_{Decay} , offset, and ASC resolution), offering much more flexibility for the spike encoding compared to conventional LCS ASCs.

B. ASC Timing

The detailed ASC timing is shown in Fig. 4. At first, the ASC operates in the *regular mode*, and, at a programmable rate T_{scan} , the ASC cycles over the different readout channels, which is done by the *Channel Select Logic* block. When a channel is selected, the ASC samples the input signal and compares it in 4 clock cycles (P1 to P4) to two thresholds that change adaptively. The digital code for these thresholds levels is generated by the *Offset Control* block, which generates the

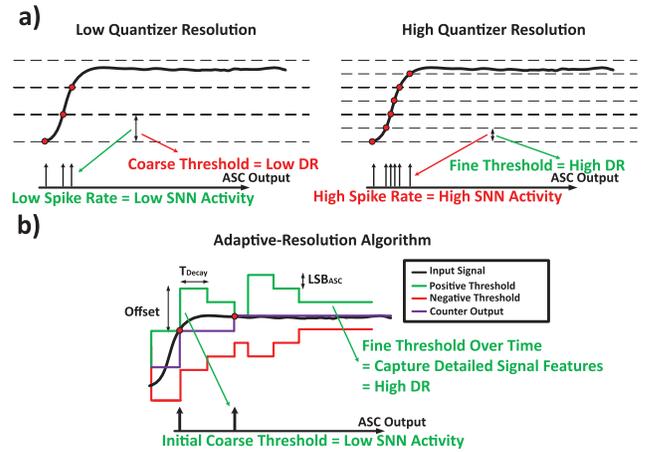


Fig. 3. a) Trade-off between spike activity and DR in a conventional LCS ASC, and b) the proposed programmable adaptive-resolution level-crossing spike encoding scheme. The programmable parameters of the ASC are: the offset, the decay time (T_{Decay}) and the ASC resolution (indicated by the LSB of the ASC, LSB_{ASC}).

current offset value, and the *Output Code Generator* block, which takes the offset value and the current value from the *Counters* block and generates a positive threshold during cycles P1 and P2, and a negative threshold during cycles P3 and P4 for the CDACs. If the sampled input signal crosses one of the thresholds, the counter value for the selected channel is updated, and the offset value for that channel is reset to the maximum value, which is programmable via the SPI. Each channel operates independently, and after 4 cycles, the next channel is selected. Once a channel's offset value is reset, the adaptive-resolution algorithm switches to a finer threshold after T_{Decay} cycles, if no new level crossing occurs. Once the offset value has reached the LSB value for the ASC, which is also programmable via the SPI, the input signal of the readout channel is assumed to be quasi-static. Hence, the ASC enters the *event mode* (see Fig. 4), and the ASC starts to skip sampling cycles to save power, until a new level crossing occurs, and the adaptive-resolution algorithm is reset and the ASC enters again the regular mode. In the event mode, the channel select logic keeps cycling over the different channels in the same order. However, if a sample cycle is skipped, the control signals will remain low. The amount of sampling cycles that need to be skipped is tracked by the *Event Logic* block. As depicted in Fig. 4, the amount of cycles that are skipped is gradually incremented until a new level crossing occurs. Compared to our previous work in [11], the event logic skips sample cycles, rather than adjusting the clock frequency in real time by means of an on-chip clock generator [11], which leads to a simpler and more robust design by using only a single clock domain.

C. Detailed Circuit Implementation

This section will describe the detailed circuit implementation of the different building blocks of the proposed ASC.

1) *Multiplexer and Sample Switches*: The multiplexer implements the time-division-multiplexing technique of the ASC. It consists of twice 8 bootstrapped switches (one for each differential channel), that are connected to the input of the sample switches of the ASC. Fig. 5 shows in more detail

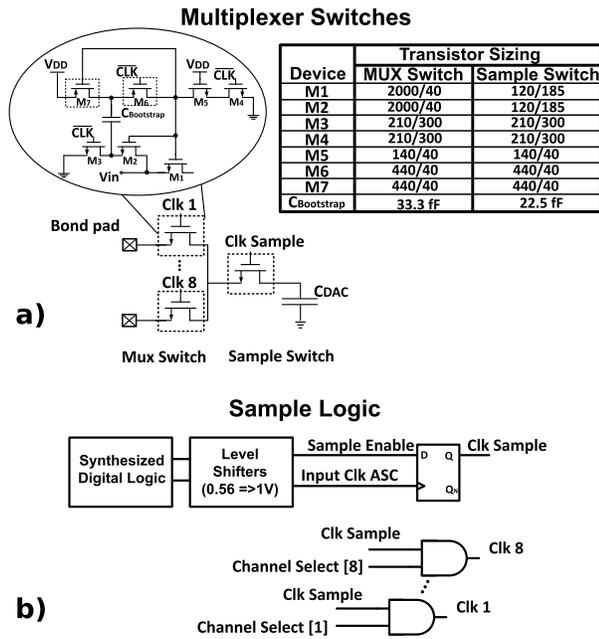
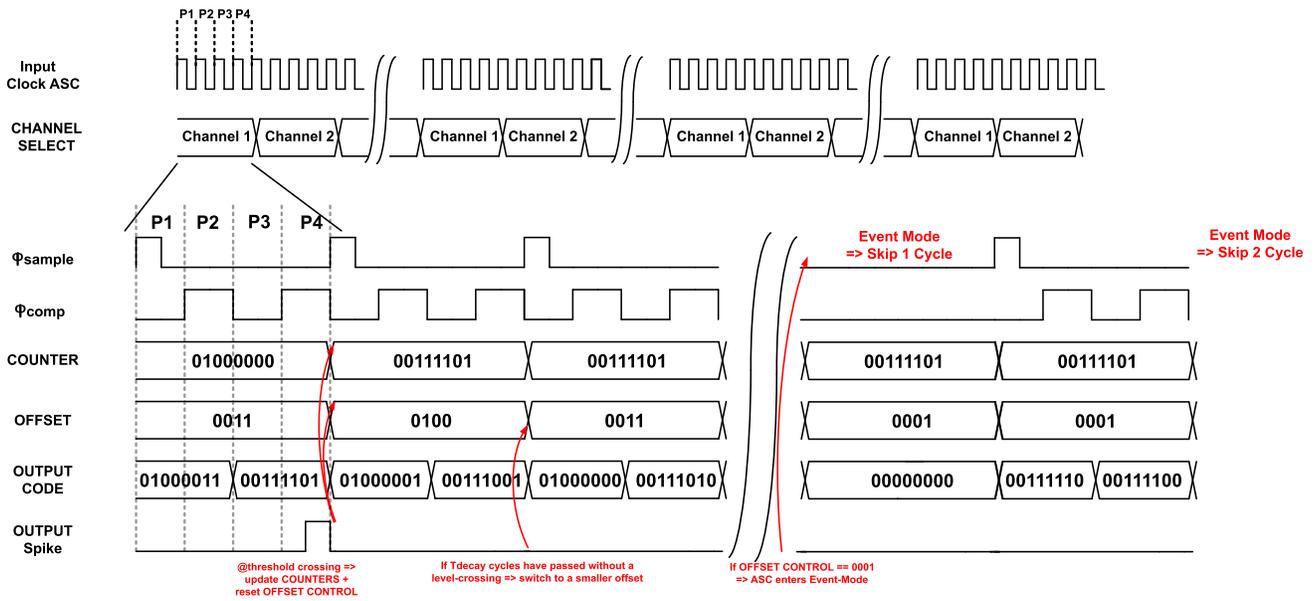


Fig. 5. a) Single-ended view of the proposed multiplexer circuit and sample switch, consisting of two rows of bootstrap switches. The device sizes are given in the table. b) Multiplexer and sample switch control logic.

the multiplexer circuit, the device sizes for the multiplexer, as well as the sample switches. The bootstrap switches are implemented as described in [26]. Also shown is the logic to control the multiplexer and the sample switches.

2) *Comparator*: To improve the ASC's power efficiency, the power consumption of the comparators needs to be reduced. Recent work on LCS ASCs has proposed continuous-time comparators with an adaptive bias current [36], that switch to a low bias current when the signal is not near one of the ASC's thresholds. Another option to reduce the comparator power has been introduced in our previous work [11]. We showed how using an adaptively clocked

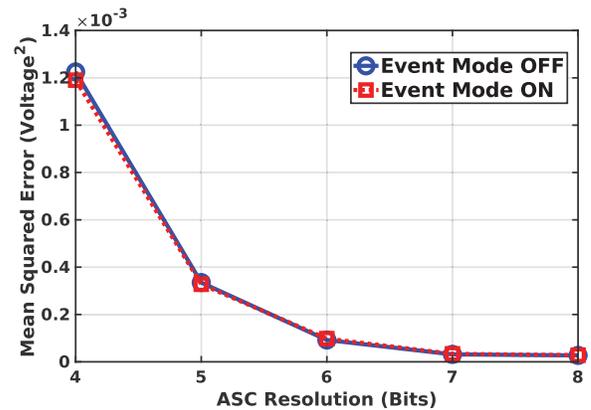


Fig. 6. Simulation of the MSE of an LCS ASC with the event mode off (working at a fixed clock rate) and the event mode on (working with an adaptive clock), for an ECG input signal.

dynamic comparator can relax the power consumption of the comparator, since it scales with the signal activity. Using such adaptive clock does not cause a degradation of the signal. In Fig. 6, the simulated mean squared error (MSE = $\frac{1}{n} \sum_{n=1}^{\infty} (\bar{x}(t) - x(t))^2$, with $\bar{x}(t)$ the output signal of the ASC and $x(t)$ the original input signal) of the LCS ASC is shown for different LCS ASC resolutions, using an ECG input. Clearly, no increase in MSE is noted when using the adaptive clock in the event mode. Compared to [11], the comparators in the current work can be clocked at a lower speed, thanks to the adaptive-resolution algorithm, which relaxes the required bandwidth constraints. To track a signal sufficiently fast (e.g. to not miss level crossings), an LCS ASC should have a sampling frequency equal to [11]:

$$f_{\text{sample}} = N_{\text{Channels}} \cdot \frac{(\frac{\partial V_{\text{in}}}{\partial t})_{\text{max}}}{\text{LSB}} \quad (1)$$

where N_{Channels} is the number of recording channels and $(\frac{\partial V_{\text{in}}}{\partial t})_{\text{max}}$ is the maximum input slope of the input signals. Since the adaptive-resolution algorithm allows to initially use

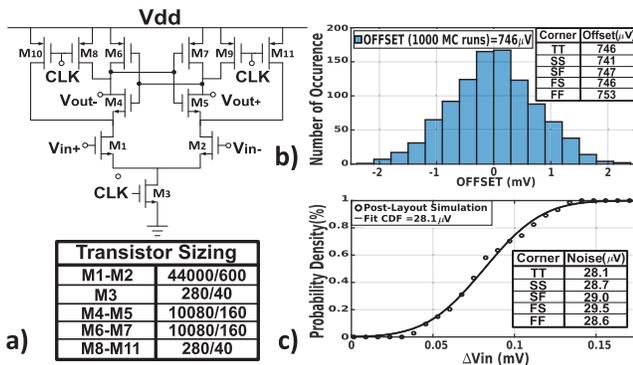


Fig. 7. a) StrongARM comparator design and device sizes used. Plotted are b) a 1000-sample Monte-Carlo simulation to determine the offset (TT corner), and c) a post-layout transient noise simulation (TT corner). The values in the other process corners are also indicated.

a coarser LSB, the sampling frequency (and thus the clocking frequency of the comparators) can be relaxed greatly. In our design, the comparator is clocked at maximum 4 MHz, half the ASC input clock, which is fast enough to track 8 channels with an input bandwidth of 1000 Hz. For comparison, without the adaptive-resolution algorithm, this clock should be 16× higher. For the comparator architecture, a StrongARM latch topology [27] has been chosen. The design and its device parameters are given in Fig. 7a. Typically, an LCS ASC uses two comparators to determine if the input signal crosses a threshold above or below the previous level [7], [8], [11], [14], [34]. The difference in input-referred offset of both comparators degrades the ASC linearity. Instead of using power-hungry calibration to compensate the offset [7], [14], the proposed ASC only uses a single comparator for both comparisons, making the comparator offset the same for both decisions. In this way, the offset introduces only a DC shift in the quantization levels, which does not degrade the linearity. The comparator offset however does reduce the ASC dynamic range. Hence, the comparator devices are sized such that the 3σ offset value is below 0.5 LSB. From post-layout simulations the σ offset value is found to be $746 \mu V_{\text{rms}}$ (TT corner). The input-referred comparator noise is found to be $28.1 \mu V_{\text{rms}}$ (TT corner). The simulated values for the offset and noise in the other corners are given in Fig. 7b and 7c, respectively.

3) *CDAC and DEM Logic*: For the CDAC design, a custom 300-aF MOMCAP (M3 and M4 metal layers) is used, similar to [28] and depicted in Fig. 8. The equivalent capacitor model used for circuit simulations is also given, which was obtained via a parasitic extraction simulation. Cbp and Ctp are 86 and 15 aF, respectively. For the layout of the CDAC, a symmetric layout along the x-axis has been chosen for each single sub-array [28], with double symmetry such that each unit cell consists of two unit capacitors that are placed symmetrically along the axis, averaging out gradients along the x-axis. This layout scheme is kept simple to reduce the fringe capacitance between the routing and the small unit capacitors [28]. For every CDAC, each sub-array is then placed along the y-axis.

LCS ASCs suffer, at low input frequencies, from harmonics that fall into the signal band [12]. These harmonics are caused by both the reconstruction scheme used to interpolate the

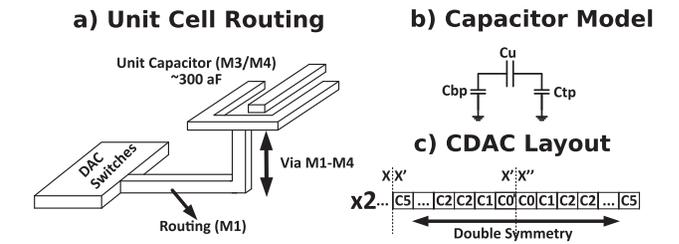


Fig. 8. a) Routing of a unit cell of the CDAC, b) the corresponding capacitor model used for circuit simulations and c) the layout floorplan for a single DAC sub-array.

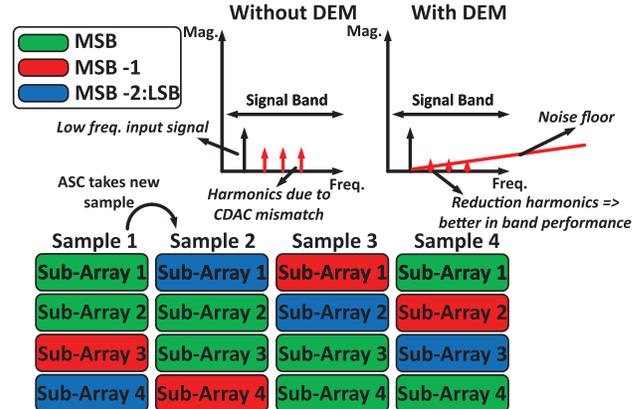


Fig. 9. Visual overview of the proposed DEM algorithm for the CDAC sub-arrays and its impact on the ASC distortion.

output signal of the non-uniformly sampling ASC, as well as by mismatches in the ASC DAC that generates the threshold values [11], [13]. To avoid increasing the sizing of the CDAC to achieve a higher linearity, which would increase the power consumption, a DEM technique is used (Fig 9). The 8-bit CDAC is split up in four sub-arrays. In each sample cycle, two sub-arrays are configured as MSB, one sub-array as MSB-1, and the other sub-array acts as a regular binary-weighted DAC. In each new sample cycle, these sub-arrays are shifted in a barrel-shifting fashion, mitigating the harmonics caused by the CDAC mismatch. Another advantage of using DEM is that possible gradient effects, which are not modeled in a standard PDK, are also averaged out along the y-axis.

IV. END-TO-END OPERATION OF THE EVENT-DRIVEN SOC

FREYA is configurable towards a wide range of time-sparse signal applications, such as ECG and EEG. The adaptive-resolution algorithm and the programmable resolution of the ASC allow tailoring the spike encoding to a particular signal of interest. To achieve the highest classification accuracy, a custom end-to-end training framework is introduced in this section, based on our previous work [18], where we showed that information-theoretic criteria can predict the settings of an LCS ASC for which an SNN will likely reach the maximum detection accuracy. The framework proposed here uses these insights to deploy FREYA in an end-to-end application, based on the flow depicted in Fig. 10. First, a high-level MATLAB model of the ASC generates realistic spike-encoding signals for each ASC setting under test, which can be used as input for the SNN training. Doing a brute-force search over all ASC parameters (i.e. the ASC resolution, offset, T_{decay}) for a given

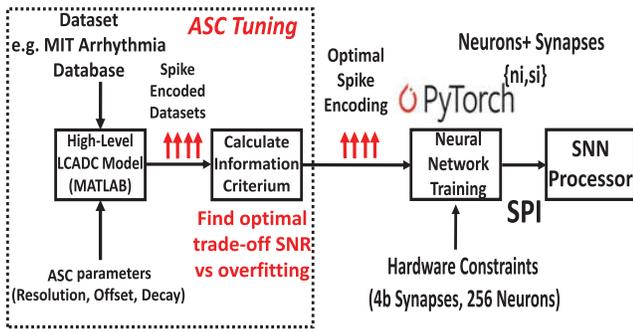


Fig. 10. Proposed end-to-end operation flow for the event-driven SoC. The relation between the information criterion and the SNN accuracy has been explored in previous work [18]; the flow presented here applies this insight to train an application end-to-end.

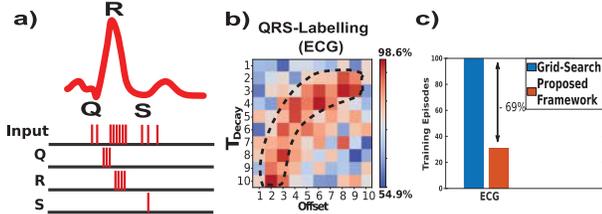


Fig. 11. a) Illustrative classification task for the SNN: event-driven QRS-labelling from an ECG input signal. b) The achieved SNN accuracy is plotted as a function of the ASC parameters (the ASC resolution is programmed to be 6 bits). Indicated in the dashed area is the region of optimal ASC performance predicted by the Corrected Akaike information criterion. c) The graph indicates the improvement in training time by using the information criterion for training, compared to a full grid search of all ASC parameters.

application would, however, yield a long optimization time for the SNN model, as a full hyper-parameter tuning would be required per ASC setting. For example, for a 10×10 grid search for the offset and T_{decay} parameters, as shown in Fig. 11, approximately 10 hours of quantization-aware training would be required per point in the grid on a computer equipped with an NVIDIA V100 GPU. Moreover, simulations show that, depending on the signal (features) of interest, there is a different optimal setting for the ASC, which might even be patient-specific. To avoid a lengthy grid search over all ASC parameters, our training framework calculates, for each spike encoding corresponding to each ASC setting under test, the Corrected Akaike information criterion [19]:

$$\text{AIC}_c = N_s \log \left(\frac{\sum_{n=1}^{N_s} (\tilde{s}_i - s_i)^2}{N_s} \right) + 2\kappa + \frac{2\kappa^2 + 2\kappa}{N_s - \kappa - 1}, \quad (2)$$

where N_s is the number of input samples over which the criterion is evaluated (the training dataset), \tilde{s}_i the reconstructed signal from the ASC output, s_i the original input signal, and κ the model complexity, which is the number of non-zero model parameters. Applied to the ASC, it is equal to the spike density of the generated dataset [18]. This criterion determines a subset of settings of the ASC for which the ideal trade-off between high DR and overfitting of a model can be reached. Hence, the criterion can be used prior to training the SNN model, to predict if the encoding scheme under test leads to a high or a low SNN accuracy [18]. By calculating this criterion for each ASC setting, the region with minimum information criteria leads to a reduced search space (dashed region in Fig. 11b), where the optimal SNN performance is expected to be found. Following this step, for each spike encoding

corresponding to an ASC setting in this reduced search space, an SNN model is trained in Pytorch. This training takes into account the hardware constraints of the SNN processor embedded in FREYA, mainly the limited amount of neurons (256) and the quantized weights (4 bits signed) [16], [18]. Once the SNN model is trained, the weights and synapses are programmed via the SPI interface of FREYA.

As a use case, we have selected the QRS-complex labeling task for ECG (Fig. 11a). The detection of the QRS-complex is often implemented for wearable ECG applications and can be used for e.g. heart rhythm estimation, arrhythmia detection, etc. [35]. The MIT-BIH Arrhythmia dataset [31] (retrieved via [32]) is used to train the SNN. This dataset contains 48 two-channel ambulatory ECG recordings of 30 minutes each, which were obtained from 47 subjects. Recording 101 is used as training sequence and recording 201 as the independent test sequence acquired from a different patient. The full 30 minutes of the recording are used for training. The SNN architecture is a recurrent neural network with a single hidden layer of 250 neurons, an input layer of three neurons and an output layer of three neurons, one for each feature of the QRS-complex, thus three classes. The SNN architecture is trained via back-propagation through time (BPTT) [20] using the SLAYER surrogate gradient technique [21] and the cross-entropy loss function, as conventionally used for classification problems [22]. The Adam optimizer [23] is utilized with learning rate $\eta = 3 \times 10^{-4}$ and decay parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, for a total of 100 training epochs with batch size 128. All weights are initialized using the uniform Xavier method [24], following prior SNN training studies such as [22]. During training, the SNN weights are quantized to a 4-bit precision via quantization-aware training [25]. Doing so, a simulated peak classification accuracy of 98.6% is reached in our QRS example. By using the proposed training framework, a reduction in NN training epochs of 69% (31 epochs) compared to the full grid search (100 epochs) is obtained for the selected use case (Fig 11c).

V. MEASUREMENT RESULTS

The proposed FREYA SoC has been implemented in a 40nm CMOS technology (see Fig. 12), and has a record-low active chip area of 0.023 mm²/channel (0.184 mm² in total, of which the on-chip SRAM takes 70%). The ASC has an area of 0.0067 mm²/channel. The chip can sample up to 8 differential inputs with a maximum bandwidth of 1000 Hz, and works at a clock frequency of 8 MHz. The embedded SNN processor can work at a clock frequency up to 50 MHz.

A. Measurement Setup

A fully automated measurement setup has been developed (see Fig. 13). It consists of a Zynq 7020 SoC FPGA, which takes care of the communication with the chip (SPI and AER interfaces), records the data coming from FREYA, and controls the on-PCB components. These consist of a 16-channel DAC (ADI - AD5766) that can provide a multi-channel input to the ASC, an optional tunable 3th-order anti-aliasing filter, on-PCB LDOs (ADI - LT3045) that provide clean reference

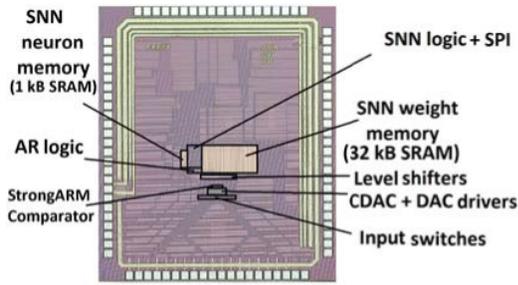


Fig. 12. Die photo of the fabricated prototype FREYA IC in 40nm CMOS.

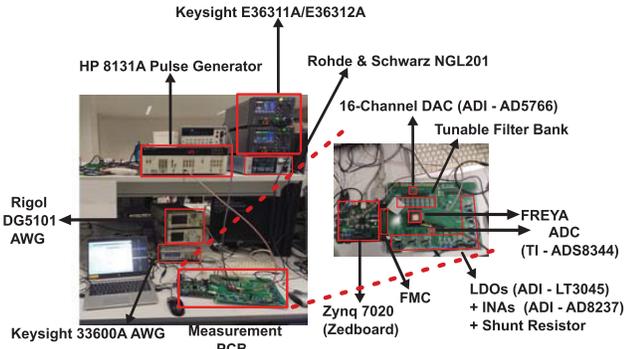


Fig. 13. Photo of the measurement setup for FREYA.

voltages for FREYA and the board components, instrumentation amplifiers (ADI - AD8237) to amplify small voltages over the current shunt resistors, and an 8-channel ADC (TI - ADS8344) that measures the output of the instrumentation amplifiers to measure the currents of the chip supplies. The clock signals for the ASC and the SNN processor are provided via an external pulse generator (HP 8131A) and an arbitrary waveform generator (Rigol DG5101).

B. Analog-to-Spike Converter Measurement Results

This section discusses the measurement results of the standalone multi-channel ASC. Fig. 14 plots an 8-channel ECG recording, where the ASC resolution was set to 6 bits. The figure shows both the reconstructed ECG waveform (Fig. 14a) and the spiking output of the ASC (Fig. 14b). For all results, the non-uniform ASC output has been interpolated via zero-order-hold (ZOH) interpolation. The DAC mismatch coefficients were corrected by a one-time ramp-based foreground calibration. Fig. 15 shows the measured signal spectrum of a single readout channel for a 600-Hz sine input. The ASC achieves 71.1 dB of SNDR and 77.9 dB of SFDR in the 1-kHz signal band. The SNDR for different input frequencies is plotted in Fig. 16: the ASC achieves a peak SNDR of 74 dB. The in-band SNDR decreases at low frequencies, which is typical for an ASC, as harmonics fall into the signal band [12]. In Fig. 17, the measured SFDR of the ASC is shown for different input frequencies, with the DEM enabled and disabled. The DEM technique improves the linearity of the ASC by 7 dB. By applying both calibration and the DEM, the ASC can achieve an even higher linearity of up to 68 dB at low frequencies, which is the theoretical maximum SFDR for ZOH reconstruction for an 8-bit resolution [12]. For higher input frequencies, an SFDR degradation is noted, which

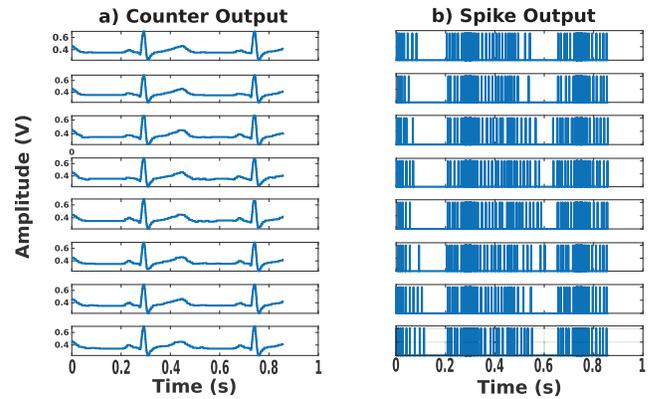


Fig. 14. 8-channel ECG recording from the ASC: a) the reconstructed output of each ASC channel, and b) the spiking output of each channel. The ASC settings used are: offset = 1 LSB, $T_{\text{decay}} = 0$ and ASC resolution = 6 bits.

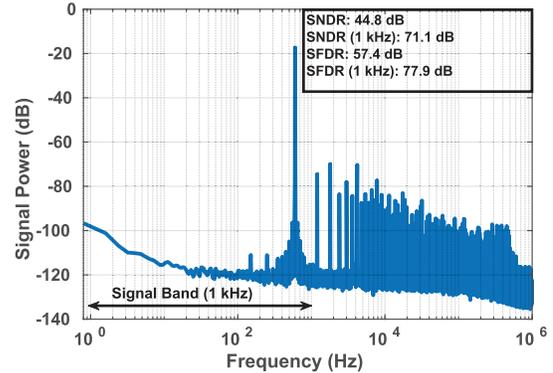


Fig. 15. Measured ASC spectrum of a single channel, for a 700-mVpp 600-Hz sine input.

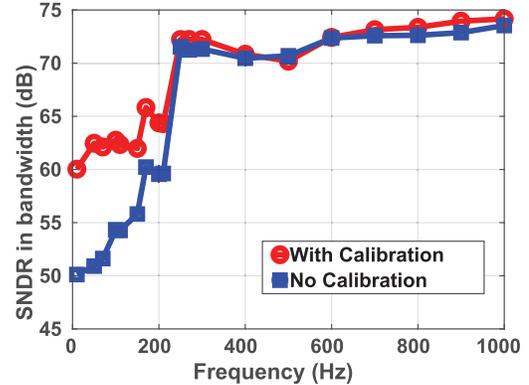


Fig. 16. Measured SNDR of the ASC output in the 1-kHz band as a function of the input signal frequency.

is has been traced back to a non-optimally designed bootstrap switch, which introduces distortion at higher input frequencies. For the application discussed in the sections below, however, this does not matter, as the achieved linearity with DEM (and thus without extra calibration) suffices for the classification task (which requires only a 6-bit linearity).

The ASC has a programmable resolution and conversion algorithm, which influence the ADC output data rate. Fig. 18 shows the output data rate for an ECG signal input for various settings of the ASC; the data rate of a classical Nyquist-rate ADC is plotted as reference. The ASC achieves a data rate that is much lower (e.g., $6\times$ at 6-bit resolution) than that of a classical, fixed-sample-rate conversion. Thanks to the dynamic encoding, the power consumption of the processing

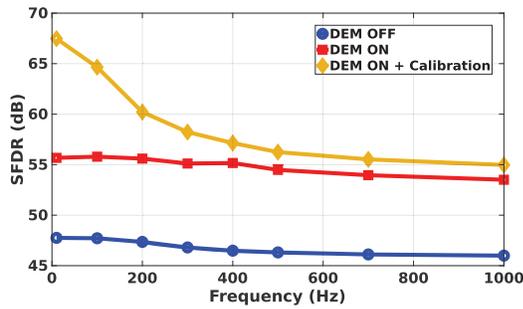


Fig. 17. SFDR of the ASC (of the full signal spectrum) as a function of the input signal frequency.

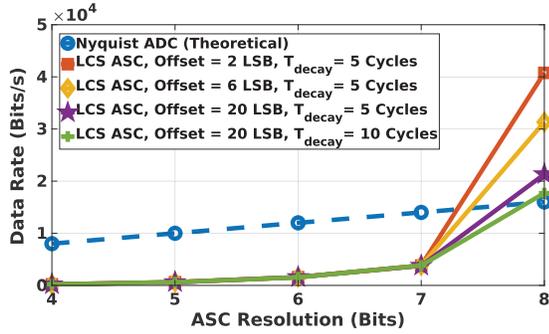


Fig. 18. Data rate of the ASC for an ECG input, plotted as a function of the ASC resolution.

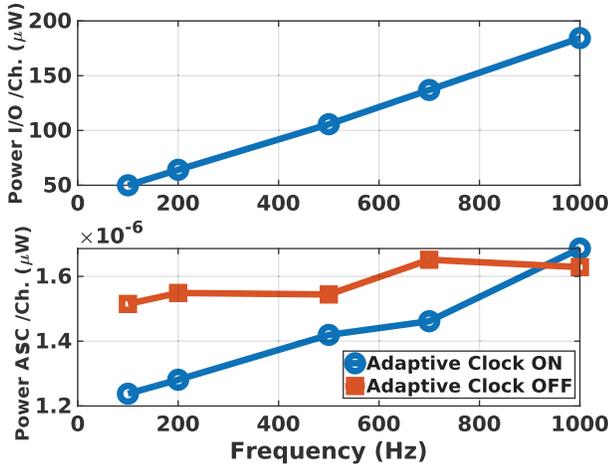


Fig. 19. I/O power of the SoC and the ASC power per channel as a function of the input frequency, with and without adaptive clocking.

and/or transmitter blocks that follow the ASC are also relaxed greatly. To highlight this, Fig. 19 shows the I/O power of the SoC as a function of the input tone frequency. The I/O power varies from 50 to 190 μ W/channel, clearly indicating how the event-driven ASC output can provide a benefit at system level. Fig. 19 also plots the power consumption as a function of the input frequency. Note that, with adaptive clocking on, the ASC power consumption scales with the input activity. The ASC consumes between 1.27 and 1.63 μ W/channel, and achieves a minimum Walden FOM of 199 fJ/conv.step. In Fig. 20, the ASC's area/channel is plotted for multiple designs from literature as a function of the DR of the ASCs. It clearly shows that the proposed ASC can significantly reduce the per-channel area. For this work a conservative estimation for the per-channel area is made; the digital circuits for the ASC and SNN are synthesized as one block, and hence, the logic circuits from the ASC and SNN are not separable (only the

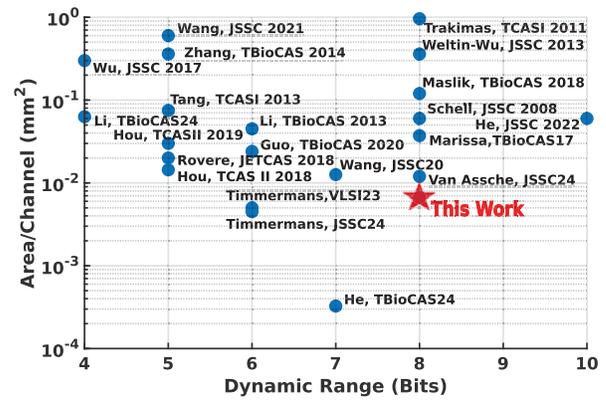


Fig. 20. Overview of area/channel of published LCS ASC designs, as a function of DR.

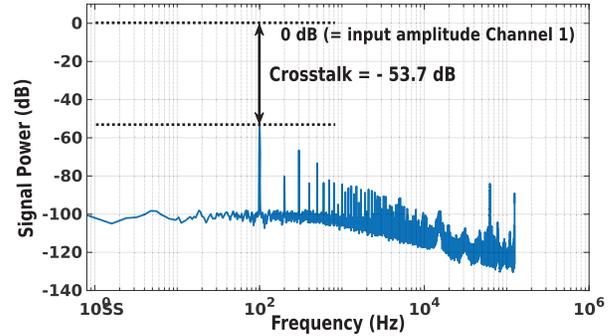


Fig. 21. Measured crosstalk in channel 2, when a full-scale, 100-Hz input signal is applied in the adjacent channel (channel 1). The crosstalk is measured by applying an input to channel 1 and grounding all other inputs.

SRAM can be distinguished). Therefore, for this result, the area of all digital logic (excluding the SRAM) is taken, which is clearly an overestimation. Moreover, since the proposed ASC has differential inputs, it uses 2 CDACs; thus, the CDAC array is twice as large compared to a single-ended design. Most designs in literature have a single-ended input. From the other designs in literature, the only reference that uses multiplexing is [37]. It uses a digital LCS technique, which can also be multiplexed, leading to a compact design that achieves an area of 0.000326 mm²/channel and a total of 128 channels. Finally, the crosstalk between the recording channels is plotted in Fig. 21. For this experiment all ASC channels have been connected to the ground, except channel 1 that has a full-scale 100-Hz input tone. The signal amplitude in the channel adjacent to the non-grounded channel has been verified, showing a crosstalk of -53.7 dB (Fig. 21).

C. SNN Measurement Results

The previous measurement results have shown how the multi-channel ASC provides a spike train for the back-end SNN in a power- and area-efficient manner. This section will discuss the measurement results for the SNN processor, for the QRS-labeling task discussed in Section IV. Fig. 22 shows the SNN power consumption (at 0.56V and 1V supplies) as a function of the input spike (or event) rate. While a sparser event rate significantly reduces the system power, the leakage and idle power set a lower bound on the benefits of the sparse signal input. Fig. 23 plots the SNN power consumption as a function of the supply voltage, the amount

TABLE I
COMPARISON WITH RECENTLY PUBLISHED SOCS FOR REAL-TIME BIOSIGNAL PROCESSING

Reference	TBioCAS23 [4]			JSSC21 [9]		JSSC22 [7]	Nat.Comm 21 [8]	ESSCIRC23 [6]	This Work
Technology (nm)	65			180		40	180	40	40
Supply Voltage (V)	0.75			0.6		0.9/1/1.1	1.8	0.6/1.1	0.56/1/0.7
Readout Channels	1			1		2	8	1	8
Area (mm ²)	1.74			13.86		0.32	1.42+77.2 (AFE+SNN)***	1.12	0.184
Total Per Channel	1.74			13.86		0.16	0.18+9.65 (AFE+SNN)***	1.12	0.023
Signal	ECG	EEG	EMG	ECG	Audio	ENG	EEG	ECG	ECG
Classification Task	Abnormality Detection	Seizure Detection	Gesture Recognition	Abnormality Detection	Keyword Spotting	De- Re- and Hyper Polarization Labeling	High Frequency Oscillation Detection	Abnormality Detection	QRS Labeling
Dataset	MIT-BIH. Arr.	Bonn University	NinaPro DB1	MIT-BIH. Arr.	Google Speech	Custom	Clinical iEEG	MIT-BIH. Arr.	MIT-BIH. Arr.
Accuracy (%)	99.3/99.16	99.84	85.13	99.7	99.4	/	78	95.31-97.81 (5b-7b ASC)	98.67**
Classes	2/5	2	8	2	2	3	2	5	3
Latency (s)	1.1	1	0.2	0.348		/	0.015	/	0.08*
Power/Channel (μ W)	46.8/86.7	32.1	30	1.68	0.378	25.3	76.75	/	20.8
Energy/Classification (μ J/class.)	2.25/4.36	2.06	5.25	0.58****	0.13****	/	9.21****	0.48-2.25 (5b-7b ASC)	13.29
Including AFE?	No			Yes		Yes	Yes	No (Off-Chip ASC used)	Yes
AFE Suited for Sensing?	/			No (5b)		Yes (10b)	Yes	/	Yes
Type of Processing	Frame Based NN			Event-Based AFE + Frame Based NN		Event-Based	Event-Based	Event-Based	Event-Based
Multipurpose?	Yes			Yes		No	No	Yes	Yes
Flexible Spike Encoding	/			No		No	No	No	Yes

*Latency measured via applying spikes via the external AER interface, SNN clock running at 15 MHz. Latency = amount of cycles between first input spike and last output spike/ $f_{clk,SNN}$
 **Software accuracy, hardware accuracy verified with 150 samples.
 ***AFE includes LNA, Filters and ASC.
 ****Value calculated from data in publication.

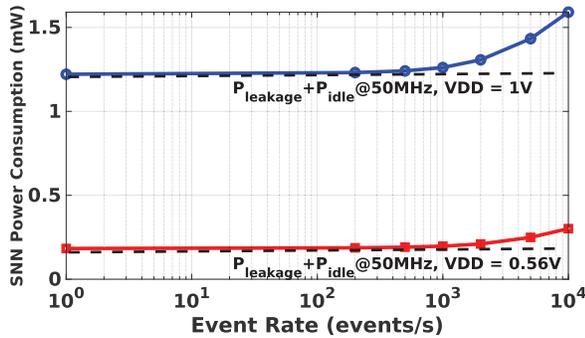


Fig. 22. Measured SNN power consumption as a function of the input signal event rate.

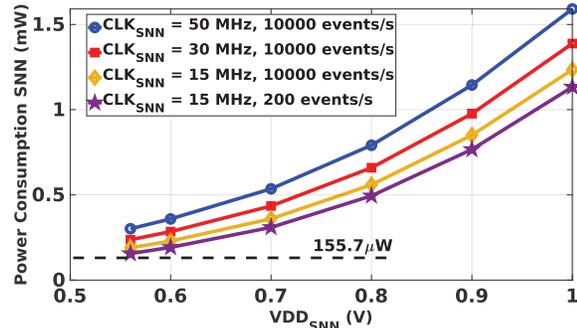


Fig. 23. Measured SNN power consumption as a function of the supply voltage, for various input signal event rates and SNN clock frequencies.

of input events (shown for 200 events/s and an accelerated input of 10000 events/s) and the clock frequency, where the programmed network architecture is the one described in Section IV. For a rate of 200 events/s, which is representative of the ECG use case, the SNN consumes a minimum total power of 155.7 μ W. The SNN core can work down to a digital supply of 0.56 V and a clock frequency of 50 MHz without losing functionality. For the SNN clock, the ASC provides a maximum spike rate of 2 MHz. It has been found experimentally that an oversampling ratio between the ASC spike rate and the SNN clock of at least 7 \times is required. Practically, an SNN clock of 15 MHz has been used.

D. System Results

Fig. 24 shows the system-level results for the QRS-labeling task. To validate the SNN accuracy obtained by software simulations in Section IV, 50 samples of each class were applied to the FREYA chip via the AER interface. For each sample, the class corresponding to the output neuron that spiked the most was taken as the classification result. These results are summarized in the confusion matrix in Fig. 24b. The figure also shows the system power breakdown of the SoC (Fig. 24a). For the classification task at hand, the total system power is 20.8 μ W/channel, of which 17.4 μ W/channel is static power consumption caused by leakage in the digital part, which is mostly due to the SRAM macros; 1.82 μ W/channel is idle power (clock on), 0.28 μ W/channel is due to the dynamic power consumption of the SNN, and 1.27 μ W/channel is from the ASC. A more detailed power breakdown for the ASC is provided as well (Fig. 24c): at low event rates, the power is mainly consumed by the always-on digital logic of the ASC, as the DAC and the comparator power scale with the signal activity. For the SoC, it is clear that the static power caused by leakage is the dominant contributor to the power consumption. It is important to note that standard SRAM cells provided by the foundry have been used, which are not optimized for low-power applications. By using custom low-power SRAMs [29], [30], or a technology with better leakage performance such as FDSOI, the power efficiency could be improved further.

The proposed SoC can greatly reduce the raw data bandwidth that is required from a medical device. Assuming that an ECG has 60 beats/s, the on-chip QRS-labeling SNN can reduce the data rate from an initial 128000 bits/s (the data rate of an 8-channel, 8-bit ADC with a sample rate of 2000 samples/s) to 6 bits/s, which corresponds to a compression of 21333 \times .

E. Comparison With the State of the Art

Table I lists recent SoCs for biosignal processing. Compared to these other systems, the FREYA SoC achieves at least a 7 \times reduction in per-channel area, thanks to the proposed

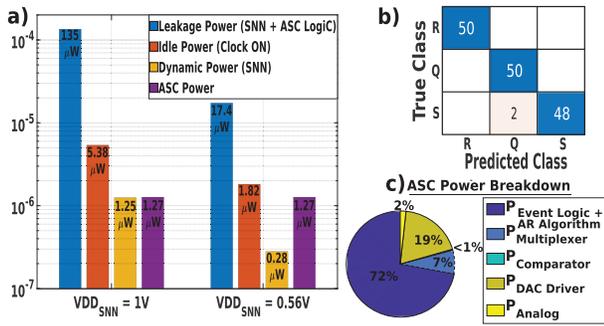


Fig. 24. a) The SoC power breakdown per channel for the QRS-labeling application, b) confusion matrix for the QRS-labeling task, and c) the detailed ASC power breakdown for the same application.

time-multiplexed ASC, which is the smallest reported in the literature, and the low-footprint digital SNN processor [16]. The SoC thus provides an area-efficient way to process the data from multiple recording channels, which is typically a bottleneck for event-driven SoCs. Compared to frame-based processing [4], [9], the proposed SNN achieves a much smaller latency (80 ms) and area footprint. The SoC consumes 20.8 μ W/channel, the lowest power consumption for all published systems; only [9] consumes less. However, as that system only uses a 5-bit LCS ASC, the DR of this LCS ASC is not sufficient, since biomedical systems usually require a DR higher than 7 bit. Note that for always-on systems, which need to continuously perform inference (such as the target ECG QRS-labeling application), the per-channel power needs to be minimized. The proposed SoC can thus provide a power-efficient LCS ASC and SNN processor, at a much smaller area footprint than other published works.

VI. CONCLUSION

This paper has presented FREYA, an 8-channel event-driven SoC for time-sparse biosignals, consisting of two main parts: a multi-channel, time-division-multiplexed LCS ASC and an integrated SNN processor. Thanks to the adaptive system clocking and the multiplexing, the proposed ASC can reuse the same hardware for each channel, resulting in an ultra-low per-channel area. The programmable adaptive-resolution algorithm makes it possible to tune the spike encoding to any signal of interest. Signal-activity-dependent power consumption is achieved due to the adaptive clocking of the ASC. To efficiently and rapidly tune the ASC and SNN towards a target application, a custom training framework for the SoC has been developed, which reduces the amount of required training epochs by 69% compared to a full grid search. The framework has been demonstrated for the QRS-labeling task based on ECG, for which a maximum accuracy of 98.67% has been reached. Finally, full measurement results of a prototype IC fabricated in a 40nm CMOS technology have been presented. The chip has a per-channel active area of 0.023 mm²/channel (0.184 mm² in total), which is a 7 \times improvement over other similar works. For the QRS-labeling task, the system consumes 20.8 μ W/channel, with an SNN clock running at 15 MHz (VDD at 0.56 V), with 80 ms of latency. This SoC therefore paves the way for multi-channel, signal-scalable, low-power event-driven SoCs for applications

that require the high-fidelity processing of multiple sensor signals.

ACKNOWLEDGMENT

The authors would like to thank the many MICAS researchers who helped during the tape-out of the chip.

REFERENCES

- [1] N. Van Helleputte et al., "A 345 μ W multi-sensor biomedical SoC with bio-impedance, 3-Channel ECG, motion artifact reduction, and integrated DSP," *IEEE J. Solid-State Circuits*, vol. 50, no. 1, pp. 230–244, Jan. 2015.
- [2] M. A. B. Altaf, C. Zhang, and J. Yoo, "A 16-channel patient-specific seizure onset and termination detection SoC with impedance-adaptive transcranial electrical stimulator," *IEEE J. Solid-State Circuits*, vol. 50, no. 11, pp. 2728–2740, Nov. 2015.
- [3] U. Shin et al., "NeuralTree: A 256-channel 0.227- μ J/Class versatile neural activity classification and closed-loop neuromodulation SoC," *IEEE J. Solid-State Circuits*, vol. 57, no. 11, pp. 3243–3257, Nov. 2022.
- [4] J. Liu et al., "An ultra-low power reconfigurable biomedical AI processor with adaptive learning for versatile wearable intelligent health monitoring," *IEEE Trans. Biomed. Circuits Syst.*, vol. 17, no. 5, pp. 952–967, Oct. 2023.
- [5] E. Forno, V. Fra, R. Pignari, E. Macii, and G. Urgese, "Spike encoding techniques for IoT time-varying signals benchmarked on a neuromorphic classification task," *Frontiers Neurosci.*, vol. 16, Dec. 2022, Art. no. 999029.
- [6] F. Tian et al., "BIOS: A 40nm bionic sensor-defined 0.47pJ/SOP, 268.7TSOPs/W configurable spiking neuron-in-memory processor for wearable healthcare," in *Proc. IEEE 49th Eur. Solid State Circuits Conf. (ESSCIRC)*, Lisbon, Portugal, Sep. 2023, pp. 225–228.
- [7] Y. He et al., "An implantable neuromorphic sensing system featuring near-sensor computation and send-on-delta transmission for wireless neural sensing of peripheral nerves," *IEEE J. Solid-State Circuits*, vol. 57, no. 10, pp. 3058–3070, Oct. 2022.
- [8] M. Sharifshazileh, K. Burelo, J. Sarnthein, and G. Indiveri, "An electronic neuromorphic system for real-time detection of high frequency oscillations (HFO) in intracranial EEG," *Nature Commun.*, vol. 12, no. 1, p. 3095, May 2021.
- [9] Z. Wang et al., "A 148-nW reconfigurable event-driven intelligent wake-up system for AIoT nodes using an asynchronous pulse-based feature extractor and a convolutional neural network," *IEEE J. Solid-State Circuits*, vol. 56, no. 11, pp. 3274–3288, Nov. 2021.
- [10] Z. Wang et al., "A software-defined always-on system with 57–75-nW wake-up function using asynchronous clock-free pipelined event-driven architecture and time-shielding level-crossing ADC," *IEEE J. Solid-State Circuits*, vol. 56, no. 9, pp. 2804–2816, Sep. 2021.
- [11] J. Van Assche and G. Gielen, "Analysis and Design of a 10.4-ENOB 0.92–5.38- μ W event-driven level-crossing ADC with adaptive clocking for time-sparse edge applications," *IEEE J. Solid-State Circuits*, vol. 59, no. 9, pp. 2858–2869, Sep. 2024.
- [12] T.-F. Wu, S. Dey, and M. S. Chen, "A nonuniform sampling ADC architecture with reconfigurable digital anti-aliasing filter," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 10, pp. 1639–1651, Oct. 2016.
- [13] T.-F. Wu, C.-R. Ho, and M. S.-W. Chen, "A flash-based non-uniform sampling ADC with hybrid quantization enabling digital anti-aliasing filter," *IEEE J. Solid-State Circuits*, vol. 52, no. 9, pp. 2335–2349, Sep. 2017.
- [14] C. Weltin-Wu and Y. Tsvividis, "An event-driven clockless level-crossing ADC with signal-dependent adaptive resolution," *IEEE J. Solid-State Circuits*, vol. 48, no. 9, pp. 2180–2190, Sep. 2013.
- [15] H. Wang, F. Schembari, and R. B. Staszewski, "An event-driven quasi-level-crossing delta modulator based on residue quantization," *IEEE J. Solid-State Circuits*, vol. 55, no. 2, pp. 298–311, Feb. 2020.
- [16] C. Frenkel, M. Lefebvre, J.-D. Legat, and D. Bol, "A 0.086-mm² 12.7-pJ/SOP 64k-synapse 256-neuron online-learning digital spiking neuromorphic processor in 28-nm CMOS," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 1, pp. 145–158, Feb. 2019.
- [17] T. Marisa et al., "Pseudo asynchronous level crossing ADC for ECG signal acquisition," *IEEE Trans. Biomed. Circuits Syst.*, vol. 11, no. 2, pp. 267–278, Apr. 2017.

- [18] A. Safa, J. Van Assche, C. Frenkel, A. Bourdoux, F. Catthoor, and G. Gielen, "Exploring information-theoretic criteria to accelerate the tuning of neuromorphic level-crossing ADCs," in *Proc. Neuro-Inspired Comput. Elements Conf.*, Apr. 2023, pp. 63–70.
- [19] J. E. Cavanaugh, "Unifying the derivations for the Akaike and corrected Akaike information criteria," *Statist. Probab. Lett.*, vol. 33, no. 2, pp. 201–208, Apr. 1997.
- [20] P. J. Werbos, "Backpropagation through time: What it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, Jan. 1990.
- [21] S. B. Shrestha and G. Orchard, "SLAYER: Spike layer error reassignment in time," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA. Curran Associates, 2018, pp. 1–10.
- [22] A. Safa et al., "Improving the accuracy of spiking neural networks for radar gesture recognition through preprocessing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 6, pp. 2869–2881, Jun. 2023.
- [23] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–15.
- [24] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.
- [25] B. Jacob et al., "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 2704–2713.
- [26] B. Razavi, "The bootstrapped switch [A circuit for all Seasons]," *IEEE Solid State Circuits Mag.*, vol. 7, no. 3, pp. 12–15, Sep. 2015.
- [27] B. Razavi, "The StrongARM latch [A circuit for all seasons]," *IEEE Solid State Circuits Mag.*, vol. 7, no. 2, pp. 12–17, Jun. 2015.
- [28] P. J. A. Harpe et al., "A 26 μ W 8 bit 10 MS/s asynchronous SAR ADC for low energy radios," *IEEE J. Solid-State Circuits*, vol. 46, no. 7, pp. 1585–1595, Jul. 2011.
- [29] B. Vanhoof and W. Dehaene, "SRAM with stability monitoring and body bias tuning for biomedical applications," *IEEE Solid-State Circuits Lett.*, vol. 5, pp. 29–32, 2022.
- [30] Y.-C. Chien and J.-S. Wang, "A 0.2 V 32-Kb 10T SRAM with 41 nW standby power for IoT applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 8, pp. 2443–2454, Aug. 2018.
- [31] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Eng. Med. Biol. Mag.*, vol. 20, no. 3, pp. 45–50, 2001.
- [32] A. L. Goldberger et al., "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
- [33] M. Sharifshazileh and G. Indiveri, "An adaptive event-based data converter for always-on biomedical applications at the edge," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Monterey, CA, USA, May 2023, pp. 1–5.
- [34] S. Narayanan, M. Cartiglia, A. Rubino, C. Lego, C. Frenkel, and G. Indiveri, "SPAIC: A sub- μ W/channel, 16-channel general-purpose event-based analog front-end with dual-mode encoders," in *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Toronto, ON, Canada, Oct. 2023, pp. 1–5.
- [35] X. Zhang and Y. Lian, "A 300-mV 220-nW event-driven ADC with real-time QRS detection for wearable ECG sensors," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 6, pp. 834–843, Dec. 2014.
- [36] M. Timmermans, K. van Oosterhout, M. Fattori, P. Harpe, Y.-H. Liu, and E. Cantatore, "A 1.8–65 fJ/Conv.-step 64-dB SNDR Continuous-time level crossing ADC exploiting dynamic self-biasing comparators," *IEEE J. Solid-State Circuits*, vol. 59, no. 4, pp. 1194–1203, Apr. 2024.
- [37] Y. He et al., "An event-based neural compressive telemetry with $>11\times$ loss-less data reduction for high-bandwidth intracortical brain computer interfaces," *IEEE Trans. Biomed. Circuits Syst.*, vol. 18, no. 5, pp. 1100–1111, Oct. 2024.
- [38] M. D. Alea et al., "A fingertip-mimicking 12×16 200 μ m-Resolution e-skin taxel readout chip with per-taxel spiking readout and embedded receptive field processing," *IEEE Trans. Biomed. Circuits Syst.*, early access, Apr. 11, 2024, doi: [10.1109/TBCAS.2024.3387545](https://doi.org/10.1109/TBCAS.2024.3387545).



Jonah Van Assche (Member, IEEE) received the joint M.Sc. degree in nanotechnology from Katholieke Universiteit Leuven (KU Leuven), Leuven, Belgium, and the Kungliga Tekniska Högskolan Stockholm (KTH), Stockholm, Sweden, in 2018, and the Ph.D. degree in electrical engineering from KU Leuven in 2024. He is currently a Post-Doctoral Researcher with the University of Stuttgart, Germany. His research interests include mixed-signal circuits for adaptive sensor readout and event-based processing circuits.



Charlotte Frenkel (Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical engineering from the Université Catholique de Louvain (UCLouvain), Louvain-la-Neuve, Belgium, in 2015 and 2020, respectively. In February 2020, she joined the Institute of Neuroinformatics, UZH, and ETH Zurich, Switzerland, as a Post-Doctoral Researcher. Since July 2022, she has been an Assistant Professor with Delft University of Technology, The Netherlands. Her current research targets neuromorphic edge intelligence, with a focus on spiking neural network processor design, embedded machine learning (tinyML), and on-chip training algorithms.



Ali Safa (Member, IEEE) received the M.Sc. degree in electrical engineering from the Université Libre de Bruxelles, Brussels, Belgium, and the Ph.D. degree in AI-driven processing for extreme edge applications from the Katholieke Universiteit Leuven (KU Leuven). Then, he joined IMEC and KU Leuven in 2020. He has been a Visiting Researcher with UC at San Diego, La Jolla, USA, in Spring 2023. He is currently an Assistant Professor with the College of Science and Engineering, Hamad Bin Khalifa University (HBKU), Doha, Qatar. His research interests include neuromorphic AI, continual learning, and sensor fusion for robot perception.



Georges Gielen (Fellow, IEEE) received the M.Sc. and Ph.D. degrees in electrical engineering from Katholieke Universiteit Leuven (KU Leuven), Leuven, Belgium, in 1986 and 1990, respectively. After a Post-Doctoral Researcher with UC Berkeley since 1991, he has been with the MICAS Research Group, Department of Electrical Engineering (ESAT), KU Leuven, where he is currently a Full Professor. His research interests include the design of analog and mixed-signal integrated circuits, such as sensor interfaces and data converters and analog and mixed-signal CAD tools and EDA.