A mixed-signal spatio-temporal signal classifier for on-sensor spike sorting.

***Abstract*—In this paper, we combine recent progress in neuromorphic computation and neuromorphic mixed-signal hardware to present the first step towards an implementation of a neuromorphic spike sorting algorithm, that has been proven able to extract and decode spikes, in real time. This implementation is based on TSMC 180nm technology. Combined with a neural recording system, we anticipate this approach to leverage efficient neuromorphic brain-machine interfaces for embedded rehabilitation prosthetic control**

I. Introduction

Real time neural activity decoding is essential for brain-machine interfaces (e.g. for prosthetics), and to enable closed-loop experiments in neuroscience. Prior work [1] has shown that neural activity in the human brain can be decoded in real-time from Multi Electrode Array (MEA), after a daily recalibration of the system. These systems require a wired connection, as it is a challenge for a wireless system to deal with the amount of recorded data while keeping heat dissipation to a required minimum. Spike sorting is a fundamental pre-processing task, providing discrimination between signals generated by different neurons, but recorded by the same electrode (or, as in this work, by a set of adjacent electrodes). This is achieved through the classification of the shapes of the recorded neural activation pulses (spikes). If done near sensor, this has the potential to improve system latency, and significantly reduce data rates, enabling wireless systems. Several hardware approaches to spike sorting have been presented [2] [3] [4].

Conventional signal processing systems use Nyquist-rate sampled and quantised signals. Introducing a different data representation can lead to significant gains in performance and power efficiency. It has been shown that an event-based approach can offer advantages in spatio-temporal pattern recognition tasks [5] [6] [7] [8] [9]. The availability of neuromorphic sensors such as silicon retinas [10] [11] and silicon cochleas [12] [13] will lead to further development of algorithms handling these event-based signal representations. Instead of the classic scheme of periodic sampling, these neuromorphic sensors only transmit data whenever there is a significant change in the signal, leading to a sparse representation and providing high temporal precision at low data bandwidth. An event-based neural recording platform has been recently introduced [14], and will be used in this work as a source of neural recording data. In this work, we present a 180nm CMOS implementation of a spike sorting algorithm presented in [15]. Our ultimate goal is to integrate the presented circuitry into the recording electrode array. Section II details the topology and behavior of the proposed system, Section III provides circuit descriptions, and Section IV shows simulations results that are compared to the expected behavior of the original algorithm.

II. Concept

A. Towards an event-based representation

Unlike its synchronous counterpart, the event-based approach, similar to Lebesgue sampling [16], asynchronously transmits *events* for a pre-set change in the signal level (Figure 1). For a signal coming from a Multi-Electrode Array, we can define an event ev as a tuple of a time of appearance t, a spatial position in the array (x, y) and a polarity p, indicating the direction of the change. $p = 1$ (ON event) indicates that the signal increased, as $p = 0$ (OFF event) indicates that the signal decreased:

$$ev_i = (t_i, x_i, y_i, p_i) \tag{1}$$

where i stands for the i-th event.

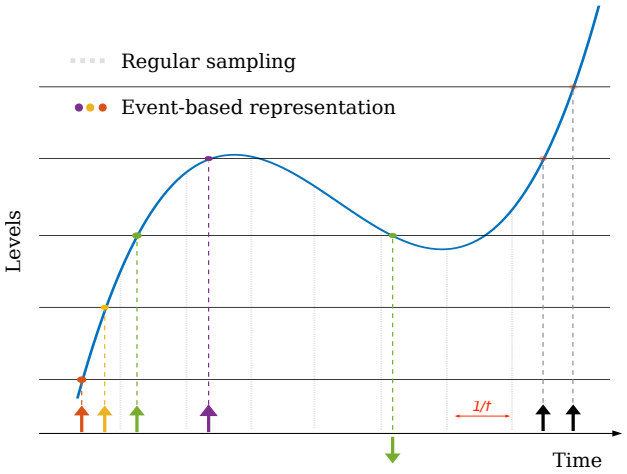

Fig. 1: Event generation. Each time the signal change, since the time of the last event (arrow), reaches a certain level, a new event is generated. The direction of this change, or polarity, is represented in the orientation of the arrow (up/down). For comparison, the grey dashed lines represent the standard, synchronous samples, uniformly distributed in time.

B. Original algorithm

Using the precise timing of the event-based representation of the input signal, we can introduce the spatio-temporal context S^i of the i-th event ev_i, representing the past activity on a given surrounding, centered around the incoming event. This

context is based on the work presented in [8] and [7], and build by sampling event traces generated through exponential decays:

$$S^i_{u,v} = \exp\left(-\frac{t_i - t_{u,v}}{\tau}\right) \; for \begin{cases} u \in [\![x_i - r \; ; \; x_i + r]\!] \\ v \in [\![y_i - r \; ; \; y_i + r]\!] \end{cases} \tag{2}$$

where $t_{u,v}$ is the timestamp of the last event at the given (u, v) position, r the surrounding size (here, $r = 1$), and τ the time constant of the event trace decaying unit.

Then, this spatio-temporal context is compared to learned templates, in order to find the closest one. Then, a classification unit can be used, in order to assign the corresponding class to the input spiking pattern.

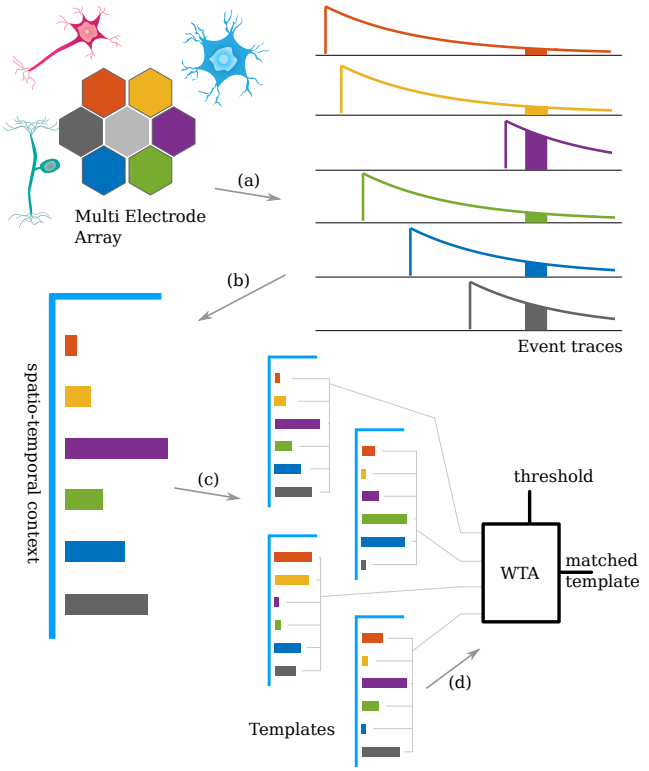

Fig. 2: Functionality of the proposed chip. (a) The hexagonal array records signals from neurons, generating events on multiple channels [14]. Each time an event is triggered, an event trace is generated. When the central event occurs (light gray electrode), the value of the traces on neighbouring channels is memorized (b), forming a spatio-temporal context. For the sake of understanding, only ON events are represented here. This context is then compared to 4 stored templates (c). The currents resulting from these comparisons are passed to a Winner-Takes-All block, in order to determine which template is the closest to the current context.

C. Constraints

The array chip (32×32 electrodes) is $3.4 \times 2.8mm$. All the circuitry for a single electrode needs to fit within $96 \times 79\mu m$ [14]. The power consumption should be as low as possible (the recording array consumes $145\mu W$ [14]), not only to allow for efficient wireless systems but also to keep tissue damage caused by heat dissipation to a minimum. Given the characteristics of biological signals, the decay time constant for the event trace should be tunable from a few μs up to ms.

III. Circuit Description

Our chip needs three main computational blocks: an exponential decay unit to form the event traces composing the spatio-temporal contexts; a comparison unit to compute a distance between the presented context and learned templates; a unit that selects the template that provides the closest match. The circuits implementing these blocks are presented in this section, followed by the overall architecture of our chip.

A. Exponential decay unit for the event trace

The time constant of the decaying unit has to range from μs to ms. A straightforward implementation of an resistor-capacitor (RC) circuit can implement exponential voltage decay, but to provide a tuneable time constant and a small circuit area, we implement this with a switched capacitor circuit, controlled by an external clock source. Figure 3a shows the basic circuit. Making the two switches (P_1 and P_2) for this switched capacitance large enough, the intrinsic drain/source capacitances are sufficiently large to avoid the need of an external capacitance (C_{ds} in Figure 3a). The non-linearity is not a problem, as it simply modifies the overall distance function of the comparison unit (see Section IV-B). The incoming spike V_{spk} resets the capacitance C to V_1 (via N_1), ensuring a quick discharge ($\sim 40ns$), that is negligible given the considered time range. Then, the capacitance charges towards V_2, and the voltage V_{trace} is fed to the next module, for comparison to the template value. Here, the capacitance C has a size of $10 \times 10\mu m$, for a value of $200fF$, while C_{ds} is approximately 24x smaller. Variations of the control clock lead to change in the time constant as shown in Table I.

Frequency	τ
5 kHz	4.3 ms
500 kHz	76 μs
f	24/f

TABLE I: Exponential decay time constant τ for the event trace, versus clock frequency, for the switched-capacitor circuitry. Extracted from post layout simulations.

B. Comparison Unit

When the central event occurs, the comparison of each event trace value to an external template value is triggered. The difference between the template value V_a and the voltage V_{trace} is obtained by a bump-antibump circuit [17], shown in Figure 4, outputting the bump-current I_{bump} defined as:

$$I_{bump} = \frac{I_b}{1 + \frac{4}{S}\cosh^2\frac{\kappa \Delta V}{2}} \tag{3}$$

where I_b is the bias current controlled by the V_b, S is the ratio between transistor sizes of (N_4, N_7) and (N_5, N_6), κ the

transconductance of the transistors and $\Delta V = V_a - V_{trace}$ the voltage difference.

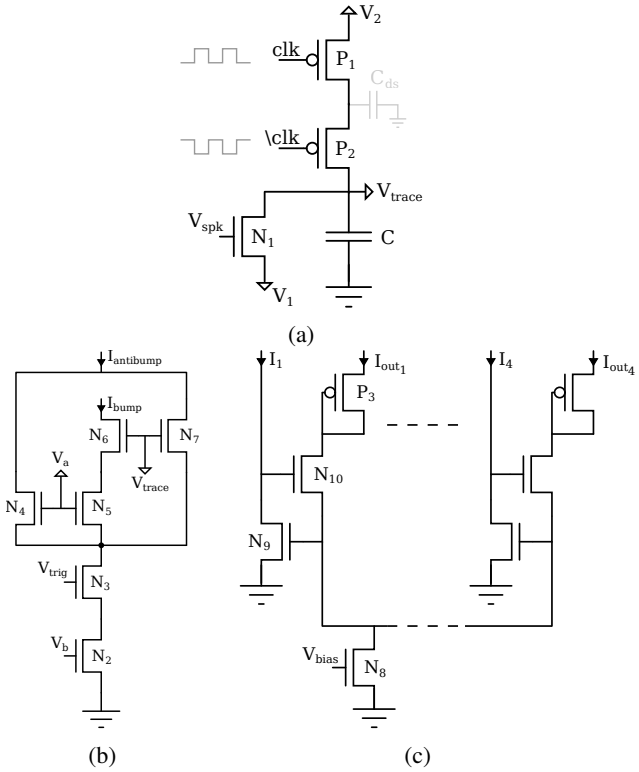

Fig. 3: Basic cells schematics: (a) Event trace circuitry; (b) Comparison circuitry [17]; (c) Basic WTA circuit [18], two inputs are shown. See text for details.

C. Template matching

The currents representing the differences between event traces and template values obtained as described above, are summed over the entire template in a trivial manner, simply adding currents. This provides the overall template match current for each of the templates. A set of 4 templates is considered sufficient to achieve desired recognition rates in our application (according to [15]). For each neighboring pixel (6 in the hexagonal case), we then need 4 comparison units, giving 32 comparison units in total. Each one of these units needs an analog reference value, which in the current design is provided externally.

D. Winner take all

The summed currents of comparison circuits represent distances between the current spatio-temporal context and the templates. The selection of the best matching template is done by a Winner-Take-All (WTA) circuit. The inputs currents will compete for activation, where only the largest one will be chosen as a winner. A simple yet efficient implementation of a current-mode WTA circuit was brought forward by Lazzaro et al. [18], using only two transistors per input channel. This design was chosen to minimize the delay of deciding for the closest matching current, as well as space efficacy. However this also makes the circuit more susceptible to mismatch. As can be seen in Figure 3c, all 4 input cells are connected to a global node that provides a bias current (N_8). This bias current is controlled by an externally provided voltage V_{bias}.

The output of the comparison units for the 4 templates (I_1 to I_4) are fed into the WTA circuitry, in order to select the one with the highest current. The WTA will set the corresponding output I_{out1} to I_{out4} to $\approx I_{bias}$, all other outputs will be suppressed (unless the multiple inputs are very closely matched). The output currents are binarised, to provide an indicator of the winning template (in a multi-layer version of the classifier, this would generate an event on the template channel). We also include a circuit to compare the winning current magnitude with a threshold current, to provide a confidence bit indicating if the winner is valid (i.e. sufficiently closely matched to a template, as opposed to a best match out of four very poor matches).

E. Implementation

All the above described blocks are assembled in our core, as shown in Figure 6. The aim of this first prototype being to validate the principle and quantify the effects of noise and variability, we replicate the core block many times, multiplexing different intermediate signals to the bonding pads, in order to be able to carry out comprehensive measurements.

The core comprises 64 blocks with digital outputs (4 binary WTA output + 1 valid bit, 64 blocks with an analog output (current output of the distance to each template), 64 blocks with access to the decaying unit voltage, and 1 full digital block. All the blocks, excepted the last one, are multiplexed to the pads, via three different 64:1 multiplexers.

The system uses 4 templates, of 6 values each. Each one of these is an analog reference value. For now, to limit the chip's complexity (at the cost or an increased number of pads), these 32 analog values are fed from an external source. In the final implementation we will use SRAM memory and 32 DACs on-chip such as the one presented in [19]. An alternative that we are exploring in this project, is to use non-volatile analog memory devices (memristors) integrated with the CMOS process. Memristors could be then also used to provide programmable decay in a modified event trace unit circuit.

IV. Simulations

A. Benchmark for performance evaluation

In order to quantify the performances of our implementation, we used artificially generated Multi-Electro-Array (MEA) recordings [20]. This generated signals where filtered accordingly to the amplification stages of the recording unit [14] and then converted to events, as shown in Figure 5.

B. Event trace and distance

Figure 4 presents the simulation of the realized event trace, with its associated template value, and the the output of the distance circuit. The exponential time constant is here set to $\tau = 1$ ms, and the template value is 1.1 V, which corresponds

to a peak response at around 1.2 ms. The implemented distance function is much sharper than traditional L_1 and L_2 norms used in the original algorithm [15], but is proven to perform well in our target application (see next section).

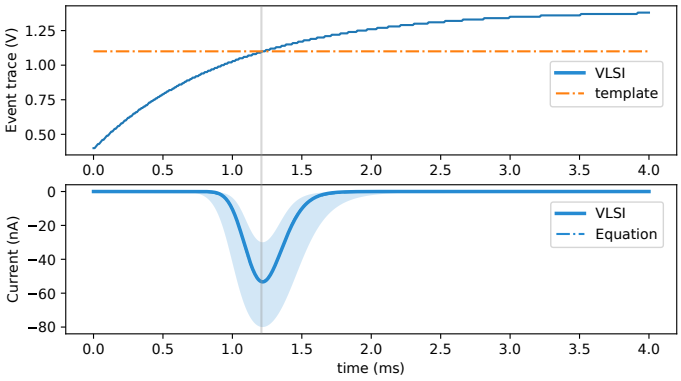

Fig. 4: Circuit simulation results: (top) event trace generated by the decay unit; (bottom) distance between the trace and the fixed template value. Blue line is the simulated distance, the blue shadowed area shows the impact of the fabrication mismatch (Monte Carlo simulation, for the bump circuit only) on the obtained distance. Dashed blue is the analytic curve that follows equation (3) and matches well the simulation (maximal error of 0.2nA; less than 1% to the average value). Distances calculated using L_1 and L_2 norms is shown in red for comparison.

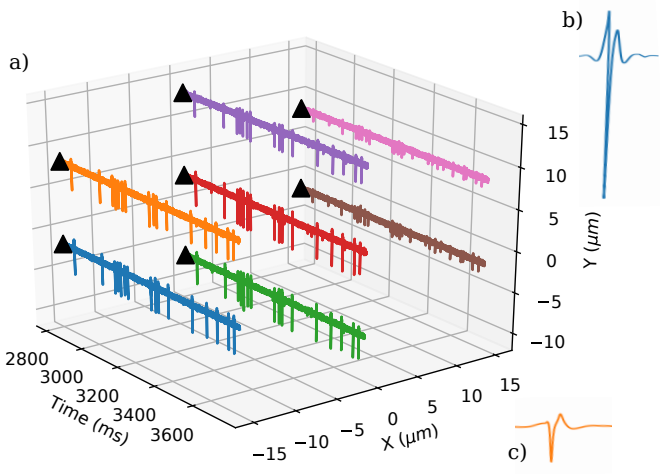

Fig. 5: a) Simulated recordings [20] for the probe topology. Here, an hexagonal probe is used. We only show 7 electrodes, in the same configuration as for our chip [21] [22]. Ground truth is available for classification performance estimation. b) and c) 2 different spike shapes extracted from the dataset.

C. Classification rate

The dataset was split in 2 sets ; one training set, containing 1370 spikes and a testing set, containing 930 spikes. Training and testing is done using the distance behavior as extracted from post-layout simulations, including variability. Classification scores are given in Table II. We can expect this chip to behave almost as well as the method introduced in the original paper, with a computational time highly reduced, specifically for hierarchical structures, where the algorithmic complexity explodes, due to the increasing number of templates [8], opposite as in our chip where the *computation time* grows lineary with the number of layers. Our simulations achieve a score of 75/88% on a 1/2-layers architecture, with a computation time of 80/160ns, which is the propagation time in the digital circuitry and that has not been optimized. Regardless the complexity of a hierarchical structure, the propagation time evolve in $\mathcal{O}(n)$. Computation time for the original algorithm was around $5\mu s$ for a single layer architecture,using Python code running on a Core i7-8700K @ 3.7 GHz computer.

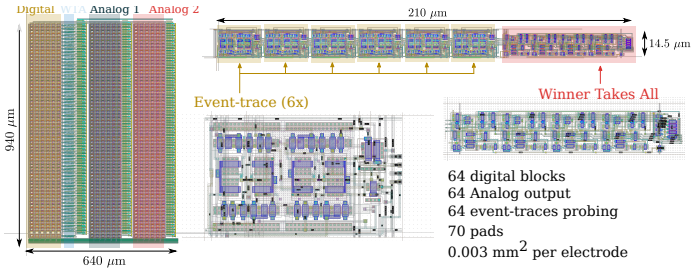

Fig. 6: Layout of the hereby described chip. The core is composed of 3×64 basic blocks (6 decaying units, 4 template matching and a WTA. The outputs are multiplexed to spare output pads. The core occupies a space of $640 \times 940 \mu m$, each individual block being $14.5 \times 210 \mu m$).

Distance	Recognition rate	
	1 layer	2 layers
$\mathcal{L}_1$	60%	68%
$\mathcal{L}_2$	73%	82%
Bhattacharyya [23]	78%	89%
This work	75%	88%

TABLE II: Recognition rate for different distance metrics, and our implemented model. We can notice that the bump distance performs almost as good as the Bhattacharyya [23], which is significantly more complex to implement on a chip.

V. Conclusion

The presented system is a first step towards a full neuromorphic signal processing pipeline for neural decoding applications. It implements the essential primitive computational blocks that will be embedded below each pixel of our recording array. Due to their event output, these computational blocks can be chained to form a hierarchical processing pipeline in more complex processing scenarios than considered in this paper. All the results were obtained via simulations (including post-layout and Monte Carlo), further work will be to fully characterize the fabricated chip, specifically to quantify the parameter variations due to fabrication mismatch and analyze the impact of those on the classification result. Our aim is to design a fully functional integrated Micro-Electrode Array

system, with on-chip spike sorting. We are also working on embedding memristive memories for parameter configuration. We anticipate this work to ultimately enable low-power embeddable brain-machine interfaces.

REFERENCES

[1] B Wodlinger, JE Downey, EC Tyler-Kabara, AB Schwartz, ML Boninger, and JL Collinger. Ten-dimensional anthropomorphic arm control in a human brain- machine interface: difficulties, solutions, and limitations. *Journal of neural engineering*, 12(1):016011, 2014.

[2] Thilo Werner, Elisa Vianello, Olivier Bichler, Daniele Garbin, Daniel Cattaert, Blaise Yvert, Barbara De Salvo, and Luca Perniola. Spiking neural networks based on oxram synapses for real-time unsupervised spike sorting. *Frontiers in Neuroscience*, 10:474, 2016.

[3] Sivylla E Paraskevopoulou, Deren Y Barsakcioglu, Mohammed R Saberi, Amir Eftekhar, and Timothy G Constandinou. Feature extraction using first and second derivative extrema (fsde) for real-time and hardware-efficient spike sorting. *Journal of neuroscience methods*, 215(1):29–37, 2013.

[4] Ian Williams, Song Luan, Andrew Jackson, and Timothy G Constandinou. Live demonstration: A scalable 32-channel neural recording and real-time fpga based spike sorting system. In *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 1–5. IEEE, 2015.

[5] Thusitha N Chandrapala and Bertram E Shi. The generative adaptive subspace self-organizing map. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, pages 3790–3797. IEEE, 2014.

[6] Amos Sironi, Manuele Brambilla, Nicolas Bourdis, Xavier Lagorce, and Ryad Benosman. Hats: Histograms of averaged time surfaces for robust event-based object classification. *arXiv preprint arXiv:1803.07913*, 2018.

[7] Gregory K Cohen, Garrick Orchard, Sio-Hoi Leng, Jonathan Tapson, Ryad B Benosman, and André Van Schaik. Skimming digits: neuromorphic classification of spike-encoded images. *Frontiers in neuroscience*, 10:184, 2016.

[8] Xavier Lagorce, Garrick Orchard, Francesco Galluppi, Bertram E Shi, and Ryad B Benosman. Hots: a hierarchy of event-based time-surfaces for pattern recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1346–1359, 2016.

[9] Saeed Afshar, Libin George, Jonathan Tapson, André van Schaik, and Tara J Hamilton. Racing to learn: statistical inference and learning in a single spiking neuron with adaptive kernels. *Frontiers in neuroscience*, 8:377, 2014.

[10] Christoph Posch, Daniel Matolin, and Rainer Wohlgenannt. A qvga 143 db dynamic range frame-free pwm image sensor with lossless pixel-level video compression and time-domain cds. *Solid-State Circuits, IEEE Journal of*, 46(1):259–275, 2011.

[11] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014.

[12] Vincent Chan, Shih-Chii Liu, and Andr van Schaik. Aer ear: A matched silicon cochlea pair with address event representation interface. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 54(1):48–59, 2007.

[13] Shih-Chii Liu, André Van Schaik, Bradley A Mincti, and Tobi Delbruck. Event-based 64-channel binaural silicon cochlea with q enhancement mechanisms. In *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pages 2027–2030. IEEE, 2010.

[14] Federico Corradi and Giacomo Indiveri. A neuromorphic event-based neural recording system for smart brain-machine-interfaces. *IEEE transactions on biomedical circuits and systems*, 9(5):699–709, 2015.

[15] Germain Haessig, Kevin Gehere, and Ryad Benosman. Spikes decoding spikes : A neuromorphic event-driven framework for real-time unsupervised spike sorting. *in Press*, 2019.

[16] Karl Johan Astrom and Bo M Bernhardsson. Comparison of riemann and lebesgue sampling for first order stochastic systems. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 2, pages 2011–2016. IEEE, 2002.

[17] Tobi Delbruck. 'bump'circuits for computing similarity and dissimilarity of analog voltages. In *IJCNN-91-Seattle International Joint Conference on Neural Networks*, volume 1, pages 475–479. IEEE, 1991.

[18] John Lazzaro, Sylvie Ryckebusch, Misha Anne Mahowald, and Caver A Mead. Winner-take-all networks of o (n) complexity. In *Advances in neural information processing systems*, pages 703–711, 1989.

[19] Tobi Delbruck, Raphael Berner, Patrick Lichtsteiner, and Carlos Dualibe. 32-bit configurable bias current generator with sub-off-current capability. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pages 1647–1650. IEEE, 2010.

[20] Alessio P Buccino and Gaute T Einevoll. Mearec: a fast and customizable testbench simulator for ground-truth extracellular spiking activity. *bioRxiv*, page 691642, 2019.

[21] Joana P Neto, Gonçalo Lopes, João Frazão, Joana Nogueira, Pedro Lacerda, Pedro Baião, Arno Aarts, Alexandru Andrei, Silke Musa, Elvira Fortunato, et al. Validating silicon polytrodes with paired juxtacellular recordings: method and dataset. *Journal of neurophysiology*, 116(2):892–903, 2016.

[22] Richárd Fiáth, Bogdan Cristian Raducanu, Silke Musa, Alexandru Andrei, Carolina Mora Lopez, Chris van Hoof, Patrick Ruther, Arno Aarts, Domonkos Horváth, and István Ulbert. A silicon-based neural probe with densely-packed low-impedance titanium nitride microelectrodes for ultrahigh-resolution in vivo recordings. *Biosensors and Bioelectronics*, 106:86–92, 2018.

[23] Anil Bhattacharyya. On a measure of divergence between two multinomial populations. *Sankhyā: the indian journal of statistics*, pages 401–406, 1946.