Hardware-Software Co-Design for Brain-Computer Interfaces

Ioannis Karageorgos*
Dept of Electrical Engineering
Arts & Sciences
Yale University
ioannis.karageorgos@yale.edu

Karthik Sriram*
Dept of Computer Science
Arts & Sciences
Yale University
karthik.sriram@yale.edu

Ján Veselý*
Dept of Computer Science
Arts & Sciences
Rutgers & Yale Universities
jan.vesely@cs.rutgers.edu

Michael Wu
Dept of Computer Science
Arts & Sciences
Rutgers University
mw811@rutgers.edu

Marc Powell
School of Engineering
Carney Institute for Brain Science
Brown University
marc_powell@brown.edu

David Borton
School of Engineering
Carney Institute for Brain Science
Brown University
david_borton@brown.edu

Rajit Manohar
Dept of Electrical Engineering
Arts & Sciences
Yale University
rajit.manohar@yale.edu

Abhishek Bhattacharjee
Dept of Computer Science
Arts & Sciences
Yale University
abhishek@cs.yale.edu

***Abstract*—Brain-computer interfaces (BCIs) offer avenues to treat neurological disorders, shed light on brain function, and interface the brain with the digital world. Their wider adoption rests, however, on achieving adequate real-time performance, meeting stringent power constraints, and adhering to FDA-mandated safety requirements for chronic implantation. BCIs have, to date, been designed as custom ASICs for specific diseases or for specific tasks in specific brain regions. General-purpose architectures that can be used to treat multiple diseases and enable various computational tasks are needed for wider BCI adoption, but the conventional wisdom is that such systems cannot meet necessary performance and power constraints.**

We present HALO (Hardware Architecture for LOw-power BCIs), a general-purpose architecture for implantable BCIs. HALO enables tasks such as treatment of disorders (e.g., epilepsy, movement disorders), and records/processes data for studies that advance our understanding of the brain. We use electrophysiological data from the motor cortex of a non-human primate to determine how to decompose HALO's computational capabilities into hardware building blocks. We simplify, prune, and share these building blocks to judiciously use available hardware resources while enabling many modes of brain-computer interaction. The result is a configurable heterogeneous array of hardware processing elements (PEs). The PEs are configured by a low-power RISC-V micro-controller into signal processing pipelines that meet the target performance and power constraints necessary to deploy HALO widely and safely.

I. Introduction

Brain-computer interfaces (BCIs) can treat neurological diseases, shed light on our understanding of the brain, and enable new brain-computer interactions [27, 42, 54, 77]. Researchers have already demonstrated BCIs that can control prostheses, treat neurological disorders (e.g., epilepsy, Parkinson's disease, anxiety, and schizophrenia), and navigate augmented realities [33, 38, 46, 52, 54, 55, 81, 82, 112, 115, 121, 122].

Many BCIs are realized as headsets or electrodes placed on the scalp, and use electromagnetic signals emanating over the skull from biological neurons to deduce brain activity [42, 54, 77]. While these devices do not require surgical deployment, the signals they collect are noisy and low-resolution, making them less ideal as a source of control signal for forward-looking BCI applications [34, 41, 74, 86]. In these cases, a better alternative—and the focus of our study—is to surgically embed BCIs directly on, around [111], and in the brain tissue [15]. Such proximity enables implantable BCIs to record from and stimulate large numbers of neurons with high signal fidelity, spatial resolution, and in real time [104]. Consequently, implantable BCIs are already being used by over 160K patients worldwide [71] and are actively being developed by companies like Kernel [6], Longeviti [7], Neuropace [12], Medtronic [14], and Neuralink [74]. The question of how to build low-power hardware for on-board processing is critical to the success of these devices.

BCIs targeting large numbers of neurons have thus far been realized with custom ASICs that treat only certain diseases or perform specific tasks in specific brain regions. Flexible hardware that supports multiple tasks and treats multiple diseases is needed for wider BCI adoption. However, the few programmable BCIs that have been built to date process only a limited number of neurons while meeting the low-power requirements necessary for implantation in the brain [3, 4]. Table I shows such limitations for a set of cutting-edge commercial and research BCIs from Medtronic [3, 4], Neuropace [106], and others [23, 56, 84].

In response, we architect BCI hardware sufficiently flexible to treat multiple disorders and enable many brain interactions, yet also adequately low-power for safe and chronic implantation in the brain. Our approach, HALO (**H**ardware **A**rchitecture for **LO**w-power BCIs), balances the flexibility of general-purpose processing with the power-efficiency of specialized hardware. Table I shows that HALO offers:

Safety: FDA, FCC, and IEEE guidelines state that implantable BCIs must not dissipate more than 15-40mW of power,

*** Joint first authors who have contributed to this work equally. Authors are listed in alphabetical order of last name.**

depending on the target brain region [37, 48, 67, 89, 102, 116]. Dissipating more power can heat surrounding brain tissue by more than 1 °C, causing cellular damage [13, 60, 116, 125].

Flexibility: Different brain regions use different neural circuits and require different processing algorithms. For example, neuropsychiatric disorders can manifest in the dorsal and orbital prefrontal cortices, the amygdala, hypothalamus, and ventral striatum (among others) [20, 24, 61, 92]. Patients with one neurological disorder often suffer from others; e.g., patients diagnosed with epilepsy are 8× likelier to develop Alzheimer's and 3× likelier to develop Parkinson's diseases [57, 96]. Table I shows that commercial devices generally target one disease, but HALO can be configured to treat any of the diseases targeted by existing BCIs.[1] HALO is also extensible, enabling support for emerging neural processing algorithms undergoing active research [24, 32, 39, 80, 90].

Performance: Many BCIs are closed-loop. For example, BCIs for epilepsy process neuronal signals to predict seizures, and then electrically stimulate neurons via on-board neurostimulators to mitigate the severity of these seizures. To be effective, the time between seizure onset and stimulation must be within tens of milliseconds [66, 106, 117]. Because such neurological disorders are influenced by structural and functional networks across brain centers [25, 29, 62], BCIs must read/stimulate many *channels* (sensors that interact with biological neurons) at high resolution and sampling frequency. It is difficult to build low-power hardware that can process large data streams in real time. As shown in Table I, Medtronic and Neuropace devices read/stimulate a handful of channels with low frequency, while Kassiri et. al. [56] and NURIP [84] support tens of channels. Emerging Neuralink devices [74] support thousands of channels, but consume 750mW and are not safe for chronic implantation. As programs like DARPA NESD [1] target recording/stimulating millions of channels, these challenges will be exacerbated. HALO is inspired by previous approaches, particularly efforts like NURIP, which implements sophisticated seizure prediction pipelines, and other similar studies (see §VII), but offers higher bandwidth brain communication in real time at 15mW.

Research contributions: To realize HALO, we first identify a list of BCI tasks to support in §III. This list includes disease treatment, signal processing, secure transmission of neuronal data (e.g., compression and encryption of extracellular voltage streams), and subsumes the capabilities of many cutting-edge BCI devices. Since BCIs are an active area of research, this list is not exhaustive. Nevertheless, it offers a viable design path for us to identify a broader set of tasks needed for a flexible BCI platform, select functional units with which to realize them, and then consider how best to integrate them.

To tame a large design space of possible architecture and integration options, we devise a hardware-software co-design approach to systematically architect HALO. Standard low-power design dictates that we realize one accelerator per BCI task (e.g., compression, seizure prediction, etc.) in the form of a dedicated ASIC. We refer to this as a *monolithic ASIC* design. We show, however, that monolithic ASICs exceed the 15mW power budget permissible for safe BCIs in many cases.

We consequently take an alternate approach, and refactor the underlying algorithm of the original BCI tasks into distinct pieces that realize different phases of the algorithm. We refer to these pieces as kernels, and show that they facilitate design of ultra-low-power hardware processing elements (or PEs) via novel hardware-software co-design approaches summarized in Table II. We round out the design with a low-power RISC-V micro-controller to configure PEs into processing pipelines and support computation for which there are currently no PEs. The result is an unconventional style of heterogeneity, where a family of accelerators operates in unrelated clock domains with ultra-low-power asynchronous circuit-switched communication. The design has the following properties:

[1]We currently envision configuring one task at a time, but HALO can be easily extended to run as many parallel BCI tasks as can fit within the power budget. We will investigate this further in future work.

	Medtronic [10]	Neuropace [106]	Aziz [23]	Chen [37]	Kassiri [56]	Neuralink [74]	NURIP [84]	**HALO**
Tasks Supported								
Spike Detection	×	×	×	×	×	×	×	✓
Compression	×	×	✓	×	×	×	×	✓
Seizure Prediction	×	✓	×	✓	✓	×	✓	✓
Movement Intent	✓	×	×	×	×	×	×	✓
Encryption	×	×	×	×	×	×	×	✓
Technical Capabilities								
Programmable	✓	Limited	×	Limited	✓	×	Limited	✓
Read Channels	4	8	256	4	24	3072	32	**96**
Stimulation Channels	4	8	0	0	24	0	32	**16**
Sample Frequency (Hz)	250	250	5K	200	7.2K	18.6K	256	**30K**
Sample Resolution (bits)	10	10	8	10	-	10	16	**16**
Safety (<15mW)	✓	✓	✓	×	✓	×	✓	✓

TABLE I: Medtronic devices support movement intent to mitigate dystonia, essential tremors, and Parkinson's disease. These devices are flexible but not power-efficient, and have limited channel counts and reduced spatial/temporal resolution. Furthermore, they are single-task. For example, Aziz delta compresses and exfiltrates data to enable offline analysis. Neuropace, NURIP and Chen et. al. implement sophisticated algorithms and are sufficiently low power for safe use, but have restricted programmability. Neuralink supports higher bandwidth communication with the brain but at the cost of 750mW, precluding real-world use in patients for now. HALO supports the functionality of all other BCIs while offering high spatial and temporal sensor resolution within 15mW.

Technique	**Direction**	**Section**
Kernel PE Decomposition	SW→ HW	IV-A
PE Reuse Generalization	SW→ HW	IV-A
PE Locality Refactoring	SW← HW	IV-A
Spatial Reprogramming	SW← HW	IV-B
Counter Saturation	SW↔ HW	IV-B
NoC Route Selection	SW→ HW	IV-D

TABLE II: Overview of hardware-software co-design techniques used in HALO. Each line indicates design influence and the paper section that discusses details of the technique.

(a) *Each PE operates in its own clock domain at the minimum frequency to sustain target performance.* This reduces power consumption versus monolithic ASICs. Research questions (see §IV-A) involve identification of kernel boundaries within BCI tasks – sometimes they are naturally available, but often they require refactoring of the original algorithm.

(b) *PEs enable hardware-software co-design.* PE decomposition allows identification of kernels that are common across BCI tasks. Adding support for generalized PE reuse via architectural configurability saves area and power (§IV-A). PEs also enable going beyond recent work on spatial programming [79, 83, 85, 126] and permitting hardware-directed software refactoring of BCI algorithms (§IV-B), decomposition of PEs into even more power efficient mini-PEs by leveraging data locality properties of the source algorithm (§IV-B), and novel counter saturation techniques (§IV-B). These innovations are orthogonal and complementary to well-known techniques like pipelining and reduced precision, which we also use.

(c) *PEs enable a simple communication fabric.* PE decomposition creates static data-flow routes that allow use of an ultra-low-power circuit-switched network for inter-PE communication, and eschew the need for conventional power-hungry packet-switched on-chip networks. The network is made up of programmable switches that permit the doctor/technician to configure the PEs to realize different BCI tasks at runtime. The network is extensible and can accommodate additional PEs for future BCI tasks.

We evaluate HALO with electrophysiological data collected in vivo from a non-human primate's arm and leg motor cortex, the brain regions responsible for arm and leg movement. Every task, from closed-loop seizure/tremor mitigation, to spike detection and extracellular voltage stream compression, fits under 15mW [89]. HALO achieves 4-57× and 2× lower power dissipation than software alternatives and monolithic ASICs respectively.

Overall, this work goes beyond single accelerator design, and offers a blueprint for ultra-low-power multi-accelerator SoC design. It also offers a suite of algorithm-to-hardware co-design techniques that are complementary to circuit-level optimizations explored in related work on low-power embedded systems in wearables and implantable BCIs [59, 64, 84, 93].

Longer-term directions: This work is a first step in a longer-term research project involving several generations of HALO tape-outs with progressively more complete implementation. At the time of writing this paper, we have performed functional validation, validation of synthesized design, and physical synthesis (multi-corner/multi-mode) to achieve timing and power closure with a margin for all HALO components. In addition, we have taped out a modified RISC-V core with all the hooks necessary to interface with the rest of the architecture. We will add components one step at a time in incremental tape-outs to manage design risk. Figure 1 shows a layout diagram of the first chip tape-out, which we have submitted for fabrication in a 28nm technology.

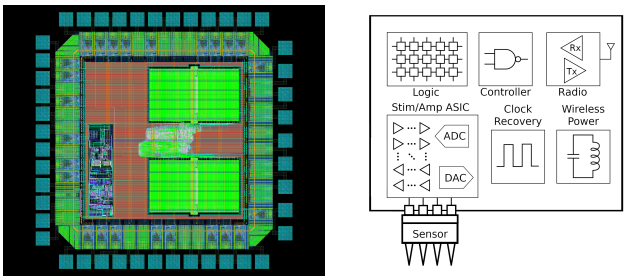

Fig. 1: On the left is a circuit diagram of our first HALO tape-out in a 28nm technology. On the right is an implantable BCI. BCIs have form factors under 1cm^2 and are placed on brain tissue, where their sensors probe millimeters into the tissue. The devices can be packaged in a hermetically-fused silica capsule or titanium capsule.

II. Background

On the right in Figure 1, we show an example implantable BCI. Building blocks beyond processing hardware include:

① Sensors: BCIs use sensors ranging from single electrodes for individual neurons to arrays of hundreds of microelectrodes, which record and stimulate 5-10 neurons individually and several hundred in total [22, 23, 58]. Going forward, sensors will record from an ever-increasing number of biological neurons. For example, widely-used Utah arrays already integrate up to 256 microelectrode channels [16, 22]. Although not immediately practical, approaches like Neuralink's "threads" and DARPA NESD performers are targeting thousands to millions of channels [1, 74].

② Analog front-end: The analog data recorded from the sensors must be amplified and digitized via analog-to-digital converters (ADCs). Different BCIs use ADCs with different sample resolution and frequency, but 8-16 bits per sample at 20-50 KHz are common [68, 74, 88, 89].

③ Communication links: BCIs use RF links that vary from the low MHz to GHz range, with 2.4GHz being typical [119]. Since RF deposition heats up brain tissue, the FCC and FDA limit the specific absorption rate to 1.6W/kg over 1g of tissue, and 1W/kg over 10g of tissue, respectively [13, 94, 110, 125]. Therefore we aim for power budgets of 15mW, while also minimizing radio transmission power.

④ Power sources: BCIs are typically powered by single-use non-rechargeable batteries, rechargeable batteries, or inductive power transfer. All must be judicious with power. Non-rechargeable batteries require service lifetimes of 12-15 years, as they require surgery for replacement [3, 26]. Rechargeable batteries and inductive powering both require transcutaneous wireless powering [43, 45, 67, 119], and must reduce the transferred power so as to prevent excessive heating.

HALO must be compatible with any variant of ①-④ and must meet two goals to be widely usable: none of the BCI tasks should exceed 15mW, and RF transmission bandwidth should be minimized to mitigate power deposited in brain tissue.

III. List of Supported BCI Tasks

We support many BCI tasks, some of which treat neurological disorders via closed-loop operation, and others that reduce radio transmission via compression of neuronal activity:

① Seizure prediction: Implantable BCIs used to treat epilepsy predict seizure onset from neuronal firing patterns and, if a seizure is predicted, electrically stimulate biological neurons in certain brain regions [14, 106]. Electrical stimulation breaks feedback loops in the neural circuits responsible for seizures, thereby mitigating seizure severity [14, 37, 56, 106]. Seizure prediction is an active area of research in the neuroscience community. State-of-the-art seizure prediction algorithms use FFTs, cross-correlation, and bandpass filters over a linear model. We implement an algorithm that combines all three complementary approaches in HALO [99]. As such, it is an exemplar of other closed-loop BCI algorithms used to treat major depressive disorder, psychosis, and obsessive-compulsive disorder [66, 117].

② Movement intent: For individuals with essential tremor, Parkinson's disease, and other movement disorders, therapeutic stimulation of the motor cortex can relieve symptoms [49]. Implantable BCIs can continually stimulate the brain, but this wastes energy when the affected limb is unused, and can lead to medical side effects [2, 49]. A better option is to stimulate brain tissue when neuronal firing indicates use of the affected limb [49]. Similarly, for paralyzed individuals, neuronal signals can be decoded to determine how to control prostheses [35, 107, 114]. Such approaches have been demonstrated on non-human primates [35], and require millisecond latency processing between detection of movement and stimulation of the brain [113]. State-of-the-art algorithms exploit the fact that movement intent is correlated with drops in neuronal firing in the 14-25Hz band in the motor cortex region, which can be detected using an FFT [49, 108].

③ Compression: Compression reduces radio transmission, and is useful for high-bandwidth brain interaction [23, 28, 118]. One may consider using lossy compression, but the brain is not understood well enough to identify what portions of the electrophysiological data can be safely discarded [98]. Apart from some specific and well-understood forms of lossy compression – including spike detection, which we discuss subsequently – lossless compression is more widely used and palatable to the neuroscience community today [91]. In HALO, we support several lossless compression schemes, as their compression ratios and power consumption can vary depending on brain region and patient activity. We support well-known LZ4 [8] and LZMA [9] compression, as well as a custom-built discrete wavelet transform (DWT) [44] compression. In §VI, we show that compression ratios vary by as much as 40% depending on compression algorithm and target brain region.

④ Spike detection: This is the first step in spike sorting pipelines that extract activity of specific neurons from the recorded signal. Spike sorting is performed on an external system, but we include spike detection on the BCI as it sends only the parts of the signal that contain a detected spike, effectively compressing transmitted data. Due to the relative rarity of spikes, spike detection lowers signal transmission bandwidth by orders of magnitude [44], reducing both device power and power deposited on the brain tissue by the radio. Spike detection is typically implemented using a non-linear energy operator (NEO) or using DWT [44].

⑤ Encryption: Although current state-of-art devices do not currently support encryption, we foresee it to be necessary for future BCIs in order to protect patient data during exfiltration off the device. HIPAA, NIST, and NSA require using AES with an encryption key of at least 128 bits [5, 11, 109].

IV. Hardware-Software Co-Design

§III describes five tasks, two of which are realizable in multiple ways. Compression can be achieved with LZ4, LZMA, and DWT, while spike detection can be achieved with NEO and DWT. Hence, HALO can be configured by a doctor/technician at runtime into one of eight distinct pipelines.

With conventional monolithic ASICs, we would implement 8 ASICs, one per task. Instead, HALO supports these tasks via PEs that realize distinct processing kernels, as shown in Figure 2. Each PE operates at a frequency catered to its specific computational needs and includes processing logic, private memory, and an adapter to communicate over the interconnect. While there may be benefits to giving PEs access to a global memory via caches, we leave this for future studies.

Decomposing BCI tasks into PEs lets us determine dataflow and enables an ultra-low-power on-chip circuit-switched network for asynchronous inter-PE communication rather than a conventional power-hungry packet-switched network. Many of our PEs, like LZ and FFT, require computational resources that scale with the number of sensor channels, increasing power/area usage. To address this, we implement a standalone interleaver that buffers and rearranges data so that these PEs can be time-multiplexed to operate on a single channel at a time. We complete HALO by integrating a low-power micro-controller, which assembles the PEs into pipelines that realize the BCI tasks via programmable circuit switches in the network and runs algorithms for which there are no PEs yet.

HALO's PE-centric approach eschews the need for a global clock or phase-locked loops. Furthermore, bundling computational kernels within PEs with private memory means that HALO is naturally modular and extensible. As we learn more about the brain's function, we may want to support more BCI tasks and our architecture will naturally permit insertion of additional PEs for emerging neuroscientific algorithms.

A. PE Decomposition

The process of decomposing BCI algorithms into PEs varies in complexity, depending on how clearly separated the algorithmic phases are. Importantly, PE decomposition must not change algorithmic functionality; i.e., there should be no change in algorithmic accuracy, compression ratio, etc.

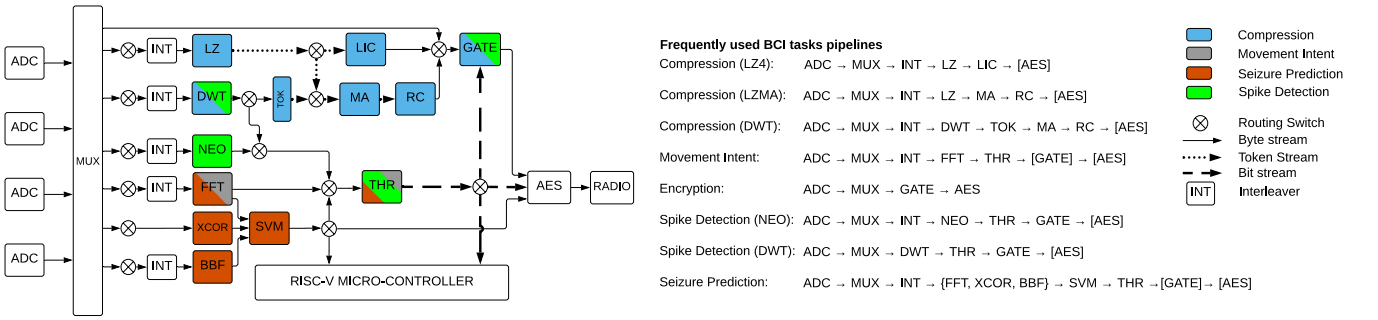

Fig. 2: ADCs digitize the analog neuronal signals and feed them into processing logic, which consists of low-power hardware PEs and a RISC-V micro-controller. PEs are configured into pipelines to realize BCI tasks, ranging from compression (in blue) to spike detection (in green). Optional PEs (e.g., AES encryption) are shown in square brackets. PEs operating in parallel (e.g., FFT, XCOR, and BBF in the seizure prediction pipeline) are shown in curly brackets. All BCI tasks are under 15mW.

Kernel PE decomposition: Some BCI tasks consist of distinct computational kernels naturally amenable to PE decomposition. For example, seizure prediction combines kernels for Fourier transform (FFT), cross-correlation (XCOR), Butterworth Bandpass Filtering (BBF), and a support vector machine (SVM). We realize each as a PE as shown in Figure 2. As FFT, XCOR, and BBF have no data dependencies, they can operate in parallel. This approach saves power because XCOR contains complex computation (e.g., divisions, square roots) that scales quadratically with channel count. In contrast, BBF is a simple filter with minimal arithmetic that scales linearly with channel count. Separating XCOR and BBF into separate PEs ensures that BBF's filtering logic is clocked over an order of magnitude slower than the logic for cross-correlation.

PE reuse generalization: Many BCI tasks use popular computational kernels in slightly different variants. To exploit this, we take inspiration from functional unit sharing from CPU microarchitecture to develop configurable PEs that can be shared among BCI tasks. Consider, for example, movement intent, which can be decomposed into FFT, followed by logic that checks whether the FFT output is in a particular spectral range. We create a threshold PE (or THR) to determine when a PEs output is within a specified numerical range (see Figure 2) and enable sharing of the FFT between movement intent and seizure prediction tasks. The FFT PE is configurable because movement intent requires 14-25-point FFTs to detect drops in signal power, while seizure prediction requires 1024-point FFTs [49, 99, 108]. We find the increased complexity of configurable PEs to be worth the cost (see §VI), as it enables reuse and provides a more versatile platform.

As another example, consider spike detection, which can be implemented with either nonlinear-energy operator (NEO) or DWT [44]. In either case, the output is fed to the THR PE, permitting reuse of THR with movement intent. Moreover, the DWT PE can be shared with one of the compression pipelines, as shown in Figure 2. Like the FFT PE, the DWT PE must be configurable to permit sharing, because spike detection requires recursive applications of DWT (usually three, four, or five times [44]), while compression requires only one.

Figure 2 shows that we also share logic for the Lempel-Ziv pattern search in a single PE for the LZMA and LZ4 compression tasks. To enable sharing of LZ, we perform intra-PE optimizations, as discussed in the next section.

Algorithm 1 LZMA pseudocode

```
1:  function LZMA_COMPRESS_BLOCK(input)
2:      output = list(lzma_header);
3:      while data = input.get() do
4:          best_match = find_best_match(data);
5:          Prob_match = count(table_match, best_match)
6:              /count_total(table_match);
7:          r1 = range_encode(Prob_match);
8:          output.push_back(r1);
9:          increment_counter(table_match, best_match);
10:     end while
11:     return output;
12: end function
```

Major refactoring: PE decomposition can, in many cases, be more complicated and require significant refactoring of the original algorithm. Consider, for example, LZMA and DWTMA compression. Both algorithms use Markov (MA) chains to calculate the probability of the current input value based on observed history, which is used to pick more efficient encoding of the input signal. We found that using the combined MA PE overshoots the 15mW power budget. To solve this problem, we refactored the original algorithm to make it more amenable for PE decomposition. To separate algorithmic phases, we realize that data locality (i.e. following routines that manipulate major data structures) is a good indicator of kernel boundaries within programs. This observation is tied to the fact that PEs in HALO have only local memories and cannot share large amounts of data. *Locality refactoring* highlights how design decisions about the architecture (i.e., use of PE-local memories) guided refactoring of our algorithms.

Algorithm 1 and Figure 3 demonstrates how we use this insight to change LZMA. The second half of this algorithm can be separated into probability calculations and frequency information updates centered around the maintenance of the core MA data structure, the frequency table (in green), as well as efficient encoding (in blue). Figure 3 shows a simplified block diagram of MA hardware before and after algorithmic

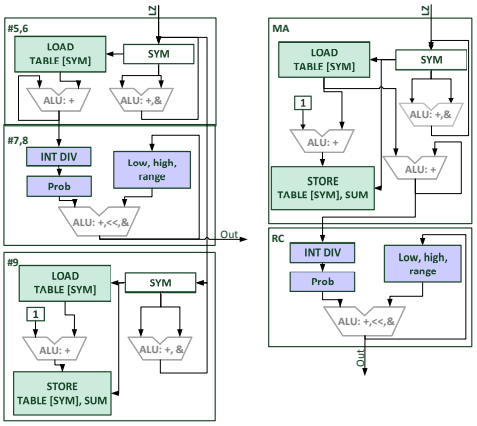

Fig. 3: (Left) Hardware for the initial version of MA, with input from LZ (see Algorithm 1). Hardware corresponding to the lines of code from the algorithm are indicated by the numbers in the diagram. The first and last steps in MA share the symbol table. Output is produced by the second block, after which control passes to the third block to maintain data structures. (Right) After refactoring, MA can be split into PEs that realize the new MA (at the top), and RC (at the bottom).

refactoring. The key memory structures, the frequency table and the encoder state, are again shown in green and blue respectively, matching their color coding in Algorithm 1. We then refactor the algorithm to bring together phases that operate on the same data structures, allowing us to separate the PEs since they can now operate independently with minimal data movement. This permits clocking each component at significantly lower frequency, leading to power savings of 2×.

B. Processing Element Optimizations

Once PEs are identified, they offer additional opportunities for hardware-algorithm co-design. Many of the optimizations do not change the functional behavior of the algorithm, but others do modify the output.

Unchanged PE output: Some of the PEs (e.g., XCOR, LZ) process data in blocks instead of samples. These PEs must wait for all inputs in the block to arrive. When all inputs arrive, computation occurs in a burst, and an output is produced. This type of bursty computation is problematic as it requires either large buffers to sink the bursts, or high PE frequency to meet data rates while sustaining periods of bursty activity. Neither option is ideal from the perspective of saving power.

In response, we note that recent work on spatial programming [79, 83, 85, 126] devised techniques to tolerate such bursty activity. While the original work focuses on hardware support to overcome this problem, we extend this work to *spatially reprogram* the original algorithm and co-design it with the hardware to achieve power improvements.

As an example, consider the XCOR PE. The original algorithm, shown in Algorithm 2, performs computation at the end once all data has been filled into the block. We refactor the algorithm to process inputs as early as they are available. The final form in Algorithm 3 reduces the amount of computation needed in the final step, as well as the number of buffers needed to store the inputs. This translates to a power savings of 2.2× over the original algorithm. This technique also extends to other PEs like LZ to achieve 1.5× power reduction.

Algorithm 2 XCOR naive implementation

```
function XCOR(input, output)
    // channel[][] stores input in appropriate channel location
    channel[channel_num][sample_num] = input
    // Calculate correlation
    if channel.filled() then
        for each i, j ∈ channels do
            data_i = 0
            for each data ∈ channel[i] do
                data_i+ = data
            end for
            data_j = 0
            for k ∈ [LAG, SIZE] do
                data_j+ = channel[j][k]
            end for
            avg_i = data_i/SIZE
            avg_j = data_j/SIZE
            output.push_back(avg_i, avg_j)
        end for
        return output;
    end if
end function
```

Additionally, HALO benefits from application of known architectural techniques to save power. These include pipelining, parallelizing computation with additional hardware, and more. For example, we use pipelining optimizations to reduce frequencies for PEs like XCOR, NEO, BBF, and SVM. Consider, for example, the XCOR PE. XCOR calculates the cross-correlation of pairs of channels using three inner loops for computing means, sums, and square roots, each of which is individually amenable to pipelining. We note that pipelining a circuit does not reduce the power it dissipates per se. However, reducing the critical path of a circuit enables us to utilize the available timing slack for gate downsizing and voltage scaling, which can enable substantial power savings. Pipelining XCOR in this manner saves 1.4× power.

Finally, LZ and MA PEs require initialization of data structures at the beginning of every compressed block. We found that dedicated circuits are necessary to meet the 15mW budget. These circuits use only combinational logic. For example, the circuit used in MA contains a few inverters and AND gates per input bit. Using these circuits instead of standalone initialization phase reduces PE power consumption by 1.8×.

Modified PE output: Although initialization circuits decrease the direct power/performance cost of starting a new compression block, there is also an indirect cost of using uninitialized internal structures, which leads to lower compression rates. This presents a problem with respect to the choice of block size. On one hand, large block sizes lead to better estimates

of frequencies, and therefore better compression ratios, which ultimately saves radio transmission power. On the other hand, small block sizes allow the use of smaller data types and reduce the memory footprint and power of the MA PE.

A traditional approach would aim to balance power/compression ratio for an ideal design. However, such an approach does not find a design point that fits within the constrained power budget. Instead we observe that the frequencies of values within a block remain largely unchanged after they have stabilized. Consequently, we allow the frequency counters to saturate and set block size independently of counter bit width.

Overall, *counter saturation* modification allows HALO to benefit both from reduced memory footprint of 16 bit counters, and better compression ratio of larger blocks. It is also yet another example of co-design where software modification can complement hardware design. Like our other software refactoring techniques, counter saturation ensures that no data is lost. In other words, compression ratio may decrease (marginally, as we find) but the data can still be correctly decoded because the frequencies used to guide encoding schemes do not affect the accuracy of what is encoded. We explore the effect of changing the block size on power and compression ratio in §VI-D.

HALO also benefits from application of known architectural techniques that trade data precision for power improvement. Where possible, we use fixed-point rather than floating point computation. We also reduce bit width for fixed-point integers. Although some of the signal processing algorithms that we study use 32-bit integers in the original studies [99], such high-resolution representation is often unnecessary and can be reduced to save power without significantly impacting accuracy. Knowing the limits of signal data, we replace floating point arithmetic with fixed point arithmetic in the BBF PE and achieve an order of magnitude reduction in power, with only $< 0.1\%$ increase in relative error. When using fixed point, reducing RC's 32-bit integers to 16-bit integers saved PE power by $1.6\times$ with no change in accuracy.

Algorithm 3 XCOR spatial programming refactoring

```
function XCOR(input, output)
    // channel[][] stores input in appropriate channel location
    channel[channel_num][sample_num] = input
    // data[] stores sums of input received so far
    data[count] += input
    // data_lag[] stores sums of input till LAG
    if count_2 == LAG then
        data_lag[count] = data[count]
    end if
    // Finish correlation computation
    if channel.filled() then
        for each i, j ∈ channels do
            avg_i = data[i]/SIZE
            avg_j = (data[j] − data_lag[j])/SIZE
            output.push_back(avg_i, avg_j)
        end for
        return output
    end if
end function
```

C. Processing Element Summary

Table III lists HALO's PEs. We offer partial PE parameterization to enable sharing (see §IV-A) and to personalize the algorithm to patients. As an example, recent studies show that it is possible to modify the number of weights and values in the SVM PE to improve seizure prediction accuracy [19, 100]. To permit such personalization, we expose as many as 5000 weights while remaining within the power budget.

PE parameters influence the total PE memory capacities. Table III quantifies the upper bound of these sizes. For example, the LZ PE's memory size is determined by finding matches of the current byte sequence in its history. The doctor/technician can reduce history size via the micro-controller if desired. In such cases, we power-gate unused memory banks.

D. On-Chip Network

We clock each PE at the lowest frequency needed to meet data processing rates, and synthesize with established synchronous design flows (see Sec. V). Local (intra-PE) synchronization is based on per-PE pausable clock generators and clock control units [123]. The clock generators use ring oscillators with a delay line which is extracted from the critical path. The ring oscillator is designed so that its frequency variation tracks the critical path [73]. One might expect ring oscillator-based clocking to inhibit circuit performance due to increased clock uncertainty. This is not the case in HALO because of the low operating frequency of PEs with respect to the achievable performance of our target process node. All PEs were synthesized with hundreds of pico-seconds of positive timing slack, minimizing – if not completely negating – the impact of increase in clock uncertainty.

While running PEs in separate clock domains saves power, it can potentially complicate design of inter-PE communication. Prior work on globally asynchronous locally synchronous (GALS) architectures [36, 63, 72] encountered these issues for packet-switched on-chip networks. Unfortunately, we cannot re-purpose their solutions as our analysis with the DSENT tool [105] estimates that a simple packet-switched mesh network consumes over 50mW, well over our 15mW power budget. Instead, we co-design inter-PE communication with the BCI algorithms. The decomposition of BCI tasks into kernels creates static and well-defined data-flows between PEs. *NoC route selection* allows replacement of a packet-switched network to a far lower-power circuit-switched network built on an asynchronous communication fabric [30, 75, 76]. It also enables HALO to accommodate new algorithms in the future by simply plugging in new PEs.

Our network uses asynchronous SEND-ACK communication over an 8-bit data bus. The receiver ACKs once it has received input and is ready to receive new data. An interconnect wrapper provides a FIFO interface for the input and output of each PE. Configurable switches assemble the interconnect so that it realizes our target pipelines. Routing is similar to FPGAs (i.e., we fix the routes in the network but allow the

PE	Functionality	Parameters
LZ	Lempel-Ziv match length-offset pair search. Hashes four input bytes to index into first array of hash-chain, which records position of previous instance of the same data. Indexes second array of hash-chain using this value and find distance to previous occurrence of data.	History length, H [256-4096B] First array size is 8KB Second array size is 2×H bytes Max memory size is 24KB
LIC	Encodes LZ output with linear integer coding. 256-byte array stores literals (bytes with no previous matches). Literals are output on matches and identified with headers/lengths.	N/A
MA	Receives data to encode from LZ and DWT. Maintains counters for each input type (literal, length, offset in LZ and predict, updates in DWT) in a Fenwick tree. Counter lookups and increments are O(logN). Emits counter values to RC, for each input.	History length, H [256-4096B] Literal counter 256 bytes; length/offset counters 2×H bytes; max memory 16.25 KB
RC	Encodes data using range encoding with the probability information from MA	N/A
DWT	Discrete Wavelet Transform, used in spike detection [44] and compression.	Levels [1-5]
NEO	Non-linear energy operator, which estimates the energy content of a signal using techniques described in prior work [44].	N/A
FFT	Fast Fourier Transform of channels.	FFT points [up to 1024]
XCOR	Accepts list of channel numbers (i.e., channel map) for which pair-wise cross-correlation is calculated. Uses input parameter LAG to control the delay between the two channels.	LAG [0-64] User-defined channel map
BBF	Butterworth bandpass filter identifies frequency bands correlated to seizures.	Frequencies up to ADC Nyquist limit
SVM	Uses outputs of FFT, BBF, and XCOR to predict seizure onset. Multiplies input values and weights to perform classification.	Up to 5000 32-bit user-defined integer weights
THR	Emits a set bit if input is below threshold	User-defined threshold value (32-bit)
GATE	Passes one input stream based on the value of the second input line (provided by THR).	N/A
AES	AES-128 bit encryption in ECB mode	Encryption key [128-bit]

TABLE III: Description of PEs, their key data structures, and interactions with one another. End-users can parameterize key attributes (e.g., the history length of LZ can be made between 256 and 4096 bytes), and show impact on memory and logic.

links to be configurable), but simplified because only a small set of connectivity patterns need to be supported, and because we route data buses rather than independent bits. Switches are implemented with programmable muxes/demuxes.

We use per-PE FIFO buffers as logical adapters to transfer data from the network into the form expected by the PE. The adapter also modifies the output created by the PE to match the fixed width interface of the interconnect. HALO's interconnect sends messages in streams of bytes, bits, and tokens (packets of multiple values). Naturally, when configuring PEs, the programmer must ensure that the output interface of a PE matches the input interface of its target PE. In practice, this gives HALO a wide configuration space and allows doctors/technicians to construct many pipelines at runtime.

E. RISC-V Micro-controller

We use a low-power micro-controller on HALO to configure the PEs and support computation not currently within PEs. We use RISC-V though any micro-controller is suitable.

① Pipeline configuration: The micro-controller assembles PEs into pipelines by configuring the programmable switches in software. We use instructions to write to general purpose IO pins that set the switches dynamically. The output interface of source PEs must match the input interface of target PEs.

② PE configuration: The micro-controller configures the PE parameters from Table III. Each PE maintains parameter variables in internal memory accessible by the micro-controller.

③ Closed-loop support: The micro-controller can configure interconnect switches so that it can receive and operate on the result of any PE. This is particularly useful for closed-loop recording/stimulation scenarios. For example, when a seizure is predicted, the micro-controller can set the microelectrode array to stimulate neurons. Stimulation logic is suitable for software execution because it occurs rarely and requires more complex decision-making (i.e., personalization of length, frequency, and amplitude of stimulation pulses to the patient) than might be appropriate for a hardware implementation. With HALO, we can stimulate as many as 16 channels under the power budget, whereas commercial BCIs today only stimulate 4-8 channels [10, 21, 69, 78, 124].

④ Safe operation: HALO realizes ultra-low power Vdd comparator circuits, running at low frequencies, to identify power overshoot. On overshoot, this circuit interrupts the micro-controller, allowing it to shut off PEs to reduce overall power.

The micro-controller must be used with care as it consumes more power than the PEs. It is, however, well-suited for low-intensity tasks at low data rates. HALO runs the micro-controller at a low frequency (25MHz) with a small amount of memory (64Kb). Even with scarce compute and memory resources, micro-controllers can perform complex communication and control services and boot real-time OSes [65].

V. Methodology

A. Target Design

HALO can operate with all sensor, ADC, amplifier, and radio technologies. For our evaluation, we assume a microelectrode array with 96 channels, each of which records the activity of groups of neurons (i.e., 5-10) in their vicinity. We allow 2× more simultaneous stimulation channels (16) than commercial designs [10, 21, 69, 78, 124]. This translates to a 0.48mW upper bound for chronic stimulation [10], which is used in the movement intent and seizure prediction pipelines. Additionally, we assume that each sample is encoded in 16 bits at a frequency of 30KHz, on par with recent work on BCIs [18, 88]. This results in a real-time data rate of ~46Mbps. Finally, we assume a radio with an operating energy of 200pJ/bit, similar to current implantable BCIs [70]. We consider a strict power budget of 15mW from the range seen in

state of the art BCIs [37, 48, 67, 89, 102, 116]. Commercial ADCs achieve 1mW per 1Msps sampling rate [97]. In line with this, we dedicate 3mW power to ADCs and amplifiers. All of HALO's processing pipelines, including the radio, must therefore consume no more than 12mW of power.

We use LZ and MA PEs with 4KB of history, 256-entry byte arrays for the literals in LIC, and 16-bit divides in RC. We use a 5000-weight SVM PE, and a 1024-point FFT. Finally, we integrate a 2-stage in-order 32-bit Ibex RISC-V core (formerly known as Zero-Riscy [40]) with the RV32EC ISA — an embedded (or reduced) version of RV32I with 16 general-purpose registers, and a "compression" feature to reduce memory requirements for storing programs (this feature is used commonly for low-power embedded devices). We fully synthesize and test the RISC-V core using our commercial synthesis flow (see § V-B).

B. Hardware Evaluations

We design and test all of HALO's components using a commercial 28nm fully-depleted silicon-on-insulator (FD-SOI) CMOS process. Synthesis and power analysis is performed using the latest generation of Cadence® synthesis tools with standard cell libraries from STMicroelectronics. Memories were generated using foundry-supplied memory macros. Relying on commercial IP (instead of academic or predictive tools) means that our power numbers are more representative of real fabricated chips. We run multi-corner, physically-aware synthesis to cover all process and environmental variation corners. To err on the conservative side, we present results for the worst variation corner. Since our design is power-limited, we define this corner at Tr_{FF}, Vdd_{MAX}, and RC_{BEST}, at 1V Vdd. While scaling voltage down can further reduce power, we focus on the improvements due to the architectural design rather than circuit-level optimizations. Adhering to HALO's strict thermal constraints, our standard cell and macro libraries have been characterized at (or interpolated to) $40\,^{\circ}\mathrm{C}$.

We compare HALO PEs against the power expended by running software versions of our PEs on the RISC-V microcontroller. To do this, we combine our hardware evaluation flow with a custom memory profiler that determines the runtime memory requirements of our target software. We simulate our software kernels in behavioral RTL to quantify these memory requirements along with the minimum required frequency necessary to meet the real-time performance requirements of the kernel. Subsequently, we synthesize the RISC-V core with the minimum frequency as a constraint and re-simulate the gate-level RTL to extract annotated switching activity factors for all gates. We then use the netlist and annotated activity factors to extract accurate power numbers for logic, and introduce memory activity factors into the memory compiler. Note that the same set of steps is used to measure PE power.

We use our floorplan data to estimate an upper bound on interconnect/switch power. As reported in prior work [63], such interconnects require relatively few gates (e.g., 0.55 kilogate equivalents) and have $< 1\%$ impact on power. We use the floorplan to assess input/output adapter overhead, upper bound on routing distance, and wire capacitance. The upper bound on total interconnect and switch power is $< 300\mu$W. We base these estimates on our experience with designing/fabricating multiple generations of FPGAs in many technologies, including 28nm.

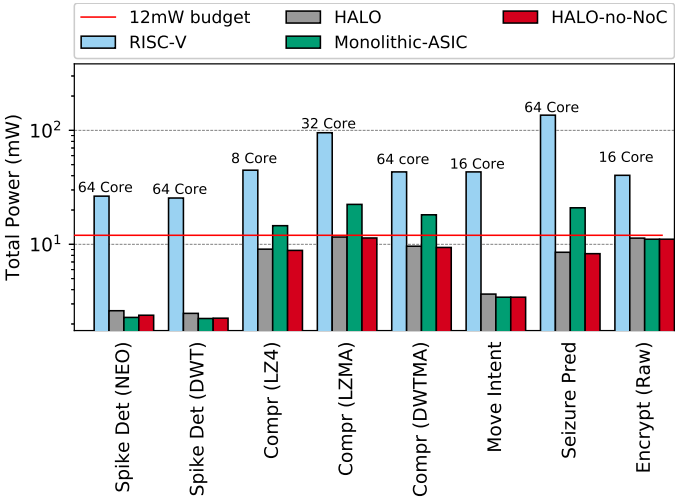

Fig. 4: Power (in log-scale) of PEs, control logic and radios for HALO versus RISC-V and monolithic ASICs. To meet the 15mW device power budget, these components (without ADCs and amplifiers) need to be under 12mW (the red line). We compare HALO against the lowest-power RISC-V and HALO-no-NoC, which shows how much power would be saved if HALO's configurability were sacrificed.

C. In-Vivo Electrophysiology from Non-Human Primate

We use electrophysiological data collected from the brain of a non-human primate. Microelectrode arrays were implanted in two locations in the motor cortex, corresponding to the left upper and lower limbs. The arrays were connected to a CereplexW™ head stage [18] for communication and data transmission to a Cerebus™ data acquisition system and signal processor [17]. Multiple antennas were used to accommodate free movement of the animal. We use recordings of brain activity while the animal performed tasks such as walking on a treadmill, reaching for a treat, and overcoming a moving styrofoam obstacle. All research protocols were approved and monitored by Brown University's Institutional Animal Care and Use Committee, and all research was performed in accordance with relevant guidelines and regulations.

VI. Evaluation

A. Power Analysis of Frequently-Used Tasks

Figure 4 compares HALO's power versus ASICs and software alternatives on RISC-V. We focus on the power consumed by the processing logic and radio, rather than the amplifier array and ADCs. Software tasks can execute on microcontroller cores in both single-core and multi-core designs, where we divide the 96 channel data streams and operate on them in parallel. We study 1-64 RISC-V core counts, in powers of two and report the best configuration per task. We

PE	Freq (MHz)	Logic (mW) Leak	Logic (mW) Dyn	Mem (mW) Leak	Mem (mW) Dyn	Total (mW)	Area (KGE)
LZ	129	0.055	1.455	0.095	1.466	3.071	55
LIC	22.5	0.057	0.267	0.006	0.046	0.376	25
MA	92	0.127	2.148	0.067	0.997	3.339	66
RC	90	0.029	0.763	0	0	0.792	12
DWT	3	0.004	0.002	0	0	0.006	2
NEO	3	0.012	0.003	0	0	0.015	5
FFT	15.7	0.057	0.509	0.085	0.356	1.007	22
XCOR	85	0.07	4.182	0.307	0.053	4.612	81
BBF	6	0.066	0.034	0	0	0.1	23
SVM	3	0.018	0.018	0.081	0.033	0.15	8
THR	16	0.002	0.011	0	0	0.013	1
GATE	5	0.003	0.006	0.067	0.054	0.13	17
AES	5	0.053	0.059	0	0	0.112	34
Tasks							
Compr (LZ4)		0.112	1.722	0.101	1.512	3.447	80
Compr (LZMA)		0.211	4.366	0.122	2.463	7.162	133
Compr (DWTMA)		0.16	2.913	0.0123	0.33	3.415	80
Seizure Prediction		0.216	4.760	0.54	0.496	6.012	111
Spike Det (NEO)		0.017	0.02	0.067	0.054	0.158	24
Spike Det (DWT)		0.009	0.019	0.067	0.054	0.149	20
Movement Intent		0.062	0.526	0.152	0.41	1.15	40
Encrypt (Raw)		0.053	0.059	0	0	0.112	34
RISC-V $\text{Control}_{\text{est}}$	25	0.341	0.137	0.248	1.080	1.800	70

TABLE IV: PE power (leakage/dynamic for logic/memory separated), frequency, and area (in kilo-gate equivalents or KGEs). All numbers assume data processing rates of 46Mbps.

also show an idealized version of HALO where the on-chip interconnect is removed to quantify the power penalty for the configurability that the network offers. Both HALO variants use the optimizations from §IV-B. HALO uses less power than monolithic ASICs and RISC-V approaches. Moreover, the inclusion of the low-power circuit-switched network consumes marginally more power than HALO-no-NoC.

B. Power Analysis of Processing Elements

Per-PE power consumption is detailed in Table IV. For each PE, we quantify the operating frequency needed to process the full stream of neuronal data at 46Mbps. We separate logic and memory power into static and dynamic components, and also show overall area. As the majority of frequently-used tasks use multiple PEs to form a processing pipeline, we also present combined numbers for all PEs that form pipelines. The full system power consumption combines the reported power consumption with that of auxiliary circuits such as ADCs, radio, interconnect, etc. We also consider the power needed to run a single RISC-V core to handle low frequency tasks such as neuronal stimulation and the communication interface.

Table IV shows that the compression and seizure prediction pipelines consume the most power, but are still within the power budget. In general, the higher the operating frequency, the higher the dynamic power. Furthermore, as expected, PEs that use more memory (e.g., LZ, XCOR, MA, etc.) also expend more dynamic and static power on the memory component.

Figure 5 quantifies the PEs in Table IV per processing pipeline. On the left, we show the total power expended by the processing pipeline including radio and interconnect, discounting ADCs and amplifiers, and separate the processing contributions into the various PEs (in blue), and on the RISC-V core (in green) that is used to perform tasks like microelectrode array stimulation control for movement intent and seizure pre-

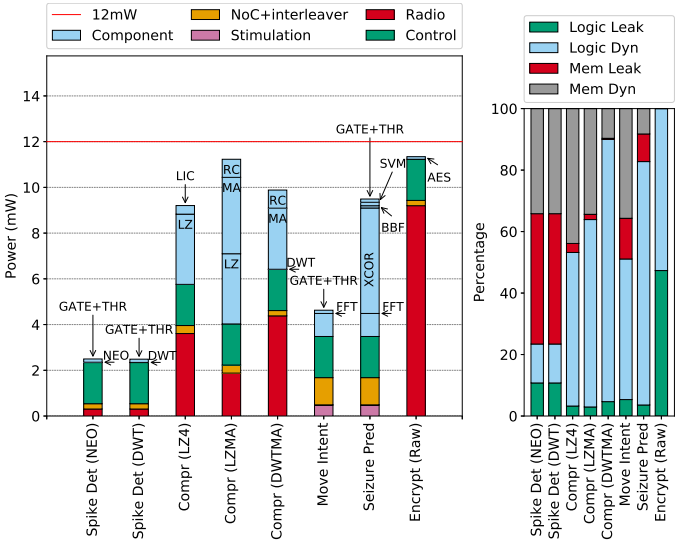

Fig. 5: (Left) Power of each task pipeline's PEs, stimulation, network and interleaver and control. Control refers to microcontroller power to set up the pipeline. Stimulation refers to power for chronic neurostimulation. (Right) Percentages of total power for compute logic and memory, separated into dynamic and leakage components.

diction. All tasks remain comfortably under the 12mW power budget for the processing pipeline and radio. As expected, spike detection tasks expend low power because they are simple (NEO, DWT, GATE, and THR require few hardware resources), and use low radio bandwidth. On the other hand, encrypting the raw data results in higher power consumption for the radio as it has to transmit the entire raw data stream from the 96 channels. The dedicated compression schemes (i.e., LZ4 and LZMA) consume roughly 9-11mW, with varying amounts going on logic versus the radio. LZMA has a higher compression factor (see §VI-C) and hence requires lower radio power. However, it requires more computation to achieve this compression ratio, which is why its logic power is higher. Finally, power consumed by the network and the interleaver remains negligible in comparison to the PEs.

The graph on the right in Figure 5 sheds light on the breakdown of processing pipeline power in terms of dynamic versus leakage components, for the logic and memory parts of the PEs. These numbers vary substantially, with techniques that use little computation (e.g., spike detection) expending the bulk of their power on memory. Most other tasks require a balance of power across compute and memory. In cases where memory accesses are more frequent (e.g., compression algorithms), dynamic memory power outweighs leakage.

C. Impact of Co-Design Decisions

Figure 6 shows the impact of the hardware-algorithm co-design discussed in §IV-B on the XCOR PE (on the left) and LZ, MA, and RC PEs for LZMA (on the right). Since these optimizations impact only the power of PEs, we elide a discussion of their impact on radios, ADCs, amplifiers and the RISC-V core. The graph on the left shows the improvement

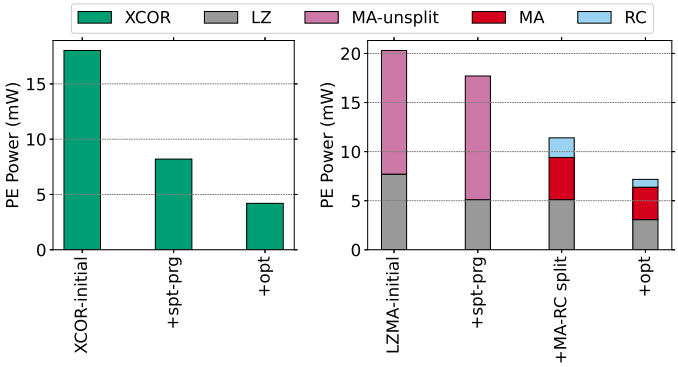

Fig. 6: (Left) Impact of spatial refactoring and other optimizations on XCOR power in seizure prediction. (Right) Impact of spatial refactoring, PE decomposition and other microarchitectural optimizations on LZMA power.

in XCOR while the graph on the right shows improvements in LZMA. All must be under the 12mW target budget. For XCOR, the unoptimized version exceeds the target budget. Spatial reprogramming saves 50% power. Optimizations such as pipelining further lower the power.

Figure 6 shows that unoptimized LZMA uses 20mW and exceeds the 12mW power budget. Spatial reprogramming saves 1.5× power. To reduce power further, we use locality refactoring to split the original MA into separate MA and RC PEs. This reduces power to 11.2mW, which is then further dropped using other optimizations such as pipelining.

D. More Design Space Studies

We introduced PE parameterization, blocking, and interleaving in §IV. The graph on the left in Figure 7 plots the compression per mW of the three BCI task pipelines as a function of history length. The compression per mW of both LZ4 and LZMA peak at history length of 4096 bytes. Longer histories enable better compression, but the gains drop after a window size of 4096 bytes. Apart from the LZMA-8192 configuration, all configurations are within the 12mW power budget. Thus, we advocate using a 4096 byte history length.

The graph on the right in Figure 7 plots compression per mW as interleaving depth is varied. For block based algorithms like LZ4 and LZMA, memory interleaving greatly reduces hardware resources within the PE at the cost of a smaller, less power hungry memory buffer. As interleaving depth increases, so too does the cost of memory. Non-block based compression like DWTMA does not face this trade-off.

Figure 8 plots the compression ratio per mW as a function of compression block size. LZMA and DWTMA benefit from larger block sizes, which enable better estimation of input frequencies until a log block size of 22 (4MB). Beyond this, the compression per mW drops slightly. This drop occurs because the saturating counters in §IV-B introduce inaccuracies in input frequency estimations. As LZ4 encoding does not depend on block size, it remains unaffected.

Finally, Figure 9 shows an example of how HALO can adapt to the needs of different brain regions. We separate compression/power results of LZ4, LZMA and DWTMA

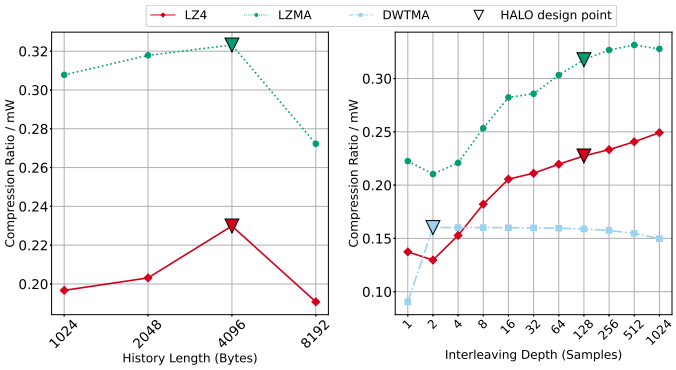

Fig. 7: (Left) Longer LZ history enables better matches but requires more memory. A history of 4KB best balances power and performance. All configurations except 8KB use <12mW. (Right) LZ match benefits only slightly from channel correlation, so larger interleaving compresses better and reduces radio bandwidth. 128 sample interleaving is a good balance. DWTMA is mostly unaffected by interleaving for ≥ 2 samples.

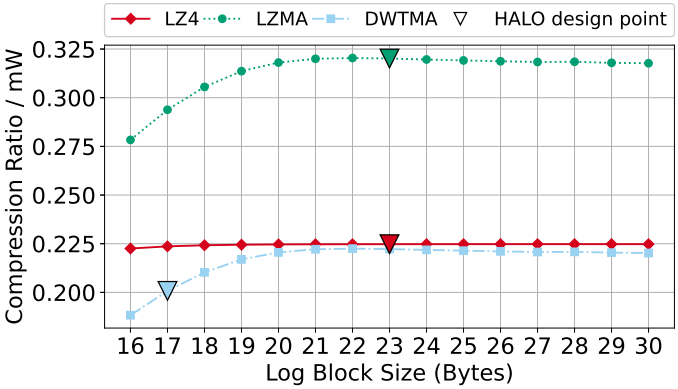

Fig. 8: Impact of blocking on compression ratio and power. The x-axis is log (block size), the y-axis is compression ratio per mW. LZ4 compression ratio does not drastically vary with the block size because it uses LIC encoding, which does suffer a penalty for smaller blocks. For LZMA and DWT, block size affects the compression ratio. As the block size increases, frequency estimations improve, improving compression ratio per unit mW from block log size 16 (64kB) to 22 (4MB). After 22 (4MB), blocks become too large and the saturated counters slowly degrade estimations of the frequencies.

for the arm and leg regions in the motor cortex. LZMA's compression ratio is superior to LZ4 and DWTMA, but LZ4 power is lower. If radio-deposited energy is the dominating concern, we advocate using LZMA. Figure 9 shows that the power budget targets are met regardless of brain region.

VII. Conclusions & Discussion

Modularity: This work performs an initial exploration of workloads that are important for neuroscience, but the list of tasks can be expanded. Future BCIs will implement other workloads, with different pipelines targeting different research and medical objectives. Because of its modular design, HALO

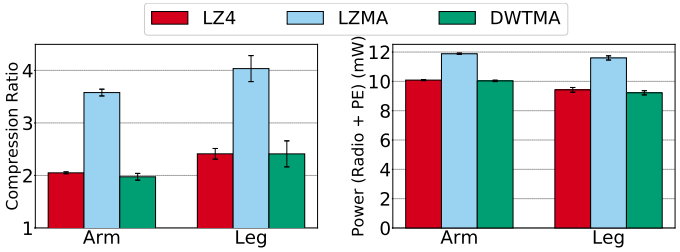

Fig. 9: (Left) Compression ratio for LZ4, LZMA, and DWTMA, separated by experiments performed on the arm and leg regions of the motor cortex. The error bars show variance of compression ratios between different experimental trials. (Right) LZ4, DWTMA, and LZMA power (excluding ADCs and amplifiers), separated by arm and leg regions.

is able to support such workloads seamlessly. In the near-term, we are further enhancing HALO's seizure prediction algorithm by implementing kernels for calculation of approximate entropy, Hann functions, and Hjorth parameters [47, 51, 87]. We have also discovered that compression based on the Burrows-Wheeler transform (e.g., Bzip2) may be particularly effective for certain classes of neural data. Implementing a monolithic ASIC for Bzip2 will be overly complex and power-hungry, but HALO's modularity offers a lower-power alternative. In particular, since Bzip2 uses range coding, we simply need to implement the Burrows-Wheeler transform, but can reuse the MA and RC PEs.

Distributed designs: Structural and functional networks across brain centers influence the manifestation of several neurological disorders. For example, the occurrence of seizures and their inter-ictal state (i.e., the period of time between successive convulsions) is a function of the activity in many parts of the brain including the hippocampus, the lateral septal nuclei and anterior hypothalamus, as well as the upper brainstem, intralaminar thalamus, and fronto-parietal association cortex [25, 29, 62, 101, 103]. In such cases, we envision the need for multiple HALO devices on different brain sub-centers, with one device determining the onset of a seizure, and another device used to stimulate tissue on another brain region, thereby mitigating (and perhaps even eliminating) the "spread" of seizures across sub-centers. Distributed designs [31, 89, 95] require particularly power-efficient and flexible platforms, making HALO a compelling starting point.

Related work: HALO is partly inspired by prior work on embedded systems for wearables. While HALO is significantly more power-constrained because it is implanted in the brain, many wearables also target signal processing and data exfiltration [59, 64, 120]. Like our work on HALO, some of the recent work in this space – e.g., applications like smart watches that can be used to monitor fitness, GPS, heart rate, hand gestures, etc. – studies the question of how to make these designs more general and flexible [50, 53, 65]. While HALO meets a different set of constraints, we expect the lessons from this work to apply to some of these domains too.

Implications on accelerator SoCs: While HALO focuses on implantable BCIs, the question of how to design multi-accelerator SoCs in ultra-power-constrained environments is a more general problem facing the systems community. This work offers one way of systematically traversing the design space by using software engineering techniques to make hardware more amenable for efficient implementation. While the details of how we decompose individual algorithms into constituent pieces, identify shared pieces among algorithms, prune these pieces to a canonical set, and then implement these pieces into distinct hardware blocks that can operate at their individual target frequency may vary across domains, their generality offers lessons for domains beyond BCIs.

VIII. Acknowledgments

We thank Guilherme Cox, Zi Yan, Caroline Trippel, Carole-Jean Wu, and Margaret Martonosi for feedback on drafts of this manuscript. We also thank Martha Kim, Sibren Isaacman, Pranjal Awasthi, Lenny Khazan, Gabe Petrov, and Anurag Khandelwal for fruitful technical discussions. We are grateful to Hitten Zaveri, Dennis Spencer, and Robert Duckrow for brainstorming future applications of HALO, particularly for their patients with epilepsy. Finally, we thank Thu Nguyen, Ricardo Bianchini, Ulrich Kremer, and Jonathan Cohen for their support and encouragement at the critical initial stages of this project. This work was done with support from NSF grant 2019529, and with partial support from DARPA grants FA8650-18-2-7850 and HR001117S0054-FP-042.

References

[1] "Bridging the Bio-Electronic Divide [online] Available: https://www.darpa.mil/news-events/2015-01-19."
[2] "DBS side effects [online] Available: https://www.mayoclinic.org/tests-procedures/deep-brain-stimulation/about/pac-20384562."
[3] "Deep Brain Stimulation Systems - Activa PC [online] Available: https://www.medtronic.com/us-en/healthcare-professionals/products/neurological/deep-brain-stimulation-systems/activa-pc.html."
[4] "Deep Brain Stimulation Systems - Activa RC [online] Available: https://www.medtronic.com/us-en/healthcare-professionals/products/neurological/deep-brain-stimulation-systems/activa-rc.html."
[5] "HIPAA Encryption Requirements [online] Available: https://www.hipaaguide.net/hipaa-encryption-requirements/."
[6] "Kernel [online] Available: https://kernel.co."
[7] "Longeviti Neuro Solutions [online] Available: https://longeviti.com."
[8] "LZ4 Explained. [online] Available: http://fastcompression.blogspot.com/2011/05/lz4-explained.html."
[9] "LZMA SDK (Software Development Kit) [online] Available: https://tukaani.org/xz/."
[10] "Medtronic Activa PC Multi-program neurostimulator implant manual [online] Available: http://www.neuromodulation.ch/sites/default/files/pictures/activa_PC_DBS_implant_manuel.pdf."
[11] "National Policy on the Use of the Advanced Encryption Standard (AES) to Protect National Security Systems and National Security Information [online] Available: https://csrc.nist.gov/csrc/media/projects/cryptographic-module-validation-program/documents/cnss15fs.pdf."
[12] "Neurospace [online] Available: https://neurospace.com.au."
[13] "Nominations from FDA's Center from Device and Radiological Health [online] Available: https://ntp.niehs.nih.gov/ntp/htdocs/chem_background/exsumpdf/wireless051999_508.pdf ."
[14] "Physician and Healthcare Payer Information: Medtronic [online] Available: http://newsroom.medtronic.com/phoenix.zhtml?c=251324&p=irol-newsArticle&id=1845602."
[15] "Spencer Probe Depth Electrodes [online] Available: http://alliancebiomedica.com/index.php?route=product/product&product_id=164."

[16] "The benchmark for multichannel, high-density neural recording [online] Available: https://www.blackrockmicro.com/electrode-types/utah-array/."

[17] "Where High-Performance DAQ Meets Unparalleled Ease of Use [online] Available: https://blackrockmicro.com/neuroscience-research-products/neural-data-acquisition-systems/cerebus-daq-system/."

[18] "Where High-Performance DAQ Meets Unparalleled Ease of Use [online] Available: https://www.blackrockmicro.com/cereplex-wireless-headstage/."

[19] U. R. Acharya, Y. Hagiwara, and H. Adeli, "Automated Seizure Prediction," *"Epilepsy & Behavior"*, vol. 88, p. 251–261, 2018.

[20] G. E. Alexander, M. R. DeLong, and P. L. Strick, "Parallel Organization of Functionally Segregated Circuits Linking Basal Ganglia and Cortex," *Annual Review of Neuroscience*, vol. 9, no. 1, pp. 357–381, 1986.

[21] M. Azin, D. J. Guggenmos, S. Barbay, R. J. Nudo, and P. Mohseni, "A battery-powered activity-dependent intracortical microstimulation ic for brain-machine-brain interface," *IEEE JSSC*, April 2011.

[22] J. Aziz, R. Genov, M. Derchansky, B. Bardakjian, and P. Carlen, "256-Channel Neural Recording Microsystem with On-Chip 3D Electrodes," *IEEE ISSC*, Feb 2007.

[23] J. N. Y. Aziz *et al.*, "256-Channel Neural Recording and Delta Compression Microsystem With 3D Electrodes," *IEEE JSSC*, March 2009.

[24] A. A. Bari *et al.*, "Charting the road forward in psychiatric neurosurgery: proceedings of the 2016 american society for stereotactic and functional neurosurgery workshop on neuromodulation for psychiatric disorders," *Journal of Neurology, Neurosurgery & Psychiatry*, 2018.

[25] F. Bartolomei *et al.*, "Defining Epileptogenic Networks: Contribution of SEEG and Signal Analysis," *Epilepsia*, vol. 58, no. 7, 2017.

[26] K. Bazaka and M. Jacob, "Implantable Devices: Issues and Challenges," *Electronics*, vol. 2, pp. 1–34, 03 2012.

[27] A. Bhattacharjee, "Using branch predictors to predict brain activity in brain-machine implants," ser. MICRO-50 '17.

[28] U. Bihr *et al.*, "Real-Time Data Compression of Neural Spikes," ser. NEWCAS '14.

[29] H. Blumenfeld, "What is a Seizure Network? Long-Range Network Consequences of Epileptic Networks," *Adv Exp Med Biol*, vol. 8, 2014.

[30] D. S. Bormann and P. Y. K. Cheung, "Asynchronous Wrapper for Heterogeneous Systems," in *IEEE ICCD*, 1997.

[31] D. A. Borton, M. Yin, J. Aceros, and A. Nurmikko, "An implantable wireless neural interface for recording cortical circuit dynamics in moving primates," *J Neural Eng*, vol. 10, no. 2, p. 026010, Apr 2013.

[32] D. Borton, S. Micera, J. d. R. Millán, and G. Courtine, "Personalized neuroprosthetics," *Science Translational Medicine*, vol. 5, 2013.

[33] D. Brandman *et al.*, "Rapid Calibration of an Intracortical Brain–Computer Interface for People with Tetraplegia," *J. Neural Eng.*

[34] G. Buzsáki, C. A. Anastassiou, and C. Koch, "The Origin of Extracellular Fields and Currents — EEG, ECoG, LFP and Spikes," *Nature Reviews Neuroscience*, vol. 13, pp. 407–420, May 2012.

[35] J. M. Carmena *et al.*, "Learning to Control a Brain–Machine Interface for Reaching and Grasping by Primates," *PLOS Biology*, vol. 1, no. 2, 2003.

[36] D. M. Chapiro, "Globally-Asynchronous Locally-Synchronous Systems (Performance, Reliability, Digital)," Ph.D. dissertation, 1985.

[37] T. Chen *et al.*, "A Hardware Implementation of Real-Time Epileptic Seizure Detector on FPGA," *IEEE BioCAS '11*, 2011.

[38] S. F. Cogan, "Neural Stimulation and Recording Electrodes," *Annual Review of Biomedical Engineering*, vol. 10, pp. 275–309, 2008.

[39] A. L. Crowell, S. J. Garlow, P. Riva-Posse, and H. S. Mayberg, "Characterizing the Therapeutic Response to Deep Brain Stimulation for Treatment-Resistant Depression: A Single Center Long-Term Perspective," *Frontiers in Integrative Neuroscience*, vol. 9, p. 41, 2015.

[40] P. Davide Schiavone *et al.*, "Slow and Steady Wins the Race? A Comparison of Ultra-Low-Power RISC-V Cores for Internet-of-Things Applications," ser. PATMOS '17, Sep. 2017, pp. 1–8.

[41] J. del R. Milan and J. M. Carmena, "Invasive or Noninvasive: Understanding Brain-Machine Interface Technology [Conversations in BME]," *IEEE Engineering in Medicine and Biology Magazine*, 2010.

[42] A. L. S. Ferreira, L. C. d. Miranda, and E. E. Cunha de Miranda, "A Survey of Interactive Systems based on Brain-Computer Interfaces," *SBC Journal on 3D Interactive Systems*, vol. 4, no. 1, 2013.

[43] D. K. Freeman *et al.*, "A Sub-millimeter, Inductively Powered Neural Stimulator," *Frontiers in Neuroscience*, 2017.

[44] S. Gibson, J. W. Judy, and D. Marković, "Spike Sorting: The First Step in Decoding the Brain," *IEEE Signal Processing Magazine*, 2012.

[45] B. Gosselin, "Recent Advances in Neural Recording Microsystems," *Sensors (Basel)*, vol. 11, no. 5, pp. 4572–4597, 2011.

[46] R. Guduru *et al.*, "Magnetoelectric 'Spin' on Stimulating the Brain," *Nanomedicine*, vol. 10, 2015.

[47] F. J. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform," *Proceedings of the IEEE*, Jan 1978.

[48] R. R. Harrison *et al.*, "Wireless neural recording with single low-power integrated circuit," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 17, no. 4, pp. 322–329, Aug 2009.

[49] J. Herron, T. Denison, and H. J. Chizeck, "Closed-loop DBS with Movement Intention," *2015 International IEEE/EMBS Conference on Neural Engineering (NER)*, 2015.

[50] J. Hester *et al.*, "Amulet: An energy-efficient, multi-application wearable platform," 11 2016, pp. 216–229.

[51] B. Hjorth, "Eeg analysis based on time domain properties," *Electroencephalography and Clinical Neurophysiology*, 1970.

[52] L. R. Hochberg *et al.*, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm," *Nature*, May 2012.

[53] H. Huang and S. Lin, "Toothbrushing monitoring using wrist watch," in *ACM SenSys*, 2016. [Online]. Available: https://doi.org/10.1145/2994551.2994563

[54] M. M. Jackson and R. Mappus, *Applications for Brain-Computer Interfaces*. London, UK: Springer, 2010.

[55] J. C. Kao, P. Nuyujukian, S. I. Ryu, and K. V. Shenoy, "A High-Performance Neural Prosthesis Incorporating Discrete State Selection With Hidden Markov Models," *IEEE Trans Biomed Eng*, April 2017.

[56] H. Kassiri *et al.*, "Closed-Loop Neurostimulators: A Survey and A Seizure-Predicting Design Example for Intractable Epilepsy Treatment," *IEEE Trans. Biomed. Circuits Syst.*, 2017.

[57] M. R. Keezer, S. M. Sisodiya, and J. W. Sander, "Comorbidities of Epilepsy: Current Concepts and Future Perspectives," *The Lancet Neurology*, vol. 15, no. 1, p. 106–115, 2016.

[58] R. Kelly *et al.*, "Comparison of Recordings from Microelectrode Arrays and Single Electrodes in Visual Cortex," *JNeurosci*, 2007.

[59] H. Kim *et al.*, "A low power ecg signal processor for ambulatory arrhythmia monitoring system," in *2010 Symposium on VLSI Circuits*, 2010.

[60] S. M. Kim, P. Tathireddy, R. Normann, and F. Solzbacher, "Thermal Impact of an Active 3-D Microelectrode Array Implanted in the Brain," *IEEE Trans. Neural Syst. Rehabil. Eng.*, Dec 2007.

[61] M. Koenigs and J. Grafman, "The Functional Neuroanatomy of Depression: Distinct Roles for Ventromedial and Dorsolateral Prefrontal Cortex," *Behav. Brain Res.*, vol. 201, no. 2, pp. 239–243, Aug 2009.

[62] M. A. Kramer and S. S. Cash, "Epilepsy as a disorder of cortical network organization," *The Neuroscientist : a review journal bringing neurobiology, neurology and psychiatry*, vol. 18, no. 4, Aug 2012.

[63] M. Krstic and E. Grass, "New gals technique for datapath architectures," in *PATMOS*, 2003.

[64] J. Kwong and A. P. Chandrakasan, "An energy-efficient biomedical signal processing platform," *IEEE JSSC*, 2011.

[65] A. Levy *et al.*, "Multiprogramming a 64kb computer safely and efficiently," ser. SOSP '17. ACM, 2017, pp. 234–251.

[66] S. Li, W. Zhou, Q. Yuan, and Y. Liu, "Seizure Prediction Using Spike Rate of Intracranial EEG," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 21, no. 6, pp. 880–886, Nov 2013.

[67] Y. Lin *et al.*, "A Battery-Less, Implantable Neuro-Electronic Interface for Studying the Mechanisms of Deep Brain Stimulation in Rat Models," *IEEE Trans. Biomed. Circuits Syst.*, vol. 10, no. 1, pp. 98–112, Feb 2016.

[68] M. Liu, H. Chen, R. Chen, and Z. Wang, "Low-Power IC Design for a Wireless BCI System," *ISCAS '08*, pp. 1560–1563, 2008.

[69] X. Liu *et al.*, "The pennbmbi: Design of a general purpose wireless brain-machine-brain interface system," *IEEE Trans. Biomed. Circuits Syst.*, vol. 9, no. 2, pp. 248–258, April 2015.

[70] Y.-H. Liu, C.-L. Li, and T.-H. Lin, "A 200-pj/b mux-based rf transmitter for implantable multichannel neural recording," *IEEE Trans. Microw. Theory Techn*, vol. 57, 11 2009.

[71] A. M. Lozano *et al.*, "Deep brain stimulation: current challenges and future directions," *Nat Rev Neurol*, Mar 2019.

[72] S. Moore, G. Taylor, R. Mullins, and P. Robinson, "Point to Point GALS Interconnect," in *ASYNC '02*, April 2002, pp. 69–75.

[73] A. Moreno and J. Cortadella, "Synthesis of All-Digital Delay Lines," in

Proc. International Symposium on Advanced Research in Asynchronous Circuits and Systems, May 2017, pp. 75–82.
[74] E. Musk and Neuralink, "An Integrated Brain-Machine Interface Platform with Thousands of Channels," *bioRxiv*, 2019.
[75] J. Muttersbach, T. Villiger, and W. Fichtner, "Practical Design of Globally-Asynchronous Locally-Synchronous Systems," ser. ASYNC 2000, April 2000.
[76] J. Muttersbach, T. Villiger, H. Kaeslin, N. Felber, and W. Fichtner, "Globally-Asynchronous Locally-Synchronous Architectures to Simplify the Design of On-Chip Systems," in *IEEE ASIC/SOC Conference*, 1999.
[77] C. Mühl, B. Allison, A. Nijholt, and G. Chanel, "A Survey of Affective Brain Computer Interfaces: Principles, State-of-the-art, and Challenges," *Brain-Computer Interfaces*, vol. 1, no. 2, pp. 66–84, 2014.
[78] T. K. T. Nguyen *et al.*, "Closed-loop optical neural stimulation based on a 32-channel low-noise recording system with online spike sorting," *J. Neural Eng.*, vol. 11, no. 4, p. 046005, jun 2014.
[79] T. Nowatzki, V. Gangadhar, N. Ardalani, and K. Sankaralingam, "Stream-Dataflow Acceleration," *ISCA '17*, 2017.
[80] B. Nuttin *et al.*, "Consensus on Guidelines for Stereotactic Neurosurgery for Psychiatric Disorders," *J. Neurol. Neurosurg. Psychiatry*, Sep 2014.
[81] P. Nuyujukian, J. C. Kao, S. I. Ryu, and K. V. Shenoy, "A Nonhuman Primate Brain–Computer Typing Interface," *Proceedings of the IEEE*, Jan 2017.
[82] P. Nuyujukian *et al.*, "Cortical Control of a Tablet Computer by People with Paralysis," *PLoS ONE*, 2018.
[83] J. Oliver *et al.*, "Synchroscalar: a multiple clock domain, power-aware, tile-based embedded processor," ser. ISCA '04, June 2004, pp. 150–161.
[84] G. O'Leary, D. M. Groppe, T. A. Valiante, N. Verma, and R. Genov, "Nurip: Neural interface processor for brain-state classification and programmable-waveform neurostimulation," *IEEE JSSC*, vol. 53, no. 11, pp. 3150–3162, Nov 2018.
[85] A. Parashar *et al.*, "Triggered instructions: A control paradigm for spatially-programmed architectures," *SIGARCH Comput. Archit. News*, 2013.
[86] B. Pesaran *et al.*, "Investigating Large-Scale Brain Dynamics Using Field Potential Recordings: Analysis and Interpretation," *Nature Neuroscience*, vol. 21, pp. 903–919, 2018.
[87] S. M. Pincus, "Approximate entropy as a measure of system complexity," *PNAS USA*, Mar 1991.
[88] M. P. Powell, W. R. Britz, J. S. Harper, and D. A. Borton, "An Engineered Home Environment for Untethered Data Telemetry from Nonhuman Primates," *Journal of Neuroscience Methods*, 2017.
[89] M. P. Powell, X. Hou, C. Galligan, J. Ashe, and D. A. Borton, "Toward Multi-Area Distributed Network of Implanted Neural Interrogators," 2017.
[90] N. R. Provenza *et al.*, "The Case for Adaptive Neuromodulation to Treat Severe Intractable Mental Disorders," *Frontiers in Neuroscience*, vol. 13, p. 152, 2019.
[91] S. Riki, "A Review on Neural Signal Compression Methods," *International Journal of Computer Science and Network Security*, 2017.
[92] R. G. Robinson, L. B. Starr, K. L. Kubos, and T. R. Price, "A Two-Year Longitudinal Study of Post-Stroke Mood Disorders: Findings During the Initial Evaluation." *Stroke*, vol. 14, no. 5, p. 736–741, 1983.
[93] R. Sarpeshkar *et al.*, "Low-power circuits for brain–machine interfaces," *IEEE Trans. Biomed. Circuits Syst.*, 2008.
[94] SCC39, "IEEE Standard for Safety Levels with Respect to Human Exposure to Radio Frequency Electromagnetic Fields, 3 kHz to 300 GHz [online] Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1626482."
[95] D. A. Schwarz *et al.*, "Chronic, wireless recordings of large-scale brain activity in freely moving rhesus monkeys," *Nature Methods*, 2014.
[96] M. Seidenberg, D. T. Pulsipher, and B. Hermann, "Association of epilepsy and comorbid conditions," *Future Neurol*, Sep 2009.
[97] J. Shen *et al.*, "A 16-bit 16-ms/s sar adc with on-chip calibration in 55-nm cmos," *IEEE JSSC*, 2018.
[98] M. A. Shetliffe, A. M. Kamboh, A. Mason, and K. G. Oweiss, "Impact of Lossy Compression on Neural Response Characteristics extracted from High-Density Intra-cortical Implant Data," in *IEEE EMBC*, 2007.
[99] H. Shiao *et al.*, "SVM-Based System for Prediction of Epileptic Seizures From iEEG Signal," *IEEE Trans Biomed Eng*, vol. 64, no. 5, pp. 1011–1022, May 2017.
[100] A. Shoeb, B. F. D. Bourgeois, S. Treves, S. C. Schachter, and J. Guttag, "Impact of Patient-Specificity on Seizure Onset Detection Performance," *IEEE EMBC*, 2007.
[101] E. H. Smith and C. A. Schevon, "Toward a mechanistic understanding of epileptic networks," *CURR NEUROL NEUROSCI*, 2016.
[102] A. M. Sodagar, K. D. Wise, and K. Najafi, "A wireless implantable microsystem for multichannel neural recording," *IEEE Trans. Microw. Theory Techn*, vol. 57, Oct 2009.
[103] S. Spencer, "Neural networks in human epilepsy: Evidence of and implications for treatment," *Epilepsia*, 2002.
[104] I. Stevenson and K. Kording, "How Advances in Neural Recording Affect Data Analysis," *Nature neuroscience*, 02 2011.
[105] C. Sun *et al.*, "Dsent - a tool connecting emerging photonics with electronics for opto-electronic networks-on-chip modeling," ser. NOC '12, May 2012, pp. 201–210.
[106] F. T. Sun and M. J. Morrell, "The RNS System: responsive cortical stimulation for the treatment of refractory partial epilepsy," *Expert Review of Medical Devices*, 2014.
[107] D. M. Taylor, S. I. H. Tillery, and A. B. Schwartz, "Direct Cortical Control of 3D Neuroprosthetic Devices," *Science*, 2002.
[108] C. Toro *et al.*, "Event-Related Desynchronization and Movement-Related Cortical Potentials on the ECoG and EEG," *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, no. 5, 1994.
[109] M. C. Tracy, W. Jansen, K. A. Scarfone, and J. Butterfield, "Guidelines on Electronic Mail Security," *NIST Special Publication 800-45*, Feb 2007.
[110] U.S. Federal Communications Commission, "FCC Policy on Human Exposure to Radiofrequency Electromagnetic Fields [online] Available: https://www.fcc.gov/general/fcc-policy-human-exposure."
[111] Y. Wang, J. Yan, J. Wen, T. Yu, and X. Li, "An Intracranial Electroencephalography (iEEG) Brain Function Mapping Tool with an Application to Epilepsy Surgery Evaluation," *Front Neuroinform*, 2016.
[112] Y. Wang and L. Guo, "Nanomaterial-Enabled Neural Stimulation," *Frontiers in Neuroscience*, vol. 10, 2016.
[113] J. Wessberg and M. A. L. Nicolelis, "Optimizing a Linear Algorithm for Real-Time Robotic Control using Chronic Cortical Ensemble Recordings in Monkeys," *Journal of Cognitive Neuroscience*, 2004.
[114] J. Wessberg *et al.*, "Real-Time Prediction of Hand Trajectory by Ensembles of Cortical Neurons in Primates," *Nature*, 2000.
[115] F. R. Willett *et al.*, "Principled BCI Decoder Design and Parameter Selection Using a Feedback Control Model," *Scientific Reports*, 2019.
[116] P. D. Wolf, "Thermal Considerations for the Design of an Implanted Cortical Brain-Machine Interface (BMI)," *Indwelling Neural Implants: Strategies for Contending with the In Vivo Environment*, 2008.
[117] T. A. Wozny *et al.*, "Effects of Hippocampal Low-Frequency Stimulation in Idiopathic Non-Human Primate Epilepsy Assessed via a Remote-Sensing-Enabled Neurostimulator," *Experimental Neurology*, vol. 294, pp. 68 – 77, 2017.
[118] T. Wu, W. Zhao, H. Guo, H. H. Lim, and Z. Yang, "A Streaming PCA VLSI Chip for Neural Data Compression," *IEEE Trans. Biomed. Circuits Syst.*, vol. 11, no. 6, pp. 1290–1302, Dec 2017.
[119] A. Yakovlev, S. Kim, and A. Poon, "Implantable Biomedical Devices: Wireless Powering and Communication," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 152–159, April 2012.
[120] R. F. Yazicioglu *et al.*, "Ultra-low-power wearable biopotential sensor nodes," in *IEEE EMBC*, 2009.
[121] M. Yin *et al.*, "Wireless Neurosensor for Full-Spectrum Electrophysiology Recordings During Free Behavior," *Neuron*, 2014.
[122] D. Young *et al.*, "Closed-Loop Cortical Control of Virtual Reach and Posture using Cartesian and Joint Velocity Commands," *J. Neural Eng.*, jan 2019.
[123] K. Y. Yun and R. P. Donohue, "Pausible Clocking: A First Step Toward Heterogeneous Systems," in *IEEE ICCD VLSI in Computers and Processors*, 1996.
[124] S. Zanos, A. G. Richardson, L. Shupe, F. P. Miles, and E. E. Fetz, "The Neurochip-2: an autonomous head-fixed computer for recording and stimulating in freely behaving monkeys," *IEEE Trans Neural Syst Rehabil Eng*, vol. 19, no. 4, pp. 427–435, Aug 2011.
[125] Y. Zhao, L. Tang, R. Rennaker, C. Hutchens, and T. S. Ibrahim, "Studies in RF Power Communication, SAR, and Temperature Elevation in Wireless Implantable Neural Interfaces," *PLOS ONE*, 2013.
[126] Zhiyi Yu *et al.*, "An asynchronous array of simple processors for dsp applications," ser. ISSCC '06, Feb 2006.