Journal of Neural Engineering

PAPER • OPEN ACCESS

Power-efficient *in vivo* brain-machine interfaces via brain-state estimation

Journal of Neural Engineering

OPEN ACCESS

RECEIVED
14 September 2022

REVISED
31 December 2022

ACCEPTED FOR PUBLICATION
16 January 2023

PUBLISHED
27 January 2023

PAPER

Power-efficient *in vivo* brain-machine interfaces via brain-state estimation

Daniel Valencia[1,2,*], **Gianluca Leone**[3], **Nicholas Keller**[1], **Patrick P Mercier**[2] **and Amir Alimohammad**[1]

[1] Department of Electrical and Computer Engineering, San Diego State University, San Diego, United States of America
[2] Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, United States of America
[3] Department of Electrical and Computer Engineering, University of Cagliari, Cagliari, Italy
[*] Author to whom any correspondence should be addressed.

E-mail: dlvalencia@sdsu.edu

Keywords: neural signal processing, brain–machine interfaces, application-specific integrated circuits

Abstract

Objective. Advances in brain–machine interfaces (BMIs) can potentially improve the quality of life of millions of users with spinal cord injury or other neurological disorders by allowing them to interact with the physical environment at their will. *Approach.* To reduce the power consumption of the brain-implanted interface, this article presents the first hardware realization of an *in vivo* intention-aware interface via brain-state estimation. *Main Results.* It is shown that incorporating brain-state estimation reduces the *in vivo* power consumption and reduces total energy dissipation by over 1.8× compared to those of the current systems, enabling longer better life for implanted circuits. The synthesized application-specific integrated circuit (ASIC) of the designed intention-aware multi-unit spike detection system in a standard 180 nm CMOS process occupies 0.03 mm^2 of silicon area and consumes 0.63 μW of power per channel, which is the least power consumption among the current *in vivo* ASIC realizations. *Significance.* The proposed interface is the first practical approach towards realizing asynchronous BMIs while reducing the power consumption of the BMI interface and enhancing neural decoding performance compared to those of the conventional synchronous BMIs.

1. Introduction

Over the past decade, researchers have studied the activities of individual neurons with respect to their neighboring neurons and their response to various stimuli [1, 2]. Neurons communicate by means of firing electric pulses, called action potentials (APs) or spikes. The electrical activity of neurons can be measured and recorded by multi-electrode arrays (MEAs) in which each intracortical electrode implanted in the motor cortex record cellular electrical activity from a small population of neurons within a few hundred micrometers of the neuron closest to the tip of the electrode [3], with added ambient noise and technical artifacts, such as electrode micro-motion or instrumentation noise. The measured electrical activity inside the gray matter of the brain can be used for muscle control, and also sensory perception, such as seeing and hearing, speech, decision making, and self-control. In a brain–machine interface (BMI) system, spikes are first detected from the background noise by comparing the recorded and filtered voltage waveforms with a threshold, which is commonly estimated as the scaled value of background noise. Neighboring neurons often fire spikes of similar shape and amplitude, however, relative to their distances to an electrode's tip, the shape of spike waveforms may differ among neurons. This fact allows the spiking activity of individual neurons to be separated through the spike sorting process [4]. It has already been verified that robust BMIs can also be implemented without employing spike sorting [5–9]. In this case, all threshold crossings (TCs) of the recorded and filtered voltage waveforms associated with an electrode (channel) are treated as spikes from one putative neuron. By considering the number of spikes over a given time (spike count) as the feature of interest, the transmission data rate is significantly reduced compared to transmitting the spike waveforms [10]. Three types of signals can be obtained by intracortical recordings: (a) local field potentials (LFPs), which are extracted by low-pass filtering (<300 Hz) of the

Table 1. The *in vivo* BMI states and the in silico signal processing of the designed brain-switch scheme.

BMI states	User's activities	Behavioral example	In silico signal processing
'Standby'	Non-BMI activities	Sleeping Eating	Intention estimation
'Active'	Only BMI activity BMI activity + Non-BMI activities	Prosthesis control Prosthesis and eating	Neural decoding

neural activity in the vicinity of an electrode tip; (a) multi-unit activity (MUA), which is obtained by high-pass filtering of the recorded neural activity and detecting APs from the ambient noise; and (c) single-unit activity (SUA), which are the detected MUA APs clustered into different groups associated with putative neurons. Wireless transmission of LFPs requires substantially more data rate, as LFPs are continuous signals whereas SUA and MUA measure the instances of single or an ensemble of spikes and hence, can be represented as discrete events. For brevity, the subsequent references to spikes refers to MUA spikes rather than sorted SUAs. For example, consider a 1000 channel neural recording system operating at 20 kS s^{-1} with 12 bits per sample. Transmitting filtered neural signals would require a data rate of 240 Mbps, while transmitting only spike counts, assuming up to one spike per millisecond per recording channel, would reduce the data rate to only one Mbps. The data rate can be even further reduced by spike binning, i.e. transmitting spike counts over a larger time intervals in the order of tens of milliseconds. Assuming a wireless transmission energy of 8.5 pJ bit^{-1} [11], the former requires 2 mW of power while the latter requires only 8.5 μW, over 99% reduction.

In synchronous BMIs the user's neural activity is processed within a predefined time frame, while in asynchronous or self-paced BMIs, the user can interact with the BMI at their leisure, which is more convenient in practical applications. When the user is not actively engaged in a BMI task, the power consumption of the *in vivo* interface can be reduced to extend the operational lifespan of implanted devices. Transitioning the BMI from a 'standby' state back to the 'active' state requires the BMI to estimate and differentiate between different mental states. In a two state 'brain-switch' approach [12–14], the brain-switch module continuously monitors neural signal features to detect whether the user would like to engage in the underlying BMI activity and hence, only processing neural signals during the 'active' state.

Both non-invasive recordings, such as electroencephalography (EEG) and near-infrared spectroscopy, and invasive recordings, such as electrocorticography (ECoG) and intracortical MEAs, have been employed for the brain-switch scheme. The non-invasive methods are more relaxed compared to the invasive methods as they do not impose stringent constraints on power consumption. In all the recordings, neural response patterns are used to differentiate between the 'standby' and 'active' mental states. In some cases, the neural signals are user modulated, such as the imagined motor movements in the EEG-based studies [15, 16], while in others the response is driven by specific stimuli, such as the event-related desynchronization in ECoG-based studies [12]. For intracortical recordings, changes in both the spike firing rates and the spectral power of the LFPs have been used for mental state estimation [13], however, no practical *in vivo* realization of the brain-switch scheme has been reported. Moreover, modulation of neural activity unrelated to the BMI application can be ignored to avoid spurious decoding outputs.

Table 1 lists the two considered BMI states, the example of the user's activities in each state, and in silico signal processing operation. The 'standby' state is related to when the user activity is not related to the underlying BMI application, such as sleeping or for example, if the BMI task is cursor control but the user is eating. The 'active' state is related to the user-modulated neural activity for either only the underlying BMI application or also consisting of unrelated activities, such as the user controlling a robotic prosthesis for eating. During the 'standby' mode, the *in vivo* BMI monitors a subset of recording channels and the in silico signal processing executes the intention-estimation algorithm. During the 'active' mode, the BMI processes all recording channels using an in silico neural decoding. While the transition from the 'standby' to 'active' mode is estimated using the designed brain-switch scheme, the transition from 'active' to 'standby' can be estimated using the outputs of the neural decoder.

For example, consider a 1000-channel implanted wireless BMI used to control assistive devices with three possible configurations. In Configuration I, the device is continuously detecting and transmitting the binned spike counts from all channels, analogous to a synchronous BMI system, which processes all recorded signals within a predefined time frame. Configurations II.a and II.b both realize an asynchronous BMI, in which a relatively small subset of channels are used to realize the designed brain-switch scheme. When the brain-switch algorithm detects BMI activities, the BMI transitions from 'standby' to 'active' state and all channels are enabled for spike detection and processing. Configuration II.a realizes the

Table 2. The energy dissipation of three alternative wireless BMI configurations. Configuration I constantly detects and transmits MUAs for all recording channels; Configuration II.a transmits and detects MUAs for a subset of channels for realizing a brain-switch algorithm on an external device; Configuration II.b detects MUAs for a subset of channels and realizes a brain-switch algorithm *in vivo*. Configurations II.a and II.b enable processing on all channels when the brain-switch algorithm denotes an 'active' mental state.

BMI Configuration	I	II.a	II.b
'Active' detection power (mW)/energy (J)	30/2592	30/108	
'Standby' detection power (mW)/energy (J)	N/A	3/248.4	
'Active'/'Standby' transmission energy (J)	0.73/ N/A	0.03/0.07	0.03/0
Total energy (J)	2622.73	389.50	389.43
Detection energy savings (%)	N/A	86	
Transmission energy savings (%)	N/A	86	96
Total energy savings (%)	N/A	85.14	85.15

brain-switch algorithm in silico on a computer or a portable computational device by processing the received neural signals, however, Configuration II.b realizes the brain-switch algorithm *in vivo*, which reduces the power consumption of the wireless transmission during the 'standby' mode, at the cost of the power consumption and silicon area of the *in vivo* brain-switch circuitry.

Table 2 gives the energy dissipation and savings of the three alternative configurations. Assuming that the power consumption of the analog front end (AFE) for amplifying and digitizing the recorded neural signals are the same for both configurations, the power consumption of the *in vivo* spike detection and wireless transmission are compared. Employing an analog-to-digital converter with 12–16 bits resolution and sampling between 10 kHz–30 kHz [17–20], the nominal power consumption for the AFE circuitry ranges between 0.75 μW and 7.3 μW per channel. Assuming that detecting spikes from 1000 channels requires 0.03 mW of power per channel [21], the power consumption of detecting spikes for the implanted BMI interface is approximately 30 mW, for each of the three configurations. If the *in vivo* BMI interface transmits the spike counts in 1 ms intervals to represent the brain's neural activities, the data rate is reduced to one Mbps with the transmission energy of 8.5 pJ bit^{-1} [11]. For configurations II.a and II.b, let us assume that 10% of the channels are sufficient to detect the user's intention and that they are engaged with the BMI for an average of 1 h per day. Energy dissipated during 'active'/'standby' BMI operation is given as $P \times \Delta_t \times 3.6$ J mW^{-1}, where P denotes the power consumption (in mW), and Δ_t denotes the amount of time (in hours) over which the power is consumed. Data transmission energy dissipation during 'active'/'standby' BMI operation is given as $\eta \times f_b \times \Delta_t \times 3600$ J W^{-1}, where η denotes the transmission energy of 8.5 pJ bit^{-1} [11], and f_b denotes the output data rate for a 1000-channel system (1000 kbps). It can be noted that Configuration II.b offers the highest potential energy savings. Moreover, it can be seen that for the TC-based BMIs, the main source of power consumption is spike detection. Given that the mean detection power of Configuration I is 30 mW, if an SAFT LS14250 battery [22], which has a nominal capacity of 4.32 Watt-hours, is used, the battery life for spike detection is 144 h. By applying spike detection to a smaller subset of channels during 'standby' periods, the mean detection power of Configurations II.a and II.b are reduced to 16.5 mW, a 45% decrease, extending the battery life to 261 h, an 81% increase. Note that table 2 does not account for the power consumption of the brain-state estimation circuitry in Configuration II.b. Since the power consumption difference between Configurations II.a and II.b is negligible, we propose to realize Configuration II.a for the BMI system and to estimate the user's intention in silico (i.e. in software running on a personal computing device).

This article presents the power-efficient realization of the *in vivo* interface for asynchronous BMIs using a neural network-based brain-switch scheme. The rest of this article is organized as follows. Section 2 discusses the signal processing and alternative brain-switch algorithms for detecting the user's mental states. The details of the designed asynchronous spike detection module in hardware is discussed in section 3. The estimated silicon area and power consumption of the spike detection application-specific integrated circuit (ASIC) is also presented. As a proof-of-concept, a system-level realization of the designed asynchronous BMI utilizing the designed brain-switch scheme and the in silico neural decoding is presented in section 4. Finally, section 5 makes some concluding remarks.

2. Neural signals and brain-switches

To drastically reduce wireless data transmission between the *in vivo* implanted circuitry and offline software processing, various compression techniques have been used, from relatively simple methods, such as difference encoding [23], to more complex techniques, such as compressed sensing [24, 25]. Wireless data transmission can be further reduced by transmitting neural information only when the user is intending to engage in a BMI-related action. The data

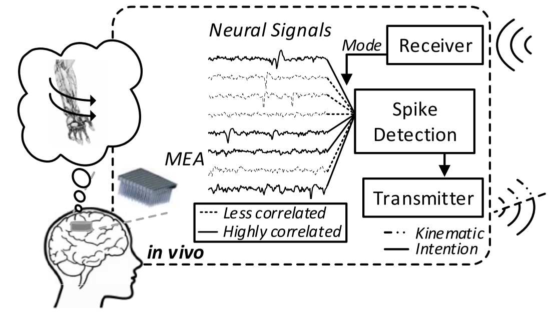

Figure 1. The block diagram of the designed intention-aware *in vivo* BMI.

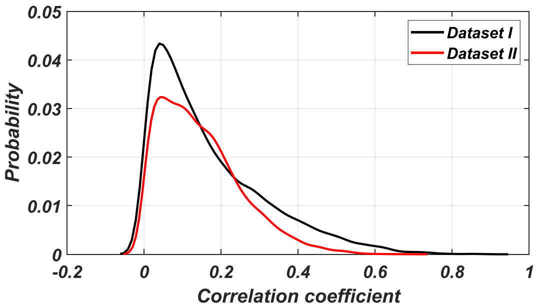

Figure 2. The probability distributions of correlation coefficients between recording channels and the state transitions over the trials in Datasets I and II.

rate can further be reduced by only transmitting the neural information for a subset of channels for estimating when the user wants to engage with a BMI task, as shown in figure 1.

Implementing a brain-switch effectively involves the detection of a mental state transition from ‘standby’ to ‘active’ modes, which controls the *in vivo* processing and wireless transmission of neural signals. Three mental state estimation schemes have been previously investigated [14]. The simplest method is a threshold-based approach, in which a specific neural signal or signal feature crossing a threshold is interpreted as a mental state transition. The threshold value is derived from the training data. The second approach is a classifier-based thresholding technique, in which the neural signals or features are passed to a classification algorithm to produce a continuous output signal, such as the probability of transition, which is then compared to a pre-defined threshold, as in the first scheme. Finally, in a classifier-only-based approach, the output of a classification algorithm is used directly. The output of the classifier may indicate a discrete mental state, such as ‘idle’, ‘planning’, and ‘movement’, which can be used to directly enable or disable further processing based on the predicted mental state. All brain-switch algorithms assume that neural signals exhibit different behavior during mental state transitions. For example, EEG-based brain-switches may detect an increase in the 1 Hz–4 Hz frequency band power [15, 16], while ECoG-based brain-switches may detect a change in the power of the mu and beta frequency bands for motor execution/imagery [12]. For spike-based signals, the number of SUA spikes during a time window [26] as well as changes in the firing rates [13] have been considered as the features for the brain-switch algorithms. Therefore, we assume that a subset of channels will exhibit enough variations in MUAs that will allow a classification algorithm to relatively reliably detect mental state transitions. We employ two publicly available neural datasets. Datasets I (I140703) and II (L101210) [27], which contain raw recordings from the motor cortex (M1 region) of two Rhesus macaque monkeys employing a 96-channel Utah Array. The recordings were sampled at 30 KHz and then downsampled to 10 KHz. The monkeys performed a cued reach and grasp task to displace an object. During the experiments, the monkeys were presented with a series of cues to indicate that a trial was beginning and to specify one of four combinations of grip and displacement forces to use. After the cues were given, the monkeys released a switch on the table to reach for the object. Upon a successful reach, grasp, and displacement, the monkeys were rewarded. For the active brain-state estimation, the switch release time is considered as the ground-truth intention time. To identify which channels exhibit greater changes during a state transition, a correlation analysis is performed. More specifically, following spike detection, the correlation between the spike counts and a known transition response (i.e. a signal that transitions from zero to one at a pre-defined time) is calculated. Then, the correlation between the spike counts for all channels and the transition from a go cue and the switch release event are calculated. Dataset I and II consist of 153 and 149 trials, respectively. A training subset consisting of the first 70% of the trials are used to analyze the correlation coefficients between the spike counts for each recording channel and the state transition. The mean correlation coefficient over Dataset I and II were 0.15 ($\sigma = 0.13$) and 0.15 ($\sigma = 0.11$), respectively. The probability distributions of correlation coefficients between recording channels and the state transitions over the trials in Datasets I and II are shown in figure 2. It can be seen that correlation values equal to or greater than 0.30 and 0.25 for Datasets I and II, respectively, constitute the top 15% percent and are considered highly correlated to the ‘active’ brain-state.

After calculating the correlation coefficients for each channel and trial, the mean correlation for the *N* highest correlated subset of channels is calculated. Since the state transition event is of interest, the response of the correlated channels during this transition time is considered. To represent the activity of the correlated channels, the sum of the correlated

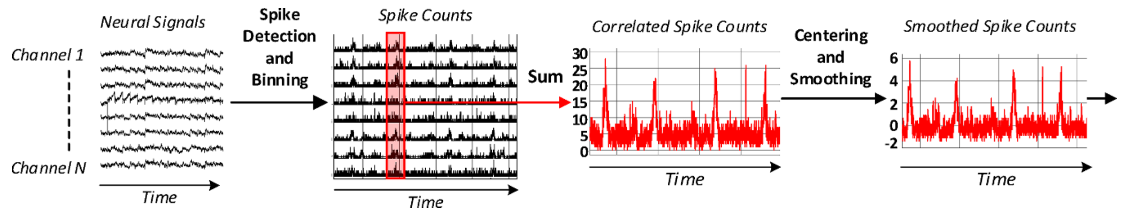

Figure 3. Generating an estimate of the correlated channel's spike density. Binned spike counts are added per time step for a subset of correlated channels, followed by centering and smoothing.

channels is computed per time step. The centering via Z-scoring is applied followed by the Gaussian kernel smoothing to estimate the spike density function [28]. The described process is shown in figure 3, where the correlated spike counts refer to the accumulated spike counts of only the N highest correlated channels and the smoothed spike counts represents the estimated spike density function after applying the centering and smoothing.

To reduce the power consumption of the *in vivo* signal processing, it is preferable to enable as few correlated channels as possible. As shown in figure 4(a), since the centering and smoothing operations perform a normalization of the signal, the response of a smaller number of correlated channels will be similar to that of a higher number of correlated channels. As shown in figure 4(a), the behavior of the estimated spike density for the correlated channels begins to deviate from the baseline during the switch release event. Note that the activity of the correlated channels during the BMI and non-BMI operation may impact the decoding performance, however, the brain-switch algorithm is only operating during the non-BMI operations. Figure 4(b) shows the boxplot of the estimated spike density for different number of correlated channels. While the mean variability is similar over different number of channels, it can be seen that the overall variability spread for eight channels is smaller than that of a higher channel counts. Thus, eight highest correlated channels are selected for the brain-switch algorithm. For Dataset I, the mean correlation of the eight highest correlated channels is 0.38 (std. 0.017) for the BMI period, and 0.25 (std. 0.016) during the non-BMI period, a 34% decrease. For Dataset II, the mean correlation is 0.28 (std. 0.02) for the BMI period, and 0.20 (std. 0.005) during the non-BMI period, a 28% decrease.

2.1. Brain-switch algorithms

For estimating the user's intention, three alternative models are considered: the hidden Markov model (HMM) [29], a feed-forward neural network (FNN), and a recurrent neural network (RNN). The HMM is a stochastic model that estimates the probability of being in a particular state based on the input data and the current state. The mental state can be modeled

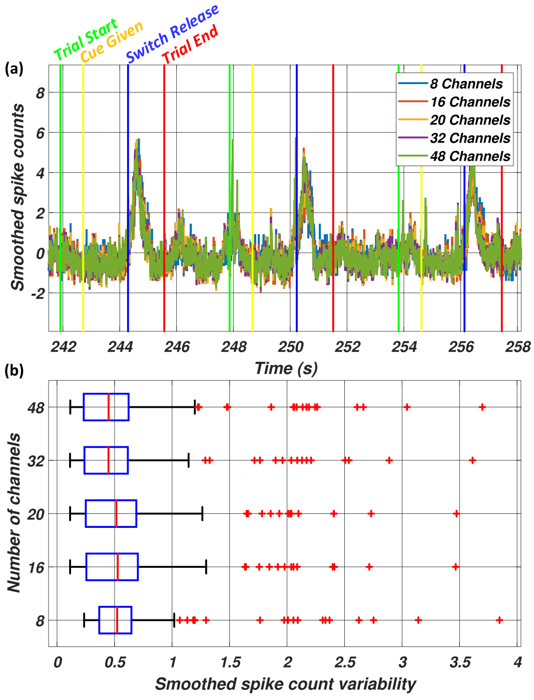

Figure 4. (a) A sample from Dataset I of the smoothed spike counts during the trials and (b) the smoothed spike count variability during the non-BMI periods over both datasets.

as a two-state HMM, where one is for the 'standby' state while the other indicates that the user is in an 'active' state. The forward algorithm [29] then calculates the likelihood of being in a Markov state X_i after a sequence of n observations $O_0(t_0 = 0), O_1(t_1 = 1), \ldots, O_n(t_n = n)$ as $P_i(t_n) = \sum_{j=1}^{M} \left[P_j(t_{n-1})\boldsymbol{A}_{ij}\boldsymbol{B}_{ij}\right]$, where M denotes the total number of Markov states, i and j denote the indices of Markov states, and $P_i(t_n)$ denotes the probability of being in state P_i at time $t = t_n$. We also consider machine learning (ML)-based methods using FNNs and RNNs for brain-state estimation. Both FNNs and RNNs use artificial neurons that each compute an output $y = f(z)$, where z denotes an accumulated weighted input and $f(\cdot)$ denotes a non-linear activation function. The weighted input z is computed by multiplying the inputs with a weight matrix $\mathbf{W_x}$ that is obtained during training. While FNNs process data in a forward direction from the input layer to the output layer, RNNs employ a self-recurrent connection which attempts to learn from temporal features in the data. As a result, RNNs employ additional weight matrices $\mathbf{W_R}$, denoting the self-recurrent weights, in addition to the input weight matrix $\mathbf{W_x}$. Both the FNN and RNNs use the rectified linear unit (ReLU) activation function at the output of the artificial neurons. The RNN also employs the ReLU function in its recurrent connections. The employed FNN and RNN models for mental state estimation have eight artificial neurons in a single hidden layer, with one unit at the output. To quantify the accuracy of the employed brain-state estimation schemes, the *F*-score and the Pearson CC metric are assessed. The CC is given as:

$$\mathrm{CC} = \frac{\sum_i (t_i - \bar{t})(\hat{t}_i - \bar{\hat{t}})}{\sqrt{\sum_i (t_i - \bar{t})}\sqrt{\sum_i (\hat{t}_i - \bar{\hat{t}})}},$$

where t and $\hat{t}$ denote the actual and estimated intention time, respectively, $\bar{t}$ and $\bar{\hat{t}}$ denote the mean of the actual and estimated intention time, accumulated over i estimations, are assessed. The *F*-score and the CC metrics can be interpreted as a measure of the accuracy of detecting the intention and as the precision of detection, respectively. Note that in the context of BMIs, an algorithm with a higher *F*-Score and an acceptable CC is preferable over an algorithm with a relatively low *F*-Score and a high CC. A highly precise intention estimator, i.e. high CC, will be of no use when the intention is not detected. Previous studies among BCI users have shown that the overall system's reliability is far more important than it is speed [30].

For training the intention detectors, two-second snippets of smoothed spike counts over various animal reaches are extracted from the Datasets I and II. For the intention-positive snippets, the switch release time was aligned at one second. The outputs of the models were a zero, indicating no intention, or one, indicating the BMI-related intention. The FNN has no temporal state and the two-second reach data was divided into sequences of three consecutive, non-overlapping smoothed spike counts. The RNN accepts one input per time step per training sample, while the FNN accepts three inputs per training sample. Both the RNN and FNN models were trained using Python's Tensorflow framework and to avoid overfitting to training data, early stopping was used to monitor the mean squared error metric and stop training when the validation error was no longer reduced. To evaluate the performance of the models after training, a 0.5 threshold was applied to the ReLU outputs to generate the predicted state, e.g. an output greater than or equal to 0.5 was interpreted as an 'active' state and as a 'standby' state otherwise. Table 3 gives the performance of the three developed brain-state estimation schemes over Datasets I and II. It is shown that both of the ML-based models outperform the HMM in detecting the state transitions. The performance of the RNN is consistent over both datasets while the FNN drops in performance has a considerable drop in performance over Dataset I. Thus, for estimating the brain-state, the RNN model is employed.

Table 3. The performance of alternative brain-state estimation methods over two datasets.

Dataset	Method	*F*-Score (std.)	CC (std.)
I	HMM	0.47 (0.08)	0.83 (0.09)
	FNN	0.72 (0.31)	0.93 (0.04)
	RNN	0.83 (0.04)	0.89 (0.05)
II	HMM	0.40 (0.02)	0.83 (0.12)
	FNN	0.92 (0.02)	0.98 (0.02)
	RNN	0.95 (0.03)	0.91 (0.05)

3. Hardware implementation of the designed intention-aware BMI

To detect the spikes of the individual neurons near the tip of an electrode, the contribution of the LFPs is first removed using a band-pass filter. A more computationally-efficient approach to remove the low-frequency components of the signal is employed by subtracting the moving average of the signal, computed over a reasonably short timing window, i.e. $\tilde{x}[n] = x[n] - \bar{x}[n]$, where $x[n]$ denotes the neural signal, k denotes the length of the timing window and $\bar{x}[n] = \frac{1}{k}\sum_n^{n-k} x[n]$ [31]. The mean subtraction filter does not require multiplications, which is a significant saving as *in vivo* spike detection is a continuous operation. It was found that removing the mean of the neural signal over a relatively small timing window of 800 μs is sufficient for removing the LFPs. With a sampling rate of $f_s = 10$ kHz, this corresponds to computing the average of eight samples, which requires eight additions and one right shift operation per input sample per channel. Figure 5 shows an example of the conventional bandpass filter and the filtered recording using the mean subtraction filter. It can be seen that the employed filter preserves the spikes in the bandpass filtered signal while the low-frequency oscillations are eliminated.

After applying the mean subtraction filter, spikes are first emphasized by applying the absolute value operation to the filtered signal. Then spikes are detected by comparing the emphasized signal with a threshold value, which is computed dynamically as a scaled value of the estimated background noise. The noise is estimated as the mean of the emphasized signal over a one-second time window. The accuracy of the detection unit was measured using the *F*-score metric as $F = 2T_\mathrm{P}/(2T_\mathrm{P} + F_\mathrm{N} + F_\mathrm{P})$ [32], which accounts for true positives (T_P), false positives (F_P), and false negatives (F_N) spikes over the widely used Wave_Clus datasets [33]. Compared to the energy-based methods, such as the non-linear energy operator (NEO) and the root-mean-square (RMS) noise

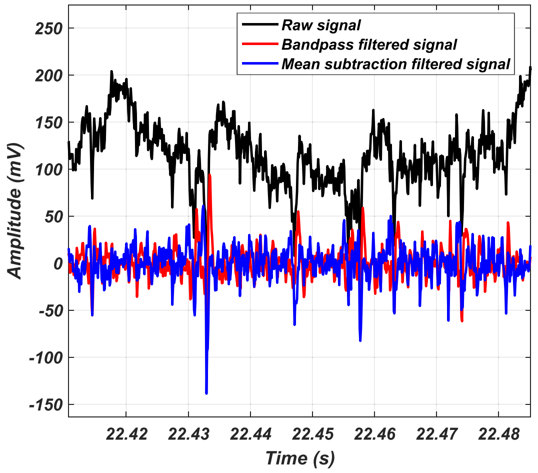

Figure 5. The raw neural signal, the bandpass filtered signal, and the mean subtraction filter outputs.

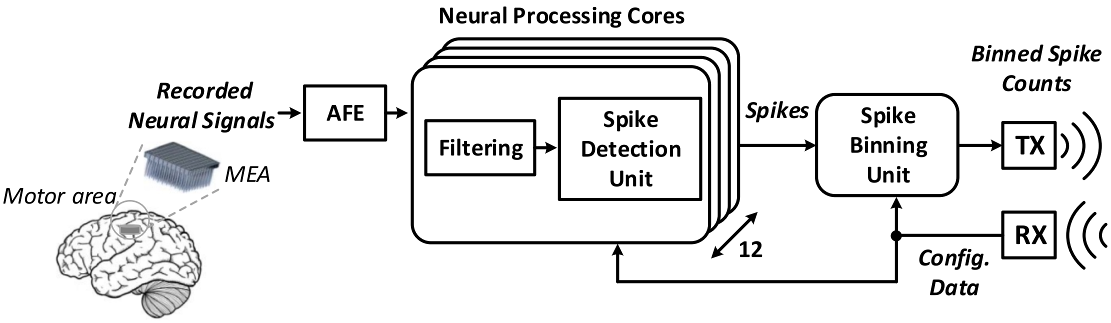

Figure 6. The top-level block diagram of the designed intention-aware *in vivo* BMI.

estimation with an F-Score of 0.94 [34], the employed method shows a comparable F-Score of 0.92 while being less computationally-complex.

The top-level block diagram of the designed asynchronous *in vivo* BMI is shown in figure 6. The raw neural signals recorded by the MEA are amplified, digitized by the AFE, and is passed to twelve *Neural Processing Cores* (NPCs), where each contains a *Filtering* unit to remove the LFPs and a *Spike Detection Unit* to detect spikes in the form of TCs. Note that the realization of the AFE is beyond the scope of this paper and hence, our subsequent power and energy estimations do not account for the AFE. The spikes are then passed to the *Spike Binning Unit*, which counts the number spikes over periods of 10 ms. Each NPC processes eight channels of interleaved recorded data and the operating frequency of the system is $f = 8 \times f_s = 80$ KHz. The NPCs interleaves data processing of eight recording channels by employing the C-Slow architectural transformation [35]. The eight highest correlated channels are configured through the configuration ports, which will enable or disable processing based on the corresponding NPCs' correlation settings. The correlation settings for each channel are obtained on a workstation prior to the system configuration using labeled data collected during the training process. Each NPC receives an 8-bit configuration word indicating which of its eight interleaved data channels are highly correlated to the user's intention.

To remove LFP components from the recorded neural signals, the moving average of the signal over the previous eight values is subtracted from the current value. To account for the interleaved data, the outputs of every eighth register of a 64-word shift register are used to compute the mean of the current data channel using an adder tree and an arithmetic right shift, as shown in figure 7. To reduce the switching activity of the uncorrelated channels, the propagation of new values are disabled by de-asserting the enable signal *EN_Channel*. If an NPC's 8-bit configuration word is equal to zero, the input register enable signal *EN_Core* is de-asserted and no signals will propagate through the delay elements.

The datapath of the *Spike Detection Unit*, as shown in figure 8, consists of an absolute value unit, a correlation shift register *Corr. Shift Reg*, an accumulation and threshold memory *Acc./Thr. Mem.*, which consists of two memory modules, and a control unit *CTRL*. The

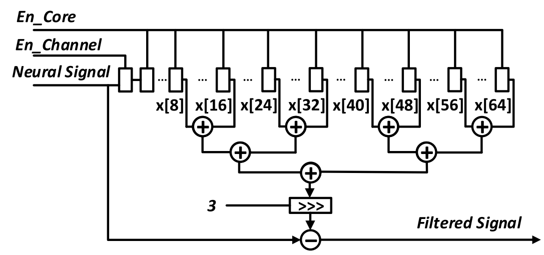

Figure 7. The datapath of the LFP-removal moving average filter.

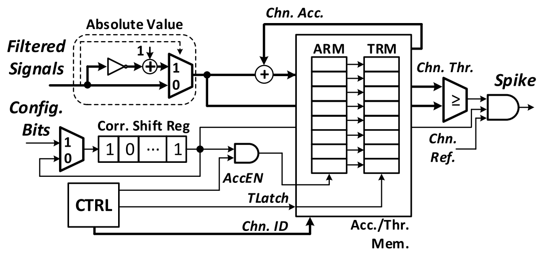

Figure 8. The datapath of the spike detection unit.

Corr. Shift Reg is used to enable or disable the accumulation of rectified filtered signals. The *Corr. Shift Reg* is initialized after the training phase to determine the correlation settings for each of the channels processed by the NPCs. The accumulation register memory *ARM* stores the accumulated rectified signal for the eight NPC channels. The threshold register memory *TRM* stores the threshold values, updated approximately every second ($2^{\lfloor \log_2 f_s \rfloor} \times \frac{1}{f_s} = 0.8192$ s, where f_s denotes the sampling rate). The *CTRL* unit enables the *TRM* to store the corresponding value from the *ARM* after shifting it 10 bits to the right to compute the scaled threshold value. To detect spikes, the rectified filtered signal is compared to the current channel's threshold value. The comparator output is gated with two signals, the *Corr. Shift Reg* output, which is used to disable transient spikes from uncorrelated channels, and a channel refractory *Chn. Ref* signal, which is used to prevent considering multiple output spikes due to the same threshold crossing event.

The spike detection unit can operate in two modes. The first mode of operation is during the system initialization in which a threshold must be derived for each channel processed by the NPC. During this period, a master enable bit is used to allow the processing of uncorrelated channels. This is also the case for when the off-chip intention estimator detects that the user intends to perform a BMI-related activity, where the neural recordings of all channels are required. In the second mode of operation, the master enable bit is de-asserted and only the highest correlated channels are processed.

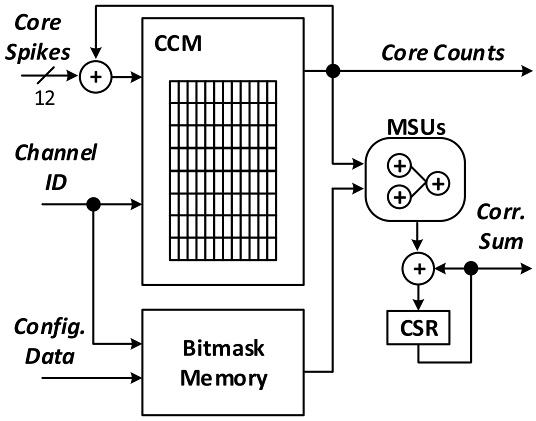

Figure 9. The datapath of the spike binning unit.

The spikes generated by the *Spike Detection Unit* are then processed by the *Spike Binning Unit* shown in figure 9, which accumulates the spikes detected by each of the NPCs in the core count memory *CCM*. The *CCM* stores eight 48-bit words of data, one for each channel of an NPC. Each word is partitioned into twelve 4-bit sections to store the spike counts produced by the 12 NPCs. To compute the accumulated spike counts of the correlated channels, the output *Core Counts* from the *CCM* are added with the masked summing units *MSUs*. The *Bitmask Memory* stores correlated channel information in twelve-bit words, which are used within the *MSUs* to mask the spike counts of uncorrelated channels. The outputs of the MSUs are then accumulated into the correlation sum register *CSR*.

The synthesized ASIC layouts of Configurations I and II.a, implemented in a standard 180 nm CMOS process, is shown in figure 10. Synthesis was performed using Synopsys Design Compiler and the place-and-route was done using Cadence Innovus. The designs were synthesized to support a 96-electrode Utah Array by implementing 12 NPCs, each processing eight channels. To optimize the area and power consumption of the ASIC, we tested the system-level accuracy of the system over various numerical resolutions. Input data, and subsequent internal digital signals, are represented using the fixed-point (WI.WF) number format, where WI and WF denote the number of integer and fraction bits. In our design, data is represented using one integer bit and F fraction bits, where F was between 11 and 2 bits. We found that the system-level accuracy dropped significantly for values of F less than 7. Table 4 gives the ASIC characteristics and implementation results for various configurations of (WI.WF). The power consumption of the 96-channel intention-aware circuitry and the power per active channel is approximately 59 μW and 0.63 μW, respectively. The power was estimated by simulating the synthesized and routed

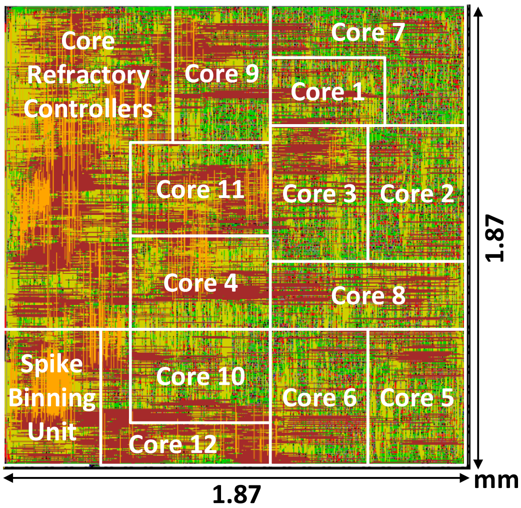

Figure 10. The ASIC layout of the designed intention-aware BMI system employing a 96-electrode Utah Array with 12 spike detection cores.

Table 4. The ASIC characteristics and implementation results of the intention-aware spike detection circuitry over various numerical formats.

	(WI.WF)			
Circuit metric	(1.11)	(1.9)	(1.7)	(1.5)
Core area (mm^2)	4.49	4.00	3.50	3.01
Intention-aware circuitry power (μW)	85.8	71.9	59.6	49.5
Power per active channel (μW)	0.86	0.66	0.63	0.43
Mean system accuracy (CC)	0.81	0.80	0.80	0.75

Table 5. The ASIC characteristics and implementation results of the state-of-the-art spike detection circuits.

Work	Ours	[36]	[37]	[38]	[39]
Technology (nm)	180	180	180	65	130
Supply voltage (V)	1.8	—	1.8	0.8	1.2
Number of channels	96	16	1	64	16
Normalized area per channel (mm^2)[a]	0.03	—	0.03	0.05	1.21
Normalized power per channel (μW)[a]	0.63	4	1.5	4.6	210
Adaptive threshold	Y	N	Y	N	Y
Intention-aware	Y	N	N	N	N
Scaled active/'standby' power consumption (mW)[ba]	0.124/0.067 [c]	0.4	0.15	0.46	21
Scaled energy dissipation (J)[da]	0.611	3.04	1.14	3.496	159.6
Relative increased energy dissipation	—	4.97 ×	1.86 ×	5.72 ×	261.21 ×

[a] Normalized to a 180 nm CMOS Process with a 1.8 V supply voltage, as described in [40].

[b] Assuming 100 channels and using 10 channels for brain-state estimation.

[c] Accounting for the power consumption of the brain-state estimation control circuitry.

[d] Assuming 30 min of real-time operation over a two hour BMI session.

design, considering the switching activity of all nodes in the design.

Table 5 gives the ASIC characteristics and implementation results for various spike detection systems. For a fair comparison, the implementation results have been scaled to a 180 nm CMOS process with a 1.8 V supply voltage, as described in [40]. Also, we compare the area and power consumption of the spike detection circuitry, when able, for a fair comparison. The design in [36] is a 16-channel BMI with a window-discriminator for spike detection. Window discriminators involve dual threshold values and detect spikes when there is a crossing of both the upper and lower threshold values. In [37], an analog implementation of a NEO-based spike detection unit is presented. In [38], the authors present a 64-channel neural signal acquisition system that detects spikes using the NEO-based pre-emphasis and a constant threshold. In [41], a 64-channel neural signal acquisition system is presented. Spikes are detected by an adaptive dual-threshold window comparator. Also, their system supports extracting spike amplitude-related features, such as the time between threshold crossing and negative peak, the time between the negative and positive peak, and the time between the positive peak and the return to baseline. In [39], the authors present a 16-channel exponential component and polynomial component (EC-PC) spike detection system, which involves representing the neural signals in the Hilbert transformed space and predicting occurrences of spikes by estimating the probability that a data point is part of an action potential. The implementation results in table 5 shows that our ASIC design consumes the least power per channel, primarily due to the fact that in an intention-aware BMI system, it is only necessary to detect spikes of a relatively small number of highly correlated channels for the majority of the time. As given in table 5, assuming that 10% of the channels are sufficient for providing adequate brain-state estimation, and assuming one hour of BMI activity in a day, it is clear that our design offers significantly less energy dissipation, 1.8 × less power compared to the state of the art.

While it is shown that the proposed asynchronous system effectively reduces the power consumption of the *in vivo* interface, system-level power saving schemes may also be applicable for further power reduction. For example, in a BMI system that a user primarily interacts by controlling a cursor on a computer screen, if eye-tracking cameras are used, the BMI system can power down the *in vivo* recording circuitry should the user not be paying attention to the screen or if the user is asleep. Note that the results given in table 5 assume that the system only estimates the mental states during the 'standby' BMI mode and transitions to the 'active' state for neural decoding if the BMI-related activity is detected. While the system employs the brain-switch estimation during the 'standby' state, which may involve non-BMI activities, as demonstrated in the next section, the false brain-switch estimation has a negligible impact on the overall decoding performance.

4. Intention-aware decoding

We integrated our designed intention-aware detection unit along with an ML-based neural decoder. During an initial training phase, neural recordings and their associated labeled events are processed to generate training data for the ML-based brain-state estimation and neural decoding algorithms. The labeled training data is then used to determine which channels are highly correlated to the BMI intention and to power-down channels that are not highly-correlated, reducing the power required to operate *in vivo* spike detection when the user is not actively engaged in the BMI activity. Figure 11 shows the system-level block diagram of the designed wireless BMI system for a motor-cortex neural decoding application. Neural activity is detected and accumulated in 10 ms bins per recording channel. During the user's 'active' mental state, multi-unit spikes are detected on each recording channel, and during the 'standby' state, only the eight highest correlated channels are processed to detect the user's mental state. The designed system employs two ML-based models in silico. One is the RNN discussed in section 2 and the second is a temporal-convolutional network (TCN) [42] used as a kinematic decoder to map sequences of binned spike counts onto the intended object displacement velocity. When the RNN identifies an 'active' mental state, the TCN is enabled for two seconds to decode the user's kinematics and the RNN is disabled and it is internal memory is cleared.

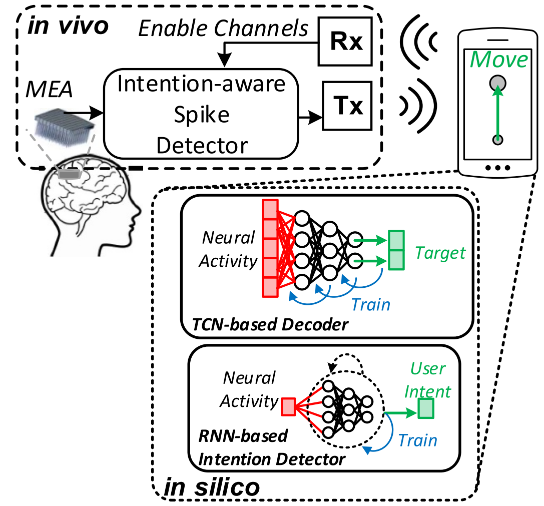

Figure 11. The system-level block diagram of our designed wireless BMI system for neural decoding applications.

To evaluate the performance of the neural decoding, the datasets described in section 2 are used. During the off-line training phase, the velocity of the displaced object is calculated based on the object's displacement and is smoothed using a moving average filter to reduce instrumentation noise. To generate training data, spikes are counted over 10 ms bins, from the time of switch release to the time that a trial ends. The spike counts are Z-scored and smoothed with a Gaussian kernel before being passed to the decoding model. The training data for Dataset I and II contain 138 and 98 trials, respectively, aligned to the switch release time. Trials that were shorter in time were padded with zeros to match the data lengths during training, however, during BMI operation, both the RNN and TCN models produce an output for intention and kinematic estimations, respectively, per input time step.

Similar to the ML-based intention estimator, the TCN-based decoder was trained using the Tensorflow

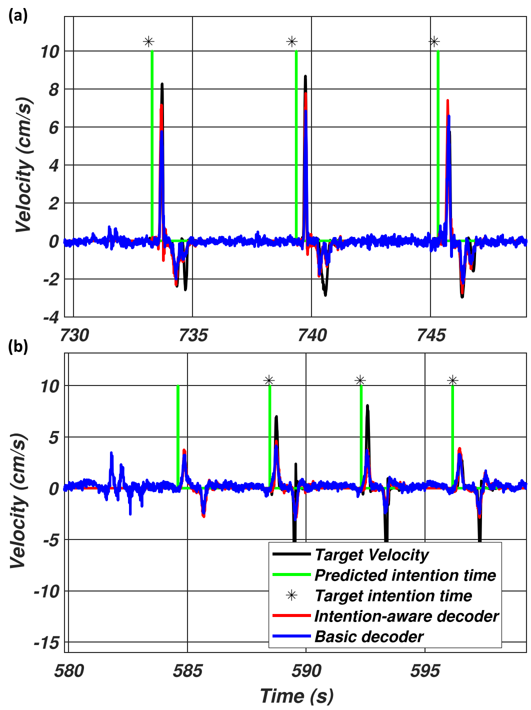

Figure 12. The outputs of the basic and intention-aware TCN-based decoders and the estimated intention times using the RNN-based intention estimator over the testing subset of (a) Dataset I and (b) Dataset II.

framework, the RMSprop optimizer, and the root mean square error loss metric. For evaluating regression performance, the CC metric was used. The model was trained for up to 300 epochs and to reduce overfitting, we employed early stopping when the CC no longer increased. The designed TCN-based decoder achieves a mean CC of 0.8 (std. 0.06) over Datasets I and II. Sample outputs of the TCN-based decoder and the estimated intention times from the RNN-based intention estimator over Datasets I and II are shown in figure 12. For reference, the output of a basic decoder which continuously decodes the neural signals is shown in figure 12. While the basic decoder will generate spurious outputs among trials, the intention-aware decoder will only generate outputs when the brain-switch algorithm enables the TCN-based decoder. Over the test set shown in figure 12, the *F*-score of the RNN-based intention detector was 0.70 and 0.78 over Dataset I and II, respectively. It was found that the mean absolute error of the intention-aware decoder is 0.21 and 0.35 over Datasets I and II, respectively. It was also found that the RNN-based intention detector is more likely to produce a false positive rather than a false negative. However, since the TCN-based decoding model is relatively accurate, the overall performance of the system is not considerably degraded. The primary goal of the proposed design is to enable power savings by decoding only during the 'active' mental state, which will inevitably reduce spurious outputs. For Dataset I and II, the error of the velocity decoding is considered as the mean absolute error. It was found that the mean absolute error of the basic velocity decoder is 0.26 and 0.51 over Datasets I and II, respectively. The intention-aware decoder thus has 18.8% and 35% smaller errors than the basic velocity decoder.

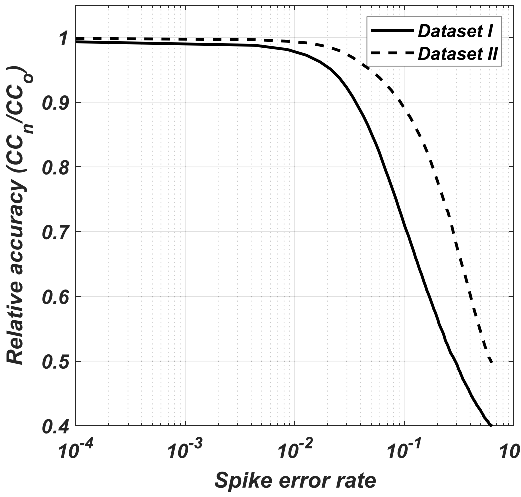

Figure 13. The relative accuracy of the TCN-based decoder over increasing values of SER.

To evaluate the performance of the BMI system in presence of spiking errors, the spiking error rate (SER) metric [43], which represents errors that can occur in the process of detecting spikes and the wireless transmission of spike counts, was used. To simulate SER, spikes were first detected using the designed detection unit and binned into one millisecond intervals. Due to a neuron's refractory period imposed by the spike detection units, it is assumed that in each one millisecond bin at most one spike can be detected per electrode. To inject errors into the bins, bit-flips were applied to the bins. For example, for an SER of 10^{-2}, there is a 1% chance that a spike or non-spike will be considered as its counterpart. Figure 13 shows the performance degradation of the TCN-based decoder over increasing values of SER. It can be seen that the model offers relatively stable performance for up to about 6% error. The robustness of the model can be further improved by increasing the numerical resolution of signals (i.e. reducing quantization noise during detection), or accounting for spiking errors when the training the model itself, for example, adding error into the spike counts.

5. Conclusion

The *in vivo* spike detection has become increasingly challenging when employing high-density multi-electrode arrays in the order of thousands of recording channels. This article demonstrated that an intention-aware BMI system can drastically reduce the power consumption of the *in vivo* spike detection. It was shown that the ML-based algorithms can be used effectively to estimate the user's intention from a relatively small subset of highly-correlated recording channels, which allows disabling the detection circuitry of the remaining uncorrelated channels. Moreover, since the user is mainly not engaged in the BMI activity throughout the day, the impact of the intention-aware BMI is even more effective. The design and implementation of a 96-channel spike detection unit in a standard 180 nm CMOS process was estimated to occupy 0.03 mm^2 of silicon area and consumes 0.63 μW of power per channel while operating at 80 kHz. The designed *in vivo* BMI system was used for neural decoding over two neural recordings. It was shown that the ML-based algorithms can reliably detect the user's intention in the presence of up to 6% spiking errors. Additionally, it was shown that compared to the state-of-the-art BMI systems, incorporating intention awareness reduced the total energy consumption by over 1.8 ×.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://doi.org/10.12751/g-node.f83565 [27].

Acknowledgment

This work was supported by the National Science Foundation Award #2007131 and the San Diego State University Graduate Fellowship Program.

ORCID iD

Daniel Valencia https://orcid.org/0000-0003-4539-8166

References

[1] Wu W, Black M J, Gao Y, Bienenstock E, Serruya M, Shaikhouni A and Donoghue J P 2003 Neural decoding of cursor motion using a Kalman filter *Advances in Neural Inf. Proc. Systems* pp 133–40

[2] Serruya M, Hatsopoulos N, Fellows M, Paninski L and Donoghue J 2003 Robustness of neuroprosthetic decoding algorithms *Biol. Cybern.* **88** 219–28

[3] Gold C, Henze D A, Koch C and Buzsaki G 2006 On the origin of the extracellular action potential waveform: a modeling study *J. Neurophysiol.* **95** 3113–28

[4] Lewicki M S 1998 A review of methods for spike sorting: the detection and classification of neural action potentials *Netw., Comput. Neural Syst.* **9** R53

[5] Todorova S, Sadtler P, Batista A, Chase S and Ventura V 2014 To sort or not to sort: the impact of spike-sorting on neural decoding performance *J. Neural Eng.* **11** 056005

[6] Stark E and Abeles M 2007 Predicting movement from multiunit activity *J. Neurosci.* **27** 8387–94

[7] Wander J D and Rao R P 2014 Brain–computer interfaces: a powerful tool for scientific inquiry *Curr. Opin. Neurobiol.* **25** 70–75

[8] Oby E R, Perel S, Sadtler P T, Ruff D A, Mischel J L, Montez D F, Cohen M R, Batista A P and Chase S M 2016 Extracellular voltage threshold settings can be tuned for optimal encoding of movement and stimulus parameters *J. Neural Eng.* **13** 036009

[9] Willett F R, Avansino D T, Hochberg L R, Henderson J M and Shenoy K V 2021 High-performance brain-to-text communication via handwriting *Nature* **593** 249–54

[10] Valencia D, Thies J and Alimohammad A 2019 Frameworks for efficient brain-computer interfacing *IEEE Trans. Biomed. Circuits Syst.* **13** 1714–22

[11] Miranda H and Meng T H 2010 A programmable pulse UWB transmitter with 34% energy efficiency for multichannel neuro-recording systems *IEEE Custom Integrated Circuits Conf. 2010* (IEEE) pp 1–4

[12] Williams J J, Rouse A G, Thongpang S, Williams J C and Moran D W 2013 Differentiating closed-loop cortical intention from rest: building an asynchronous electrocorticographic BCI *J. Neural Eng.* **10** 046001

[13] Williams J J, Tien R N, Inoue Y and Schwartz A B 2016 Idle state classification using spiking activity and local field potentials in a brain computer interface *Annual Int. Conf. IEEE Engineering in Medicine and Biology Society* (IEEE) pp 1572–5

[14] Han C-H, Müller K-R and Hwang H-J 2020 Brain-switches for asynchronous brain–computer interfaces: a systematic review *Electronics* **9** 422

[15] Bozorgzadeh Z, Birch G E and Mason S G 2000 The LF-ASD brain computer interface: on-line identification of imagined finger flexions in the spontaneous EEG of able-bodied subjects *IEEE Int. Conf. on Acoustics, Speech and Signal Processing* vol 4 (IEEE) pp 2385–8

[16] Borisoff J F, Mason S G and Birch G E 2006 Brain interface research for asynchronous control applications *IEEE Trans. Neural Syst. Rehabil. Eng.* **14** 160–4

[17] Samiei A and Hashemi H 2021 A bidirectional neural interface SoC with adaptive IIR stimulation artifact cancelers *IEEE J. Solid-State Circuits* **56** 2142–57

[18] Shen L, Lu N and Sun N 2018 A 1-V 0.25-μw inverter stacking amplifier with 1.07 noise efficiency factor *IEEE J. Solid-State Circuits* **53** 896–905

[19] Chandrakumar H and Marković D 2018 A 15.2-ENOB 5-kHz BW 4.5-μW chopped CT $\Delta\Sigma$-ADC for artifact-tolerant neural recording front ends *IEEE J. Solid-State Circuits* **53** 3470–83

[20] Yoshimoto S, Araki T, Uemura T, Nezu T, Sekitani T, Suzuki T, Yoshida F and Hirata M 2016 Implantable wireless 64-channel system with flexible ECoG electrode and optogenetics probe *2016 IEEE Biomedical Circuits and Systems Conf. (BioCAS)* (IEEE) pp 476–9

[21] Zhang Z, Savolainen O W and Constandinou T G 2022 Algorithm and hardware considerations for real-time neural signal on-implant processing *J. Neural Eng.* **19** 016029

[22] Simeral J D *et al* 2021 Home use of a percutaneous wireless intracortical brain-computer interface by individuals with tetraplegia *IEEE Trans. Biomed. Eng.* **68** 2313–25

[23] Sedgewick R and Wayne K 2011 *Algorithms* (Boston, MA: Addison-Wesley Professional)

[24] Wang A, Jin Z, Song C and Xu W 2015 Adaptive compressed sensing architecture in wireless brain-computer interface *Proc. Design Automation Conf.* pp 1–6

[25] Shrivastwa R R, Pudi V, Duo C, So R, Chattopadhyay A and Cuntai G 2020 A brain–computer interface framework based on compressive sensing and deep learning *IEEE Consum. Electron. Mag.* **9** 90–96

[26] Achtman N, Afshar A, Santhanam G, Yu B M, Ryu S I and Shenoy K V 2007 Free-paced high-performance brain–computer interfaces *J. Neural Eng.* **4** 336–47

[27] Brochier T, Zehl L, Hao Y, Duret M, Sprenger J, Denker M, Grün S and Riehle A 2018 Massively parallel recordings in macaque motor cortex during an instructed delayed reach-to-grasp task *Sci. Data* **5** 1–23

[28] Szűcs A 1998 Applications of the spike density function in analysis of neuronal firing patterns *J. Neurosci. Methods* **81** 159–67

[29] Rabiner L and Juang B 1986 An introduction to hidden Markov models *IEEE Acoust. Speech Signal Process. Mag.* **3** 4–16

[30] Kübler A, Holz E M, Riccio A, Zickler C, Kaufmann T, Kleih S C, Staiger-Sälzer P, Desideri L, Hoogerwerf E-J and Mattia D 2014 The user-centered design as novel perspective for evaluating the usability of BCI-controlled applications *PLoS One* **9** e112392

[31] Zhang Z and Constandinou T G 2021 Adaptive spike detection and hardware optimization towards autonomous, high-channel-count BMIs *J. Neurosci. Methods* **354** 109103

[32] Van Rijsbergen C J 1979 *Information retrieval* 2nd edn (Newton, MA: Butterworth-Heinemann)

[33] Quiroga R Q, Nadasdy Z and Ben-Shaul Y 2004 Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering *Neural Comput.* **16** 1661–87

[34] Valencia D, Mercier P P and Alimohammad A 2022 *In vivo* neural spike detection with adaptive noise estimation *J. Neural Eng.* **19** 046018

[35] Leiserson C E, Rose F M and Saxe J B 1983 Optimizing synchronous circuitry by retiming *Caltech Conf. on Very Large Scale Integration* (Springer) pp 87–116

[36] Liu X, Zhang M, Richardson A G, Lucas T H and van der Spiegel J 2016 Design of a closed-loop, bidirectional brain machine interface system with energy efficient neural feature extraction and PID control *IEEE Trans. Biomed. Circuits Syst.* **11** 729–42

[37] Koutsos E, Paraskevopoulou S E and Constandinou T G 2013 A 1.5 μw NEO-based spike detector with adaptive-threshold for calibration-free multichannel neural interfaces *IEEE Int. Symp. on Circuits and Systems* pp 1922–5

[38] Biederman W, Yeager D J, Narevsky N, Leverett J, Neely R, Carmena J M, Alon E and Rabaey J M 2015 A 4.78 mm^2 fully-integrated neuromodulation SoC combining 64 acquisition channels with digital compression and simultaneous dual stimulation *IEEE J. Solid-State Circuits* **50** 1038–47

[39] Wu T, Xu J, Lian Y, Khalili A, Rastegarnia A, Guan C and Yang Z 2015 A 16-channel nonparametric spike detection ASIC based on EC-PC decomposition *IEEE Trans. Biomed. Circuits Syst.* **10** 3–17

[40] Stillmaker A, Xiao Z and Baas B 2011 Toward more accurate scaling estimates of CMOS circuits from 180 nm to 22 nm *Technical Report* ECE-VCL-2011-4 VLSI Computation Lab, ECE Department, University of California, Davis vol 4 p m8

[41] Delgado-Restituto M, Rodriguez-Perez A, Darie A, Soto-Sánchez C, Fernández-Jover E and Rodriguez-Vazquez A 2017 System-level design of a 64-channel low power neural spike recording sensor *IEEE Trans. Biomed. Circuits Syst.* **11** 420–33

[42] Lea C, Vidal R, Reiter A and Hager G D 2016 Temporal convolutional networks: a unified approach to action segmentation *European Conf. on Computer Vision* (Springer) pp 47–54

[43] Even-Chen N, Muratore D G, Stavisky S D, Hochberg L R, Henderson J M, Murmann B and Shenoy K V 2020 Power-saving design opportunities for wireless intracortical brain–computer interfaces *Nat. Biomed. Eng.* **4** 984–96