31st Annual International Conference of the IEEE EMBS
Minneapolis, Minnesota, USA, September 2-6, 2009

VLSI Architecture of NEO Spike Detection with Noise Shaping Filter and Feature Extraction Using Informative Samples

Linh Hoang, Zhi Yang, Wentai Liu
School of Engineering, University of California at Santa Cruz, CA 95064
{linh, yangzhi, wentai}@soe.ucsc.edu

Abstract— **An emerging class of multi-channel neural recording systems aims to simultaneously monitor the activity of many neurons by miniaturizing and increasing the number of recording channels. Vast volume of data from the recording systems, however, presents a challenge for processing and transmitting wirelessly. An on-chip neural signal processor is needed for filtering uninterested recording samples and performing spike sorting. This paper presents a VLSI architecture of a neural signal processor that can reliably detect spike via a nonlinear energy operator, enhance spike signal over noise ratio by a noise shaping filter, and select meaningful recording samples for clustering by using informative samples. The architecture is implemented in 90-nm CMOS process, occupies 0.2 mm^2, and consumes 0.5 mW of power.**

I. Introduction

Recent advancements in neural signals recording systems [1] enable neuroscientists and clinicians capture simultaneous activity of many neurons in the brain for analysis and studies. By using implantable microelectromechanical systems (MEMS) multielectrode arrays [2] placed in the cerebral cortex, neuroscientists able to observe neurons communicate with one another by way of electrical activity, which is known as action potentials or simply as spikes.

The direct applications for these multi-channel neural recording and processing capable systems are the enabling technologies for neuroprosthetic devices—devices those can be controlled by thoughts. As reported in literature, neural signals recorded from monkey's motor cortex were analyzed to build a relationship between neural activities and intended limbs movements then used to control a cursor on a computer screen or a robotic arm [3]. The positive achievements of the emerging technologies in brain-machine interface yearn for feedback mechanisms enabling the brain perceive information through prosthetic sensors. Building a realistic bionic arm [4] is an example research that incorporates multi-channel neural recording and stimulation technologies for actuating a robotic arm and perceiving senses from the prosthetic sensors respectively.

This work was supported in part by UCOP. The authors wish to acknowledge the support of TMSC for chip fabrication and ARM for physical IP.

To surmount the challenging requirements of an implantable neuroprosthetic device that is low power, small footprint, high performance signal processing and limited wireless data rate great efforts are aimed at developing hardware efficient algorithms and architectures. On-chip signal processing can reduce the wireless data transmission, provide real-time computing solutions to complex spike sorting problem and enable a closed-loop neuroprosthetic framework. In this paper, we will briefly present our spike detection with noise shaping filter and feature extraction algorithms using informative samples along with a detail description of a cost-effective hardware architecture. Section II reviews our new spike sorting algorithm through each of the processing steps. Section III describes the architecture and hardware implementation. The results and future works are presented in Section IV.

II. Algorithms

Spike sorting is a process of assigning spikes to different neurons. The process can be broken down into three major steps as follow: spike detection, feature extraction and clustering. In this section, we present our spike detection and feature extraction algorithms those are implemented in hardware as describe in Section III. For the next step in spike sorting process, we briefly discuss our new clustering algorithm that uses results from feature extraction for grouping neurons.

A. Spike Detection

The purpose of spike detection is to identify a neural spike from ambient noise or idle period of a neuron. Unfortunately, signal-to-noise ratio (SNR) can be as low as 0dB making it difficult to detect accurately with a simple amplitude thresholding. A solution for this 0dB SNR spike detection is to employ a nonlinear energy operator filter [5].

1) Nonlinear Energy Operator: NEO was originally invented by Teager [6] and was used for the amplitude and frequency demodulation and speed analysis. It computes the energy function by using both the amplitude and the frequency characteristics as formulated by (1). Since a spike is typically characterized by localized high frequency and instantaneous energy, NEO is an appropriate candidate functions as a spike detector (because NEO outputs a spike when $x(n-1)$ and $x(n+1)$ are small, which represent a fast change from a high instantaneous energy $x^2(n)$.) Although the equation is deceptively simple, comprehensive formulation and interpretation are complicated; a simplified formulation applied with spike signals is presented in [7].

$$\Psi(x(n)) = x^2(n) - x(n+1)x(n-1). \tag{1}$$

By using NEO as a spike detector, it is possible to detect 99.5% spikes as a worst case when used spike data from waveclus [7].

B. Feature Extraction

Two goals of feature extraction are: to remove commonalities information between different spikes and emphasize their uniqueness. This is achievable by carefully examining the neurons' signatures and noise shaping for feature extraction. Further data reduction is possible through the selecting of a subset informative samples in the waveforms are extracted as the features. The result can be used for designing a frequency and noise shaping filter.

1) Frequency and Noise Shaping Filter: The spikes from neurons with similar ion channel populations and distances to the recording electrode have similar waveforms. A solution for sorting similar spikes is to differentiate the neuronal geometry signatures. Assume $W_1(t)$ and $W_2(t)$ are the geometry kernel functions of two neurons and $j_m(\tau)$ is the transmembrane current profile, the difference between two spikes is:

$$\Delta V(t) = \int j_m(\tau)\,[W_1(t-\tau) - W_2(t-\tau)]\,d\tau. \tag{2}$$

A small waveform difference appears if $\int (W_1(t) - W_2(t))\,dt \approx 0$. To differentiate the waveforms we can rewrite (2) in the frequency domain as

$$\mathscr{F}(\Delta V) = \mathscr{F}(j_m)\mathscr{F}(W_1 - W_2), \tag{3}$$

where $\mathscr{F}(\cdot)$ denotes the Fourier transform. The previous condition of $\int (W_1(t) - W_2(t))\,dt \approx 0$ is equivalent to $\mathscr{F}(W_1 - W_2) \approx 0|_{f=0Hz}$, which suggests that the waveform difference caused by the geometry kernel functions locate at a higher frequency spectrum. A frequency-shaping filter that emphasizes on high frequency spectrum can help differentiate similar spikes but adversely amplify high frequency thermal noise. A compromise solution is to use derivative as the most effective frequency-shaping filter that is linearly emphasized the signal according to its spectrum. As a result, this noise shaping technique serves as a filter that outputs the derivative of the spike waveforms for differentiating neurons' signatures and as a bandpass filter.

2) Informative Samples: For a given M spikes with each spike is represented by N samples, it is necessary to extract subset samples with the most information as features, informative samples, for reducing the complexity in clustering. To quantify the information carried by individual spike samples, a kernel density estimation can be used as a non-parametric way of estimating the probability density function of N^{th} sample of M spikes [8]. Given $x_1, x_2, \cdots, x_M$ are independent and identically distributed of a random variable, the kernel density used to approximate the probability density function is

$$f(x) = \frac{1}{h^d} \sum G\left(\frac{x - x_j}{h}\right), \tag{4}$$

where $G(\cdot)$ is an arbitrary isotropic kernel with a convex profile, h is a smoothing parameter called bandwidth and d is a dimension of the data space. Under smoothing, a small h, artifices modes in probability density function while over smoothing, a large h, obscures most of the structure of the data. One of the solutions is to estimate local bandwidths with a pilot kernel density estimation as

$$f_0(x_i) = \frac{1}{h_0^d} \sum_{j \neq i} G\left(\frac{x_i - x_j}{h_0}\right), \tag{5}$$

where h_0 is an initial specified global bandwidth. Based on the pilot density, local bandwidths are updated as

$$h_{x_i} = h_0 \left[\frac{\lambda}{f_0(x_i)}\right]^{0.5} \tag{6}$$

where λ is a constant, which is assigned to be a geometric mean of $f(x_i)|_{i=1,\ldots,N}$. By using the updated local bandwidths in (6), a density estimate is constructed as

$$f(x) = \frac{1}{h_{x_j}^d} \sum G\left(\frac{x - x_j}{h_{x_j}}\right). \tag{7}$$

The result from $f(x)$ are peaks and valleys those can be used for partitioning spikes into clusters.

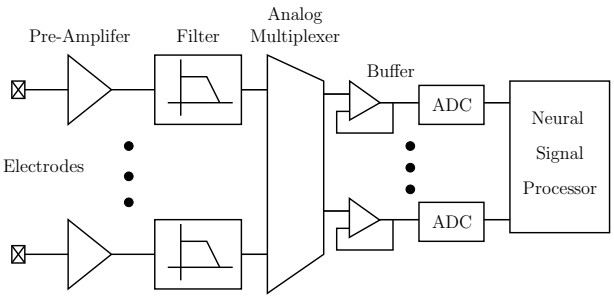

Fig. 1. An integrated multi-channel neural recording system with real-time neural signal processing.

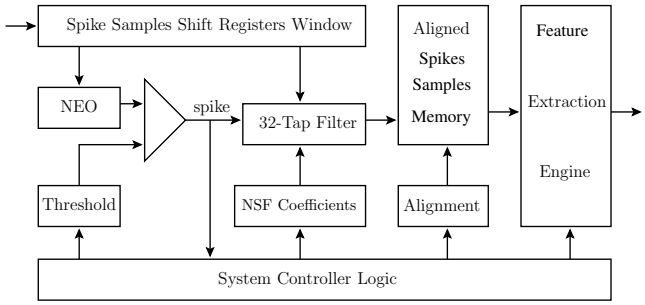

Fig. 2. An architecture of a neural signal processor with a NEO based spike detection and a feature extraction engine using informative samples.

C. Clustering

Based on the extracted features, clustering is used for classifying spikes into different groups. The most widely used clustering method for spike sorting is *k*-means due to its low computation. Nevertheless, we found it is unsatisfactory due to its sensitive to initial seed selection and outliers, produce erroneous results with irregularly shaped clusters, and it requires a number of *a-priori* clusters because it is a parametric algorithm. A solution to this deficiency is a non-parametric clustering algorithm we have developed based on mean shift algorithm. The novel clustering algorithm is an energy based evolving mean shift (EMS) algorithm with kernel scope obtained through nearest neighbor search [7]. The performance of EMS is superior to its related non-parametric clustering mean shift and blur mean shift, and the popular *k*-means. The hardware implementation for EMS is, not presented in this paper, closely resemble to finding the informative samples for feature extraction due to the required computation for a non-parametric density estimator.

III. System Architecture and Structure

A system architecture for a typical multi-channel neural recording is composed of analog front-ends for signals amplifying and conditioning, and a sophisticated neural signal processor for spike sorting as shown in Fig. 1.

A. Neural Recording Front-End Interface

The analog front-end of a typical multi-channel neural recording system is composed of pre-amplifiers, filters, analog multiplexers, buffers and analog-to-digital (ADC) converters. Power and chip area are often the most important parameters in a design. In such system, each ADC is shared by several recording channels via an analog multiplexer as shown in Fig. 1. As a result, the neural signal processor for spike sorting needs a large memory input buffer for rearranging channel-interleaving spike samples.

B. Spike Detection Architecture

Fig. 2 depicts an architecture of the spike detection implemented using NEO algorithm with a noise shaping filter and a peak alignment. The input is a serial stream of 9-bit recording samples from the ADC [1], however, only eight bits are used to reduce the area and computing power. The output from the spike detection process is an array of filtered spikes those are aligned according to their peaks. These aligned spikes are used in the feature extraction step for selecting a subset of samples, informative samples, for clustering.

Three major processing units for a spike detection step are: a NEO spike detector, a noise shaping filter, and a peak alignment. NEO spike detector calculates the energy from the input recording samples according to (1) and identifies the peak sample within a 5-sample window. A sample that indicates a neuron firing is when its energy exceeds the programmed threshold and it is the peak sample. Its time-index is saved for spike alignment; and it triggers the convolution process of a 32-tap filter. Both the 32-tap filter coefficients and threshold value are reprogrammable via the system controller. The spike alignment stores and aligns the middle 32 samples output of 32-tap filter to memory for a spike feature extraction engine.

C. Feature Extraction Engine

The feature extraction engine is structurally built to compute the information carried by the spike samples as shown in Fig. 3. It comprises two expensive operators such as divide and square, multiplexers, and storage elements. The overall area consumption due to combinational logic, however, is smaller than the required area for memory.

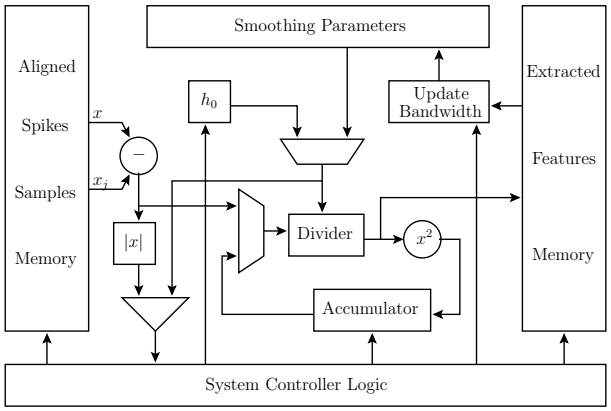

Fig. 3. An architecture of a feature extraction engine using informative samples.

The core of this engine is employed to compute the kernel density algorithm, which estimates the probability density function of the spikes' samples. To simplify the kernel density computation, an isotropic kernel $G(\cdot)$ from (4) is implemented as a square function. Note the training process for estimating smoothing parameters in (5) and the actual kernel density estimation in (7) is similar, the same hardware can be used with the help of a multiplexer. The top multiplexer is used for selecting between the initial h_0 and the estimated h_j local smoothing parameters. While, the other multiplexer shares a divider unit between calculating the $G(\cdot)$ term and the final $f(x)$. To save more area and power, update of the smooth parameters as described in (6) can be offload to a general purpose microprocessor since the smoothing parameters do not change often for a given recording. The system controller monitors the inputs x and x_j to ensure proper results can be included in the accumulator. The results from computing the probability density function are stored in a SRAM module, named as extracted features memory, that has the same amount of storage space as the input SRAM module for the aligned spikes samples memory.

IV. Results and Future Works

A neural signal processor including both the NEO-based spike detection and feature extraction using informative samples is implemented in a VLSI CMOS 90nm processor. Table I shows a breakdown of area and power consumptions for each of the modules. The spike detection together with a noise shaping filter use the most area and power. However, the actual contribution to this is a 32-tap filter, which occupies 78% of the area. The SRAM modules for storing 100 spikes, which each of the spike is composed of 32 samples occupy more than half of the total chip area. The storage for the smoothing parameters in this design requires only 100 8-bit words, however, the minimum allowable by a memory compiler is 256 words. Much of the latency is due to the feature extraction engine because the kernel density estimation has a complexity of $\Theta(n^2)$. This delay can be shorten by increasing the level of parallelism and clock rate in the feature extraction engine.

TABLE I

A 10MHz neural signal processor synthesized in 90nm technology at 1 V.

Components Listing	Area μm^2	Time *cycles*	Power μW
Spike Detection & NSF	75,712	3,500	460
Aligned Spikes Memory	50,581	—	15
Feature Extraction Engine	3,913	10,240,000	33
Smoothing Parameters	12,978	—	10
Extracted Features Memory	50,581	—	15
Total	193,765	10,243,500	533

The next step is to design an architecture for the EMS clustering algorithm and incorporate with a neural signal processor. A challenge will be finding the trade-offs between the accuracy, area, computing latency and power. In addition, the complexity of the EMS algorithm is $\Theta(dn\log n)$ where d is the dimension of the input feature score. As a result, a principle component analysis step is needed to reduce the dimensionality for a tractable hardware implementation.

References

[1] M. Chae, W. Liu, Z. Yang, T. Chen, J. Kim, M. Sivaprakasam, and M. Yuce, "A 128-channel 6mW wireless neural recording IC with on-the-fly spike sorting and UWB transmitter," in *Proc. Digest of Technical Papers. IEEE International Solid-State Circuits Conference ISSCC 2008*, 3–7 Feb. 2008, pp. 146–603.

[2] A. C. Hoogerwerf and K. D. Wise, "A three-dimensional microelectrode array for chronic neural recording," *IEEE Trans. Biomed. Eng.*, vol. 41, no. 12, pp. 1136–1146, Dec. 1994.

[3] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows, and J. P. Donoghue, "Instant neural control of a movement signal," *Nature*, vol. 416, pp. 141–142, 2002.

[4] University of Utah, "University of Utah to help build realistic bionic arm," *ScienceDaily*, April 2006.

[5] K. H. Kim and S. J. Kim, "Neural spike sorting under nearly 0-db signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 10, pp. 1406–1411, Oct. 2000.

[6] H. Teager and S. Teager, "Evidence of nonlinear sound production mechanisms in the vocal tract," *In Speech Production and Speech Modeling*, pp. 241–261.

[7] Z. Yang, Q. Zhao, and W. Liu, "Neural signal classification using a simplified feature set with energy based non-parametric clustering," *To be appear in Neurocomputing 2009*.

[8] Z. Yang, Q. Zhao, and W. Liu, "Spike feature extraction using informative samples," in *Advances in Neural Information Processing Systems 21*, 2009, pp. 1865–1872.