128-Channel Spike Sorting Processor with a Parallel-folding Structure in 90nm Process

Tung-Chien Chen [1,2], Wentai Liu[2], Liang-Gee Chen[1]
[1]National Taiwan University, Taipei, Taiwan; [2]University of California, Santa Cruz, CA, USA;

***Abstract*—An emerging class of neural prostheses aims to provide more aggressive performance by realizing advanced realtime signal processing algorithms in particular the spike sorting on chips. To support realtime spike sorting for 128 channels, the traditional fully parallel approach duplicating 128 processing units results in a large burden on chip area. The fully folding approach sharing one processor over 128 channels consumes large dynamic power in data caching. We propose to use the parallel-folding structure to optimally tradeoff the area and power. Our 128-channel spike sorting processor consumes 1.36 mm^2 area and 1.87 mW power in 90 nm process. 91.1% and 63.4% of the hardware resources (area×power) are reduced compared to the fully parallel and the fully folding approaches respectively.**

I. Introduction

Successful proof-of-concept laboratory experiments on cortically controlled motor prostheses [1] motivate continued development for neural prosthetic devices. Advances in implantable electrode arrays and miniaturized multi-channel recording ICs [2]–[4] drive the research field toward the direction of long-duration and wireless experiments on freely moving animals. In order to further improve neural prostheses, the bulk associated with the external computing systems has to be eliminated. The realtime signal processing tools in particular spike sorting should be implemented on chips [5].

Several state-of-the-art neural signal processing hardwares have been proposed. In [2], [3], comparator-based spike detection modules are implemented along with the analog frontend interface circuitry (AFIC) for the multi-channel neural recording systems. However, they transmit only the binary event streams of the detected spikes, and the resulting significant loss of information limits the ability of classification of individual neural sources. In [4], the feature extraction tool after the spike detection has been integrated in a 128-channel neural recording system. However, it can only support realtime processing for one of the 16 channels, which is still far from the expectation in terms of realtime processing capability for multiple channels.

In this paper, a spike sorting processor (SSP) performing spike detection and feature extraction in a real time for 128 channels are implemented in 90 nm process. The sorting performance, the chip area, and the power consumption are important issues for the on-chip SSP. On algorithm level, we adopt the cost effective algorithms that are robust over noise and have small penalty in computational complexity [6]–[8]. On hardware level, to support realtime signal processing for 128 channels, the traditional fully parallel approach duplicating 128 processing units results in a large burden on chip area. The fully folding approach sharing one processor over 128 channels consumes large dynamic power in data caching. We propose to use the parallel-folding approach to optimally tradeoff the area and power. The rest of this paper is organized as follows. The algorithm and the proposed architecture are described in Section II and Section III. The implementation results are shown in Section IV. Finally, Section V concludes this work.

II. Spike Sorting Algorithm

Most neurons in the brain communicate by firing action potentials, or spikes. These electrical voltage signals can be recorded extracellularly with very thin electrodes implanted into animals' or humans' brains. Very often an implanted electrode records the signals from more than one neurons. The recorded waveform is the superimposed neural potentials from these surrounded neurons. Spike sorting is a kind of reverse process to differentiate which firing spikes corresponds to which of these close-by neurons from the superimposed neural potentials.

Spike sorting generally consists of three main parts—the spike detection to allocate the spike events, the feature extraction to do the dimension reduction, and the clustering to classify the detected spikes into groups according to the extracted features. In this paper, we will focus more on the spike detection and feature extraction parts.

In the previous state-of-the-art designs [2], [3], the comparator-based spike detectors using an voltage threshold are implemented. This spike detection algorithm is not accurate and may result in many miss detections or false positives when the SNR is low. Therefore, another algorithm called nonlinear energy operator (NEO), first characterized by Kaiser [6], is suggested by [7], [9]. The NEO algorithm has a small penalty in hardware compared to the comparator-based spike detector. It can identify the spikes using the localized instantaneous energy and is robust over the low-frequency noise.

As for the feature extraction, the most intuitive method would be to take basic characteristics of spike shapes, such as their peak-to-peak amplitude and timing interval. This simple algorithm can be implemented in a very small application-specific integrated circuit (ASIC), but has poor accuracy in differentiating the spikes especially when the noise is large. Other algorithms based on principal component analysis (PCA) [10] and discrete wavelet transformation (DWT) [11] are investigated with better sorting accuracy. However the relatively high computation complexity is required [7]. On-chip parameter modeling to calculate eigenvectors for PCA and to do coefficient selection after DWT also requires significant

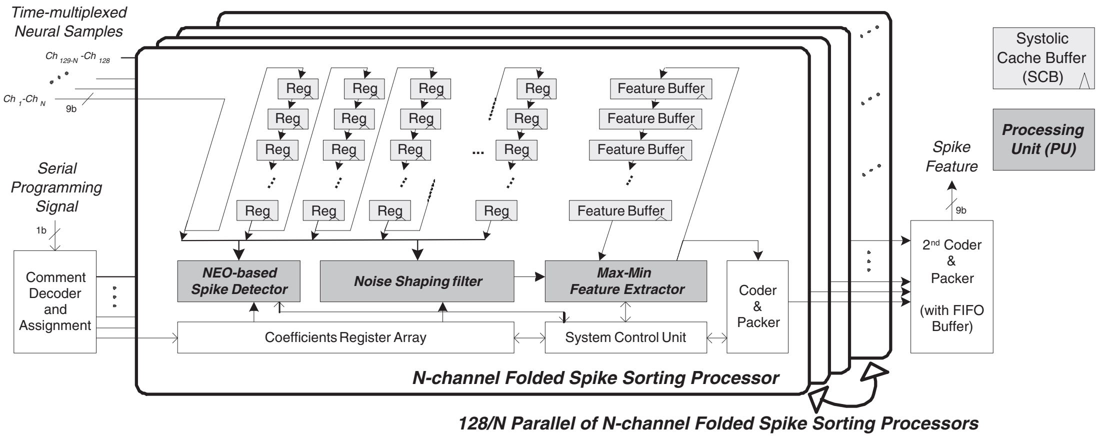
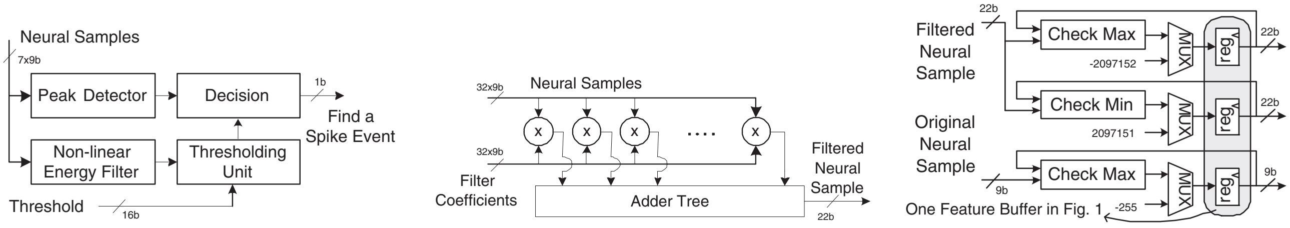


Fig. 1. The overall architecture of the proposed 128-channel SSP that employs both the parallel and folding approaches.



(a) NEO-based Spike Detector

(b) Noise Shaping FIR Filter

(c) MaxMin Feature Extractor

Fig. 2. Three main processing units for the NEO spike detection and DD-based feature extraction algorithms.

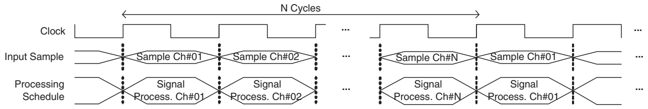


Fig. 3. The channel-interleaved processing schedule for each NFSSP.

hardware resources as well [12]. To have a good tradeoff between sorting accuracy and computational complexity, discrete derivatives (DD) is recommended by [7], [8]. Compared to DWT, the DD performs similar functionality emphasizing the high frequency neural characteristics with relatively low computational complexity [7]. The similar sorting accuracy compared with PCA is also reported in [8] for the DD method combining with a bandpass filter and a maximum-minimum (MaxMin) detector.

In summary, the NEO and DD algorithms are chosen because of their robustness over noise and the small penalty in hardware. For the spike detection, the NEO filter is first performed. Then, the spike events are recognized when the filtered samples are over a threshold at the peaks of the convex curves of the original neural waveform. For the feature extraction, the DD is first applied along with a bandpass filter. Then, the MaxMin detector calculates the feature scores by extracting the maximum and minimum values from the filtered spike waveforms and the maximum values from the original spike waveforms. For the more detailed descriptions and performance results of the NEO spike detection and the DD-based feature extraction algorithms, please refer to [6], [8].

III. Proposed Architecture

A. Overall Architecture

Low power consumption and miniaturized area are two primary design issues to implement an 128-channel spike sorting processor (SSP). The fully parallel approach results in a large area by using 128 single-channel SSPs in parallel. The folding approach [13] can significantly reduce the chip area by sharing one SSP over multiple channels. This area saving also contributes to the reduction on leakage power. However, additional dynamic power for data caching and scheduling is required by the folding approach. If all 128 channels are fully folded on a single SSP, prohibitively large dynamic power will be consumed, which violates the original design purpose.

Clearly, the tradeoff between the parallel and folding approaches should be utilized in order to balance the leakage and dynamic power as well as to minimize the chip area. Figure 1 shows the overall architecture of the proposed 128-channel SSP that employs both the parallel and folding approaches. The spike sorting operations of N channels are first folded on one processor. Then 128/N of such N-channel folded SSP (NFSSP) are used in parallel to support the 128-channel processing capability. The input of each NFSSP is 9-bit bit-parallel sample-serial channel-interleaved neural samples from N different channels. The outputs after the second coder&packer consist of 48 bits of three features, 16 bits of timing information, and 8 bits of channel index for each detected spike event. If the sampling rate of neural data is 40k samples/sec, N×40 kHz operation frequency is required by the system to perform realtime processing for N-channel neural data on each NFSSP.

B. Processing Units

There are three main processing units in each NFSSP—a NEO-based spike detector, a noise shaping filter, and a MaxMin feature extractor. The spike detector and noise shaping filter are operated on cyclic basis. After the system is reset, these two processing units continuously operate and output the results every cycle. The block diagram of the NEO-based spike detector is shown in Fig. 2 (a). It calculates the energy function for the input samples. Once the energy exceeds the threshold and a peak is detected in a convex curve of the spike waveform, a find-spike-event signal is activated. The noise shaping filter is a parallel FIR filter as shown in Fig. 2 (b). It performs the inner product between the 32 input neural samples and the filter coefficients in parallel. This filter serves two folded purposes. First, it is used as a band-pass filter to reject the low-frequency background noise and high-frequency thermal noise. Second, the filter outputs the first derivative of the neural signals to intensify the high-frequency part of neural characteristics. Note that the threshold value for spike detection and the coefficients for noise shaping filter are programmed into the coefficient register array during the hardware initial configuration. These values can be trained off-chip based on few minutes of initial recordings.

The feature extractor as shown in Fig. 2 (c) is operated on event basis. When a spike event is recognized by the spike detector, the feature buffer is reset to the initial values, and the feature extractor is activated. If the length of spikes is programmed to 32 samples, the feature extractor continuously monitors the raw and filtered neural samples in the next 32 cycles. The corresponding maximum and minimum values are stored and updated in the feature buffer. After the extraction, the final feature scores along with the event timing information are packed and output by the coder&packer engine.

C. Systolic Cache Buffer

We use systolic cache buffers (SCBs) [13] to fold multi-channel spike sorting tasks on a single set of processing units. Two SCBs are used for each NFSSP. The first SCB stores raw spike samples for the spike detection and filtering. The second SCB stores the intermediate maximum, minimum values and timing information during the event-triggered feature extraction operation. In each SCB, there are N rows of registers to store the data for N channels. The data in the Nth row are fed to the processing units to perform spike sorting for the corresponding channel. At each cycle, the data of N channels are shifted row by row in the SCB. In this manner, the data caching and fetching of N-channel neural samples is automatically arranged by SCB with the minimum storage space. Figure 3 summarizes the processing schedule. The neural samples are fed into the NFSSP channel by channel in an interleaved manner. With the SCB, the signal processing tasks of multiple channels are able to be performed in a channel-interleaved way on cyclic basis on a shared processor.

D. Optimization for the Parallel-folding Structure

Determining the optimal value of N in terms of area and power consumption is highly related to the fabrication process.

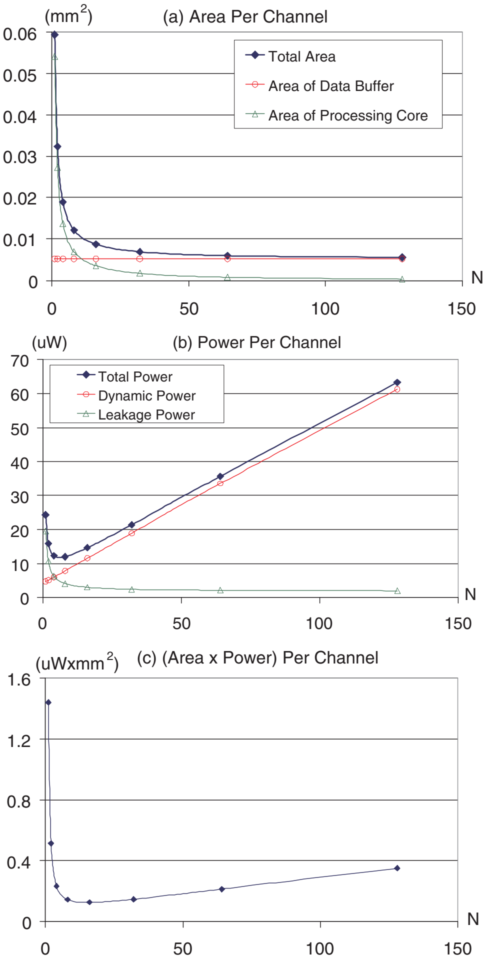


Fig. 4. The area and power tradeoff over different N-channel folded SSPs. (N = 1, 2, 4, 8, 16, 32, 64, and 128)

Focusing on the 90 nm technology, several NFSSPs with N ranging from one to 128 are implemented. We normalize the synthesized results of power and area by N to get the unit hardware requirement for each single channel. Figure 4 (a) shows the trend of area reduction with the increasing folding levels. The average memory space of SCB for each channel is about the same. The area reduction comes from the sharing of the processing units. Therefore, when N is increased, the overall area saving is less significant and will eventually saturate to the average SCB area.

Figure 4 (b) shows the tradeoff between the dynamic and leakage power with different folding levels. In general, the leakage power is roughly proportional to the chip area, while the dynamic power is proportional to the product of the operation frequency and the area. When N is one, the area per channel is largest, and the operation frequency is slowest. In this case, the large leakage power in deep sub-micro process dominates the overall power consumption. When the folding level becomes larger, the area per channel saturates, and a faster operation frequency is required. In this case, the leakage power is reduced. The dynamic power consumed by the processing units is about the same, but N times of dynamic power is consumed by the SCB for the data caching. The overall power consumption is finally dominated by the

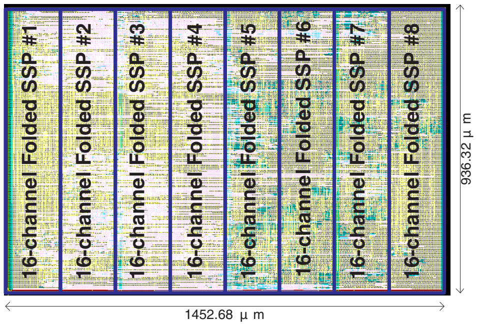


Fig. 5. The layout of the proposed 128-channel SSP in 90 nm CMOS process.

dynamic power. In 90 nm process, the hardware has the minimum power consumption for each channel when the folding level, N, is equal to eight.

We try to optimize the hardware by minimizing the product of chip area and power consumption. Figure 4 (c) shows the area-power product per channel for different NFSSPs. In the optimal case, the spike sorting of 16 channels are first folded on a single SSP. Then eight of such 16-channel folded SSP are used in parallel to support spike sorting for 128 channels in a real time. The system operation frequency is 640 kHz with 40k sps neural sampling rate. Note that this parallel-folding structure has the flexibility that 64-channel and 256-channel systems can also be optimally supported by parallelizing 4 and 16 of 16-channel folded SSPs respectively.

IV. Implementation Results

The 128-channel spike sorting processor with the parallel-folding structure is implemented in TSMC 90 nm 1P9M CMOS process. Figure 5 shows the chip layout, while Table I summarize the chip specification. The chip can detect the spike events with the NEO-based algorithm and extract the MaxMin features after the noise shaping filter. It can do realtime processing for 128 channels with 1.36mm^2 of chip area. If the input neural data are sampled at 40k sps, the chip consumes 1.87 mW of power with 640 kHz operation frequency and 1.08 volt supply voltage according to the gate-level simulation result.

Figure 6 shows the comparison of the fully parallel, the fully folding, and the parallel-folding approaches for the 128-channel SSP. In the fully parallel approach, 128 of single-channel SSPs are implemented in parallel with 40 kHz operation frequency. It consumes the largest chip area. The power consumption is dominated by the large leakage power. In the fully folding approach, one set of processing units is shared by 128 channels with 128 levels of SCB. The processor is operated at 5.12 MHz operation frequency. It consumes the smallest chip area, but has the largest dynamic power for data caching. The proposed parallel-folding approach with eight of 16-channel folded SSPs can optimally tradeoff the area and power consumption. 91.1% and 63.4% of the hardware resources (area×power) can be reduced compared to the fully parallel and the fully folding approaches respectively.

V. Conclusion

TABLE I
Chip Specification of 128-channel SSP in 90 nm CMOS Process

Process	90 nm 1P9M CMOS
Supply Voltage (Volt)	1.08
Core Area (mm^2)	1.36
Operation Frequency (kHz)	640
Power Consumption(mW)	1.87

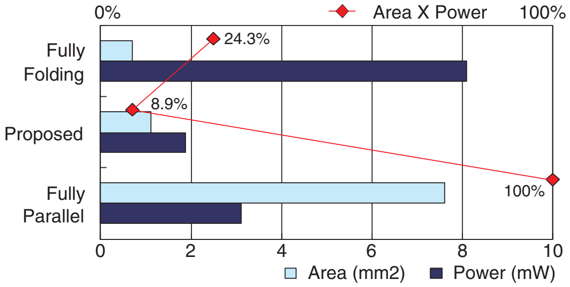


Fig. 6. The comparison of the fully parallel, the fully folding, and the proposed parallel-folding approaches for the 128-channel SSP.

In this paper, a 128-channel spike sorting processor is implemented to support NEO spike detection and DD-based feature extraction algorithms for 128 channels. According to the implementation result, the chip consumes 1.36 mm^2 area and 1.87 mW power in 90 nm process. The proposed parallel-folding approach with eight of 16-channel folded SSPs can optimally tradeoff the area and power. 91.1% and 63.4% of the hardware resources (area×power) are saved compared to the fully parallel and the fully folding approaches respectively.

References

[1] M. Velliste and et al., “Cortical control of a prosthetic arm for self-feeding,” *Nature*, vol. 453, pp. 1098–1101, 2008.

[2] R. H. Olsson and K. D. Wise, “A three-dimensional neural recording microsystem with implantable data compression circuitry,” *IEEE J. Solid State Circuits*, vol. 40, no. 12, pp. 2796–2804, 2005.

[3] R. R. Harrison and et al., “A low-power integrated circuit for a wireless 100-electrode neural recording system,” *IEEE J. Solid State Circuits*, vol. 42, no. 1, pp. 123–133, 2007.

[4] M. Chae and et al., “A 128-channel 6mw wireless neural recording ic with on-the-fly spike sorting and uwb transmitter,” in *ISSCC Dig. Tech. Papers*, Feb 2008, pp. 146–603.

[5] M.D. Linderman and et al., “Signal processing challenges for neural prostheses,” *IEEE Signal Process. Mag.*, vol. 25, no. 1, pp. 18–28, 2008.

[6] 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. on Biomed. Eng.*, vol. 47, no. 10, pp. 1406–1411, 2000.

[7] S. Gibson and et al., “Comparison of spike sorting algorithm for future hardware implementation,” in *Proc. Conf. IEEE EMBS*, Aug. 2008, pp. 5015–5020.

[8] Z. Yang and et al., “A neuron signature based spike feature extraction algorithm for on-chip implementation,” in *Proc. Conf. IEEE EMBS*, Aug. 2008, pp. 1716–1719.

[9] I. Obeid and P.D. Wolf, “Evaluation of spike-detection algorithms fora brain-machine interface application,” *IEEE trans. on Biomed. Eng.*, vol. 51, no. 60, pp. 905–911, 2004.

[10] M. Abeles and M. H. Goldstein, “Multispike train analysis,” in *Proc. of IEEE*, May. 1977, vol. 65, pp. 762–773.

[11] J. C. Letelier and P. P. Weber, “Spike sorting based on discrete wavelet transform coefficients,” *J. of Neurosci. Methods*, vol. 101, no. 2, pp. 93–106, 2000.

[12] T.-C. Chen and et al., “VLSI architecture of leading eigenvector generation for on-chip principal component analysis spike sorting system,” in *Proc. Conf. IEEE EMBS*, Aug. 2008, pp. 3192–3195.

[13] T.-C. Chen and et al., “NEUSORT2.0: a multiple-channel neural signal processor with systolic array buffer and channel-interleaving processing schedule,” in *Proc. Conf. IEEE EMBS*, Aug. 2008, pp. 5029–5032.