DESIGN AND IMPLEMENTATION OF A LOW POWER SPIKE DETECTION PROCESSOR FOR 128-CHANNEL SPIKE SORTING MICROSYSTEM

Tsung-Chuan Ma, Tung-Chien Chen, Liang-Gee Chen *

DSP/IC Design Lab, Graduate Institute of Electrical Engineering,
National Taiwan University, Taipei, Taiwan
Email:{tcm, djchen, lgchen}@video.ee.ntu.edu.tw

ABSTRACT

It is impractical to apply a general spike sorting algorithm for every subject because of the individual characteristics of brain signal. Furthermore, extracting more neural activities for higher accuracy of spike sorting requires more input electrodes as well as large power consumption and chip area. Therefore, several practical constraints are considered in this work when implementing a programmable spike sorting hardware with large number of input channels. In this paper, we provide a 128-channel spike detection processor for spike sorting microsystem without compromise of the power efficiency. This chip consumes only 87.02uW and 9.7uW/mm^2 of power density, fabricated with 90nm low-leakage CMOS process.

Index Terms— Spike Detection, Spike Sorting, Neural Signal Processing

1. INTRODUCTION

Spike sorting plays an important role on both neuroprothetics and neuroscientific researches. Many spike detection algorithms, such as simple threshold, non-linear energy operation(NEO), and root-mean-square power (RMSP), are used in different researches [1–3]. However, due to the individual characteristics among each subject's brain signal, it is impractical to presume which algorithm is more appropriate for distinguishing spike classes in subject's neurons. Another critical issue is that even the most optimal spike detection algorithm is given, the accuracy of spike sorting methods decreases because the brain is a time-variant system [4]. According to this characteristics, it is difficult to predict the most appropriate strategy for the spike sorting task. Therefore, a programmable and flexible spike sorting system is necessary for neural decoding application.

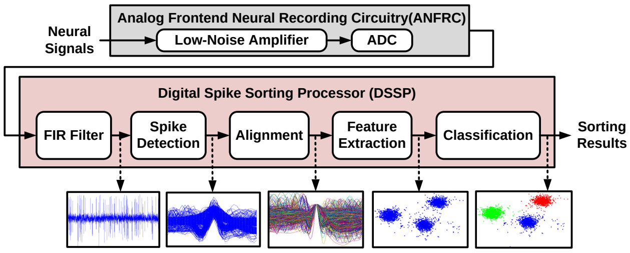

Fig. 1. The hardware operation of neural recording and spike sorting

In addition to the programmability for overcoming subjects' difference, an integrated and implantable system for experimenting on free-moving subjects is one of the mainstream research trend [2, 5]. Several design challenges for these implantable and programmable systems must be conquered. A study in [6] showed that the power dissipation density of an implantable neural system should be under 800μW/mm^2 to avoid damaging the tissue owing to overheating. On the other hand, in order to acquire more spike information from more neurons, a neural processing system in [5], which targeted on more than 100 channel with high input data rate, dramatically increased power density over 800μW/mm^2 and overall power consumption. These power constraints present difficult design challenges to a programmable spike sorting system. Therefore, new algorithm and architecture are needed.

In this article, we propose a new spike detection algorithm and its VLSI architecture design to overcome aforementioned design challenges. In Section 2, we analyze the complexity of spike sorting system and several spike detection algorithms used in state-of-the-art laboratory apparatus. The proposed spike detection algorithm and its VLSI design are discussed in Section 3 and 4, respectively. Section 5 shows the implementation result, and Section 6 concludes this work.

2. SPIKE SORTING MICROSYSTEM

2.1. Illustration of the Spike Sorting

Fig.1 shows the microsystem comprising neural recording circuit and digital spike sorting processor (DSSP). For a spike

*The authors thank UMC University Program for IC fabrication, Chip Implementation Center of Taiwan for chip testing, and National Science Council of Taiwan for grant support (NSC100-2221-E-002-248-MY3, NSC100-2811-E-002-049)

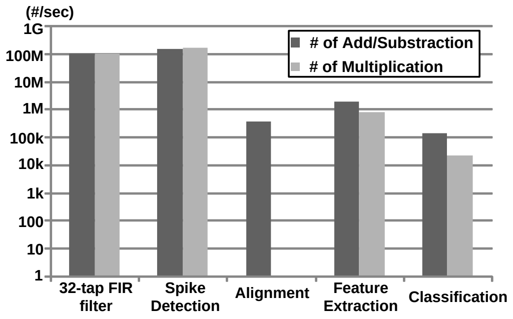

Fig. 2. Complexity Estimation of Spike Sorting Algorithm. The estimated algorithms at each stage are 32-tap FIR Filter, RMSP spike detection, DWT-PCA feature extraction, and k-means classification.

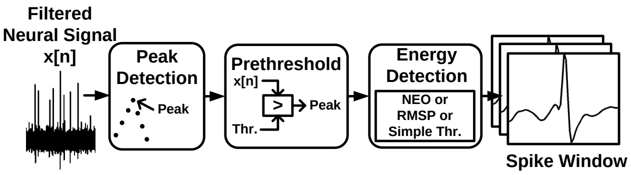

Fig. 3. Proposed spike detection flow on single channel spike sorting system.

sorting microsystem, raw neural activities are recorded, amplified, and digitalized in analog front-end circuit, AFNRC, and transmitted to the input of DSSP for further signal processing operation. In DSSP, the spike signal is filtered by a band-pass Finite Impulse Response (FIR) filter to remove low frequency component, local field potential, and environmental noise. Neural spikes are first detected from digital neural signals according to their localized instantaneous energy. Thus the distinct factors, namely the features, of the detected neural spike are extracted after the waveform alignment. Spikes which have the analogous features are presumed to be fired from one particular neuron. Hence, the classifier separates spikes into different clusters according to its waveform characteristics in the feature space.

2.2. Complexity Estimation of Spike Sorting Algorithm

Two of the mentioned design challenges are reducing power consumption and power density of an implantable hardware within a prescribed limit. The computational complexity, given in Fig.2, shows that the most dominant operation is the spike detection stage. The result of figure 2 is derived by the following settings of a 128-channel spike sorting system, 25k-sample/sec of each electrode, 30 spike/sec of each neuron, and 32 data sample of each spike. If each electrode is surrounding by three neurons, the numbers of addition and multiplication per second in spike detection stage are more than 100M with RMSP spike detection algorithm in [3] as the analyzing target. In addition, the computational complexity of spike detection stage will be dramatically increased in the future if a higher sampling frequency is needed. In consequence, a new spike detection algorithm is required to alleviate the burden of the complexity due to the higher sampling frequency.

3. PROPOSED SPIKE DETECTION ALGORITHM

3.1. Review of Spike Detection Algorithms

Several neuroscientific research groups have published spike detection algorithms [1–3]. These algorithms have three basic parts, including energy detection, threshold, and peak detection. As shown in Eq.(1a), the energy detection of RMSP requires considerable computation, which is directly proportional to sampling frequency and the number of input channels, to detect large energy $Y[n]$ in spike windows. If the energy of spike window is large enough (i.e., $T[n]$ is 1 at threshold part), a spike is observed as soon as a peak is detected (i.e., $P[n]$ at peak detection part).

$$\text{Energy Detection: } Y[n] = \sqrt{\frac{1}{32}\sum_{i=0}^{31} x^2[n+i]} \tag{1a}$$

$$\text{Threshold: } T[n] = 1, \text{if } Y[n] > threshold \tag{1b}$$

$$\text{Peak Detection: } P[n] = 1, \text{if } x[n] > x[n+k], -3 < k < 3 \tag{1c}$$

The colossal computation in previous spike detection algorithms results from performing energy detection on each neural sample. Nonetheless, evaluation on each sample is not efficient since the spike firing rate of each neuron is very low, normally less than one hundred spikes per second, compared with sampling rate of the system. Hence, a modification of the spike detection algorithm is necessary to improve system efficiency in order to reduce overall power consumption and power density for the implantable applications.

3.2. Proposed Spike Detection Algorithm

Fig.3 reveals the flow of the proposed spike detection algorithm. Due to the sparsity of neural activities, the proposed algorithm lowers the computational complexity of spike detection algorithm by reversing the arrangement of computing stages. First, peak detection stage observes a local peak of neural waveform by comparing the filtered data $x[n]$ and its neighboring samples. If $x[n]$ is a local maximum (i.e., $x[n] > x[n-k], k = 1, 2, 3, -1, -2, -3$), the location of $x[n]$ is labeled as a peak. Once a local peak has been identified, prethreshold stage examines whether the amplitude of detected peak is large enough to be a peak of a spike. Following the prethreshold stage, which screens a possible neu-

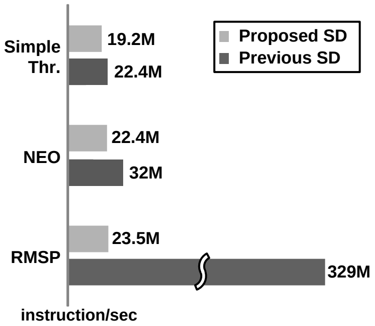

Fig. 4. Instructions per second of proposed spike detection algorithm and previous ones. Peak detection and Prethreshold can effectively reduce the computational complexity of spike detection module. This result only considers the computation of ALU, such as addition, substraction, and multiplication.

ral spike, energy detection, such as NEO and RMSP, checks whether the neural data from previous stages is a spike signal.

Previous spike detection algorithms [1–3] result in high computational complexity because they compute the energy of every neural sample. Since peak detection and prethreshold already labels possible location of spikes, energy detection only checks these possible locations instead of searching all the neural samples. Hence, this mechanism can lessen the overall computation of spike detection algorithm. Fig.4 shows that the proposed algorithm reduces the computation of spike detection in DSSP. Directly implementing previous RMSP algorithm as in Eq.1, the spike detection stage consumes more than 300M instructions per second. After applying the proposed algorithm, peak detection and prethreshold take advantage of the sparsity of neural signal and reduce the overall computation from 300M to 23.5M. There are also 14.28% and 30% computational reductions when implementing simple threshold and NEO algorithm, respectively. Moreover, after testing neural data "C_Easy1" in [7], the worst accuracy drop is less than 0.1% compared to previous spike detection algorithm.

4. PROPOSED ARCHITECTURE

4.1. Overall Architecture Design

For neurascientific and neuraprothetics researches, recording more electrodes provides more neural information and higher accuracy of spike sorting. Nonetheless, directly implementing 128-channel biosignal processor with fully parallel scheme results in larger silicon area and higher power consumption when developing VLSI architecture of DSSP. According to previous study in [5], the parallel-folding technique optimizes the area and power consumption of 128-channel spike detection processor with 8-channel folding processing elements (PE) in UMC 90nm low-leakage CMOS process as shown in Fig.5. Sixteen 8-channel folding PEs assemble a 128-channel spike detection processor for DSSP.

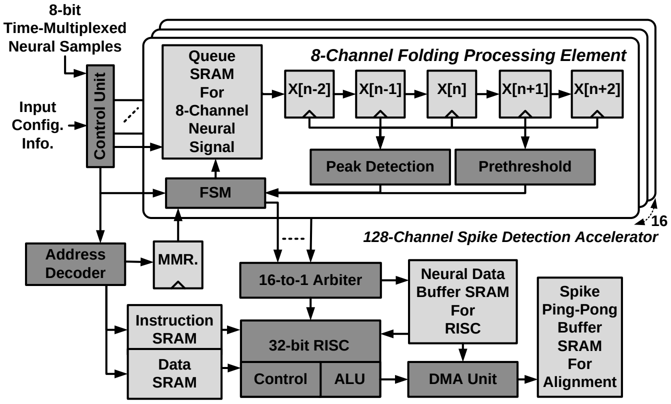
8-bit Time-Multiplexed Neural Samples
Input Config. Info.
Control Unit
Queue SRAM For 8-Channel Neural Signal
8-Channel Folding Processing Element
X[n-2]
X[n-1]
X[n]
X[n+1]
X[n+2]
Peak Detection
Prethreshold
FSM
16
128-Channel Spike Detection Accelerator
Address Decoder
MMR.
16-to-1 Arbiter
Neural Data Buffer SRAM For RISC
Spike Ping-Pong Buffer SRAM For Alignment
Instruction SRAM
Data SRAM
32-bit RISC
Control
ALU
DMA Unit

Fig. 5. The overall architecture of the proposed 128-channel spike detection processor.

In Fig.5, the control unit receives 8-bit time-multiplexed neural data, sampling at 12.5k/sec, and writes these samples with a channel-interleave scheme as in [5] into each PE. A 32-bit RISC with 4kB instruction memory and 4kB data memory provides programmability and flexibility for accommodating the individual characteristic. Programmer can implement different spike detection algorithms and convert it to machine code in instruction SRAM. Users can also set parameters for each channel in data SRAM through the control unit. The memory-mapped register (MMR) keeps presetting parameters of each channel for prethreshold module. Once possible neural spikes are detected, each PE requests the 16-to-1 arbiter to write these data into neural data buffer SRAM for energy detection. After accepting neural data from PEs, the 16-to-1 arbiter triggers the 32-bit RISC to perform the energy detection algorithm to check whether these neural data is a real spike. If the RISC detects spikes, it issues command to the direct memory access unit (DMA) to automatically transfer spike data from the neural data buffer SRAM to the spike ping-pong buffer SRAM for alignment process. After finishing data transfer, DMA sends acknowledgement back to RISC to indicate a successful output of spike.

4.2. Processing Elements

Processing Element comprises a queueing SRAM for neural signal, the register array, and the computation units. The filtered neural samples are written from the control unit with a channel-interleave scheme into queueing SRAM. A finite state machine (FSM) loads neural data from queueing SRAM to the register array for the peak detection and the prethreshold modules. The peak detection and the prethreshold check whether each local maximum data is large enough to be a peak of a spike. If a local peak with high amplitude is found, the FSM transmits these neural samples to the 16-to-1 arbiter.

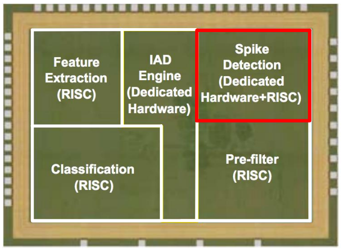
Feature Extraction (RISC)
IAD Engine (Dedicated Hardware)
Spike Detection (Dedicated Hardware+RISC)
Classification (RISC)
Pre-filter (RISC)

Fig. 6. Chip micrograph of the spike sorting processor with proposed spike detection hardware. The spike detection hardware is implemented as dedicated hardware, including peak detection and prethreshold, and 32-bit RISC.

Table 1. Synthesized Result in 90nm CMOS Process

Process	UMC 90nm 1P9M Low-leakage CMOS
Supply Voltage	1.0 Volt
Operation Frequency	20MHz in Maximum
Core Area	5.92mm^2
Chip Area	8.89mm^2
Power Consumption	87.02uW (0.68uW/channel)*
Power Density	9.7uW/mm^2

*The algorithm running on the microprocessor is NEO spike detection, DWT-PCA feature extraction, and k-means classifier.

Storing 128-channel neural data in the register array causes large dynamic and leakage power because all registers need to be active to pass data to neighboring register. Since all registers are active to transmit neural data, this mechanism in [5] violates the constraint of power density and causes heat damage to subject's tissue. Therefore, in the proposed PE, neural data are written into queueing SRAM one by one to reduce overall power consumption and power density.

5. IMPLEMENTATION RESULT

As shown in Fig.6, the die micrograph of the chip, the proposed spike detection processor is one important module of 128-channel DSSP. Table 1 summarizes the implementation result of the silicon chip. The RISC-based spike detection hardware supports the programability for various spike sorting algorithm in order to accommodate the individual characteristics. Each spike detection algorithm is coded in assembling language, compiled to the machine code, and programmed into the spike sorting processor. For each algorithms, the required parameters of prethreshold are pretrained off-line in the PC. The spike detection hardware requires 1.19mm^2 area including 33.5k logic gates and 137.6kb SRAM. The SRAMs are used to queue filtered neural data, buffer neural data for RISC, store instruction and data for RISC, and transmit detected spikes for alignment module. The spike detection hardware can handle at most 125k spikes per second with 20MHz operation frequency. This specification is able to perform on-line processing for 4k neurons with spike firing rate of 30 spikes/neuron.

Table 2. Comparison with Previous Work

Reference	[5]	[2]	This Work
Programmability	No	No	Yes
No. of Channels	128	64	128
Power (uW/channel)	14.6	2.03	0.68
Area (mm^2/channel)	0.01	0.06	0.06
Power Density (uW/mm^2)	1460	30	9.7
Process (nm)	90	90	90
Core Voltage (V)	1.08	0.55	1.0

Table 2 compares this work with previous ones. Our chip provides programmability for different spike detection algorithms. The power per channel is 87.02uW and 30% of previous works [2] if performing same spike detection algorithm. The power density is 9.7uW/mm^2, only 32% of [2].

6. CONCLUSION

In this paper, a spike detection processor is proposed with flexible programmability for 128 channel spike sorting microsystem. The proposed algorithm is implemented and fabricated in 90nm low-leakage CMOS process. The implementation result shows that this spike detection processor operates on-line processing up to 4k neurons without compromise of the power efficiency.

7. REFERENCES

[1] J.P. Donoghue, "Bridging the brain to the world: a perspective on neural interface systems," *Neuron*, vol. 60, no. 3, pp. 511–521, 2008.

[2] Vaibhav Karkare, Sarah Gibson, and Dejan Markovic, "A 130-μw, 64-channel neural spike-sorting dsp chip," *IEEE journal of solid-state circuits*, vol. 46, no. 5, pp. 1214–1222, 2011.

[3] K.D. Harris and et al., "Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements," *Journal of Neurophysiology*, vol. 84, no. 1, pp. 401–414, 2000.

[4] S. Gluth and et al., "Deciding when to decide: time-variant sequential sampling models explain the emergence of value-based decisions in the human brain," *The Journal of Neuroscience*, vol. 32, no. 31, pp. 10686–10698, 2012.

[5] T.C. Chen and et al., "128-channel spike sorting processor with a parallel-folding structure in 90nm process," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, 2009, pp. 1253–1256.

[6] K.M Silay and et al., "Numerical analysis of temperature elevation in the head due to power dissipation in a cortical implant," in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, 2008, pp. 951–956.

[7] R.Q Quiroga and et al., "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural computation*, vol. 16, no. 8, pp. 1661–1687, 2004.