A Coarse-Grained Reconfigurable Approach for Low-Power Spike Sorting Architectures

Nicola Carta*, Carlo Sau, Danilo Pani, Francesca Palumbo, Luigi Raffo

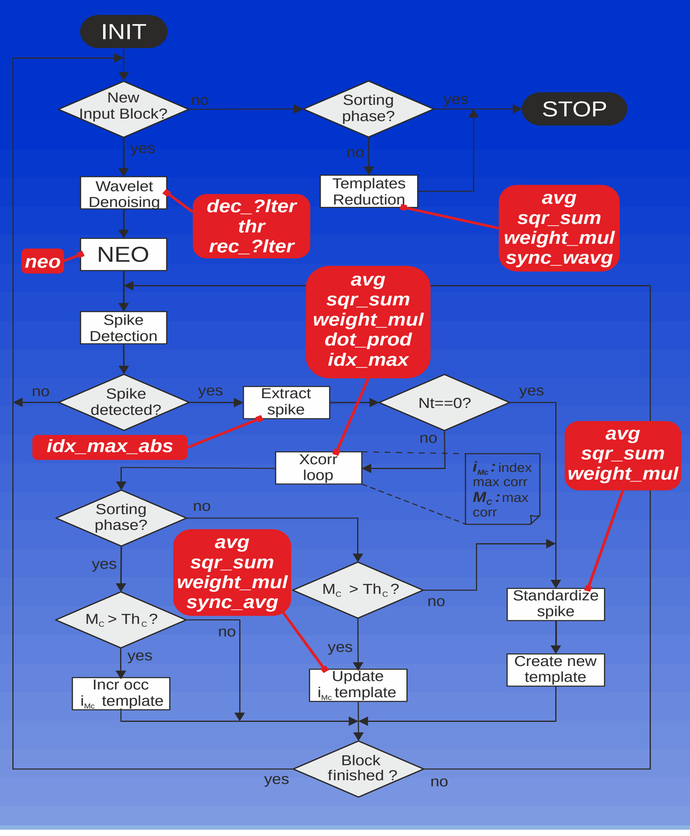
DIEE – Dept. of Electrical and Electronic Engineering – University of Cagliari - Italy

{nicola.carta, carlo.sau, danilo.pani, francesca.palumbo, luigi}@diee.unica.it

Abstract. Spike sorting is a critical task in neural signal decoding because of its computational complexity. From this perspective, the research trend in the last years aimed at designing massively parallel hardware accelerators. However, for implantable system with a reduced number of channels, as could be those interfaced to the Peripheral Nervous Systems (PNS) for neural prostheses, the efficiency in terms of area and power is in contrast with such a parallelism exploitation. In this work, a novel approach based on high-level dataflow description and automatic hardware generation is presented and evaluated on an on-line spike sorting algorithm for PNS signals. Results in the best case revealed a 71% of area saving compared to more traditional solutions, without any accuracy penalty. With respect to single kernels execution, better latency performance are achievable still minimizing the number of adopted resources.

Overview

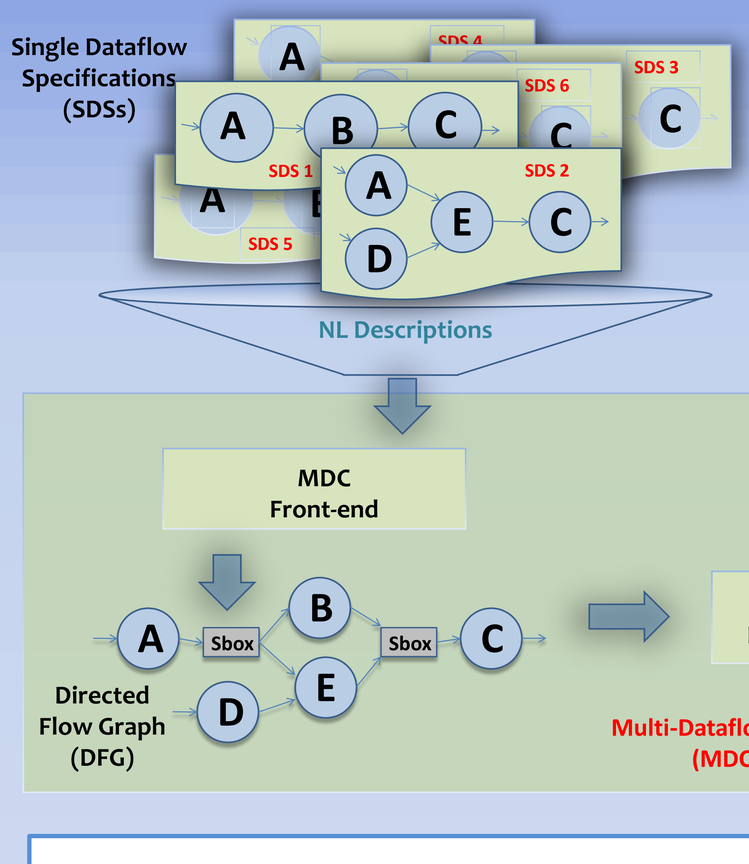
- Spike sorting is a well-known technique for the identification of single neurons activity from either multi-site or single-channel recordings with limited selectivity.
- When the number of channels is supposed to be low, such as in PNS interfaces, massive parallelization to satisfy real-time constraints is in contrast with the low-power requirements.
- In this work, we propose the adoption of a novel approach in the specification and implementation of the algorithm.
- An automatic tool, called Multi-Dataflow Composer (MDC), has been adapted to create a coarse-grained reconfigurable architecture allowing the reuse of different computational Functional Units (FUs) for different parts of the algorithm.
- In this way, the final implementation contains the minimum number of required FUs, therefore leading to area and power saving, which are characteristics of paramount importance for implantable devices.



The algorithm

- A modified version of the spike sorting algorithm presented in [1], that showed to be real-time implementable on a DSP [2], has been adopted, exploiting the smaller Haar and introducing the Non-Linear Energy Operator (NEO) to improve the spike detection performance .
- The algorithm is composed of the three typical parts of any spike sorting algorithm (**Spike Detection**, **Alignment**, **Sorting**) with a preliminary stage of **Wavelet Denoising**, required to work on the real data with low SNR.
- It works on-line both when the spikes of the single neurons are being identified (*training*) and when they have been identified (*sorting*), exploiting an unsupervised template matching strategy.
- The unsupervised approach relies on a matrix of templates progressively extracted thanks to the NEO, centered on their maximum value and updated through synchronized averaging when the maximum correlation M_c with an incoming spike is above a threshold Th_c.
- After the templates have been created, their number is reduced merging those with high similarity and keeping only the N most representative ones.
- During the sorting phase, only template matching is performed without any update of the templates.

Run-time Reconfigurable Hardware Synthesis and FPGA Implementation



HDL Components library, i.e. the implementation of the FUs:

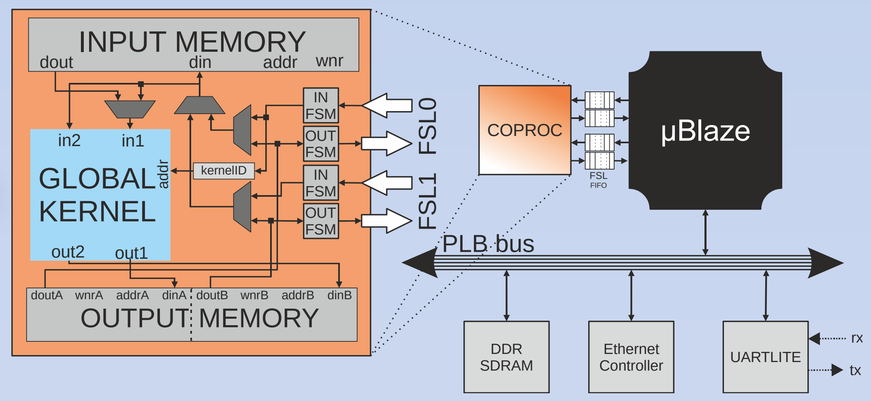
- IEEE 754-1985 compliant arithmetic modules;
- control flow modules.

All the FUs obey to the same FIFO-based communication protocol.

Global Kernel

a Coarse-Grained Reconfigurable Hardware Platform with the minimum number of FUs

- The algorithm is segmented in computational *kernels*, described in terms of their dataflows where several **FUs** (possibly shared, whose implementation is provided in the HDL components library) are interconnected. The FUs network is described with the formalism of the **Network Language (NL)**, a dialect derived from XML. A single reconfigurable hardware platform is provided as output by the adopted tool.
- Runtime reconfiguration is allowed by the use of low-overhead switching boxes which activate a desired dataflow path according to the specified kernel id.
- The overall datapath includes 3 adders, 3 multipliers, 1 substractor, 1 comparator, and 1 absolute value calculator to perform the various kernels.
- The MDC-based Global Kernel has been integrated in a coprocessor. A Xilinx Spartan-3E 1600 Development Board has been used for testing.



Experimental Results

Algorithm Performance

A publicly available dataset of synthetic mixtures of real neural signals [3] has been used as inputs for the spike sorter. The *Easy1* signal has been selected because the action potentials from different neurons present a correlation value between them lower than 0.9.

The signal includes the activity of 3 neurons and, in 60s, each neuron fires approximately 1100 times.

Noise	Template 1	Template 2	Template 3	Template 4	Template 5	Template 6	Template 7	Template 8	Template 9	Template 10
0.05	1080	1016	992	25	22	11	10	9	6	5
0.10	1120	1002	998	130	46	17	9	8	7	7
0.15	1079	1039	979	246	11	9	6	5	1	1
0.20	1010	788	723	469	12	6	4	4	3	3
0.25	1065	870	868	60	12	3	3	0	0	0
0.30	1053	930	582	67	0	0	0	0	0	0
0.35	1124	854	593	83	0	0	0	0	0	0
0.40	742	561	262	45	0	0	0	0	0	0

Hardware Synthesis Results

SGK: Static Global Kernel (the architecture resulting from the implementation of all the kernels without resource sharing).

MGK: Merged Global Kernel (the unique dataflow implementation by the MDC tool).

Whole **Coprocessor** including the MGK wrapped by an external layer that allows its easy exploitation.

FPGA (Xilinx FPGA Spartan-3E 1600)

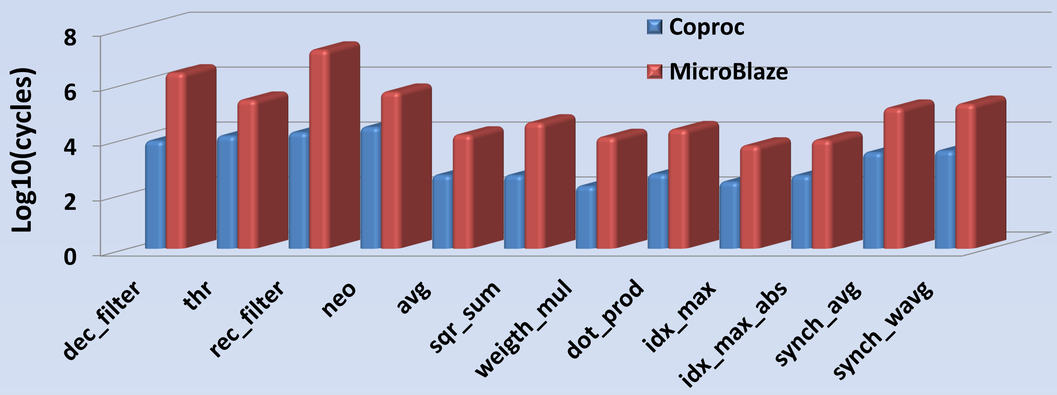
IMPL	Slices [%]	FFs [%][a]	MULs [%][b]
SGK	81	23	100
MGK	26	7	33
Coprocessor	32	9	36

[a] Flip Flop Slices [b] MULT18X18SIO dedicated multipliers

ASIC (90nm low-power CMOS library)

IMPL	Area [mm^2]	Power [mW]	freq [MHz]
SGK	0.463	61.56	357.71
MGK	0.134	14.69	312.5
Coprocessor	0.159	15.16	208.33

Latency Analysis



Conclusions

- Spike sorting is a critical task in neural signal decoding because of its computational cost. In most cases, a straightforward HDL implementation would lead to very inefficient solutions in terms of area and power.
- The MDC tool allows developing coarse-grained reconfigurable architectures, minimizing the hardware resources needed to accomplish the required tasks.
- Synthesis results confirm the effectiveness of the proposed approach for obtaining a very small and low power architecture for spike sorting.
- The performance in terms of latency can also be improved by tuning of the implemented parallelism in the light of a defined number of channels and real-time constraints, which is a promising result for implantable devices.

The research leading to these results has received funding from the Region of Sardinia, Fundamental Research Programme, L.R. 7/2007 "Promotion of the scientific research and technological innovation in Sardinia" under grant agreement CRP-18324 RPCT Project (Y. 2010) and CRP-60544 ELoRA Project (Y. 2012). Carlo Sau gratefully acknowledges Sardinia Regional Government for the financial support of his PhD scholarship (P.O.R. Sardegna F.S.E. Operational Programme of the Autonomous Region of Sardinia, European Social Fund 2007-2013 - Axis IV Human Resources,Objective l.3, Line of Activity l.3.1).

[1] L. Citi et al., "On the use of wavelet denoising and spike sorting techniques to process electroneurographic signals recorded using intraneural electrodes," *Journal of Neuroscience Methods, vol.* 172, pp. 294–302, 2008.

[2] D. Pani et al., "Real-time processing of tfLIFE neural signals on embedded dsp platforms: A case study," in *Neural Engineering (NER), 2011 5th International IEEE/EMBS Conference on, 2011, pp. 44–47.*

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