Neuromorphic Architectures for Spiking Deep Neural Networks

Giacomo Indiveri, Federico Corradi, and Ning Qiao
Institute of Neuroinformatics, University of Zurich and ETH Zurich, Zurich, Switzerland
e-mail: giacomo@ini.uzh.ch

Abstract

We present a full custom hardware implementation of a deep neural network, built using multiple neuromorphic VLSI devices that integrate analog neuron and synapse circuits together with digital asynchronous logic circuits. The deep network comprises an event-based convolutional stage for feature extraction connected to a spike-based learning stage for feature classification. We describe the properties of the chips used to implement the network and present preliminary experimental results that validate the approach proposed.

Introduction

Deep Neural Networks (DNNs) are typically composed of many layers of neurons coupled among each other via weighted connections. They have recently shown state-of-art performance in multiple benchmarks tasks such as computer vision, machine translation, and speech recognition [1]. However, most of these applications are executed on conventional computing systems, which are not ideally suited to implementing such massively parallel architectures.

Neuromorphic computing platforms represent a new generation of non von Neumann massively parallel architectures that are ideally suited to implementing DNNs: neuromorphic systems are built with electronic circuits using design principles that are based on those of biological nervous systems [2, 3]. They typically comprise mixed-mode analog/digital Very Large Scale Integration (VLSI) circuits, and are ideally suited for the exploitation of emerging memory technologies and memristors [4, 5, 6]. They process information using energy-efficient asynchronous, event-driven methods which are characterized by the co-localization of memory and computation [7].

Here we demonstrate a multi-chip neuromorphic system that implements spiking DNNs efficiently, using low-latency, low-power, and compact circuits. The neuromorphic chips in the system comprise both analog and digital circuits. The analog circuits implement compact and faithful models of biological synapses and spiking neurons, and carry out the neural computation operations of the network. The digital circuits include both asynchronous designs, used to transmit the spike-events among neurons and across devices, and standard digital logic elements, used to configure the on-chip network topology, and to program the routing schemes for implementing a wide variety of different DNN architectures. Although the analog circuits are affected by device mismatch, the overall architecture carries out robust computation, because it uses multiple (mismatched) silicon neurons in the learning process as weak classifiers to implement a well known machine learning ensemble method, which has been shown to be provide substantial gains in accuracy and recognition rates [8].

We show a convolutional neural network application example, performing in real-time, and implemented using exclusively full-custom neuromorphic devices, from the visual input stages to the output classifier one, without requiring any additional computing or memory resources. The system we present is compact, low-power, and has *in-situ* real-time on-line learning capabilities.

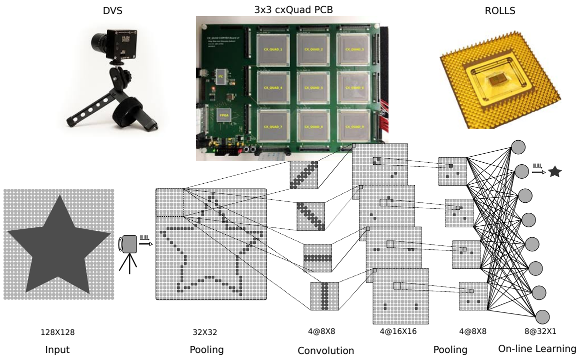

Fig. 1.: Experimental setup: a Dynamic Vision Sensor (DVS) converts moving visual stimuli displayed on a screen into streams of Address-Events (AEs). The retina AEs are subsampled and re-mapped onto the first chip of the cxQuad Printed Circuit Board (PCB) via a direct cable connection. The spiking output of the first chip is decomposed into multiple streams to implement a 2-layer convolutional network. The spiking output of this network is then mapped onto the classification layer implemented on the ROLLS neuromorphic processor, which is configured to implement ensembles of perceptrons for classifying the extracted features.

The neuromorphic DNN setup

The experimental setup used to demonstrate the spiking convolutional network is depicted in Fig. 1. It comprises a Dynamic Vision Sensor (DVS), a custom Printed Circuit Board (PCB) with 9 “cxQuad” multi-neuron chips, and the Reconfigurable On-Line Learning Spiking (ROLLS) neuromorphic processor [9] configured to implement a classification layer.

The DVS retina responds only to changes in the scene contrast and provides in output continuous streams of Address-Events which report the location of the pixels that sensed the changes. The DVS has been fully characterized in [10]. The cxQuad chip is a novel multi-core device that we recently designed, which comprises analog neuron and synapse circuits, as well as an asynchronous digital routing fabric which optimized for minimizing memory requirements, while maximizing scalability and re-configurability features [11, 12]. The cxQuad asynchronous digital circuits employ the Address-Event Representation (AER) to route spikes among neurons both within a core, across cores, and across chip boundaries; the neuromorphic analog neuron and synapse circuits implement biophysically realistic temporal dynamics using log-domain Differential Pair Integrator (DPI) filters and adaptive-exponential Integrate-and-Fire (I&F) neuron circuits [9]. Each cxQuad chip has 1k neurons and 64k synapses distributed among four cores. Specifically, each core has 16×16 units, which contain one neuron and one synapse block. The synapse block comprises a linear integrator circuit which integrates input events from 64 12-bit programmable Content Addressable Memory (CAM) cells. Output events generated by the neurons can be routed to the same core, via a Level-1 router, to other cores on the same chip, via a Level-2 router, or to cores on different chips, via a Level-3 router. The

978-1-4673-9894-7/15/$31.00 ©2015 IEEE

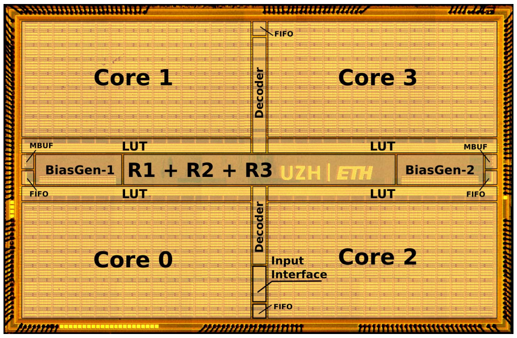

Fig. 2.: cxQuad chip, fabricated using a 180 nm 1P6M CMOS process. It occupies an area of 43.79 mm^2 and comprises 1k neurons and 64k×12-bit CAM programmable synapses subdivided among 4 cores. In addition it integrates 4k×12-bit SRAMs, 3-level hierarchical routers, two temperature compensated bias generator circuits, and one input pre-decoder block.

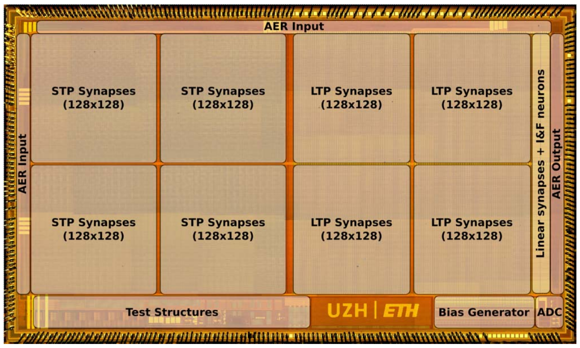

Fig. 3.: ROLLS chip, fabricated using a 180 nm 1P6M CMOS process. It occupies an area of 51.4 mm^2 and comprises 64k STP programmable synapses, 64k LTP learning synapse, 256 shared synapses, and 256 analog neurons. Furthermore, it comprises digital AER input/output blocks, an on-chip temperature compensated bias generator, and analog current/voltage to spike-frequency converters.

memory used by the routers to store post synaptic destination addresses is implemented using 4k 12-bit Static Random Access Memory (SRAM)s blocks distributed among the Level-1, 2, and 3 router circuits. Thanks to the scalable architecture and to the on-chip programmable routers, the routing of all 9k neurons on the board can be easily configured to implement a wide range of connections schemes, without requiring external mapping, memory, or computing support. Furthermore, since the CAM and SRAM circuits are co-localized with, and embedded within the neuron and synapse arrays, the cxQuad architecture lends itself well to the exploitation of emerging memory technologies, e.g., by replacing the Complementary Metal-Oxide-Semiconductor (CMOS) CAM or SRAM cells with Resistive Random Access Memories (R-RAMs) or memristors. A micro-graph of the cxQuad chip is shown in Fig. 2.

The classification layer is implemented by the ROLLS chip, which comprises 256 neurons and 133,120 synapses (see Fig. 3). The neuron circuits are the same ones integrated onto the cxQuad chip. They implement a type of Adaptive Exponential (AdExp) Integrate and Fire model [13], which has been shown to be a reliable predictor of real neuron voltage traces, and can exhibit a wide range of dynamic behaviors [14]. A simplified schematic diagram of the neuron circuit is shown in Fig. 4. The sum of all synaptic input currents I_{syn} is integrated onto the capacitor C_M, in parallel with a possible DC current I_{dc}. A differential pair at the input (see NMDA block of Fig. 4) models the voltage gating mechanism of NMDA-type synapses, while the LEAK block implements the leak conductance of conductance-based neuron models. As the input currents are integrated, the neuron's output current I_{mem} increases. Eventually, the NA block activates a positive feedback mechanism that produces a sharp increase in the output and generates a fast spike. After the spike reset, the membrane capacitor C_M is actively kept discharges by the K block, thorough the I_K current for a set *refractory period*. With every spike, an adaptation current I_{ah} is increased. As this current, which models the neuron's after-hyper-polarizing current, is subtracted from the input, the output spike frequency decreases. This negative feedback mechanism models the neuron's *spike-frequency adaptation* effect. At the same time, with every spike, a digital REQ signal is produced to activate the asynchronous digital AER circuits, for the chip Input/Output (I/O) communication circuits.

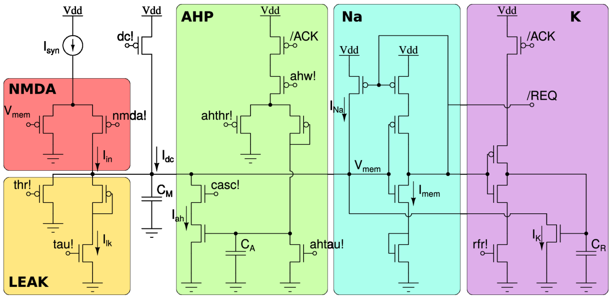

Fig. 4.: Simplified schematic diagram of the cxQuad and ROLLS chips neuron circuits. Input currents produced by the synapses I_{syn} are injected into the neuron membrane capacitance C_M, in parallel with a programmable constant DC current. The NMDA block models the voltage-gating mechanisms of NMDA synapses. The LEAK block models the neuron's leak conductance. The AHP block models the generation of the after hyper-polarizing current in real neurons, responsible for their spike-frequency adaptation behavior. The Na and K block model the effect of Sodium and Potassium channels, responsible for generating action-potentials (spikes) in real neurons. The REQ and ACK signals represent the digital voltages used to communicate Address-Events to the output AER circuits. All signals ending with "!" represent global variables (shared parameters) used to set the neuron firing properties. The I_{mem} and I_{ah} currents represent the fast and slow variables in the AdExp model, respectively.

The synapse circuits of the ROLLS chip are of three different types: linear time-multiplexed, Short-Term Plasticity (STP) synapses, and Long Term Potentiation (LTP) synapses. The STP synapses have analog circuits that can reproduce short-term adaptation dynamics and digital circuits that can set and change the programmable weights. The LTP synapses contain analog learning circuits and digital state-holding logic. The learning circuits implement the synaptic plasticity model proposed in [15], which lends itself well to VLSI implementation. The model requires only bi-stable synapses, and it is naturally robust to mismatch and variability. Our circuits update voltages continuously during the learning phase, and slowly drive the capacitor voltage that represents the synaptic weight to one of two stable states, depending on the analog synapse's weight value, compared to a threshold bias (see [9] for a thorough description and characterization of these circuits).

Table I summarizes the main features and specifications of both cxQuad and ROLLS chips.

	cxQuad	**ROLLS**
Technology	180 nm 1P6M	180 nm 1P6M
Supply voltage	1.3 V-1.8 V(core) 1.8 V-3.3 V(I/O)	1.8 V(core) 1.8 V-3.3 V(I/O)
Die Size	43.8 mm	51.4 mm
Number of Neurons	1k	256
Number of Synapses	64k	128k
Type of Neurons	Analog I&F	Analog I&F
Type of Synapses	Digital	Analog
Size of Neuron	116 μm^2	91 μm^2
Size of Synapse	12-bit CAMs: 125.8 μm^2	LTP: 252.1 μm^2 STP: 267.6 μm^2
LUT Read Speed	62.5M*Events/s	
Broadcast intra core	36M*Events/s	
Latency passing chip	15.4 ns	
Energy per spike	2.8 pJ @1.8 V	3.7 pJ @1.8 V
Energy per synaptic Event	417 fJ @1.8 V 134 fJ @1.3 V	77 fJ @1.8 V @1kHz
Power Dissipation @ 30Hz	945uW @1.8 V 360 uW @1.3 V	4 mW @1. 8V

TABLE I: Comparison of cxQuad and ROLLS features. The cxQuad and ROLLS are both fabricated using 180nm 1P6M process with a silicon area of 43.8 mm^2 and 51.4 mm^2, respectively. The cxQuad comprises 1k analog I&F neurons and 64k programmable synapse circuits; the ROLLS comprises 256 analog I&F neurons and 128k reconfigurable analog plastic synapse circuits. The energy required to produce an output spike is 3.7 pJ for the ROLLS and 2.8 pJ for the cxQuad. The cxQuad chip requires 417 fJ to implement a synaptic input event, which is broadcast to all neurons within a one core, with a 1.8 V core power supply, and 134 fJ with a 1.3 V core supply. The ROLLS only needs 77 fJ for point-to-point synaptic input event (no broadcasting), including the implementation of weight adaptation (plasticity) and conversion of digital pulse into an analog current, to feed into the connected neuron. With 3-level hierarchical routers, the cxQuad can easily transmit events to specific desired target synapses/cores/chips with an extremely low latency: events passing from one cxQuad chip to another one only need 15.4 ns. With all 1k neuron active (worst case scenario) at a mean firing rate of 30 Hz, and each event is broadcast to one core, the average power dissipation of cxQuad is 945 uW at 1.8 V power supply, and can be reduced to 360 uW at 1.3 V power supply. For similar conditions (which are typical of experiments involving classification, deep-networks, and attractor networks), the average power consumption of the ROLLS chip is approximately 4 mW.

Results

Experimental data measured from one of the ROLLS silicon neurons in response to a step input current signal is shown in Fig. 5. The data represents the silicon neuron currents sensed by a custom on-chip Analog to Digital Converter (ADC) which converts currents ranging from pico Amperes to micro Amperes into a Pulse Frequency Modulation (PFM) signal [9]. The large dots represent the I_{mem} current of Fig. 4, while the gray line represents the spike-frequency adaptation current I_{ah}, which is integrated with every output spike.

To demonstrate an application of the two chips to spike-based DNNs, we implemented the architecture described in Fig. 1: we sub-sampled the 128×128 DVS output and mapped it to the 32×32 neurons of the first chip of the 3×3 cxQuad PCB; we then set the network connectivity profile by programming the on-chip routing tables of the cxQuad chips. The first projections were configured to extract oriented edges using four 16×16 feature maps; the subsequent projections were configured to extract shapes that matched the types of stimuli used in the experiment, and to perform pooling operations, by having convergent inputs. The output of the last cxQuad chip was mapped onto the ROLLS chip, which was configured to have 8 different pools of neurons trained to perform binary classification of the 8 visual different patterns used. The training of the ROLLS neurons was supervised: it consisted in driving the neurons belonging to the true class ensemble with a high-frequency teacher signal to induce increases in the weights of the stimulated synapses. Conversely, synapses belonging to neurons that did not receive the teacher signal during that stimulus presentation tended to decrease their efficacies.

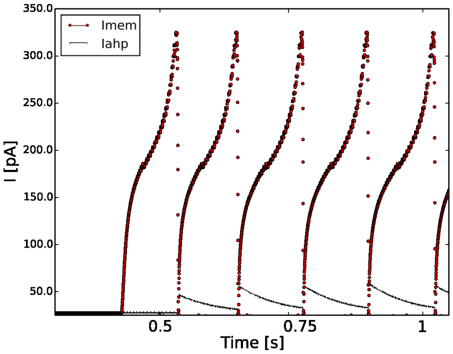

Fig. 5.: Silicon neuron output current I_{mem} and adaptation current I_{ah} measured from one of the ROLLS neuron circuits in response to a step input current activated at t=0.3 s.

Figure 6 shows the measurements made from the different stages of the network during the experiment. The visual stimuli were flashed on a monitor and sensed by the DVS. The DVS output address events were sent to the cxQuad PCB via a direct cable connection, with extremely low latency. All convolution and pooling operations took place in the cxQuad PCB in parallel and in real-time. Already after a few milliseconds (depending on the neuron time constants) from the first spikes produced by the DVS in response to the visual input, the results of the convolutional network were being transmitted to the ROLLS chip for classification. Also in this case, the cxQuad outputs were transmitted to all 256 neurons of the ROLLS chip in parallel. After the first few spikes received in input, the pool of 32 neurons that was trained to recognize the input stimulus presented responded with an average firing activity that was higher than all other pools (see bottom right spiking activity in Fig. 6).

In this toy-problem example we tested the classification with the same training-set patterns, so performance is close to 100%. This experiment is only meant to demonstrate the feasibility of this study, and to highlight the low-latency and low-power performance figures of the DNN architecture proposed (see also Table I). In general, we argue that by using a combination of slow, low-power, sub-threshold analog circuits, and fast programmable asynchronous digital circuits, and by implementing spike-based neural processing architectures in which memory and computation are co-localized, we can potentially solve the von Neumann bottleneck problem [7] for dedicated DNN computing platforms. In general, these types of neuromorphic architectures are useful for real-time sensory processing applications [3] and for being integrated in embedded systems.

Acknowledgments

This work was supported by the neuroP EU project, under ERC grant number 257219. The cxQuad hierarchical asynchronous routing methods and designs were developed by Saber Moradi and us, and the overall architecture of the cxQuad device was developed together with Fabio Stefanini

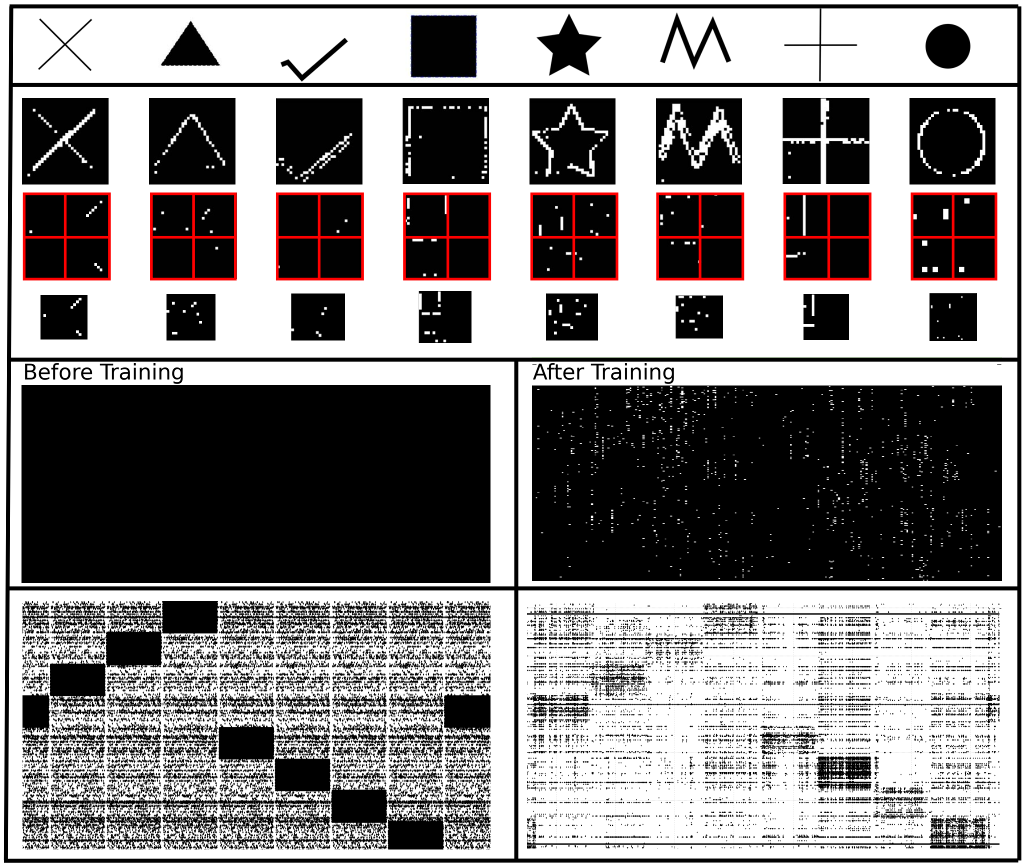

Fig. 6.: Convolution and learning with the cxQaud and ROLLS chips. The top row shows the eight visual input stimuli used in the experiment. The second row shows activations of the three input convolution and pooling layers, implemented by the cxQuad board (white dots represent neuron spiking activity). The third row shows the state of the ROLLS LTP bi-stable synapses before and after training. Black pixels represent low weights and white pixels represent high weights. The last row shows the spiking activity during the training phase (left), and the test phase (right). Each dot in the plot represents a spike. The horizontal axis represents time and the vertical axis the neuron address. High activity regions in the left plot denote the presence of the teacher signal superimposed to the input; high activity in the right plot shows how the ensemble of neurons stimulated by the stimulus it was trained to recognize produces a higher activity, compared to all other neurons in the same vertical region.

and Saber Moradi.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton. “Deep learning”. In: *Nature* 521.7553 (2015), pp. 436–444.

[2] C. Mead. “Neuromorphic Electronic Systems”. In: *Proceedings of the IEEE* 78.10 (1990), pp. 1629–36.

[3] E. Chicca, F. Stefanini, C. Bartolozzi, and G. Indiveri. “Neuromorphic electronic circuits for building autonomous cognitive systems”. In: *Proceedings of the IEEE* 102.9 (Sept. 2014), pp. 1367–1388.

[4] G. Indiveri, B. Linares-Barranco, R. Legenstein, G. Deligeorgis, and T. Prodromakis. “Integration of nanoscale memristor synapses in neuromorphic computing architectures”. In: *Nanotechnology* 24.38 (2013), p. 384010.

[5] S. Yu, Y. Wu, R. Jeyasingh, D. Kuzum, and H.-S. Wong. “An electronic synapse device based on metal oxide resistive switching memory for neuromorphic computation”. In: *Electron Devices, IEEE Transactions on* 58.8 (2011), pp. 2729–2737.

[6] L. Deng et al. “Complex Learning in Bio-plausible Memristive Networks”. In: *Scientific reports* 5 (2015).

[7] G. Indiveri and S.-C. Liu. “Memory and information processing in neuromorphic systems”. In: *Proceedings of the IEEE* 103.8 (2015), pp. 1379–1397.

[8] L. Breiman. “Bagging predictors”. In: *Machine Learning* 24 (1996), pp. 123–140.

[9] N. Qiao et al. “A Re-configurable On-line Learning Spiking Neuromorphic Processor comprising 256 neurons and 128K synapses”. In: *Frontiers in Neuroscience* 9.141 (2015).

[10] P. Lichtsteiner, C. Posch, and T. Delbruck. “A 128x128 120 dB 15 μs Latency Asynchronous Temporal Contrast Vision Sensor”. In: *IEEE Journal of Solid-State Circuits* 43.2 (Feb. 2008), pp. 566–576.

[11] S. Moradi, N. Imam, R. Manohar, and G. Indiveri. “A Memory-Efficient Routing Method for Large-Scale Spiking Neural Networks”. In: *Circuit Theory and Design, (ECCTD), 2013 European Conference on.* IEEE, 2013, pp. 1–4.

[12] S. Moradi, G. Indiveri, N. Qiao, and F. Stefanini. *Networks and hierarchical routing fabrics with heterogeneous memory structures for scalable event-driven computing systems.* European patent application EP 15/165272. Filed 27.04.2015. Apr. 2015.

[13] R. Brette and W. Gerstner. “Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity”. In: *Journal of Neurophysiology* 94 (2005), pp. 3637–3642.

[14] R. Naud, N. Marcille, C. Clopath, and W. Gerstner. “Firing patterns in the adaptive exponential integrate-and-fire model”. In: *Biological Cybernetics* 99.4–5 (Nov. 2008), pp. 335–347.

[15] J. Brader, W. Senn, and S. Fusi. “Learning real world stimuli in a neural network with spike-driven synaptic dynamics”. In: *Neural Computation* 19 (2007), pp. 2881–2912.