A 43pJ per Inference CBNN-based Compute-in-sensor Associative Memory in 28nm FDSOI

Benoit Larras, Antoine Frappé

A 43pJ per Inference CBNN-based Compute-in-sensor Associative Memory in 28nm FDSOI

Benoit Larras and Antoine Frappé
Univ. Lille, CNRS, Centrale Lille, Junia, Univ. Polytechnique Hauts-de-France, UMR 8520 - IEMN, F-59000 Lille, France
benoit.larras@junia.com

***Abstract*—Distributed smart sensors are more and more used in applications such as biomedical or domestic monitoring. However, each sensor broadcasts data wirelessly to the others or to an aggregator, which leads to energy-hungry sensor nodes and an increased latency at the network level. To tackle both challenges, this work proposes to distribute part of the processing elements in each sensor node and presents an ASIC implementation of an associative memory using clique-based neural networks (CBNNs) coupled with an integrated SRAM memory, in a 28nm FDSOI technology node. It consumes 43pJ for a single inference, which is 6.5 times better than state-of-the-art associative memories implementations, for the same memory size.**

I. Introduction

In the context of healthcare monitoring, wireless body area sensor networks (WBAN) are used to gather data in multiple places of the body. The targeted applications include cardiac monitoring, posture or gesture recognition, etc. Traditionally, a WBAN is structured as in [1], Figure 1.a. All the sensors in the network acquire data through their sensing interface and broadcast it wirelessly. Then, an aggregator gathers information from all the sensors and processes the whole data flow with one or more processing elements (PEs), usually using Artificial Intelligence (AI) algorithms. They are connected to a dedicated memory designed to store all the sensors' data, connections map, and system states.

However, several drawbacks of this architecture need to be highlighted for the case of autonomous sensors in a WBAN. First of all, the sensors transmit continuously the acquired raw data, which consumes most of the energy budget of the sensor. Besides, PEs in the aggregator have to compute the concatenated data flow, either by the means of multiple PEs operating in parallel or by increasing the computation time. The latency of the network is also impacted since the PEs have to wait to receive all the data to begin their computation. Finally, a larger bus is needed to communicate with the global memory in the aggregator, which requires increased time and energy budgets.

In the context of sensor networks, the "Near-sensor computing" paradigm tends to move at least a part of the PEs within the sensors, instead of computing multiple data flows in the same block, as described in [2], Figure 1.b. Even though the processing capability becomes more limited in each sensor if the PEs are divided among the network, there is an opportunity to decrease the sensors' energy consumption. At the sensor level, pre-processing data allows decreasing the number of bits to transmit to the rest of the network, which also decreases the sensor's total energy consumption. The memory exchanges and global latency are also reduced at the network level, which increases the capabilities of real-time operation.

In WBANs, a commonly used computation is the convergence to the closest *a priori* stored configuration of physiological parameters. It is implemented, in [3] in the case of hand gesture recognition, and in [4] in the case of posture recognition, by the means of associative memories. This type of AI paradigm is interesting as it does not require off-line training of the entire network, but storing the different configurations to output. In [3], the minimal Hamming distance with the stored gestures is computed with a global digital logic block from surface electromyography sensors on the arm. The memory and PE are implemented in an aggregator and are not easily distributed next to the sensors. In [4], a

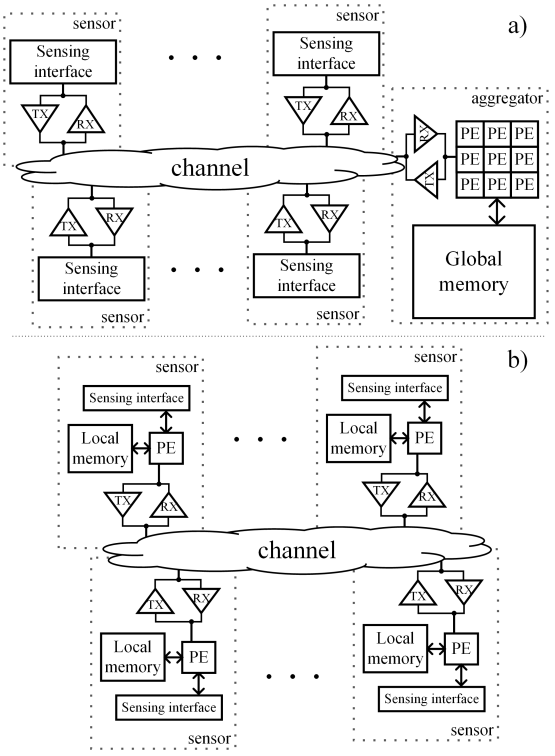

Figure 1. Structural description of 2 versions of a WBAN. a) Architecture with centralized computations in an aggregator. b) Architecture with distribution of the processing elements in the sensors.

mixed-signal implementation of a clique-based neural network (CBNN), distributed in clusters, is used to determine the posture by combining data from accelerometers on the body. This type of associative memory is strongly compatible with near-sensor distributed computing but the referenced circuit does not implement on-chip storage for sensor and external data. Other non-CMOS implementations of associative memories are possible using opto-electronic [5] and memristive [6] components.

This work proposes the implementation of a distributed PE integrating a cluster from a CBNN and its on-chip memory in 28nm FDSOI technology node. The benefits are:

- a reduced energy consumption per inference (less than 43pJ under 0.7V supply) and silicon occupation of the block {PE+memory} (27,600μm^2);
- a decreased circuit complexity by the use of transistor body biasing;
- a standalone operation by the means of memory integration and organization.

This paper is organized into the following sections. Section II describes the architecture of the proposed localized PE, Section III presents the measurement results, from individual characterization as well as applicative use-cases, and Section IV concludes the paper.

II. Processing element (PE) implementation

A. CBNN behavior and distribution

Clustered CBNN networks are formed by grouping neurons per category of information, Fig. 2. There are N_C clusters, N_N neurons per cluster and N_S synapses per neuron. One cluster corresponds to the information contained by one sensor. One neuron corresponds to a set of parameter values from this sensor. Neurons from different clusters are connected to form a clique, and all the cliques represent the convergence options of the network. An in-cluster "Winner-Takes-All" (WTA) rule activates or not the neurons. Thus, only one neuron is activated per cluster.

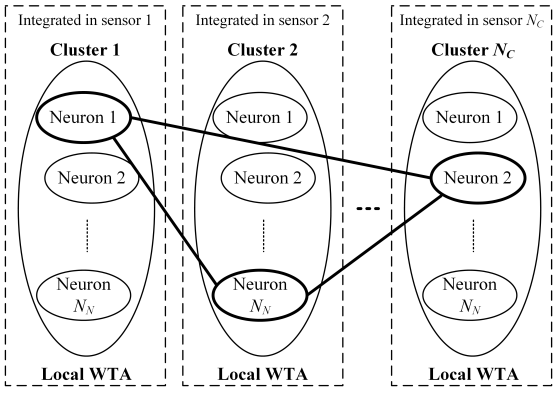

Figure 2. Structure of a CBNN implementing N_C clusters of N_N neurons each. Each cluster is integrated in an independent sensor. An example of a clique is highlighted in black.

The clustered structure of the CBNN is compatible with the "Near-sensor computing" paradigm, as the computations are localized per cluster. The clusters can therefore be dispatched next to the sensors as the PEs in Fig. 1.b. The transmitted data from each sensor become the output of the integrated cluster, plus its index (reference address in the network), instead of raw data from the sensors. The total number of bits transmitted by a sensor node is $log_2(N_C) + log_2(N_N)$. In terms of power consumption and latency, this transmission scheme is more efficient than computing all the sensor features with the whole CBNN in the aggregator, for a value of N_N greater than 27 [4]. Moreover, the embedded memory depicted in Fig. 1 can be distributed as well in each sensor node. Each local memory can therefore be reduced to store only the incoming connections to the concerned integrated cluster. With smaller memories, the access time for each cluster is faster and thus more energy efficient.

B. PE mixed-signal circuit

Analog CMOS circuits implement a cluster of neurons with the lowest circuit complexity and, thus, lead to low power consumption [7]. N_S synapses feed the input contributions into each neuron, then the WTA operation is triggered between the neurons in the cluster, as shown in Fig. 3. The synapses are implemented as switched current sources converting binary input voltages into binary currents, *i.e.* 0A or I_{UNIT}, which are summed at node A.

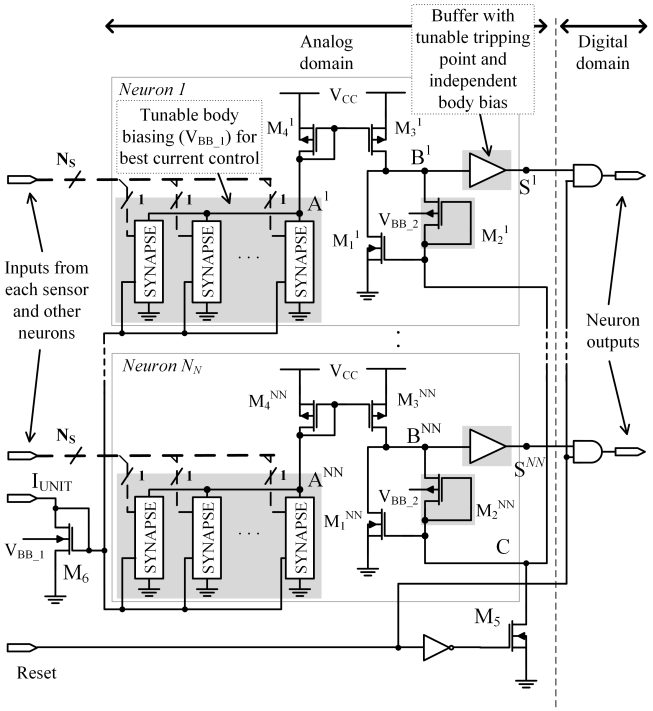

Figure 3. Schematic of a mixed-signal cluster of N_N neurons in a CBNN. N_S is the number of possible input connection per neuron. Shaded areas show independently body-biased transistors.

The WTA circuit, separated in the neurons, is composed of transistors M_1 and diodes M_2, and receives the input current copied by the M_3 - M_4 current mirror. The current flowing in the WTA element sets the drain-source voltage of M_1 depending on the currents flowing in the other neurons, connected by the means of node C. The result of the operation is output as a binary voltage at node B. If the input current of a neuron is the highest, the drain-source voltage of the corresponding transistor M_1 is set higher than its saturation voltage. Otherwise, the drain-source voltage of M_1 is set under its saturation voltage. A buffer translates the voltage at node B in a standard binary voltage at node S. The activated neuron's index is output to the rest of the network.

The circuit, designed for the 28nm FDSOI technology node, benefits from the body biasing feature in 3 ways. First, in order to optimize the energy consumption of the PE, the transistors in the synapses operate in the subthreshold region. A better control of the I_{UNIT} current value is achieved through body biasing of the transistors in the synapses. Second, the buffer threshold at node B is affected by PVT variations and typically requires a feedback circuit to calibrate the threshold by current starving. Here, the buffer is body-biased independently so that the threshold is always comprised between the high and low levels of the WTA circuit, and thus no external circuitry is needed. Finally, leakage currents flowing in each blocked diode M_2 are drawn from the synapses currents and limit the number of WTA elements in parallel. A second stage of WTA operation is needed for higher number of neurons in a cluster [7]. By body biasing separately the diodes, the leakage current is reduced from 3.6nA down to 41pA, which can allow either further reducing I_{UNIT}, or increasing the number of WTA elements in parallel by a factor of 100.

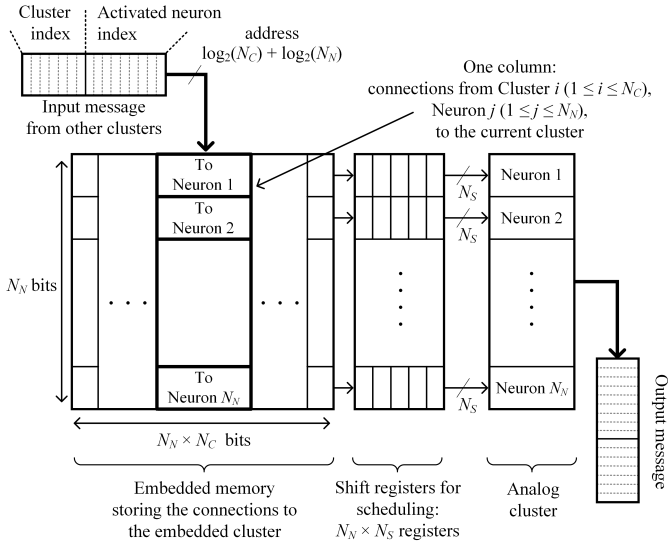

Figure 4. Organization of the embedded memory. It contains $N_C \times N_N$ words of N_N bits each, displayed as columns. Each square represents a bit storing the connection state between 2 neurons from a distant cluster to the concerned integrated cluster. Each row represents a neuron in the concerned cluster (*destination neuron*), and each column a neuron in the whole network (*source neuron*).

C. Memory organization

The embedded memory stores the binary connections between the neurons from different clusters in the network. The only implemented connections are those from neurons of other clusters to the considered integrated cluster, as shown in Fig. 4. The memory stores $N_C \times N_N$ words. It corresponds to the number of neurons in the whole network. Each word contains N_N bits, each bit stimulating one neuron in the considered cluster.

The reading process is scheduled as follows. The circuit receives $log_2(N_C) + log_2(N_N)$ bits from another cluster. It corresponds to the index winning neuron in that cluster, concatenated with the cluster index. This message is used as an address to output the activated connections from the embedded memory to the current cluster. The output word is pushed in a shift register in order to gather the following messages from the other clusters. The selected bits are input in the analog cluster at the same time, as soon as all the clusters inputs have been received. Therefore, the order in which the messages are received does not matter. A cluster can also time out before sending data, a binary timeout flag is triggered. In that case, the shift register outputs the received data as-is into the analog cluster. If half of the neurons are stimulated, a stored clique can be recovered thanks to the high inherent redundancy of CBNNs.

III. Measurement results

The proposed design is implemented on a 1-mm^2 ASIC in the 28nm FDSOI technology node, as depicted in Fig. 5. Three independent instances of the block {PE+memory}, named C1, C2 and C3, are implemented on-chip. They can be used for different sensors in the same network.

A. Characterization of the block {PE+memory}

Each block implements a cluster of 32 neurons, with 16 synapses per neuron. There are 512 scheduling registers per block, and the memory stores 512 words of 32 bits each. The total silicon area per block {PE+memory} is 27,600μm^2.

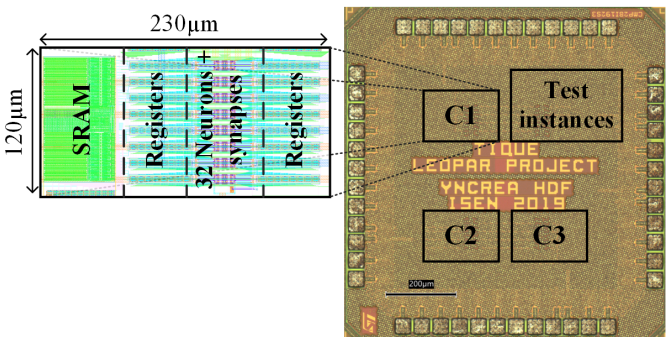

Figure 5. Photograph of the fabricated chip in 28nm FDSOI technology node. It implements 3 independent blocks {PE+memory} C1, C2 and C3.

Table I
ELECTRICAL CHARACTERISTICS OF A BLOCK {PE+MEMORY}

Technology process	28nm FDSOI	
Unitary synapse current I_{UNIT}	30nA	
Supply voltage V_{CC}	**1V**	**0.7V**
Static power consumption	7μW	0.8μW
Energy per memory read	3.6pJ	2.5pJ
Energy consumption: registers	max. 19pJ	max. 2.4pJ
Energy consumption: analog cluster	max. 820fJ*	max. 550fJ*
Analog cluster inference time	101ns	

* obtained for 25% of the number of inputs activated [7].

The electrical characteristics are shown in Table I. The circuit operates at a supply voltage V_{CC} between 0.7V and 1V, with a unitary current I_{UNIT} of 30nA. The measured static power consumption for one instance is of 7μW for 1V, and 0.8μW for 0.7V. Once the memory is loaded, a series of up to 16 32-bit vectors is loaded in the shift registers, and the data bits are loaded in the cluster, starting its convergence phase. The power consumption of the analog WTA depends on the number of '1' in the memory, however most of the dynamic power is used in the shift registers. Including the memory read operations, the block {PE+memory} consumes at most 43pJ per inference under 0.7V supply.

B. State of the art of associative memories for WBAN

The individual characteristics of the proposed implementation are compared to state-of-the-art associative memories for WBAN. Their features are summed up in Table II. In [3], the associative memory used for hand gesture recognition computes the Hamming distance between the sensors' inputs and the stored configurations. However, the associative memory is implemented on an external FPGA, which constrains the system for embedded operation and is not relevant for performance comparison. A state-of-the-art associative memory implementing a multi-context ternary content addressable memory (T-CAM) is integrated in an ASIC used for speech recognition in [8]. It is combined with an SRAM storing 96kb of data. The association operation consumes 80pJ, but the main drawbacks of this implementation are linked to the use of a global memory storing all the convergence options. The static energy consumption of the global memory is 20 times more important than that of the associative operation itself, and its silicon area increases dramatically the circuit surface. Distributing the memory mitigates these drawbacks, which is not demonstrated in [4] because the memory is implemented externally on a FPGA. Globally, for a distribution in 16 sensor nodes, the energy consumption normalized to the embedded memory size is reduced in this work by a factor of 6.5. In addition, it has been shown in [4] that the energy consumption overhead due to the wireless interface is mitigated since a lower number of bits is transmitted in the distributed scheme.

IV. CONCLUSION

This paper presents a distributed mixed-signal implementation of clique-based neural networks, combined with an SRAM memory. The circuit is used as an associative memory in WBAN applications, and has been fabricated in 28nm FDSOI technology node. It consumes at most 43pJ per inference, which is 6.5 times the energy consumption of state-of-the-art implementation of associative memories such as T-CAMs for the same memory size. Further work will include real-life demonstration of the complete WBAN with distributed PEs and memories in the sensors.

Table II
ASSOCIATIVE MEMORIES STATE OF THE ART

	[3]	[8]	[4]	**This work**
Application	Hand gesture recognition	Speech recognition	Posture recognition	**WBAN**
Integration process	FPGA	ASIC 65nm bulk	ASIC 65nm bulk + FPGA	**ASIC 28nm FDSOI**
Computation type	Hamming distance digital computation	Multi-context T-CAM	CBNN	**CBNN**
Topology	Localized	Localized	Distributed	**Distributed**
Available memory	–	96kb	8kb (40kb for 5 dist.nodes)	**16kb** (256kb for 16 dist. nodes)
Silicon area	n.a.	174,000μm^2 without SRAM	210,000μm^2 without SRAM	**27,600μm^2**
Core energy consumption	n.a.	80pJ	max. 18pJ	**max. 3pJ***
Total energy consumption	–	1.68nJ	–	**max. 43pJ** per node*
Energy consumption per stored bit	–	17.5pJ/bit	–	**max. 2.7pJ/bit***

* obtained for 0.7V.

REFERENCES

[1] S. Movassaghi *et al.*, "Wireless Body Area Networks: A Survey," *IEEE Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1658–1686, 2014.

[2] W. Yu *et al.*, "A survey on the edge computing for the internet of things," *IEEE Access*, vol. 6, pp. 6900–6919, 2018.

[3] A. Moin *et al.*, "A wearable biosensing system with in-sensor adaptive machine learning for hand gesture recognition," *Nature Electronics*, vol. 4, pp. 54–63, 2021.

[4] B. Larras and A. Frappé, "On the Distribution of Clique-Based Neural Networks for Edge AI," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, pp. 1–1, 2020.

[5] Z. Wang and X. Wang, "A novel memristor-based circuit implementation of full-function pavlov associative memory accorded with biological feature," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 7, pp. 2210–2220, 2018.

[6] Y. Alkabani, M. Miscuglio, V. J. Sorger, and T. El-Ghazawi, "Oe-cam: A hybrid opto-electronic content addressable memory," *IEEE Photonics Journal*, vol. 12, no. 2, pp. 1–14, 2020.

[7] B. Larras *et al.*, "A Fully Flexible Circuit Implementation of Clique-Based Neural Networks in 65-nm CMOS," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 5, pp. 1704–1715, May 2019.

[8] R. Arakawa *et al.*, "Multi-Context TCAM-Based Selective Computing: Design Space Exploration for a Low-Power NN," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 1, pp. 67–76, 2021.