Design and Architectural Co-optimization of Monolithic 3D Liquid State Machine-based Neuromorphic Processor

Bon Woong Ku[†], Yu Liu[§], Yingyezhe Jin[§], Sandeep Samal[‡], Peng Li[§], and Sung Kyu Lim[†]
[†]School of ECE, Georgia Institute of Technology, Atlanta, GA
[§]Department of ECE, Texas A&M University, College Station, TX
[‡]Intel Corp., Hillsboro, OR
{bwku,limsk}@ece.gatech.edu

ABSTRACT

A liquid state machine (LSM) is a powerful recurrent spiking neural network shown to be effective in various learning tasks including speech recognition. In this work, we investigate design and architectural co-optimization to further improve the area-energy efficiency of LSM-based speech recognition processors with monolithic 3D IC (M3D) technology. We conduct fine-grained tier partitioning, where individual neurons are folded, and explore the impact of shared memory architecture and synaptic model complexity on the power-performance-area-accuracy (PPAA) benefit of M3D LSM-based speech recognition. In training and classification tasks using spoken English letters, we obtain up to 70.0% PPAA savings over 2D ICs.

CCS CONCEPTS

• **Computer systems organization → Neural networks**; • **Hardware → Neural systems**;

KEYWORDS

Liquid State Machine; Neuromorphic Processor; Monolithic 3D ICs

ACM Reference Format:
Bon Woong Ku[†], Yu Liu[§], Yingyezhe Jin[§], Sandeep Samal[‡], Peng Li[§], and Sung Kyu Lim[†]. 2018. Design and Architectural Co-optimization of Monolithic 3D Liquid State Machine-based Neuromorphic Processor. In *DAC '18: DAC '18: The 55th Annual Design Automation Conference 2018, June 24–29, 2018, San Francisco, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3195970.3196024

1 INTRODUCTION

The liquid state machine (LSM) [3] is a model of recurrent spiking neural networks (SNNs) constructed with a recurrent reservoir and a training unit. In the standard LSM model, the recurrent reservoir consists of a set of spiking neurons randomly connected with non-trainable synapses, and exhibits complex non-linear dynamics as a pre-processor mapping input patterns to a higher-dimensional transient response. The training unit receives the reservoir responses for the final classification through trainable synapses, referred to as output synapses.

While SNNs hold a lot of promise due to their bio-plausibility and hardware implementation efficiency, the training of SNNs still remains challenging. It is difficult to develop a powerful gradient-based learning mechanism for SNNs, particularly recurrent SNNs. To this end, the LSM is envisioned as a good tradeoff between the ability in tapping the computational power of recurrent SNNs and engineering tractability. Recently, cost-effective hardware implementations of the LSM have been investigated along with bio-inspired training algorithms to tune both the reservoir and training unit. For example, [9] proposed a supervised probabilistic spike-dependent output tuning algorithm, [8] proposed an LSM-based learning processor with runtime programmable arithmetic precision and data-dependent reconfiguration. [1] proposed a self-organizing LSM architecture with hardware-friendly spike-timing-dependent-plasticity rules for reservoir tuning.

Monolithic 3D (M3D) is an emerging 3D technology enabled by the sequential integration of device layers. This technology uses miniscule monolithic inter-tier vias (MIVs) (<100nm diameter, <1fF), which achieves massive vertical integration density with no silicon-area overhead from 3D vias. These 3D connections help in reducing wirelength and power with potentially better performance and memory access options [5]. In particular, M3D IC design offers great benefits in neural network designs due to the neuromorphic architecture with a huge number of connections at both intra-neuron and inter-neuron levels. In this work, for the first time, we explore the benefits offered by M3D ICs in LSM-based speech recognition processors.

The major contributions of this paper are (1) We carry out ASIC design for LSM neural processors in 2D and M3D ICs with detailed design comparison. (2) We explore the impact of different synapse models and memory distributions on the power-performance-area-accuracy (PPAA) benefit of M3D LSM neural processors. (3) We conduct vector-based functional verification and PPAA analysis for the real-world task of speech recognition.

2 LSM ARCHITECTURE DESCRIPTION

2.1 Processor Architecture

The overall LSM processor architecture is adopted from [1], and there are 135 digital reservoir neurons (RNs) in the reservoir unit (RU) and 26 digital output neurons (ONs) in the training unit (TU) as depicted in Fig. 1. External input spikes are fed to their targeted reservoir neurons through the crossbar interface with a pre-defined

DAC '18, June 24–29, 2018, San Francisco, CA, USA

ACM ISBN 978-1-4503-5700-5/18/06…$15.00
https://doi.org/10.1145/3195970.3196024

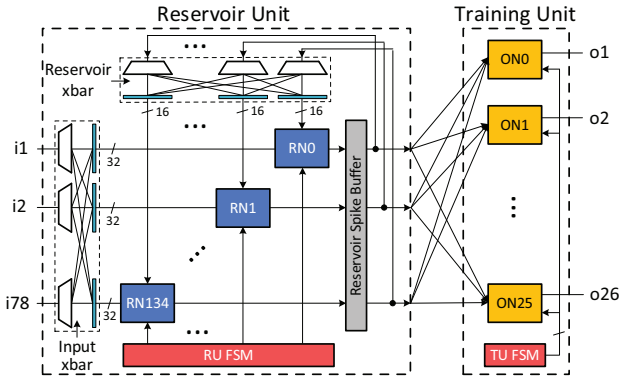

Figure 1: Our LSM-based neuromorphic processor architecture. There are 135 reservoir neurons (RNs) in the reservoir unit, and 26 output neurons (ONs) in the training unit. Each RN receives up to 32 external input spikes and up to 16 pre-synaptic reservoir spikes. Each ON has a full connection to the individual RNs to receive the reservoir response.

connectivity pattern. The spikes generated from reservoir neurons are registered (i.e. Reservoir spike buffer[134:0]) and propagate to the TU. Meanwhile, these spikes are also sent back to other reservoir neurons in the RU recurrently through reservoir crossbar interface. The operations of neurons at the same layer are executed in parallel under the control of a global finite state machine (FSM).

Our on-chip training of the LSM processor is divided into two phases unlike the standard LSM training model. First, we train RU based on a hardware-friendly spiking-timing dependent plasticity (STDP) algorithm [1] until its synaptic weight distribution converges. Then, a bio-plausible supervised spike-based learning algorithm [2] is employed on the TU for the main classification function. In this second phase, the reservoir maintains its synaptic weights while producing spike responses for the TU.

2.2 Digital Spiking Neuron Implementation

The proposed LSM neural processor operates through a series of computational steps that are controlled by the corresponding states of the global FSM in the RU and TU, respectively. Based on the architectural and functional properties, we partition the implementation of a single digital neuron into three functionally dependent modules: the synaptic input processing module, the spike generation module, and the learning module. At each time step, these three modules activate in order, controlled by the global FSMs.

The synaptic input processing module computes synaptic responses upon arrival of spike inputs. As a baseline, we implement 2nd-order dynamic synaptic model [9], in which the excitatory and inhibitory synapses have their separate state variables:

$$\begin{aligned} EP(t+1) &= EP(t)(1-1/\tau_{EP}) + \sum w_i \cdot S_+(i) \\ EN(t+1) &= EN(t)(1-1/\tau_{EN}) + \sum w_i \cdot S_+(i) \\ IP(t+1) &= IP(t)(1-1/\tau_{IP}) + \sum w_i \cdot S_-(i) \\ IN(t+1) &= IN(t)(1-1/\tau_{IN}) + \sum w_i \cdot S_-(i) \end{aligned} \quad (1)$$

where $EP(t)$ and $EN(t)$ are excitatory state variables of a neuron at the tth biological time step, while $IP(t)$ and $IN(t)$ are for inhibitory ones. τ_{EP}, τ_{EN}, τ_{IP}, τ_{IN} are the decay constants of the corresponding state variables, w_i is the synaptic weight. $S_+(i)$ and $S_-(i)$ is the spike of the i-th excitatory/inhibitory synapse.

When updating the state variables in a neuron, the input synapses are examined in serial. If there is an input spike at the current time step, the synaptic weight of the associated synapse will be added to the corresponding state variables. After the synaptic responses are generated, the spike generation module updates the membrane potential V_{mem} with the response based on the widely used leaky integrate-and-fire (LIF) model and generates a spike if the membrane potential exceeds a pre-defined threshold. The calculation of membrane potential follows below:

$$V_{mem}(t+1) = V_{mem}(t)(1-1/\tau_m) + \frac{EP-EN}{\tau_{EP}-\tau_{EN}} - \frac{IP-EN}{\tau_{IP}-\tau_{IN}} \quad (2)$$

where $V_{mem}(t)$ is the membrane potential at the tth biological time step, τ_m is the decay constant of membrane voltage.

At last, the learning module activates in each emulation time step after the spike generation module finishes the process and tunes the afferent pre-synaptic weights of the associated neurons with a bio-inspired supervised spike-based algorithm [9]. In our LSM neural processor, we implement the activity-dependent clock gating adopted from [2] and directly gate on the clock signals inside each neuron. The clock signal of each functional module only toggles when the module needs to be activated.

3 DESIGN FLOW AND METHODOLOGIES

3.1 Baseline RTL-to-GDS Flow

In this work, we implement full-chip RTL-to-GDSII ASIC LSM neural processors using commercial 28nm process design kit at the block-level with 135 reservoir neurons and 26 output neurons to reduce the design complexity and facilitate IP reuse. While using the conventional hierarchical design flow for 2D IC design, we adopt the state-of-the-art M3D design flow named Shrunk-2D [5], and extend it to build the top-down hierarchical M3D IC design.

In Shrunk-2D flow, a pseudo-3D design called Shrunk2D is built where the dimensions of the cells, wire pitches/widths are scaled down by a factor 0.707 ($1/\sqrt{2}$) to emulate 50% footprint reduction in 3D IC. The idea is to account for the wirelength reduction impact on timing and optimization in 3D IC, while using the same standard cell timing/power information as the original technology. The optimized placement of the Shrunk2D design is used as an initial solution for tier partitioning where we maintain the (X,Y) coordinates and just change the tier location. MIV insertion is carried out subsequently by using a 3D metal stack and the cell pins defined in proper layers based on the tier location. The diameter of the MIV used is 50nm and the RC parasitics are (10Ω, 0.2fF) based on 28nm PDK metal pitches, via-sizes, and via aspect ratio.

3.2 Hierarchical Shrunk-2D

We carry out two-level folding where each individual neuron is partitioned into two tiers, and top-level cells are partitioned into two tiers incrementally. First, we decide the top-level floorplan based on

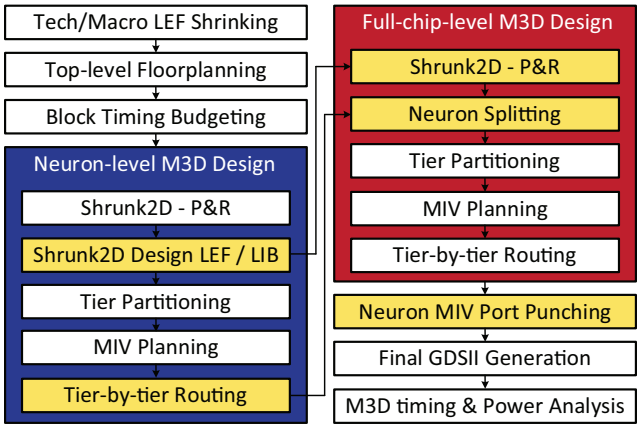

Figure 2: Our hierarchical Shrunk-2D flow to enable two-level design folding: individual neuron is partitioned into two tiers, and top-level design is also tier partitioned.

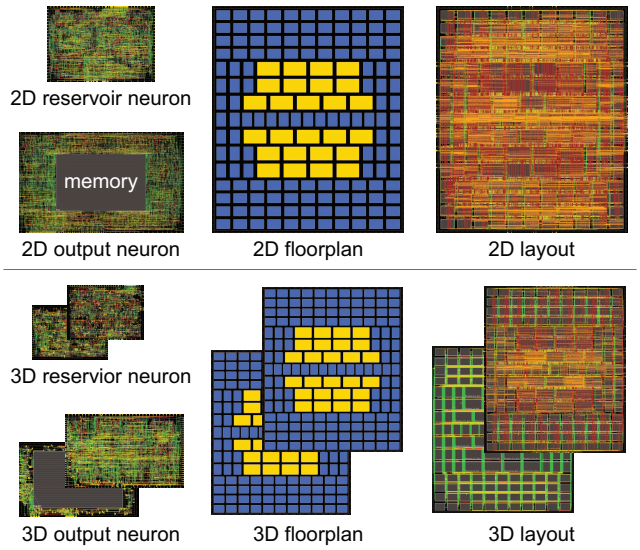

Figure 3: 2D vs. M3D designs of a reservoir neuron, a output neuron, and full-chip. Reservoir neurons are in blue, and output neurons in yellow in the flooplan.

the shrunk layout geometry, and derive the timing budget for the reservoir and output neuron blocks. Then, we follow the Shrunk-2D flow for each neuron, and build two-tier folded M3D neuron designs. To build top-level Shrunk2D design, we use Shrunk2D design for individual neuron blocks. Although the top-level Shrunk2D design finds the neurons unfolded in this step, the individual neuron is actually folded, and fully occupy the placement area in both tiers. Therefore, we need to split the Shrunk2D neuron blocks into two different blocks that share the same X,Y location but placed on the separate tiers. This is called neuron splitting.

The top-level netlist and placement result also should be updated in accordance with the neuron splitting. Then, we build the top-level M3D design. Once the tier-by-tier routing for the top-level is done, we replace the neuron macros into the ones with MIV ports, and revise the Verilog and routing results to support full connectivity including both top-level and neuron-level 3D connections. This is called neuron MIV port punching. Lastly, we generate GDSII file for our M3D LSM neuromorphic processer, and proceed the signoff M3D timing and power analysis.

3.3 Design Methodology Enhancements

We use 6 metal layers in 2D IC while only four metal layers are allowed inside each neuron. For the M3D IC, 4+4 metal layers are used inside the folded neuron to provide the same routing resources as the 2D neuron, and additional two routing layers on the top tier are dedicated to the inter-neuron routing. In a reservoir neuron, we use flip-flops to store synaptic weights considering relatively limited pre-synaptic fanins. For a output neuron, however, we use register-file modules to store the weights since they have trainable synapses in full connection to the reservoir unit. Memory modules are generated using a commercial memory compiler for the used 28nm technology node, and occupy up to four metal.

For the tier partitioning, we place the cells and pins of the neuron block to maximize the area and power benefit leveraged from M3D IC. For reservoir neurons, we put all functional cells in the synaptic input processing module and the action potential (spike) generation module on the top tier so that they are on the same layer with the global nets and closer to the external connections to package pins. Then, we separate the 16-bit reservoir spike input pins into two groups and put the 8 lower bits of the reservoir spike inputs and their peripheral logic cells on the bottom tier. All other input and output pins are assigned to the top tier for simplicity. Since the reservoir spike input pins are connected to the synaptic input processing module, by having half of the reservoir spike inputs on the bottom tier, we increase the vertical connections inside each neuron.

The memory inside each output neuron takes a large part of the layout. Considering that the routing across the memory is costly, we put the memory and its peripheral logic cells on the bottom tier while all other cells (i.e. synaptic input processing module, action potential (spike) generation module and the learning module) on the top tier. Similar to the reservoir neuron, we also partition the spike input pins of an output neuron into two evenly sized groups and put one group on the bottom tier to increase the vertical connections.

Figure 3 demonstrates the M3D and 2D LSM neuromophic processors. The two-tier M3D IC footprint is half that of 2D IC. Therefore, the total silicon area used is the same. Since output neurons communicate with all reservoir neurons, the 26 output neurons are uniformly arranged in the center of the floorplans.

4 DESIGN/ARCHITECTURE OPTIMIZATION

4.1 Memory Sharing

In the proposed LSM processor, a large number of memory resources are required for weight storage, thus an efficient memory design scheme is important for the hardware cost and energy efficiency. The straightforward way is to distribute the memory module inside each neuron. The depth of the memory depends on the number of pre-synapses of the neuron, which is set to be 16 for reservoir

neurons and 135 for output neurons. The memory width represents the synaptic weight bit resolution, which is 2 and 8 for reservoir and output synapses, respectively.

Although the distributed memory architecture is easy to implement, it results in large peripheral overhead due to a large number of memory modules. To improve the memory efficiency, we replace the individual weight storage inside the neuron with a large shared memory at reservoir and output layer, respectively. This is based on that, at each emulation time step, all neurons at the same layer work in parallel; The synaptic weights are accessed in serial following the same order based on their index. Therefore, the neurons at the same layer are actually accessing the same address of their own memory, although the values stored at that address might be different. Given that, in the shared memory architecture, we store all synaptic weight values in a row that are previously at that same address in the distributed memory, and the values are associated with different neurons by the bit index. When updating the weight value, the updated synaptic weights from all neurons will first be concatenated to one word then write to the intended address. When reading the weights, different parts of the memory output are assigned to their targeted neurons.

4.2 Synaptic Model Complexity Reduction

Reducing synaptic model from the 2nd-order dynamics to the 1st-order dynamics is another approach to optimize the overall power-performance-area-accuracy benefit. In the 1st-order synapse model, there is only one state variable E in each neuron, which represents the overall synaptic response among all its input spikes:

$$E(t+1) = E(t)(1 - 1/\tau_E) + \sum_i w_i \cdot S_i \tag{3}$$

where $E(t)$ is the 1st-order state variable at the tth biological time step, τ_E is the decay constant of the synaptic response.

The calculation of membrane voltage in the 1st-order synaptic model is also different from the 2nd-order:

$$V_{mem}(t+1) = V_{mem}(t)(1 - 1/\tau_m) + \frac{E}{\tau_E} \tag{4}$$

In the following sections, we will show that these two approaches will effectively reduce the area and power of the M3D LSM neural processor without hurting the classification accuracy too much.

4.3 Individual Neuron Results

First, we compare the 2D neuron designs of the shared memory architecture to those of the baseline distributed memory 2nd-order synaptic model architecture. The distributed memory modules occupy huge placement area and internal power inside the individual neuron. Using shared memory architecture, these modules are now located at the top-level hierarchy, and it leads to 14% and 54% footprint area savings for reservoir and output neurons, respectively. The reduced number of flip-flops and the absence of memory module allows to 24%, and 48% internal power savings, and reduced footprint leads to 15%, and 23% switching power savings in the reservoir and output neuron, respectively. For output neuron, eliminating the memory module not only helps to reduce the huge internal power, but also removes the routing blockage over the memory module, resulting in the efficient routing.

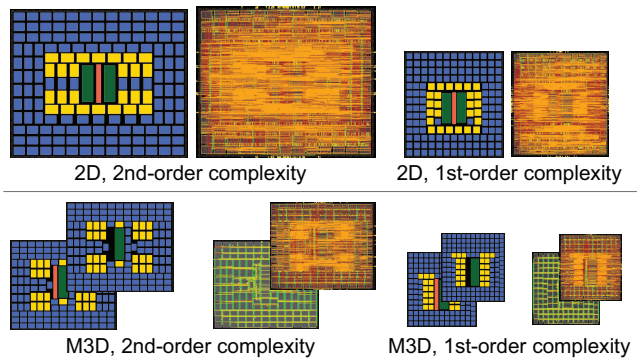

Figure 4: 2D vs. M3D LSM processors with memory sharing & synaptic model complexity reduction schemes. In red is shared memory for the reservoir neurons (yellow), and in greens are for output neurons (blue).

On top of this huge benefit, reducing synaptic model complexity enables more compact neuron design by reducing the cell count from the relaxed synaptic weight precision. This results in 57% and 75% footprint savings from shared memory 1st-order synaptic model architecture compared to the baseline architecture, and 65% and 69% of total power savings for the reservoir and output neuron, respectively.

We observe that M3D designs offer even more savings in terms of footprint, and power consumption for all neuron designs on top of the architectural optimization benefit. Assuming no silicon area overhead, 50% footprint savings of M3D design lead to additional 9% and 4% total power savings in the reservoir neuron and 15% and 4% for output neurons in two-different architectures as shown in Figure 5. It is note worthy that the shared memory 2nd-order synaptic model architecture maximizes the M3D power benefits in both neuron designs. This is because, targeting 1GHz, the neurons of 1st-order synaptic model architecture have large timing margin in the path, and meet the timing easily without the need for buffer insertion. Since the neurons are pin-capacitance and internal-power dominant designs, reducing the buffer count in M3D design plays an important role in the power savings.

4.4 Full-Chip Results

Figure 6 shows how the smaller individual neuron enabled by architectural optimization impacts on the full-chip footprint, wirelength and static power consumption. Compared to the baseline architecture, full-chip footprint of the shared memory 2nd-order and 1st-order architecture is reduced by 21% and 53%, respectively while keeping the same spacing between the neuron blocks at the top-level placement. However, in shared memory 2nd-order architecture, we observe that this footprint savings does not lead to the wirelength savings because of the routing overhead from the shared memory to the individual neurons. Instead, the shared memory helps to reduce the full-chip internal power by 23%, and this leads to 18% of total power savings. On the other hand, shared memory 1st-order architecture has both wirelength and power savings by 35% and 55%, respectively.

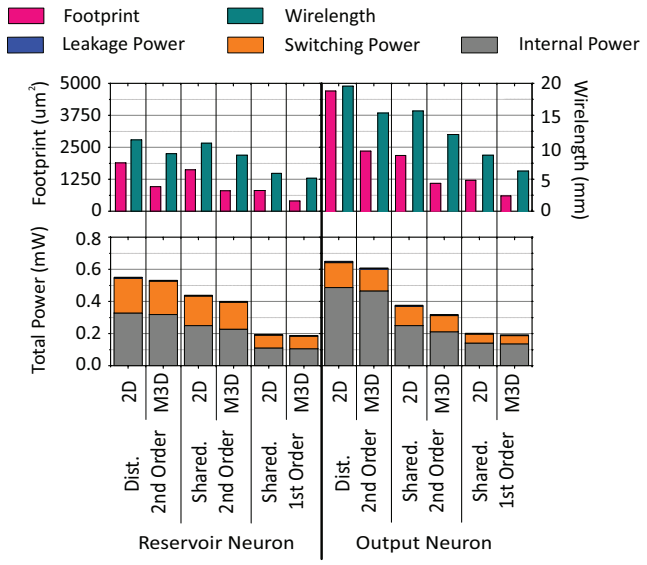

Figure 5: Individual 2D and M3D neuron design results with the architectural combinations of the proposed memory sharing and the synaptic model complexity reduction.

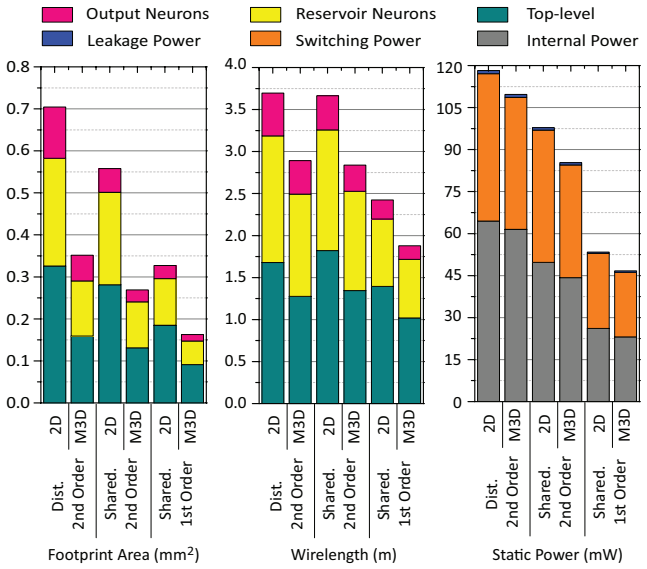

Figure 6: The impact of shared memory and synaptic models on the full-chip design results.

At the top-level, M3D ICs have clear wirelength savings from the 2D counterparts at the same architecture thanks to a large number of inter-neuron connectivities. In every architecture, M3D designs offer more than 24% inter-neuron wirelength savings. However, we observe that this inter-neuron wirelength savings do not guarantee the huge full-chip switching power savings because of the sparse communications between the neurons in the LSM processer. Nonetheless, combining all the power savings from both individual neurons and the top-level, we find that both architectural optimization approaches help to increase the M3D power savings from 9% to 13%.

5 APPLICATION-BASED ANALYSIS

We carry out the real-world application of speech recognition on the implemented LSM neural processors and explore the practical 3D IC benefits. The benchmark is adopted from the TI46 speech corpus [7], which contains read utterances from 16 speakers of the English letters 'A' through 'Z'. Without loss of generality, we select one representative speech for the letter 'R' and evaluate the power dissipation in our designs. The continuous temporal samples are preprocessed by Lyon's ear model [4] and encoded into 78-channel spike trains using the BSA algorithm [6]. The labeled 26 output neurons correspond to the 26 letters in the English alphabet and the output spike trains of the intended output neuron ('R' in this case) is observed as expected.

5.1 Full-Chip Power Breakdown

Figure 7 shows the power consumption results for the reservoir and output training, and classification of the letter 'R' from three-different architecture presented in this work. Thanks to the clock gating implementation, the different activation of reservoir and training unit effectively reduces the total power consumption. In the reservoir training phase, there is no power consumption of the training unit as its clock is completely gated out. During the output training and testing phases, the power of reservoir unit is much smaller than the reservoir training phase because reservoir synaptic weights do not change. Architectural optimization has a great impact on the total power savings. Compared to 2D ICs with distributed memory, 2D shared memory design with 2nd- and 1st-order architecture offer 36% and 57% power savings for reservoir training, and 4% and 27% for output training, and 7% and 38% for testing, respectively.

The major source of these huge power savings are derived from the individual reservoir neuron optimization. Regarding the M3D power savings, we find that M3D designs always reduce the top-level power consumption by more than 20%. However, as a part of the overall bio-inspired computation models, the recurrent SNN inherently operated with sparse firing activities, therefore power savings at the top-level inter-neuron communications have been generally consistent and small. Another benefit from M3D is the output neuron power savings. We observe that the training unit have a maximum of 12% power savings in M3D compared to the 2D counterpart, and this leads to clear power savings in M3D for output training and actual classification.

5.2 Power-Performance-Area-Accuracy Benefit

The energy dissipation is dependent on the power as well as the number of clock cycles of operation. Although the shared memory architecture offers huge footprint and power savings, the shared reservoir memory requires additional clock latency to access compared to the flip-flops in the distributed reservoir weight storage. The design with 1st-order synaptic model also largely saves the power and footprint, but this hurts the classification accuracy from

Figure 7: Vector-based power consumption analysis in different operation steps

92.3% to 91.9%. Therefore, we compare the final power-performance-area-accuracy (PPAA) benefits of the design and architectural co-optimization in LSM neuromorphic processor to measure the trade-off among different design criteria.

In general, the overall spike density is roughly the same over various samples. Therefore, the average power remains the same and we use the power consumption values for each phase from Section 5.1. To get good learning performance over the entire benchmark, 25 epochs of reservoir training and 250 epochs of output training are conducted and these numbers of iterations are taken into account when calculating the total energy consumption.

Targeting 1GHz clock operation, Table 1 summarizes the overall energy savings for 2D ICs and 3D IC LSM neuromorphic processor based on the three-different architectures, with two-different design approaches, respectively. Although the reservoir training energy is actually large in shared memory architecture, it has little impact on the total energy dissipation considering its small number of training iterations than the output training. Also, the power and footprint savings are significantly large over the accuracy degradation when using 1st-order synaptic model. This implies that the power and footprint savings from our co-optimization approaches are well preserved in the energy consumptions for the speech recognition. On average, for the LSM neural processor, M3D IC design offers up to 19% less energy consumption than its 2D IC counterparts for training and inference of a speech sample. Overall, we observe a 70% PPAA benefit from using design and architectural co-optimization compared to the 2D baseline design.

6 CONCLUSION

In this work, we implemented monolithic 3D (M3D) IC design for an LSM-based neuromorphic processor and devised various design and architectural co-optimizations to minimize the area and the energy consumption in the speech recognition. We presented the impact of shared memory architecture and the synaptic model complexity on the individual neuron and full-chip design. We measured the energy dissipation for speech recognition application with TI46 corpus spoken English speech samples, and achieved up to 70.0% reduction in the power-performance-area-accuracy overhead. This work serves as an important step towards realizing bio-inspired neuromorphic processors utilizing 3D IC design advantages.

Table 1: Power × Operation Time Period × Silicon Area ÷ Accuracy (PPAA) benefits of design and architectural co-optimization proposed in this work.

	Distributed 2nd-order		Shared 2nd-order		Shared 1st-order	
	2D	M3D	2D	M3D	2D	M3D
Silicon Area (mm^2)	0.070	0.070	0.056	0.054	0.033	0.033
Res. Tr. Period (ms)	1.35		3.42			
Res. Tr. Power(mW)	87.76	76.93	56.39	53.68	37.84	35.32
Res. Tr. Energy(mJ)	0.119	0.104	0.193	0.184	0.129	0.121
Out. Tr. Period (ms)	109.40		109.41			
Out. Tr. Power(mW)	35.92	33.70	34.46	28.70	26.17	23.28
Out. Tr. Energy(mJ)	3.929	3.687	3.770	3.140	2.863	2.547
Training Energy (mJ)	4.048	3.791	3.963	3.323	2.993	2.668
Test Period (ms)	0.21		0.24			
Test Power (mW)	46.37	41.85	43.22	36.92	28.85	26.05
Testing Energy (mJ)	0.009	0.008	0.010	0.009	0.007	0.006
Total Energy (mJ)	4.058	3.799	3.973	3.333	2.999	2.674
Accuracy (%)	92.3				91.9	
Normalized PPAA	1	0.93	0.77	0.62	0.34	0.30

7 ACKNOWLEDGEMENT

This research is partially funded by the Laboratory Directed Research & Development program of the Oak Ridge National Laboratory. Also, the authors would like to thank High Performance Research Computing (HPRC) at Texas A&M University for providing computing support.

REFERENCES

[1] Y. Jin, Y. Liu, and P. Li. SSO-LSM: A Sparse and Self-Organizing architecture for Liquid State Machine based neural processors. In *Nanoscale Architectures (NANOARCH), 2016 IEEE/ACM International Symposium on*, pages 55–60. IEEE, 2016.

[2] Y. Liu, Y. Jin, and P. Li. Exploring sparsity of firing activities and clock gating for energy-efficient recurrent spiking neural processors. In *Low Power Electronics and Design (ISLPED, 2017 IEEE/ACM International Symposium on*, pages 1–6. IEEE, 2017.

[3] M. LukošEvičIus and H. Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, 2009.

[4] R. F. Lyon. A computational model of filtering, detection, and compression in the cochlea. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'82.*, volume 7, pages 1282–1285. IEEE, 1982.

[5] S. Panth, K. Samadi, Y. Du, and S. K. Lim. Design and CAD Methodologies for Low Power Gate-Level Monolithic 3D ICs. In *ISLPED*, pages 171–176, Aug 2014.

[6] B. Schrauwen and J. Van Campenhout. BSA, a fast and accurate spike train encoding scheme. In *Proceedings of the International Joint Conference on Neural Networks*, volume 4, pages 2825–2830. IEEE Piscataway, NJ, 2003.

[7] TI46. The TI46 Speech Corpus. http://catalog.ldc.upenn.edu/LDC93S9.

[8] Q. Wang, Y. Li, and P. Li. Liquid state machine based pattern recognition on FPGA with firing-activity dependent power gating and approximate computing. In *International Symposium of Circuits and Systems (ISCAS), 2016 IEEE*, pages 361–364. IEEE, 2016.

[9] Y. Zhang, P. Li, Y. Jin, and Y. Choe. A Digital Liquid State Machine With Biologically Inspired Learning and Its Application to Speech Recognition. *IEEE transactions on neural networks and learning systems*, 26(11):2635–2649, 2015.