This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

Power-aware Neuromorphic Architecture with Partial Voltage Scaling 3D Stacking Synaptic Memory

Ngo-Doanh Nguyen, *Member, IEEE,* Akram Ben Ahmed, *Member, IEEE,*
Abderazek Ben Abdallah, *Senior Member, IEEE,* and Khanh N. Dang, *Member, IEEE,*

***Abstract*— The combination of Neuromorphic Computing and 3D Integrated Circuits - the 3D stacking neuromorphic system can be the most advanced architecture that inherits the benefits of both computing and interconnect paradigms. However, simply shifting to the third dimension cannot exploit the 3D structure and also end up with a low yield rate issue. Therefore, in this paper, we propose a methodology to design 3D stacking synaptic memory for power-efficient operations and yield rate improvement of Neuromorphic Systems (NCs). In this proposed methodology, the synaptic weights are stacked on top of the processing elements, and these weights are split into multiple subsets placed in different layers. Furthermore, with the support of 3D technology, the supply voltage of each layer can be controlled independently which leads to power reduction by scaling down or turning off the supply voltage of the memory layer(s) containing the Least Significant Bits (LSBs) while maintaining acceptable accuracy. On top of that, this work also proposes a methodology to deal with the low-yield rate issue by treating the defective memory cells as noises. In our evaluation with the CMOS 45nm technology, the energy per synaptic operation for MNIST classification, when undervolting two upper memory layers (from 1.1V to 0.8V), reduces by 21.62% while the accuracy only reduces sightly by 0.51%. This energy reduction increases to 66.77% with 6.58% accuracy loss when our system uses both power-gating and undervolting for all memory layers. Furthermore, the system can also improve the yield rate by 0.18% or 12.4% while suffering 0.38% or 1.7% of accuracy loss, respectively.**

***Index Terms*—Spiking Neural Networks, Neuromorphic System, 3D-IC-based stacking memory, Low-power, Voltage Scaling, Power-gating**

I. Introduction

SPIKING Neural Network (SNN) is a well-known Artificial Intelligence (AI) model for low-power solutions because it has lighter weight inferences compared to other neural network models' ones [1]–[3]. SNNs, which mimic the activities of the biological brain, transfer information under trains of spikes that are spatial and temporal sparse [4]. Simultaneously, the computations inside of SNNs are rather simplified from the biological models, especially with Integrate-and-Fire-like models [5], which are easier to implement into neuromorphic hardware. As a result, it allows the implementation to significantly reduce the area cost and energy consumption, which benefits the vast majority of resource-intensity and power-hungry applications such as Internet-of-Things (IoTs) or wearable devices [6].

Moreover, the power consumption of NC systems can further break down into the memory and the processing elements (PEs) and the former takes more attention in order to reduce overall power. It is because the memory usually consumes most of the power in hardware systems, and it is about 50-75% of the total power in NC systems [7]. The reason is that neural network models, in general, require a large number of weight computations to achieve high accuracy and those weights are transferred back and forth between memory and PEs with limited bandwidth capability under Von-Neumann-based architectures. There are currently several design approaches to solve this problem. The first solution is to perform the computations near the memory [8]–[10], or better, to merge PEs and memory, which is so-called **In-Memory Computing** [11]–[15], in order to reduce data movements and terminate delivery power. However, the drawback of this approach is the technology's scalability and reliability degradation over time. Another approach is to reduce the supply voltage of memory, as known as **voltage scaling technique**, to gain the energy-per-access reduction, which saves a large number of total energy consumption [7], [16]–[19]. The disadvantage of this method is that the hardware still requires power to transfer data and has a long transmission distance in large-scale systems since the memory weight takes up the major part of the hardware. The reason is that the previous works are conducted under only two dimensions, where memory segments and computing segments are in the same layer. However, if the voltage scaling technique is applied to 3D-IC-based architecture, which has memory on logic, the data movement problem is expectedly solved by the natural characteristics of the 3D design. Another disadvantage of these works is the great reduction of logic correctness when the supply voltage is near the circuit's threshold voltage [19]–[22]. It is because all calculation bits are put into the same voltage domain. Hence, the meaningful bits are affected by noise caused at the near-threshold voltage. Therefore, our main motivation is isolating the meaningful active bits and the inactive bits using 3D-IC technology to solve these issues.

Although bringing multiple benefits such as the low footprint or energy efficiency, one of the most critical issues of 3D-IC-based architectures is their reliability, especially during the manufacturing phase. Stacking 3D-IC chips usually have

Ngo-Doanh Nguyen, Abderazek Ben Abdallah, and Khanh N. Dang are with the Graduate School of Computer Science and Engineering, The University of Aizu, Aizu-Wakamatsu 965-8580, Fukushima, Japan. Akram Ben Ahmed is with Digital Architecture Research Center, National Institute of Advanced Industrial Sciences and Technology, 2-3-26 Aomi, Koto-ku, Tokyo, Japan.
E-mail: {m5262108,benab,khanh}@u-aizu.ac.jp; akram.benahmed@aist.go.jp

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

low yield rates, since the silicon layers are not tested before bonding. Therefore, we may end up stacking defective layers on top of the non-defective ones which generally destroy the correctness of the chip. As a result, simply stacking multiple layers to have 3D-IC chips is too naive due to the low yield rate and costly manufacturing process.

Starting from the facts mentioned above, in this paper, we propose a novel low-power neuromorphic architecture with *in-situ* quantization and undervolting 3D stacking synaptic memory. With 3D-IC technology, a memory word can be split into small subsets, and each subset is stacked in a separate layer on top of the computing segment. These stacking layers, which are called the memory layers hereafter, contain and represent only the synaptic weights of the SNN model. In addition, each memory layer can represent one, two, or multiple-bit precisions of synaptic weights. Hence, the NC system has the ability to reduce the supply voltage of memory layers containing the Least Significant Bits (LSBs) or completely turn off the power supply to save overall energy consumption while maintaining accuracy. Moreover, our proposed architecture is able to adapt to multiple different power scenarios by using *in-situ* dynamic quantization.

Furthermore, by splitting and stacking in this mechanism, we can help the system deal with the low yield rate by accepting defective memory at the top layers. Since defects in memory usually lead to stuck-at or bridging faults, they are treated as noises in our architecture. While defects in the computing logic could lead to incorrect results, they must be well-tested. Here, we can only need to ensure the correctness of the logic layer and the MSBs layers. Defects in the LSBs can be ignored which leads to a much better yield rate and reduced manufacturing cost.

The main contributions of this paper are summarized as follows:

- A novel low-power methodology to implement neuromorphic architectures with 3D stacking synaptic memory, where the memory word is split into multiple subsets and placed in separate layers.
- With 3D-IC technologies, the under-voltage technique is applied separately to each memory layer in 3D architecture based on the significant bits of synaptic weights. It aims to reduce overall power consumption with acceptable accuracy.
- Consequently, an *in situ* dynamic quantization for synaptic weight is implemented in this work as the next level of undervolting. The weights are configured in the design phase and stay unchanged during inference. Therefore, the bit precision of synaptic weights is dynamically modified by removing completely the supply voltage of memory layer(s).
- A novel stacking memory mechanism which helps improve the yield rates by accepting imperfection at the top layers.

The rest of this paper is organized as follows. Section II presents the related works. Section III introduces the methodology for 3D-IC implementation. The hardware architecture is shown in section IV. In Section V, the performance and power consumption of our spiking computing core in each supply voltage scenario is evaluated. In Section VI, we discuss the challenges and issues of this work and potential solutions. Finally, we end the paper with conclusions in Section VII.

II. Related Works

A. Neuromorphic Systems for low-power applications

To exploit the potential of SNNs as low-power AI solutions, numerous academic literature and research works have been conducted under the name of Neuromorphic Computing (NC) systems [1]–[3], [10], [23]–[28]. Especially, the power consumption of these NC systems is minimized optimally by implementing specialized hardware, such as Field-Programmable Gate Arrays (FPGAs) [25], [26] or Application-Specific Integrated Circuits (ASICs) [23], [24]. In practice, NC systems have three main implemented approaches, which are (1) **2D-IC based Analog Mixed-signal Hardware Designs** [2], [3], (2) **2D-IC based Digital Hardware Designs** [1], [10], and (3) **3D-IC based Hardware Design** [27], [28]. Analog mixed-signal systems are able to emulate the activities of our biological brains accurately with low power consumption; however, this type of system is difficult to scale in different fabrication technologies. This is because the activities of analog circuits tend to extremely vary and require careful calibration in different technology nodes. The digital approach, on the other hand, is robust and scalable and it is easier to prototype and debug systems compared to the analog mixed-signal ones. However, this approach consumes the most portion of power in the aforementioned three approaches, if the same technology node is used [29]. Meanwhile, the 3D-IC-based approach brings great benefits in terms of power consumption, hardware footprint, and signal transmission. The disadvantage of this approach is that it lacks solid architecture at the system level, which brings out the potential of 3D-IC for SNN applications. This premise leads to the necessity of designing a low-power and high-efficiency 3D-IC-based neuromorphic architecture.

At the end of Moore's Law for a single monolithic die, hardware architectures, especially memory architectures, begin to transform into 3D packages or **3D-IC based Hardware Design**. The SNN architecture is not an exception [30]. For example, the Loihi-2 architecture [28] currently has appealed to support 3D multi-chip scaling, which begins to switch to the next generation of hardware architectures. NeuroSIM [27], a 3D neuromorphic system, is integrated with two-layer memristors as the electronic synapses of SNN. It results in reducing half of the hardware area, 1.48$\times$ times in terms of power consumption and 2.58$\times$ times in terms of latency compared to the traditional 2D one-layer configuration. MigSpike [31], a 3D-IC-based SNN architecture used for fault tolerance, reduces the migration cost from remapping in NoC by 10.19$\times$ compared to 2D approaches. Therefore, 3D-ICs promise many significant benefits compared to the two above approaches, such as hardware footprint, cost, and power consumption. As a premise, a 3D SNN system would have even greater leverage for reducing power consumption and hardware area.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

B. Power-Optimal Memory for low-power AI-oriented applications

The **voltage scaling technique** is one of the famous techniques that are widely used for low-power systems. In fact, previous works proved that by applying the under-voltage technique power consumption related to memory could be greatly reduced. For example, Salami *et al.* [18] reduces power consumption by 39% on FPGA on-chip memories, Leng *et al.* [32] saves 20% of power in GPUs, and power consumption of DRAMs in [33] is dropped by 16%. In addition, Minerva [34] lowers the supply voltages of SRAMs to save a total of 2.7x power consumption. In order to accomplish the voltage transformation, the system is required to have an off-chip voltage regulator (VR) with a power switching technique [35], [36] or an on-chip one (i.e.: low-dropout VR [37], [38], switched capacitor VR [39], [40]). Moreover, the under-voltage technique could also be applied to internal components of FPGAs [19] or HBMs (High Bandwidth Memory) [41] to gain around 3× and 2.3× power efficiency, respectively. However, due to the supply voltage reduction, the noise margin of a memory cell is also reduced, which leads to an increase in the probability of errors such as read stability failure, write stability failure, or access time failure [42]. As a result, such small errors could lead to a huge impact on the accuracy of conventional 2D neural network architectures [19]. It is because there is a chance that the MSBs of weights are affected by reducing the supply voltages of SRAMs. However, with 3D technology, the weights can be split into multiple subsets placed in separate layers with isolated supply voltage, which is able to protect the memory layers containing MSBs and reduce the supply voltage of memory layers containing LSBs.

Another way to improve the power efficiency of memory is to apply new technologies to restructure the memory cells such as **In-Memory Computing** (IMC), and **3D stacking memory**. For instance, the emergence of IMC methods can be divided into analog IMC [43]–[45] and digital IMC [46]–[48]. Analog IMC may not be suitable for high-precision applications such AI because as it has the disadvantage of low conversion accuracy limited by the low-cost analog-to-digital converters (ADCs), while digital IMC has the advantage of high computational accuracy. Moreover, the analog IMC is also vulnerable to noise caused by temperature, sneak currents, and many other sources of variations [49]. On the other hand, although the digital IMC has robustness and precision, it consumes more power compared to the analog IMC [50]. For the 3D stacking memory in chips, there are several proposed works [51], [52] to shorten the data movements, which reduces power consumption. With a high bandwidth and a large capacity, 3D stacking of SRAMs has drawn attention for being a large cache in CPUs and a large memory in DNN inference accelerators [53], [54]. The data communication between 3D layers can be wired integration using through-silicon vias (TSVs) [51], [52] or a wireless integration using inductive coupling known as ThruChip Interface (TCI) [55]. However, despite these great benefits of 3D stacking technology, the challenge of this approach is that it has a low yield rate and low reliability. In this paper, to tackle one of these problems, we propose a 3D architecture, which is able to improve the yield rate, by accepting defective layers while maintaining tolerable accuracy.

TABLE I
DIFFERENCE BETWEEN BIT FLIPPING POSITIONS

Value	Original	Flipped bit position			
		MSB	3^{rd} bit	5^{th} bit	LSB
Binary	10101100	00101100	10001100	10100100	10101101
Float	-0.34375	0.34375	-0.09375	-0.28125	-0.3515625
Diff. (%)	0 (0%)	+0.6875 (+200%)	+0.25 (+72.727%)	+0.0625 (+18.182%)	+0.0078125 (+2.273%)

III. Methodology of 3D Stacking Synaptic Memory

Before presenting the implemented architecture, in this section, we would like to illustrate the methodology of 3D Stacking Synaptic Memory. To the best of our knowledge, this is the first work that utilizes both voltage scaling and power gating partially for memory without a significant drop in accuracy. It is because the prior works [19]–[22] put all bits into the same voltage domain. As a result, the noise caused by dropping supply voltage to the subthreshold affects the meaningful active bits or MSBs. However, by taking advantage of 3D-IC and multiple power rails through TSV, we can isolate the meaningful active bits and the inactive bits into different layers. Hence, we can reduce the supply voltage below the subthreshold or completely power-gate the inactive bits without greatly affecting the final accuracy, unlike the prior works. Another difference between our work and the prior dynamic-voltage-scaling 3D-IC architecture [56] is that we also utilize the power-gating technique for the memory layers. Here, assuming that the synaptic weights consist of n-bit and are in fixed point and quantized from the floating point in the case of off-chip training. These bit configurations are unchanged after manufacturing.

A. Different Important Levels of Bits

Conventionally, all bits are treated as same as each other regardless of their position in the weight. However, we can simply realize that in terms of value, they are definitely not the same. Although spike neural networks application can be noise resilient, flipping bits due to undervolting or power gating still has different impacts on different positions of the bit. Assuming the weight of $n = 8$ bit: $W[0:7] = 10101100$ with one signed bit and seven bits fractional, the differences in values are shown in Table I. In summary, flipping bit in the LSBs gives a lesser impact on the value of the weight itself.

Motivated by this, this work presents a method to allow power-reduction targeting LSBs. However, we can quickly notice that power-gating or voltage scaling for LSBs is mostly not possible with the native 2D memory architecture. On the other hand, the 3D architecture is different. It provides different power nets to each stacking layer. Therefore, the voltage-scaling and power-gating techniques could be applied to the memory layers consisting of LSBs to reduce power consumption while maintaining acceptable accuracy.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

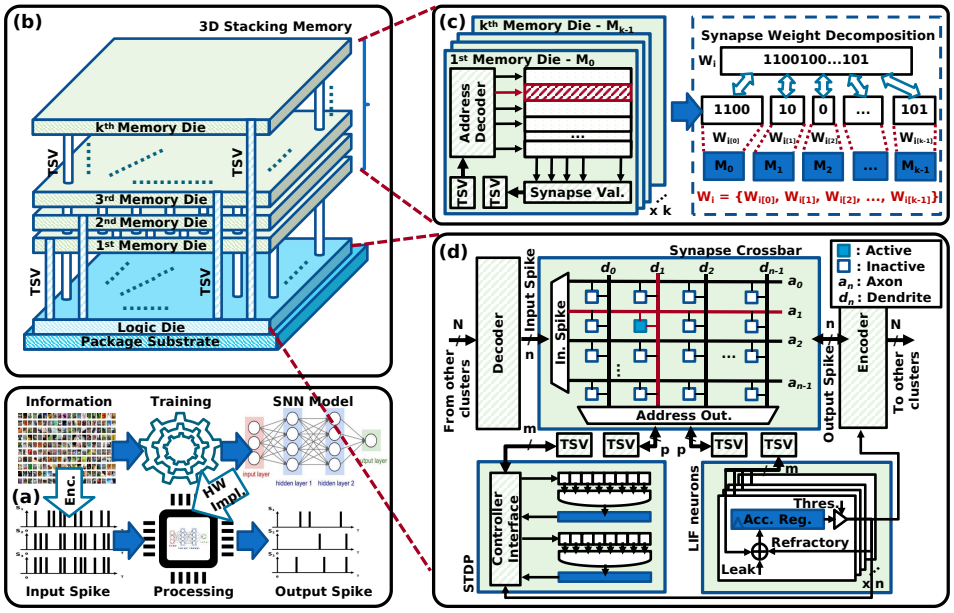

Fig. 1. The overview hardware architecture with 3D stacking memory for the proposal methodology. (a) The overview of SNN hardware implementation. (b) The overview architecture with n stacking memory layers. (c) The synapse weight decomposition with 3D stacking memory layers. (d) The hardware architecture of each neuromorphic computing core.

B. Dynamic Low-power Memory Structure

In this proposed methodology, the n-bit weights are distributed into different memory layers stacked on each other. It could be also treated as a set of subset bits $\{m_0, m_1, ...m_{M-1}\}$ where m_i is the i^{th} subset of synaptic weights and M is the number of subsets. In this case, m_0 contains the most significant bits, and m_{M-1} contains the least significant bits. The number of bits in each subset could be different and can be modified during the design phase. The strategy for *in-situ* low-power structure is acquired by the three following modes (I, II, III), which represent the corresponding low-power techniques. We define those three modes for easier mentioning in the explanation and evaluation.

- **Normal power mode:** The neuromorphic systems operates without power-gating or voltage-scaling.
- **Low-power mode I:** Voltage-scaling is applied to the neuromorphic systems.
- **Low-power mode II:** Power-gating is applied to the neuromorphic systems.
- **Low-power mode III:** Both voltage-scaling and power-gating are applied to the neuromorphic systems.

If the system is currently at low-power mode and the **normal power mode** is detected, the system gradually restores the supply voltage to every inactive memory layer. The order will be bottom-up, which starts from MSBs among all inactive bits. One of the drawbacks of splitting memory weights is having smaller memory cells which lead to lower density and high power consumption. However, we could solve this issue by merging multiple adjacent weights into a single memory cell [10], [23]. Moreover, we utilize multiple power rails for every memory layer to change their power supply. Hence, it is the hardware overhead compared to the traditional voltage scaling. However, our hardware architecture is implemented in 3D and every memory layer has the same hardware area. As a result, compared to the implementation in 2D architecture, there is no overhead in hardware footprint. Another concern of this method is that the number of combinations for configuring and deciding low-power mode for each layer is huge. As a result, a standalone optimization algorithm is required to decide the best operating mode in a specific situation. In this paper, the decision is based on our experimental experience.

TABLE II
NOTATIONS AND PARAMETERS DEFINITION.

Symbol	Definition	Symbol	Definition
L	# LIF Modules	M	# Stacking Memory Layers
n	# Bit of Weights	m_i	i^{th} Memory Layer
t	# Turned-off Bits	D	# Die
W	Synaptic Weights	P	Power Consumption
BER	Bit Error Rate	f_{sw}	Switching Frequency
V_{DD}	Supply Voltage	Y	Yield Rate
SNM	Signal Noise Margin	I	Current of Circuits
T	# Accepted Layers	V_T	Voltage Threshold
α	Ratio between logic components and memory	V_{m_i}	Supply Voltage for i^{th} Memory Layer
C, K, N	Technology Dependent Parameters for Power Dissipation	k, r, q, β, V_s, V_r,	Technology Dependent Parameters for SNM

IV. Implementation Hardware Architecture

The overview hardware architecture for our proposed methodology is shown in Fig. 1. In detail, Fig. 1(a) shows our software-hardware design methodology with the abstract hardware architecture, shown in Fig. 1(b), where we split the logic components and the memory components into separated layers. In addition, Fig. 1(c) illustrates our synaptic weights' arrangement in each memory layer while Fig. 1(d) presents the

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

block diagram of the logic components in our neuromorphic system. For ease of understanding, the hardware architecture is illustrated as a neuromorphic system consisting of $L = 16$ Leaky Integrate-and-Fire (LIF) neurons with four synaptic memory layers stacking on top. However, the number of neurons and stacking memory layers could be configured during the design phase. In addition, the output of one neuron could either be transferred to the neurons in the same cluster or in other clusters. On the other hand, the input of one neuron signals the crossbar to extract the corresponding synaptic weight from the memory layers via TSV for LIF computations. The memory layers are stacked on top of the logic computational layer and each memory layer contains a subset of synaptic weights. Those synaptic weights could be either updated from a broadcast message via the address decoder or from the internal Spike-Time Dependent Plasticity (STDP) with self-learning and self-updating functions. Moreover, by dividing the synaptic weights into subsets and placing them on different memory layers, our hardware is able to offer the *in-situ* dynamic quantization for synaptic weights with voltage-scaling and power-gating schemes. These techniques are famous and influential to reduce significantly power consumption in low-power systems. In the following subsections, we explain the strategy for *in-situ* dynamic quantization with voltage-scaling and power-gating and its insight. The parameters, which we use to ease the explanation, are introduced in Table II.

A. 3D Stacking Synaptic Memory

For a better explanation, the sample hardware for the proposed methodology uses the 8-bit synaptic weights. In addition, it has four memory layers ($M = 4$) stacked on top of PEs and each memory layer contains a 2-bit subset of 8-bit synaptic weights. The LSBs are placed on the top layer and the MSBs on the bottom layer ($\{m_0, m_1, m_2, m_3\} = W[0:7]$). As a result, when applying the voltage-scaling technique or power-gating one to the top memory layer, the power consumption could be reduced from the original power consumption while suffering a small fractional loss in accuracy (flip LSBs). It is only available because of the bit-loss resilience of SNNs [57], where other neural network systems usually drop their accuracy sharply when reducing bit-operation on-fly [19]. In conclusion, there are three benefits to the hardware architecture. First, it takes advantage of 3D implementation, which reduces the transferring distance between memory and PEs. As a result, the power for data transferring can be reduced. Second, the bit-weight quantization could be dynamically activated during the inference without any interruption in system operations. Last, the hardware can partially apply the voltage-scaling technique and the power-gating technique to the memory layer(s), which keeps the MSBs unchanged and only affects LSBs. In addition, LSBs can be reloaded during system operations because the supply voltage is dynamically controlled.

B. Power Efficiency with 3D stacking memory

The power consumption of our hardware is similar to other conventional neural network architectures, which is the sum of power consumption by memory storage P_{mem} and power consumption by PEs P_{pe}. In practice, the power consumption from memory is usually dominant, which is about 75% of the total power [7]. It is because the neural network models often require millions of weights to acquire high accuracy and those weights are transferred back and forth in long-distance between memory and PEs. This leads to the huge size of memory, which prolongs the transferring distance and requires more power to transfer those weights in the conventional 2D systems. However, as mentioned above, the 3D design of memory-on-logic brings the two most benefits: distance reduction, and footprint reduction, for neural network models in general, and the SNNs in particular.

On the other hand, the power consumption of CMOS-based circuits could be further expressed as P_{total}, a sum of two components, the dynamic power P_{dyn} (or active power) and the leakage power P_{leak} (or static power).

$$P_{total} = P_{leak} + P_{dyn} \tag{1}$$

Furthermore, those two power consumptions are mathematically represented by the following equations:

$$P_{dyn} = C \times f_{sw} \times V_{DD}^2 \tag{2}$$

$$P_{leak} = K \times N \times I_{leak} \times V_{DD} \tag{3}$$

These equations clearly show that power consumption could be significantly reduced by adjusting the supply voltage. In the case of dynamic power, Eq. 2 expresses the power reduction in quadratic-fold when scaling down the supply voltage. Moreover, the dynamic power consumption could be further reduced with the power-gating technique, which completely removes the supply voltage. It can only happen in our 3D hardware architecture because of the multiple-layer memory and the noise resilience of SNNs. Likewise, the leakage power consumption is also reduced linearly, as shown in Eq. 3, by implementing the same techniques. Each technique applied to the hardware architecture is explained in the following subsections.

C. Partial Voltage-scaling for 3D Stacking Synaptic Memory

In this subsection, the power efficiency and the Bit Error Rate (BER) of voltage scaling for stacking synaptic memories in our hardware are analyzed. In addition, since the synaptic memory of our hardware is implemented using SRAM models, the analysis will focus on the BER of SRAM cells. The BER of an SRAM cell is the probability that the Static Noise Margin (SNM) appears to be close to zero [58], [59]. Assuming that SNM has a normal distribution, the BER of an SRAM cell is analytically expressed by the following equation:

$$BER = f(SNM) = \frac{1}{\sqrt{2\pi\sigma_{SNM}}} \exp - \frac{(SNM - \mu_{SNM})^2}{2\sigma_{SNM}^2} \tag{4}$$

where σ_{SNM} is the standard deviation of SNM and μ_{SNM} is the mean value of SNM. In practice, these two values vary from one technology to another. It is because SNM depends

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

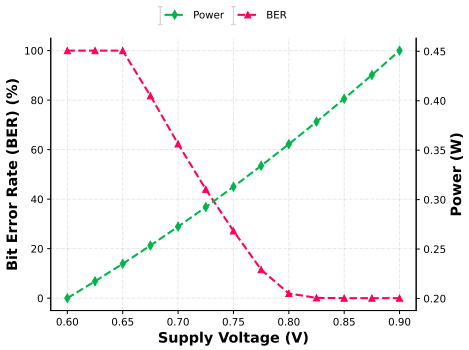

Fig. 2. The bit error rate vs. power consumption of memory (45-nm 6T SRAM cell) at near-threshold voltage.

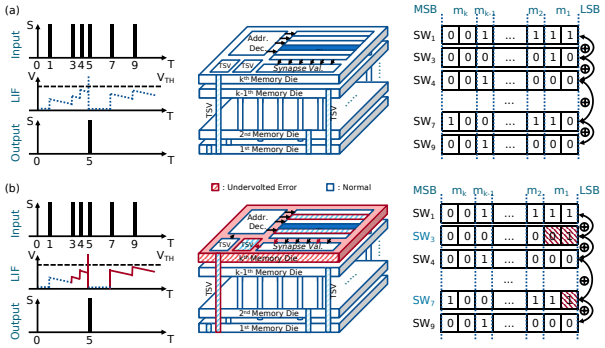

Fig. 3. Example of 8-bit synaptic weights' operation with undervolting memory layer(s). (a) The operation of our hardware under normal conditions. (b) The operation of our hardware with undervolting for the top memory layer.

on the threshold voltage V_T, the supply voltage V_{DD}, and the ratio β, which vary depending on the doping profile, the manufacturing process, and the transistor sizing [60]. Fig. 2 shows the BER of 45-nm 6T SRAM with multiple supply voltages near the threshold region. According to Seevinck *et al.* [60], the SNM is estimably calculated by the following equation:

$$SNM = V_T - \left(\frac{1}{k+1}\right)\left[\frac{V_{DD} - \frac{2r+1}{r+1}V_T}{1+\frac{r}{k(r+1)}} - \frac{V_{DD}-2V_T}{1+k\frac{r}{q}+\sqrt{\frac{r}{q}\left(1+2k+\frac{r}{q}k^2\right)}}\right] \quad (5)$$

where $r = \beta_p/\beta_a$ is the ratio of β between pull-up transistors and access transistors and $q = \beta_d/\beta_a$ is the ratio of β between pull-down transistors and access transistors. k is calculated by the following Eq. 6.

$$k = \left(\frac{r}{r+1}\right)\left(\sqrt{\frac{r+1}{r+1-V_s^2/V_r^2}} - 1\right) \quad (6)$$

where $V_s = V_{DD} - V_T$ and $V_r = V_s - \left(\frac{r}{r+1}\right)V_T$ [60]. As a result, the BER of an SRAM cell from a specific technology can be approximately obtained. In practice, Reviriego *et al.* [58] evaluated the BER of SRAM cells approximately around 3.99×10^{-2} and 2.29×10^{-3} at the half of normal supply voltage, $0.4V$, at 16nm CMOS and FinFET technologies, respectively. This BER usually accumulates over time which steadily causes the collapse of memory. It is because the conventional architecture does not support partial voltage-scaling or power-gating the memory. However, our hardware architecture takes advantage of 3D design to separate the MSBs and LSBs of synaptic weights. Since the MSBs are kept at a different layer with full-voltage protection, the collapse of all memories does not happen. As a result, with the noise resilience of SNNs, the accuracy of our hardware only suffers a fraction of loss, yet its energy efficiency is able to gain up to twice or threefold depending on the dropping voltage.

In the examplary model shown in Fig. 1, our hardware has $M = 4$ memory layers, $\{m_0, m_1, m_2, m_3\}$. Therefore, the total power consumption of the memory P_{mem} could be expressed as the following equation:

$$P_{mem} = \sum_{i=0}^{M-1} P_{m_i} \quad (7)$$

where P_{m_i} represents the power consumption of the i^{th} memory layer. In addition, each memory layer has its own dynamic power consumption and leakage power consumption, as shown in Eq. 2 and Eq. 3, respectively. Assuming that the supply voltages in all four memory layers are the same voltage, V_{DD}, in the **normal power mode**. With the voltage-scaling, those four memory layers then have their specific supply voltages, $\{V_{m_0}, V_{m_1}, V_{m_2}, V_{m_3}\}$. Combining with Eq. 1, the power consumption reduction of memory using undervolting could be expressed as the following equation:

$$P'_{mem} = \sum_{i=0}^{M-1} \left(C_i \times f_{sw_i} \times V_{m_i}^2 + K_i \times N_i \times I_{leak_i} \times V_{m_i}\right) \quad (8)$$

where P'_{mem} is the power consumption of all four memory layers when the undervolting is implemented. As a result, the ratio between the power consumption of the undervolting hardware and the power consumption of the normal hardware is approximately equal to the following equation:

$$\frac{P'_{mem}}{P_{mem}} = \frac{\sum_{i=0}^{M-1}\left(C_i \times f_{sw_i} \times V_{m_i}^2 + K_i \times N_i \times I_{leak_i} \times V_{m_i}\right)}{C \times f_{sw} \times V_{DD}^2 + K \times N \times I_{leak} \times V_{DD}} \quad (9)$$

To illustrate the power mode I, Fig. 3 shows our hardware with undervolting only for the top memory layers and provides the normal supply voltage for the remaining memory layers. In detail, Fig. 3(a) shows the normal LIF operation without voltage scaling, and Fig. 3(b) demonstrates the LIF operations with the effect of voltage scaling at near-threshold voltage. Here, the red-square areas are the flip-bits due to undervolting. As a result, the flip-bit fault only causes the error in LSBs of synaptic weights and the output spike will not be affected. We first assume that the supply voltage of the top memory layers is reduced by half and there are four stacked memory layers. The total C is $6nF$, $K = 1$, the total number of transistors

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

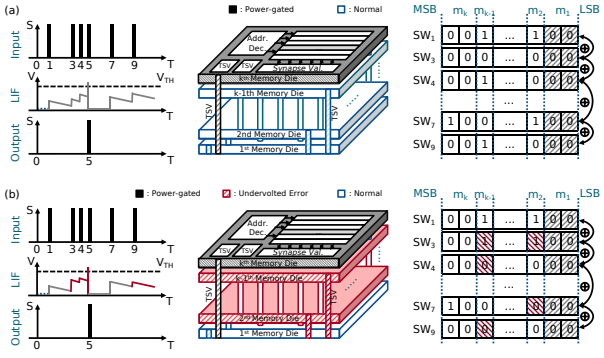

Fig. 4. Example of 8-bit synaptic weights' operation with undervolting and power-gating memory layer(s). (a) The operation of our hardware with power-gating the top memory layer. (b) The operation of our hardware with power-gating the top memory layer and undervolting two memory layers.

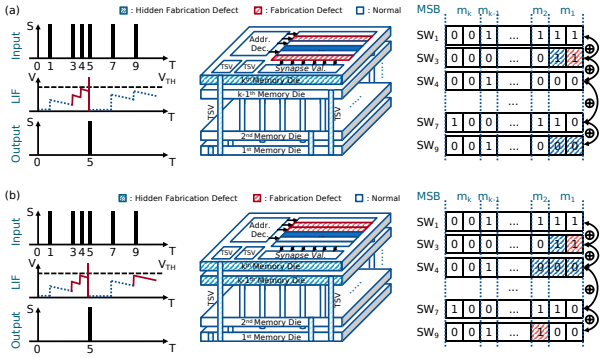

Fig. 5. Example of 8-bit synaptic weights' operation with fabrication defects. (a) The operation of our hardware with the top memory layer defected by fabrication. (b) The operation of our hardware with two upper memory layers defected by fabrication.

is 10^9, the normal voltage supply is $1.1V$, and the leakage current is $I_{leak} = 50pA$. Hence, our hardware, which has a switching frequency of $50MHz$, theoretically could save about 17.92% power consumption on the memories while the accuracy of our hardware drops insignificantly because of the noise resilience of SNNs. The drop in accuracy will be later evaluated in Section V. In practice, it could extend approximately the operating time of edge devices by 20%, which is in a power-hungry situation without changing its neural network model and hardware components. Moreover, the accuracy is only trade-offed by a marginal volume.

D. Power-gating for 3D Stacking Synaptic Memory

With the power-gating, our hardware proceeds the *in situ* synaptic weight quantization by turning the memory layer(s) off if the **low-power mode II** is detected and turning it on if the **normal power mode** is detected. Therefore, the alternation of the total power consumption is from the memory. For example, with the n-bit synaptic memory from the architecture in Fig. 1, we can define the total power consumption of synaptic memories based on Eq. 1.

$$P_{mem} = P_{mem_{leak}} + P_{mem_{dyn}} \tag{10}$$

where $P_{mem_{leak}}$ is the leakage power of synaptic memories and $P_{mem_{dyn}}$ is the power consumption of synaptic memories from switching activities. Assuming that the power supply is divided equally into synaptic memories. Hence, when one or more memory layers consisting of t LSB bits, are turned off, the power consumption of synaptic memories theoretically reduces by t/n.

$$P'_{mem} = \frac{n-t}{n} \times (P_{mem_{leak}} + P_{mem_{dyn}}) \tag{11}$$

This is because all the memories in the layers are unified and have the same switching activities when the input spike event occurs. With $n = 8$ as in Fig. 1, the expected power reductions are 25% and 50%, for $t = 2$ and $t = 4$, respectively. Therefore, for each possible value of t, we can define a power-aware mode. In addition, we can also use the voltage-scaling technique for the non-power-gated memory layer(s) to further decrease the overall power consumption. In this case, the system enters the **low-power mode III**.

Fig. 4 shows the example of both **low-power mode II** and **low-power mode III**. With the power-gated top layer, the LSBs of synaptic weights are treated as zeros. It leads to a slight decrease in the value of synaptic weights but our architecture still receives the correct output spike, as shown in Fig. 4(a). On the other hand, in the **low-power mode III** (Fig. 4(b)), the synaptic weights in undervolted layers are randomly flipped because of the lack of supply voltage. It also leads to a transformation in the output value of the LIF neuron but the output spike is still correct. It is because the memory layer containing MSBs is untouched. However, the number of untouched MSBs also needs to be considered for the correctness of the SNN model. Despite the noise resilience of SNNs, further dropping the power supply out of the remaining memory layers will cause the spiking computing core to collapse, unable to operate correctly. The evaluation section will demonstrate the experimental results for each operating power-aware mode.

E. Improving the yield rate by accepting LSBs layers' defects

As we mentioned earlier in Section I, the low yield rate is one of the most critical issues in stacking 3D-ICs technology. Assuming the yield rate for a single layer (die) is $Y_{1\_layer} < 1.0$, the yield rate of D layers is Y_{D_layers} which is much smaller than Y_{1_layer}. It is because each layer has its own defection and, by stacking multiple layers, the defect probability increases exponentially since we do not know the die quality before stacking. This yield rate can be represented by the following equation:

$$Y_{D\_layers} = \prod_{i=0}^{D-1} Y_i \tag{12}$$

where D is the number of layers and Y_i is the yield rate of the i^{th} layer. For example, assuming that all layers have the same yield rate, $Y_{layer} = 0.9$ and the stacked layer is $D = 4$. Therefore, the actual yield rate of the 3D-stacked chip is reduced to 0.6561 and the defect rate is increased to 0.3439.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

In detail, the defective layer will cause errors in the logic functions of transistors, which are usually the stuck-bit or bridging faults. Without the correctness of logic functions, the fabricated chip cannot work as designed. However, in our architecture, we split the memory and stack them on top of processing elements. As a result, the yield rate of the second layer onward can be categorized generally into two types, which are for the control-logic region in memory, Y_{logic}, and the memory cell region, Y_{mem}.

$$Y_{layer} = Y_{layer_{logic}} \times Y_{layer_{mem}} \tag{13}$$

Moreover, the memory cell region takes the most area in memory. On the other hand, in our architecture, fabrication defects in memories are considered noises, as shown in Fig. 5. The LIF operations with the defects of the top memory layer and the two upper memory layers are presented in Fig. 5(a) and Fig. 5(b), respectively. Assuming that we have stuck-at defects in the memory cells of the top layer(s), the bit values at defected regions always stay at $'0'$ or $'1'$. With the noise resilience of SNNs, the output spike is still correct even with defective synaptic weights. We assume that the defects that appeared in the wafer have a uniform distribution. Therefore, the probability that the defects occur in memory is equal to the ratio of hardware area between logic components and memory components multiplied by the yield rate. Assuming that this ratio is approximately one-ninth ($\alpha = 1/9$) and the total number of layers is $D = 5$. We can have the actual yield rate if we accept defects in $T = 2$ upper memory layers as follows:

$$Y_{D\_layers} \approx \prod_{i=1}^{D-T-1} Y_{layer_i} \prod_{j=D-T}^{D-1} \left[1 - \frac{\alpha}{1+\alpha}(1 - Y_{layer_j})\right] \tag{14}$$

Substituting numbers into the equation, the actual yield rate is $Y_{actual} \approx 0.7145$, not 0.5904, which leads to an improved overall yield rate. Therefore, we can accept the manufacturing defects to improve the overall yield rate while reducing a fraction of accuracy.

V. Evaluation Results

A. Evaluation Methodology

The proposed hardware architecture was implemented in Verilog-HDL, synthesized, and evaluated with commercial CAD tools from Cadence and Synopsys (Cadence Innovus, Synopsys Design Compiler, PrimeTime, Custom Compiler, HSPICE). The physical design of our hardware is implemented with the NANGATE 45-nm library [61] and NCSU FreePDK3D45 TSV [62]. The system memory is 6T SRAM generated from OpenRAM [63] and its BER characteristic, when undervolting is applied, is calculated from Python based on Eq. 4 and is evaluated by HSPICE. In order to evaluate the transformation of power consumption and accuracy, we implemented our hardware as a neuromorphic core with $M = 4$ memory layers stacked on top of $L = 48$ LIF modules. The SNN model embedded into the hardware is configured with a neural network of three layers *(784:48:10)* for the MNIST dataset. We also evaluate the hardware system with the VGG16 model under the CIFAR-10 dataset [64]. Since the hardware design for VGG16 is not available in this work, we estimate the energy consumption via CACTI SRAM's model [65]. The images were encoded into spikes using the rate-coding scheme under the Poisson distribution. In addition, the synaptic weights are trained as $n = 8$-bit values for MNIST, and $n = 16$-bit values for CIFAR-10. They are split equally into four memory layers of the hardware, which is two bits per layer. Please take note that the configurations of the SNN model and our hardware architecture can also be modified into different ones during the design phase.

First, for the **low-power mode I**, we examine the Signal Noise Margin (SNM) of SRAM cells at near-threshold supply voltages to extract the BER or probability of faults according to materials presented in previous works [58]–[60]. The BER is exported through Monte Carlo simulations with PrimeSim HSPICE and mathematical calculation at multiple supply voltages. After that, we insert the faults according to the extracted probabilities into synaptic weights trained from the software model. The position of faults is distributed randomly using the Monte Carlo simulation again with uniform distribution. Because we implement the hardware with four memory layers, the undervolting evaluation is then categorized into four settings. The modified synaptic weights are then loaded into hardware to evaluate the power consumption and the accuracy of the SNN model affected by undervolting.

Second, the transformation of power consumption and accuracy at **low-power mode II** are evaluated. Similar to the **low-power mode I**, the power-gating hardware also has four settings to inspect. However, the accuracy of our hardware is broken when the supply voltage of the third memory layer is turned off. Therefore, in this paper, the evaluation only covers three settings which are: normal setting without power-gating any layers, power-gating one layer, and power-gating two layers. In this case, our hardware treats the bit values of synaptic weights as zero(s) and uses them to perform LIF computations. Similarly, the switching activities of power-gating hardware are then loaded into Synopsys PrimeTime to extract power consumption. Third, the **low-power mode III** are evaluated. Because of the time-consuming simulation, we only pick one case out of all combinations to evaluate the power-accuracy transformation. Finally, we evaluate the hardware complexity and compare our system with other works [1], [10], [23], [24], [66]–[69].

B. Undervolting Hardware (Low-power Mode I)

As shown in Fig. 6, the evaluation of power transformation and accuracy transformation are taken with supply voltages from $0.7V$ to $0.85V$ with downing $0.025V$ per step. Particularly, Fig. 6(a) is the evaluation of accuracy transformation, Fig. 6(b) is for energy transformation, and the BER of our SRAM is shown in Fig. 6(c). According to the NANGATE 45-nm library [61], the voltage threshold of a transistor is around $0.65V$. As a result, we evaluate the transformation from $0.7V$ to $0.85V$ to capture the best affective region of SNM in the 6T SRAM. Here, the bit order of synaptic weights, as mentioned

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

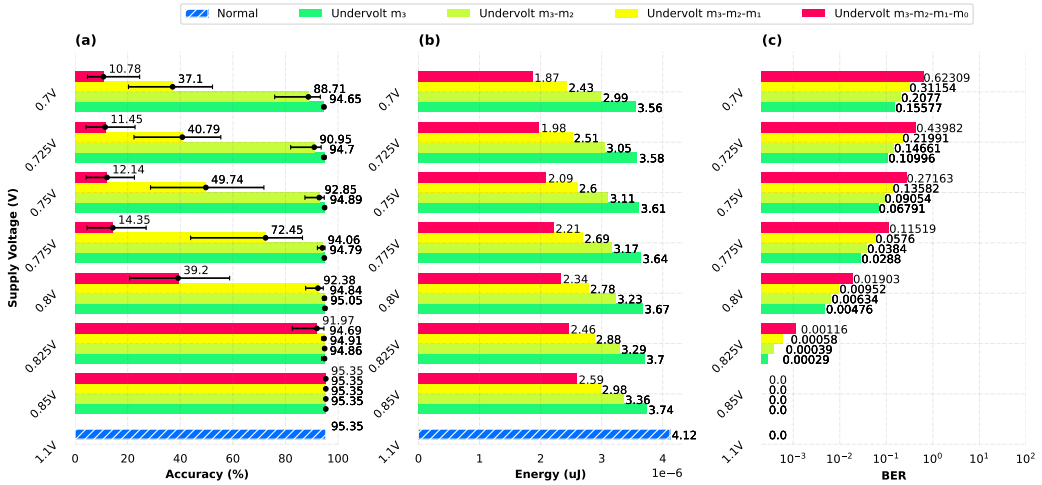

Fig. 6. The transformation of BER and accuracy and energy with undervolting memory layer(s). (a) Accuracy when undervolting each combination of memory layer(s). (b) Energy when undervolting each combination of memory layer(s). (c) BER when undervolting each combination of memory layer(s).

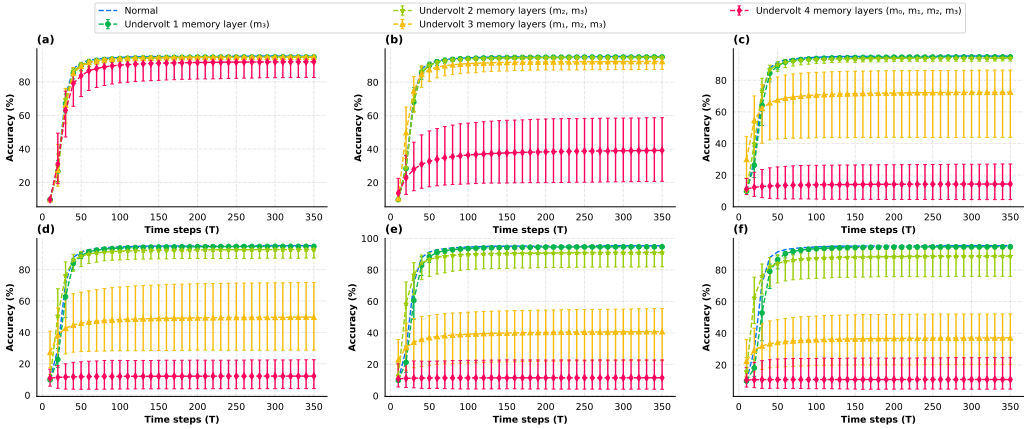

Fig. 7. Accuracy with undervolting memory layer(s) in every time step. (a) $V_{DD} = 0.825V$; BER = 0.00116. (b) $V_{DD} = 0.8V$; BER = 0.01903. (c) V_{DD} = $0.775V$; BER = 0.11519. (d) V_{DD} = $0.75V$; BER = 0.27163. (e) $V_{DD} = 0.725V$; BER = 0.43982. (f) $V_{DD} = 0.7V$; BER = 0.62309.

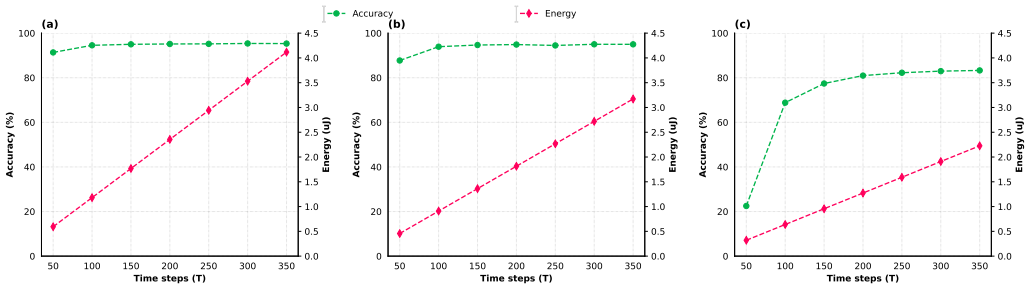

Fig. 8. Accuracy and Energy Consumption of our hardware in different power-gating modes. a) Trade-off Accuracy vs. Energy at normal operations (no power-gating). b) Trade-off Accuracy vs. Energy when power-gating m_3 c) Trade-off Accuracy vs. Energy when power-gating m_3, m_2.

in Section IV, is that the memory layer m_0 contains the MSBs and the memory layer m_3 contains the LSBs. Furthermore, we synchronize all four memory layers ($\{m_0, m_1, m_2, m_3\}$) with the same supply voltage ($V_{m_0} = V_{m_1} = V_{m_2} = V_{m_3} = V_{DD}$). Please take note that the supply voltages could be independent of each memory layer.

Fig. 6 shows that the energy per prediction could be reduced $1.4\times$ times when scaling down the supply voltage to $0.85V$ all four memory layers compared to the scaling down of only one memory layer, m_3. However, with the supply voltage going down, which is near to threshold voltage region, the BER of SRAMs starts to increase exponentially. For example, when undervolting only the memory layer m_3, the BER is approximately 0.00029 and 0.001557 at a supply voltage of $0.825V$ and $0.7V$, respectively. The numbers increase to 0.00116 and 0.623 when undervolting to all four memory layers. However, the accuracy of our hardware greatly reduces when undervolting is applied to the third memory layer m_1

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

TABLE III
THE SETTINGS FOR THE EVALUATION OF **LOW-POWER MODE II**.

Name	Setting II-1	Setting II-2	Setting II-3
Defination	Normal operation	Power-gating one memory layer	Power-gating two memory layers
Power-gated layer	-	m_3	m_2, m_3
# Active bits	8 bits	6 bits	4 bits

TABLE IV
THE SETTINGS FOR THE EVALUATION OF **LOW-POWER MODE III**.

Name	Setting III-1	Setting III-2	Setting III-3
Defination	Undervolting two memory layers	Power-gating one memory layer, Undervolting two memory layers	Power-gating two memory layers, Undervolting two memory layers
Power-gated layer	-	m_3	m_2, m_3
Under-volted layer	m_3, m_2	m_1, m_2	m_0, m_1
Supply Voltage $\{V_{m_0}; V_{m_1}; V_{m_2}; V_{m_3}\}$	$1.1V$; $1.1V$; $[0.675-0.8V]$; $[0.675-0.8V]$	$1.1V$; $0.8V$; $[0.675-0.8V]$; $0V$	$0.825V$; $[0.675-0.8V]$; $0V$; $0V$
# Active bits	8 bits	6 bits	4 bits

$(0.75-0.8V)$. It is because the MSBs of synaptic weights start to be affected. In this case, the average accuracy drops from 92.38% to 49.74% with the supply voltage at $0.8V$ and $0.75V$, respectively. In addition, the accuracy swing $(Max_{Accuracy} - Min_{Accuracy})$ also increases greatly, which is from $6.7\%\big|_{V_{DD}=0.8V}$ to $43.12\%\big|_{V_{DD}=0.75V}$.

To illustrate the transformation of accuracy under the voltage-scaling, Fig. 7 shows the accuracy of our hardware per time step, up to 350 time steps. As seen in Fig. 7, the average accuracy in all four undervolting modes at a supply voltage of $0.825V$ is around 92%. The noticeable transformation is that the accuracy significantly swings when undervolting all four memory layers. It is because the MSBs of synaptic weights are affected. However, the BER of SRAMs at this supply voltage is low (0.00116). Therefore, the number of modified synaptic weights is low and the worst case for accuracy is around 82.58%. With the supply voltage scaling down, the average accuracy curves of undervolting three memory layers and undervolting all memory layers are steadily dropped, while the ones from undervolting two memory layers and undervolting one memory layer are only changed slightly. Consequently, undervolting memory layers containing LSBs can lead to achieving high energy efficiency while maintaining acceptable accuracy.

C. Power-gating Hardware (Low-Power Mode II)

In this section, we evaluate the power transformation and accuracy transformation of our hardware when power-gating the memory layer(s). Our hardware architecture can gain power efficiency by power-gating the memory layers containing LSBs depending on the power situation. Moreover, with the proposed architecture, the *in-situ* dynamical quantization for synaptic weights was achieved without modifying the hardware components. Therefore, we evaluate with two factors: (1) the accuracy when removing the LSBs by power-gating memory layer(s) and (2) the energy efficiency when power-gating. In this paper, we evaluate the accuracy of our hardware and its energy consumption in three operation settings, as shown in Table III.

As shown in Fig. 8, the accuracy of our power-gated hardware at the 350^{th} computing time-step reaches 95.32%, 94.98%, and 83.28% for each power setting, respectively. This is a very strong indicator that we may be able to offer low-power modes in the trade-off of accuracy loss. In fact, at the 100^{th} computing time-step, the accuracy of our system drops to 94.49%, 93.96%, and 68.71% in each power-gating setting. The accuracy of 4-bit synaptic operations (Fig. 8(c)), when applying the setting II-3, loses about 15% compared to the 8-bit operations (Fig. 8(a)). On the other hand, the accuracy is only reduced slightly by 1% when applying the setting II-2 (Fig. 8(b)). Here, we can observe that power consumption could be also reduced greatly with the right time step while maintaining a reasonable accuracy. In terms of energy, this reduction in computing time-step leads to a reduction in energy per prediction and energy per Synaptic OPeration (SOP). For the total energy consumption per time-step with the same bit-width synaptic operation, it increases from the 50^{th} time-step to the 350^{th} one approximately by $7\times$ fold.

D. Undervolting and Power-gating Hardware (Low-Power Mode III)

In this section, we investigate the power-accuracy transformation of our hardware when mixing the voltage-scaling and power-gating techniques for memory layer(s). For the power-gating, the supply voltage of the power-gated memory layer is treated as zero. In this paper, we have four stacked memory layers. Therefore, the configuration of supply voltage for each layer is $\{V_{m_0}, V_{m_1}, V_{m_2}, V_{m_3}\}$. Due to the time-consuming simulation, we choose to evaluate only three settings out of all combinations with 1,000 tests from the Monte-Carlo simulation each. The configurations are defined in TableIV and its evaluation is illustrated in Fig.9.

As shown in Fig. 9(a), the average accuracy of setting III-1 in 1,000 tests at the supply voltage $V_{DD} = 0.8V$ is similar to the normal operation of our hardware and this accuracy reduces by 1-2% per undervolting step. In the worst test, the accuracy drops about 20% compared to the one at the normal operation condition. However, the energy efficiency gains 25%. The energy continues to drop when power-gating is applied to the top layer and undervolting two middle layers (Fig. 9(b)). Compared to the normal operation, it is reduced by half yet the average accuracy only reduces slightly. The only noticeable concern is that the range of accuracy is expanded, and the worst accuracy is 55.27% (dropped about 40% of accuracy compared to the normal operation). As we continue to drop the supply voltage (Fig. 9(c)), the accuracy swings stronger. Consequently, the worst accuracy is 22.76% at $V_{m_1} = 0.675V$ and $V_{m_0} = 0.825V$. However, at $V_{m_1} = 0.8V$, we can see that the energy is reduced four times compared to the normal operation while reducing 6.57% in accuracy.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

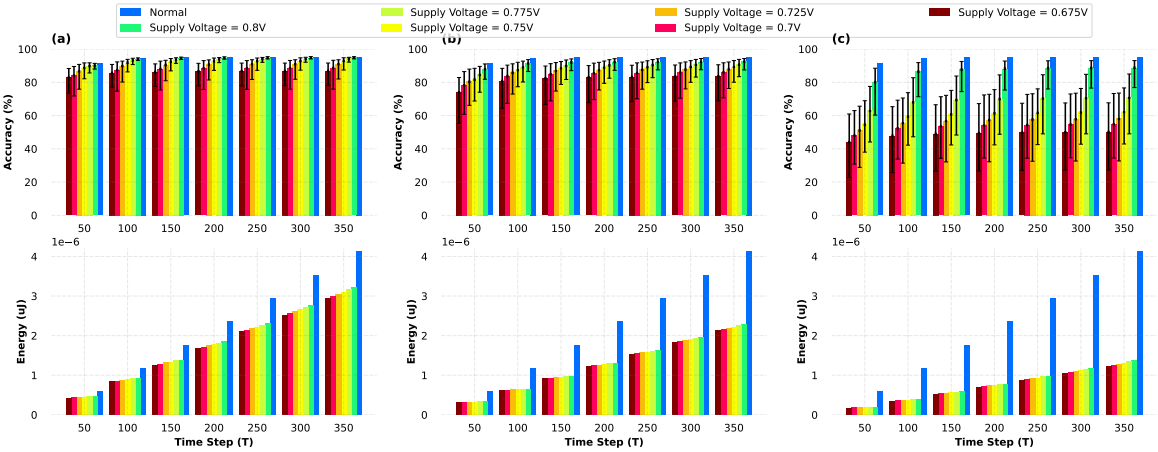

Fig. 9. The evaluation of accuracy and energy with both power-gating and undervolting. The supply voltage of the power-gated layer is treated as zero. a) Accuracy transformation and Energy transformation with setting III-1. b) Accuracy transformation and energy transformation with setting III-2. c) Accuracy transformation and energy transformation with setting III-3.

TABLE V
THE ACCURACY AND THE YIELD OF OUR HARDWARE WITH TWO UPPER DEFECTED MEMORY LAYERS (THE NORMAL ACCURACY = 95.35%).

Yield Rate per Layer	Avg. Acc.	Min. Acc.	Max. Acc.	Avg. Acc. Loss	Normal Yield	Yield Improv.
$Y_1 = 0.999$	94.97%	94.45%	95.38%	0.38%	0.995	0.9968 (+0.18%)
$Y_2 = 0.99$	94.71%	93.25%	95.45%	0.64%	0.951	0.9683 (+1.73%)
$Y_3 = 0.9$	93.85%	91.38%	95.05%	1.70%	0.5905	0.7145 (+12.40%)

TABLE VI
HARDWARE COMPLEXITY OF THE PROPOSED ARCHITECTURE.

Technology		45*nm*
Frequency		100*MHz*
# LIF		48 LIFs
# Stacking Memory		4 layers
# bit of Synaptic Weights		8 bits
Bit Configuration in Memory Layer		2-2-2-2
Gate Count	Total	809.98*KGEs*
	Memory Blocks	791.76*KGEs*
	Crossbar & Address Decoder	9.68*KGEs*
	LIFs	8.52*KGEs*

E. Accuracy with defected memory layers

As explained in Section IV-E, the defective memory caused by fabrication is treated as noise for our proposed architecture and we accept these manufacturing defects to increase the yield rate. In this section, we evaluate the accuracy of our design with three different yield rates in one wafer, which are $Y_1 = 0.999$, $Y_2 = 0.99$, and $Y_3 = 0.9$. With the assumption in Section IV-E, the defects that appeared in the wafer have a uniform distribution. Therefore, we insert the stuck-bits events into memory with the corresponding probabilities to evaluate the trade-off between accuracy and yield rate. In this case, the yield rate improvement is calculated based on Eq. 14.

Table V shows the accuracy of our hardware over 1,000 Monte-Carlo simulation tests. In each yield rate, we evaluate the accuracy with $M = 4$ stacking memory layers and one computing layer, which represents our evaluated architecture. Overall, the average accuracy in all cases drops by $0.38\% - 1.7\%$ compared to the accuracy in normal conditions (95.35%). In addition, the result in the worst case drops 3.97%, which we could consider accepting the manufacturing defect to increase the yield rate. Furthermore, in some cases, the stuck-bit event even leads to an increase in the accuracy of our hardware, which is maximally about 0.1%. In conclusion, the yield rate of the 3D-stacked chip is recently low (e.g.: $Y = 0.5904$ when $D = 5$ and $Y_{layer} = 0.9$). On the other hand, our architecture is able to improve this yield rate by 12.40% with the acceptance of defective memory layers. The trade-off comes with a reduction of about 1.7% in accuracy.

F. Hardware Complexity and Comparison

As shown in Table VI, the area cost of our synthesized hardware is about 809.98*KGEs* at the operating frequency of 100*MHz*. In detail, the synaptic SRAM-based memory occupies the largest part of the hardware area, which is around 97% because it is necessary to store a large number of synaptic weights for high accuracy. For the rest, the processing elements and control units occupy about 3% of the total area of our hardware.

Table VII represents the comparison results between our work and other existing works [1], [10], [23], [68], [69], which are all based on the MNIST benchmark. In terms of accuracy, the result shows that our system has an accuracy of 95.32% in normal conditions. Furthermore, we pick two other configurations (case 1 and case 2), which use undervolting and power-gating for memory layers. The configurations of supply voltage for each memory layer are: case 1 is $\{V_{m_0} = 1.1V;\ V_{m_1} = 1.1V;\ V_{m_2} = 0.8V;\ V_{m_3} = 0.8V\}$, case 2 is $\{V_{m_0} = 0.825V;\ V_{m_1} = 0.8V;\ V_{m_2} = 0V;\ V_{m_3} = 0V\}$, and case 3 is $\{V_{m_0} = 0.825V;\ V_{m_1} = 0.8V;\ V_{m_2} = 0.8V;\ V_{m_3} = 0V\}$. As shown in Table VII, in case 2, with the

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

TABLE VII
COMPARISON RESULTS BETWEEN THE PROPOSED ARCHITECTURE AND EXISTING WORKS.

Parameters	TrueNorth [10]	Loihi [1]	ODIN [23]	NASH [68]	Karimi *et al.* [69]	**This work**					
						Normal Case	Case 1[1]	Case 2[2]	Normal Case	Case 1[1]	Case 3[3]
Benchmark	MNIST	MNIST	MNIST	MNIST	MNIST	MNIST (784:48:10)			CIFAR-10 (VGG16)*		
Accuracy (%)	91.94	96	84	79.4	99.2	95.35	94.84	88.77	91.38	91.26	69.50
Neuron Model	IF	DenMem	LIF & Izhikevicz	LIF	LIF	LIF					
Synaptic Weight Storage	1-bit SRAM	1-to-9-bit SRAM	4-bit SRAM	8-bit SRAM	CTT twin-cell	8-bit SRAM			16-bit SRAM		
Interconnect	2D	2D	2D	3D	2D	3D					
Implementation	Digital	Digital	Digital	Digital	Mix-signal	Digital			Software simulation		
Learning Rule	Un-supervised	On-chip STDP	On-chip Stochastic SDSP	On-chip STDP	Off-chip	Off-chip					
Technology	28nm	14nm FinFET	28nm FD-SOI	45nm	22nm FD-SOI	45nm					
Supply Voltage	0.7-1.05V	0.5-1.2 V	0.55-1 V	1.1 V	0.8 V	0.65V - 1.1V					
Energy per SOP (pJ)	26 (0.775V)	23.6 (0.75V)	8.4	189.3	8	244.28 (1.1V)	191.46[1]	81.16[2]	475.20 (1.1V)	372.13[1]	205.55[3]
Energy per SOP (pJ) (in 14nm)	4.902	23.6	1.078	10.86	4.32	14.02 (1.1V)	10.98[1]	4.65[2]	27.27 (1.1V)	21.35[1]	11.79[3]

[1] Case 1: $\{V_{m_0} = 1.1V;\ V_{m_1} = 1.1V;\ V_{m_2} = 0.8V;\ V_{m_3} = 0.8V\}$ (Low-power Mode I)
[2] Case 2: $\{V_{m_0} = 0.825V;\ V_{m_1} = 0.8V;\ V_{m_2} = 0V;\ V_{m_3} = 0V\}$ (Low-power Mode III)
[3] Case 3: $\{V_{m_0} = 0.825V;\ V_{m_1} = 0.8V;\ V_{m_2} = 0.8V;\ V_{m_3} = 0V\}$ (Low-power Mode III)

operation of 4-bit synaptic weights, the accuracy drops by 6.58% compared to the normal operation (8-bit). However, this accuracy is similar to the works of *Kim et al.* [24] and *ODIN* [23], which also operates at 4-bit synaptic weight precision.

In terms of power, we compare our work with others using the energy per synaptic operation parameter. Due to the gap in technology, we use the well-known scaling equation from *Stillmaker et al.* [70] to scale down the 14-nm technology node. As shown in Table VII, our hardware consumes 244.28*pJ*, 191.46*pJ*, and 81.16*pJ* at the 45-nm technology node in three cases for 350 time-steps, respectively. After scaling down to the 14-nm technology, our energy per synaptic operation achieves the values, which accordingly are 14.02*pJ*, 10.98*pJ*, and 4.65*pJ*. Furthermore, we also evaluate our methodology with the 16-bit VGG-16 using the CIFAR-10 dataset. As shown in Table VII, the accuracy only drops slightly by 0.12% while the energy per SOP decreases significantly by 21.68% in case 1. However, in the case 3, despite the energy reduction of 56.74%, the accuracy is also reduced seriously by 21.88%.

In conclusion, these results show that our architecture with 3D stacking memory has an advantage in terms of reducing energy consumption when applying voltage-scaling and power-gating techniques for memory layers. For the MNIST dataset, switching from the normal mode to the low-power mode I, the accuracy drops by 0.51% to trade-off the energy reduction of 21.62%. When our hardware switches to the low-power mode III, the accuracy drops by 6.58% to reduce the energy consumption by 66.77%. In the case of the CIFAR-10 dataset, with the software simulation, the accuracy also drops by a small fraction (0.12%) to reduce 21.68% energy per synaptic operation when switching from the normal mode to the low-power mode I. Moreover, at the low-power mode III, the accuracy decreases by 21.88% saving 56.74% of energy consumption.

VI. DISCUSSION

In this section, we provide some discussions related to the limitations of our work and potential solutions. First, besides the reliability issue of stacking layers, Through-Silicon-Via's (TSV) reliability is also one of the major concerns. There are numerous works on dealing with TSV defects by using redundancies. Therefore, these techniques can be embedded into our architecture to deal with TSV defects. Unlike TSV defects which can be dealt with by using redundancies, defects on stacking memory dies are mostly unrepairable; therefore, we focus on this type of defects in this work.

Second, thermal dissipation is another critical issue of 3D-ICs as stacking multiple layers prevents the heat transmission to the heatsink. Although the thermal issue is still an open problem in this work, by lowering the power consumption; our work has the potential to alleviate this issue of 3D-ICs.

Third, as we show in the evaluation section that there are numerous combinations of different voltages and power gating. Also, the scaling step of the voltage can also be adjusted which leads to more voltages being chosen. Moreover, the splitting method of the memory can be also different between designs (i.e., 16-bit can be 4×4bit or 2×8bit or 8×2bit) or can be asymmetric (i.e., 8-bit can be two subsets of $3 + 5$bit or $4+4$bit or $5+3$bit) to isolate the meaningful bits and to reduce the power of inactive bits. Because of this, it is not possible to cover all possible cases to specify the standard of faulty-energy-accuracy trade-off. Hence, our picks of configuration in the comparison in Table VII may be suboptimal. To solve this issue, one of the methods is to perform an optimization process

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

(i.e. Genetic Algorithm or Particle Swarm Optimization). However, in combination with the Monte-Carlo simulation, as we have shown in the evaluation, the number of searching values can be overwhelming.

Fourth, although our work focuses on SRAM which is easily accessible, there is a possibility to apply our methodology to advanced memory technologies (eDRAM, STT-RAM, ...). In fact, this could be even more power efficient as non-volatile memories are more efficient in terms of power and can retain their value after the power gating period.

Fifth, our work focuses on an array of LIF array; however, this method can also be applied for large-scale Network-on-Chip-based architecture [31]. As each NoC core can be undervolted and power-gated separately, this could open a more fine-grained control for the system. Furthermore, the power of spike generation and spike transmission are two other factors that can affect the power consumption of the chip and must be considered in the future.

Sixth, our work utilizes multiple power rails through TSVs to supply power for every memory layer, which is dependent on an off-chip voltage regulator. However, an on-chip voltage regulator can also be implemented into the neuromorphic systems for better scalability. In this case, the hardware overhead is also needed to consider when applying multiple supply voltages for every memory layer. For example, the hardware area of the voltage regulator in [38] is around $0.375\mu m^2$ ($0.111\mu m^2$ without wired area) with the UMC 1.1V 40-nm CMOS technology. Hence, by putting this regulator into our memory layer under 45-nm CMOS technology and ignoring the wired area, the hardware overhead is mathematically about 27.05%, where the total area of memory blocks in one memory layer without wired is $0.337\mu m^2$. As a result, it could add up to a significant hardware area for voltage scaling in every memory layer. However, the hardware footprint is unchanged compared to the traditional 2D DVS one. It is because our hardware architecture is implemented in 3D and every memory layer has the same hardware area.

Although there are several drawbacks in this work, the proposed methodology and its implemented architecture have shown the potential to be able to reduce power consumption with graceful performance degradation.

VII. Conclusions

In this paper, we have proposed a methodology to split and stack the synaptic memories for low-power operation. With the 3D technology, the memory can be isolated into different layers, which allows the possibility to separately control the supply voltage of each layer. As a result, the proposed architecture can apply the voltage-scaling technique and also further turn on/off the power supply of one or multiple layer(s) inside it to save the overall energy consumption. In addition, by splitting the synaptic weights into multiple memory layers, the accuracy can be maintained by protecting the memory layer(s) containing the MSBs while dropping the supply voltage of the memory layer(s) containing LSBs. Our future works will extend this work into a very large-scale system using Network-on-Chips with an optimal power-saving strategy.

References

[1] M. Davies and *et al.*, "Loihi: A Neuromorphic Manycore Processor with On-Chip Learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.

[2] B. V. Benjamin and *et al.*, "Neurogrid: A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 699–716, 2014.

[3] W. Guo and *et al.*, "Neural Coding in Spiking Neural Networks: A Comparative Study for Robust Neuromorphic Systems," *Frontiers in Neuroscience*, vol. 15, 2021.

[4] C. Mead, "Neuromorphic electronic systems," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1629–1636, 1990.

[5] A. Burkitt and *et al.*, "A Review of the Integrate-and-Fire Neuron Model: I. Homogeneous Synaptic Input." *Biol. Cybern*, vol. 95, pp. 1–19, 2006.

[6] J. Stuijt and *et al.*, "uBrain: An Event-Driven and Fully Synthesizable Architecture for Spiking Neural Networks," *Frontiers in Neuroscience*, vol. 15, 2021.

[7] R. V. W. Putra and *et al.*, "EnforceSNN: Enabling resilient and energy-efficient spiking neural network inference considering approximate DRAMs for embedded systems," *Frontiers in Neuroscience*, vol. 16, 2022.

[8] D. Elliott and *et al.*, "Computational RAM: implementing processors in memory," *IEEE Design & Test of Computers*, vol. 16, no. 1, pp. 32–41, 1999.

[9] M. Kang and *et al.*, "An energy-efficient VLSI architecture for pattern recognition via deep embedding of computation in SRAM," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 8326–8330.

[10] F. Akopyan and *et al.*, "TrueNorth: Design and Tool Flow of a 65 mW 1 Million Neuron Programmable Neurosynaptic Chip," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 10, pp. 1537–1557, 2015.

[11] M. Prezioso and *et al.*, "Training and operation of an integrated neuromorphic network based on metal-oxide memristors," *Nature*, vol. 521, pp. 61–64, 2015.

[12] A. F. Vincent and *et al.*, "Spin-Transfer Torque Magnetic Memory as a Stochastic Memristive Synapse for Neuromorphic Systems," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 9, no. 2, pp. 166–174, 2015.

[13] S. Ambrogio and *et al.*, "Unsupervised Learning by Spike Timing Dependent Plasticity in Phase Change Memory (PCM) Synapses," *Frontiers in Neuroscience*, vol. 10, 2016.

[14] I. Boybat and *et al.*, "Neuromorphic computing with multi-memristive synapses," *Nature Communications*, vol. 9, p. 2514, 2018.

[15] S. R. Nandakumar and *et al.*, "Experimental Demonstration of Supervised Learning in Spiking Neural Networks with Phase-Change Memory Synapses," *Scientific Reports*, vol. 10, p. 8080, 2020.

[16] M. Matsumiya and *et al.*, "A 15 ns 16 Mb CMOS SRAM with reduced voltage amplitude data bus," in *1992 IEEE International Solid-State Circuits Conference Digest of Technical Papers*, 1992, pp. 214–215.

[17] B. Mohammad and J. Abraham, "A reduced voltage swing circuit using a single supply to enable lower voltage operation for SRAM-based memory," *Microelectronics Journal*, vol. 43, no. 2, pp. 110–118, 2012.

[18] B. Salami and *et al.*, "Comprehensive Evaluation of Supply Voltage Underscaling in FPGA on-Chip Memories," in *2018 51st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2018, pp. 724–736.

[19] ——, "An Experimental Study of Reduced-Voltage Operation in Modern FPGAs for Neural Network Acceleration," in *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2020, pp. 138–149.

[20] R. G. Dreslinski and *et al.*, "Near-Threshold Computing: Reclaiming Moore's Law Through Energy Efficient Integrated Circuits," *Proceedings of the IEEE*, vol. 98, no. 2, pp. 253–266, 2010.

[21] J. Zhang and *et al.*, "Thundervolt: Enabling Aggressive Voltage Underscaling and Timing Error Resilience for Energy Efficient Deep Learning Accelerators," in *Proceedings of the 55th Annual Design Automation Conference*, ser. DAC '18, New York, NY, USA, 2018.

[22] P. Pandey, P. Basu, K. Chakraborty, and S. Roy, "GreenTPU: Improving Timing Error Resilience of a Near-Threshold Tensor Processing Unit," in *2019 56th ACM/IEEE Design Automation Conference (DAC)*, 2019, pp. 1–6.

[23] C. Frenkel and *et al.*, "A 0.086-mm2 12.7-pJ/SOP 64k-Synapse 256-Neuron Online-Learning Digital Spiking Neuromorphic Processor in 28nm CMOS," *IEEE Transactions on Biomedical Circuits and Systems*, pp. 1–1, 2018.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Very Large Scale Integration (VLSI) Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVLSI.2023.3318231

[24] J. K. Kim and *et al.*, "A 640M pixel/s 3.65mW sparse event-driven neuromorphic object recognition processor with on-chip learning," in *2015 Symposium on VLSI Circuits (VLSI Circuits)*, 2015, pp. C50–C51.

[25] M. Ambroise and *et al.*, "Biorealistic spiking neural network on FPGA," in *2013 47th Annual Conference on Information Sciences and Systems (CISS)*, 2013, pp. 1–6.

[26] R. Wang and *et al.*, "An FPGA Implementation of a Polychronous Spiking Neural Network with Delay Adaptation," *Frontiers in Neuroscience*, vol. 7, 2013.

[27] H. An and *et al.*, "Three-Dimensional Neuromorphic Computing System With Two-Layer and Low-Variation Memristive Synapses," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 3, pp. 400–409, 2022.

[28] G. Orchard and *et al.*, "Efficient Neuromorphic Signal Processing with Loihi 2," in *2021 IEEE Workshop on Signal Processing Systems (SiPS)*, 2021, pp. 254–259.

[29] S. Furber, "Large-scale neuromorphic computing systems," *Journal of Neural Engineering*, vol. 13, 08 2016.

[30] A. Ben Abdallah and K. N. Dang, "Toward Robust Cognitive 3D Brain-Inspired Cross-Paradigm System," *Frontiers in Neuroscience*, vol. 15, 2021.

[31] K. N. Dang and *et al.*, "MigSpike: A Migration Based Algorithms and Architecture for Scalable Robust Neuromorphic Systems," *IEEE Transactions on Emerging Topics in Computing*, vol. 10, no. 2, pp. 602–617, 2022.

[32] J. Leng and *et al.*, "Safe limits on voltage reduction efficiency in gpus: A direct measurement approach," in *2015 48th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2015, pp. 294–307.

[33] K. K. Chang and *et al.*, "Understanding reduced-voltage operation in modern dram devices: Experimental characterization, analysis, and mechanisms," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 1, no. 1, jun 2017.

[34] B. Reagen and *et al.*, "Minerva: Enabling Low-Power, Highly-Accurate Deep Neural Network Accelerators," in *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA)*, 2016, pp. 267–278.

[35] L. Di and *et al.*, "Power switch characterization for fine-grained dynamic voltage scaling," in *2008 IEEE International Conference on Computer Design*, 2008, pp. 605–611.

[36] Z. Bai and *et al.*, "A Cascaded Multilevel Battery Energy Storage Based Parallel Dynamic Voltage Compensator for Medium Voltage Industrial Distribution Systems," *IEEE Transactions on Industrial Informatics*, pp. 1–10, 2023.

[37] N. Adorni and *et al.*, "A 10-mA LDO With 16-nA IQ and Operating From 800-mV Supply," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 2, pp. 404–413, 2020.

[38] C.-H. Huang and W.-C. Liao, "A High-Performance LDO Regulator Enabling Low-Power SoC With Voltage Scaling Approaches," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 5, pp. 1141–1149, 2020.

[39] P. H. McLaughlin and *et al.*, "A Monolithic Resonant Switched-Capacitor Voltage Regulator With Dual-Phase Merged-LC Resonator," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 12, pp. 3179–3188, 2020.

[40] D. Lutz and *et al.*, "12.4 A 10mW fully integrated 2-to-13V-input buck-boost SC converter with 81.5International Solid-State Circuits Conference (ISSCC), 2016, pp. 224–225.

[41] S. S. N. Larimi and *et al.*, "Understanding Power Consumption and Reliability of High-Bandwidth Memory with Voltage Underscaling," *CoRR*, vol. abs/2101.00969, 2021.

[42] S. Mukhopadhyay and *et al.*, "Statistical design and optimization of SRAM cell for yield enhancement," in *IEEE/ACM International Conference on Computer Aided Design, 2004. ICCAD-2004.*, 2004, pp. 10–13.

[43] E. Lee and *et al.*, "A Charge-Domain Scalable-Weight In-Memory Computing Macro With Dual-SRAM Architecture for Precision-Scalable DNN Accelerators," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 8, pp. 3305–3316, 2021.

[44] M. E. Sinangil and *et al.*, "A 7-nm Compute-in-Memory SRAM Macro Supporting Multi-Bit Input, Weight and Output and Achieving 351 TOPS/W and 372.4 GOPS," *IEEE Journal of Solid-State Circuits*, vol. 56, no. 1, pp. 188–198, 2021.

[45] S. Jain, L. Lin, and M. Alioto, "±CIM SRAM for Signed In-Memory Broad-Purpose Computing From DSP to Neural Processing," *IEEE Journal of Solid-State Circuits*, vol. 56, no. 10, pp. 2981–2992, 2021.

[46] H. Kim and *et al.*, "A 16K SRAM-Based Mixed-Signal In-Memory Computing Macro Featuring Voltage-Mode Accumulator and Row-by-Row ADC," in *2019 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, 2019, pp. 35–36.

[47] A. Agrawal and *et al.*, "X-SRAM: Enabling In-Memory Boolean Computations in CMOS Static Random Access Memories," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 12, pp. 4219–4232, 2018.

[48] W. Simon and *et al.*, "A Fast, Reliable and Wide-Voltage-Range In-Memory Computing Architecture," in *2019 56th ACM/IEEE Design Automation Conference (DAC)*, 2019, pp. 1–6.

[49] M. Hu and *et al.*, "Dot-Product Engine for Neuromorphic Computing: Programming 1T1M Crossbar to Accelerate Matrix-Vector Multiplication," in *Proceedings of the 53rd Annual Design Automation Conference*, ser. DAC '16. New York, NY, USA: Association for Computing Machinery, 2016.

[50] M. R. Haq Rashed and *et al.*, "Hybrid Analog-Digital In-Memory Computing," in *2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, 2021, pp. 1–9.

[51] K. Cho and *et al.*, "SAINT-S: 3D SRAM Stacking Solution based on 7nm TSV technology," in *IEEE Hot Chips Symposium*, 2020, pp. 1–13.

[52] N.-D. Nguyen and *et al.*, "An In-Situ Dynamic Quantization With 3D Stacking Synaptic Memory for Power-Aware Neuromorphic Architecture," *IEEE Access*, vol. 11, pp. 82 377–82 389, 2023.

[53] M. Evers and *et al.*, "The AMD Next-Generation "Zen 3" Core," *IEEE Micro*, vol. 42, no. 3, pp. 7–12, 2022.

[54] K. Ueyoshi and *et al.*, "QUEST: Multi-Purpose Log-Quantized DNN Inference Engine Stacked on 96-MB 3-D SRAM Using Inductive Coupling Technology in 40-nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 1, pp. 186–196, 2019.

[55] K. Shiba and *et al.*, "A 96-MB 3D-Stacked SRAM Using Inductive Coupling With 0.4-V Transmitter, Termination Scheme and 12:1 SerDes in 40-nm CMOS," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 2, pp. 692–703, 2021.

[56] J. Zhao and *et al.*, "An energy-efficient 3D CMP design with fine-grained voltage scaling," in *2011 Design, Automation & Test in Europe*, 2011, pp. 1–4.

[57] T. Wunderlich and *et al.*, "Demonstrating Advantages of Neuromorphic Computation: A Pilot Study," *Frontiers in Neuroscience*, vol. 13, 2019.

[58] P. Reviriego and *et al.*, "Error-Tolerant Data Sketches Using Approximate Nanoscale Memories and Voltage Scaling," *IEEE Transactions on Nanotechnology*, vol. 21, pp. 16–22, 2022.

[59] P. Royer and M. López-Vallejo, "Using pMOS Pass-Gates to Boost SRAM Performance by Exploiting Strain Effects in Sub-20-nm FinFET Technologies," *IEEE Transactions on Nanotechnology*, vol. 13, no. 6, pp. 1226–1233, 2014.

[60] E. Seevinck and *et al.*, "Static-noise margin analysis of MOS SRAM cells," *IEEE Journal of Solid-State Circuits*, vol. 22, no. 5, pp. 748–754, 1987.

[61] N. Inc. Nangate Open Cell Library 45 nm. [Online]. Available: http://www.nangate.com/

[62] N. E. D. Automation. FreePDK3D45 3D-IC Process Design Kit. [Online]. Available: http://www.eda.ncsu.edu/wiki/FreePDK3D45

[63] M. R. Guthaus and *et al.*, "OpenRAM: An open-source memory compiler," in *2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2016, pp. 1–6.

[64] A. Krizhevsky, "Learning multiple layers of features from tiny images," Canadian Institute for Advanced Research, Tech. Rep., 2009. [Online]. Available: http://www.cs.toronto.edu/ kriz/cifar.html

[65] N. Muralimanohar and *et al.*, "CACTI 6.0: A tool to model large caches," *HP laboratories*, vol. 27, p. 28, 2009.

[66] J. Schemmel and *et al.*, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," in *2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2010, pp. 1947–1950.

[67] J.-s. Seo and *et al.*, "A 45nm CMOS neuromorphic chip with a scalable architecture for learning in networks of spiking neurons," in *2011 IEEE Custom Integrated Circuits Conference (CICC)*, 2011, pp. 1–4.

[68] O. M. Ikechukwu and *et al.*, "On the Design of a Fault-Tolerant Scalable Three Dimensional NoC-Based Digital Neuromorphic System With On-Chip Learning," *IEEE Access*, vol. 9, pp. 64 331–64 345, 2021.

[69] M. Karimi and *et al.*, "Ctt-based scalable neuromorphic architecture," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, pp. 1–1, 2023.

[70] A. Stillmaker and B. Baas, "Scaling equations for the accurate prediction of CMOS device performance from 180nm to 7nm," *Integration*, vol. 58, pp. 74–81, 2017.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.