Architectural Design of 3D NAND Flash based Compute-in-Memory for Inference Engine

Wonbo Shim
Georgia Institute of Technology

Hongwu Jiang
Georgia Institute of Technology

Xiaochen Peng
Georgia Institute of Technology

Shimeng Yu
Georgia Institute of Technology

ABSTRACT

3D NAND Flash memory has been proposed as an attractive candidate of inference engine for deep neural network (DNN) owing to its ultra-high density and commercially matured fabrication technology. However, the peripheral circuits require to be modified to enable compute-in-memory (CIM) and the chip architectures need to be redesigned for an optimized dataflow. In this work, we present a design of 3D NAND-CIM accelerator based on the macro parameters from an industry-grade prototype chip. The DNN inference performance is evaluated using the DNN+ NeuroSim framework. To exploit the ultra-high density of 3D NAND Flash, both inputs and weights duplication strategies are introduced to improve the throughput. The benchmarking on a variety of VGG and ResNet networks was performed across technological candidates for CIM including SRAM, RRAM and 3D NAND. Compared to similar designs with SRAM or RRAM, the result shows that 3D NAND based CIM design can achieve not only 17-24% chip size but also 1.9-2.7 times more competitive energy efficiency for 8-bit precision inference.

CCS CONCEPTS

• **Hardware** → Integrated circuits; Semiconductor memory; Nonvolatile memory.

KEYWORDS

Deep neural network, hardware accelerator, compute-in-memory, 3D NAND Flash

ACM Reference Format:
Wonbo Shim, Hongwu Jiang, Xiaochen Peng, and Shimeng Yu. 2020. Architectural Design of 3D NAND Flash based Compute-in-Memory for Inference Engine. In *The International Symposium on Memory Systems (MEMSYS 2020), September 28–October 01, 2020, Washington, DC, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3422575.3422779

1 INTRODUCTION

Deep learning is one of the most interesting and spotlighted fields in recent years. Deep neural network (DNN) has achieved great successes in various tasks such as image classification and speech recognition. State-of-the-art learning algorithms tend to grow larger size and deeper network to achieve high accuracy. Larger and deeper networks have to perform tremendous number of computations. Therefore, huge volume of data movements between the processor and off-chip memory are required in conventional von-Neumann architecture. Commonly used architectures such as CPUs/GPUs and/or FPGA are disadvantageous in terms of energy efficiency for DNN workloads. There have been lots of efforts from industry and academia to propose alternative computing platforms. Several CMOS-based application specific integrated circuits (ASIC) accelerators such as Google TPU [1] were designed to alleviate the problem by data reuse on-chip, but the memory wall problem still remains where the model parameters are stored in global buffer and the actual computation is performed at the digital multiply-and-accumulation (MAC) arrays. In aforementioned platforms, DRAM access is still frequent due to the limited global buffer capacity.

To overcome these challenges, compute-in-memory (CIM) has been emerged as an alternative paradigm owing to its high throughput and energy efficiency [2]. CIM utilizes the conductance of the memory cell to represent the weight, and conduct MAC operations by activating multiple rows and reading out the analog current summed up along the column. High parallelism can be achieved since the dense array of millions of memory cells performs computation simultaneously. In addition, the computation is performed within the memory array so the energy consumption caused by data movement between processor and memory is reduced.

Most of the existing nonvolatile memory (NVM) devices have been investigated as synaptic devices for vector-matrix multiplication (VMM) or weighted sum computation. Hardware accelerators based on emerging devices such as resistive random access memory (RRAM) [2-8] and phase change memory (PCM) [9-10] have been actively researched because of its logic compatibility and nonvolatility, but their relatively small on/off current ratio and large on-current are not suitable for large array configuration. Floating gate type NOR Flash [11] technology is another candidate owing to its large on/off current ratio which can help activating a large number of rows in a column, but the large on-current still makes sense amplifier design of summed readout current exceedingly challenging. Moreover, embedded NOR Flash structure is hard to scale down beyond the 28 nm node, so it is less competitive than other device technologies in terms of the memory density.

MEMSYS 2020, September 28–October 01, 2020, Washington, DC, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8899-3/20/09…$15.00
https://doi.org/10.1145/3422575.3422779

Very recently, NAND Flash has been proposed as a high-density and high-bandwidth CIM candidate [12-14]. Since 3D NAND Flash has the highest density among all the memory devices [15], the weights of large DNN can be stored in a small form factor. Furthermore, it is quite advantageous that 3D NAND Flash is already matured today and is based on a widely commercialized fabrication technology.

In this work, we present the architectural design of 3D NAND Flash based CIM accelerator that is optimized to the inference of DNN, with the benchmarking results using the DNN+ NeuroSim [16] framework. We used the electrical parameters and the physical dimensions of a 3D NAND-CIM prototype by Macronix [13] for the baseline of this work. This prototype of 3D NAND [13] is of industry-grade and has been customized to support CIM paradigm. The peripheral circuits and chip level hierarchy configuration has been adapted to support DNN models such as VGG [17] and ResNet [18] for image classification. Finally, we report the performance results (energy efficiency and throughput) across technological options for CIM.

2 DESIGN OF 3D NAND BASED INFERENCE ENGINE

2.1 3D NAND Array Level Design

Generally, CIM architecture performs mixed-signal computation, i.e., analog current is summed up along the column or row, then the currents are converted from analog to digital at the edge of the array. Figure 1 shows the VMM operation with the schematic of the CIM in 3D NAND Flash memory. Prior Flash-based CIM topologies [11, 12, 14] activates the wordlines (WLs) as the input vectors of MAC, while the bitlines (BLs) are activated as input vectors in this work similarly as proposed in [13]. While the BL voltages are applied to the multiple BLs respective to the input vectors, the read voltage is applied to the selected WL of the NAND string and the pass voltage is applied to the unselected WLs. The read-out is performed WL by WL. Because the source line (SL) of the NAND strings within a single block are entirely connected through the bottom substrate, the summed current can be sensed at the end of the SL of the block. Since NAND Flash has exceptionally large page size (several kB of BLs) in a single block, this topology has advantage in that the huge number of string currents can be summed at once using a single analog-digital converter (ADC).

We designed the 3D NAND block having 3 string select lines (SSLs) to store the 2-bit weight and being computed at once. The most significant bit (MSB) of weights are stored in the cells of SSL1 and SSL2, while the least significant bit (LSB) of weights are stored in SSL3 only. Since the SL is connected to the all SSL together, the output current follows the equation in Figure 1 which means 2-bit weights are computed in analog manner without the digital adder and shifter. By using this MSB weight duplication, the total computing operation steps decrease by half.

The electrical characteristics of NAND Flash device of our work is based on the measured experimental data of a 32-layer 3D NAND chip reported in [13]. The 3nm-thick polysilicon channel could produce extremely low on current (~2 nA) and off current (below 1 pA) owing to the low mobility and high on/off ratio of NAND Flash device. The low on/off current are suitable for activating the very large number of BLs at the same time with proper summed current to design the ADC.

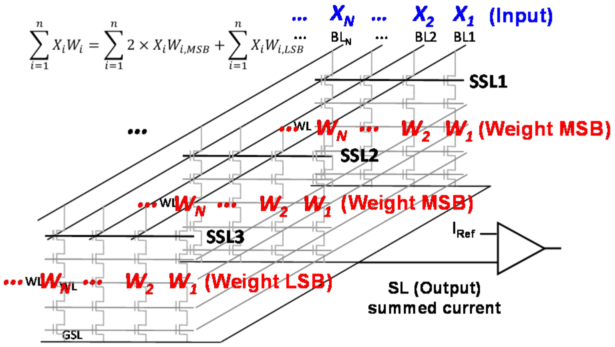
$\sum_{i=1}^{n} X_i W_i = \sum_{i=1}^{n} 2 \times X_i W_{i,MSB} + \sum_{i=1}^{n} X_i W_{i,LSB}$
... X_N ... X_2 X_1 (Input)
... BLN
... BL2 BL1
SSL1
...
...WL W_N ... W_2 W_1 (Weight MSB)
SSL2
...WL W_N ... W_2 W_1 (Weight MSB)
I_{Ref}
SSL3
...WL W_{NL} ... W_2 W_1 (Weight LSB)
SL (Output)
summed current
GSL

Figure 1: Vector-matrix multiplication (VMM) operation in 3D NAND Flash in one block with MSB weight duplication.

To determine the physical dimensions of the 3D NAND array for one block, we used the standard 3D NAND pitch size [19], e.g., BL pitch 40 nm and SSL pitch 0.75 μm. The number of BLs of the array is decided according to the kernel size of the largest convolution layer of the DNN to unroll each kernel and map into the single array. The kernel size of the largest layer of VGG-8 and ResNet is 3×3×512, then we will duplicate the weights to other BLs 3 times as will be explained in section 3.1 later, so one block has total 13824 BLs and 553 μm width. Along the vertical direction, one block consists of 3 SSLs as shown in Figure 1, so the height of a block is 2.25 μm. As a result, the number of the physical cells in one block is 13824 BL×32 WL×3 SSL, e.g., 1.27 Mb. This block size configuration follows the general design guidelines of the commercial 3D NAND.

2.2 Subarray Architecture Design

The subarray is the minimum unit of the chip hierarchy with multiple 3D NAND blocks. In our design, the subarray has 64 blocks with 144 μm BL length which is relatively much shorter than conventional NAND Flash (generally several millimeters long) to achieve small RC loading with faster BL charging speed and less energy consumption. Following the method that 2-bit weight is represented in one block, then 8-bit weight needs 4 blocks to be stored, so 16 kernels can be stored in a subarray with 8-bit weight. The total capacity of a subarray is 81Mb.

We estimated the latency and energy parameters of the NAND array with RC delay model. Owing to the small dimension of WL and BL than the conventional 3D NAND, the estimated latency of WL setup and BL setup are 303 ns and 12 ns respectively as shown in Table 1. We ran the HSPICE simulation to achieve the SL charging time, and the result shows that the summed current for the ADC sensing (in the case of max-bit input duplication which will be introduced in section 3) requires relatively long (hundreds of ns) stabilizing time because of the large capacitance (~16 pF) of the SL of the 3D NAND array. Although WL setup requires 43.5 nJ per each time, it needs to be conducted only once for each of the DNN layer so the ratio over the total energy of WL setup energy is only 1-5%.

Table 1: Latency and energy estimation for parallel read operation for CIM in 3D NAND subarray (64 block).

	Latency	Energy
WL setup	303 ns	43.5 nJ
BL setup @ 50% sparsity	12 ns	35.9 pJ (5.2 fJ per BL)
SL setup @ max-bit input	530-750 ns	41.7 pJ

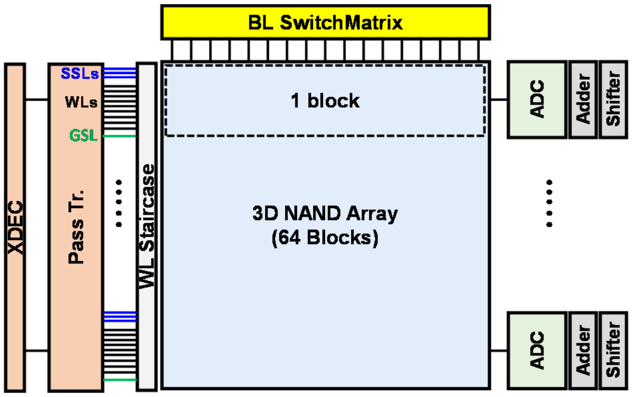

Figure 2: Designed subarray configuration.

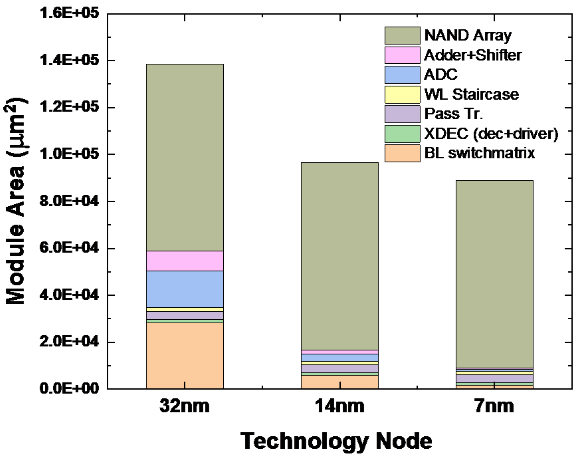

Figure 3: Area estimation of the subarray modules with various technology nodes for peripheral logic circuits.

Figure 2 shows the subarray configuration with 3D NAND array and peripheral circuits. Throughout this work, we assumed that the 3D NAND array cells and peripheral logic transistors are fabricated on same wafer. As an architectural design work, we leave the fabrication and integration challenges for future work. Input vectors are fetched to the BLs through the BL switch matrix, and x-decoder (XDEC) enables the operating blocks. WL voltages are applied to the 3D NAND of the selected blocks through the pass transistors. The summed current flowing through the SL is sensed at the edge and converted to digital value through ADC. Here 7-bit ADC is used to guarantee the computational accuracy considering the large number of BLs summation along the SL [13]. In our design, the SAR-ADC is selected over the flash-ADC due to the relatively high resolution required here to trade-off the latency over the power and area [20]. The digital adder and shifters process the multi-cycle outputs to achieve final output.

The estimated areas of each module for various peripheral logic technology nodes are shown in Figure 3. The areas of the pass transistors and XDEC drivers are fixed because we assumed the high voltage transistor process (1 μm pitch) is applied here to drive the programming voltage (~20V) to the WLs. The area of WL staircase is also fixed because it is determined by the number of WL stacks and the staircase design rule irrespective to the logic transistor technology node.

2.3 Chip Level Architecture Design

The top-down hierarchy of the proposed 3D NAND based CIM architecture is defined as chip, tile, processing element (PE) and subarray as shown in Figure 4. The chip-level consists of tiles, global buffer and neural functional computation logics for max pooling, activation (ReLU) and accumulation. The tile-level is composed of several PEs, input/output buffers, and accumulation module. Similarly, the processing elements (PEs)-level are built with multiple subarrays, PE input/output buffers and accumulation module. H-tree based routing is used for the output accumulation at each level. The number of subarrays per PE and the number of PEs per tile can vary by optimization with respect to the topologies of various DNNs.

3 WEIGHT MAPPING TO 3D NAND

Owing to its ultra-high memory density, 3D NAND is not only able to store the large size DNNs, but also able to support duplicating the weights many times for parallel computation. Therefore, we can take advantage of the duplication for improving the performance. In section 2.1, we already introduced that the MSB weight duplication makes twice as fast, so we introduce the other two types of duplication into 3D NAND in this section.

3.1 Multi-bit Input Duplication

Different from the CIM arrays with other NVM devices, the NAND CIM array has huge number of inputs (BLs) in a single array so in most of the cases it is larger than the input vector size. Since the remaining empty BLs compose same block and share the ADC with used BLs, storing (or duplicating) the weights of different kernels to the empty cells is not meaningful in terms of latency improvement. But we can duplicate the input and the respective weight of the same kernel to the empty BLs and take advantage of input cycle

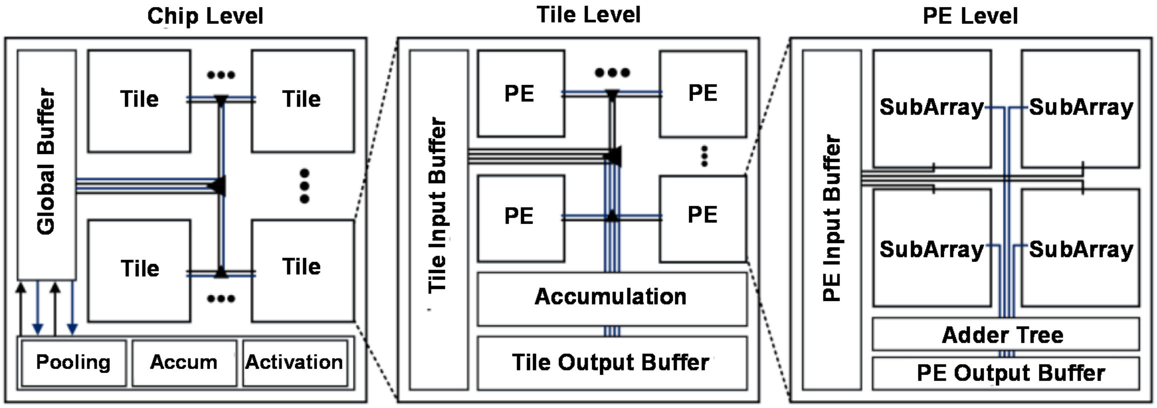

Figure 4: Hierarchy of the 3D NAND based CIM chip architecture.

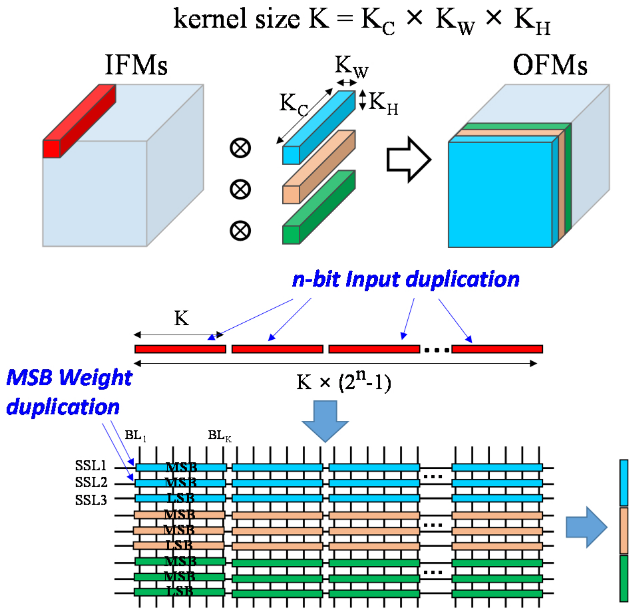

Figure 5: Mapping scheme of a convolutional kernel with MSB weight duplication and multi-bit input duplication.The next subsections provide instructions on how to insert figures, tables, and equations in your document.

reduction. Figure 5 shows the mapping scheme of a convolutional kernel with the MSB weight duplication and n-bit input duplication. Similar to the MSB weight duplication with SSLs, the n-bit of input can be represented by activating 0 to 2n-1 BLs. For example, 2-bit input can be represented with 3 BLs using one BL as LSB input and two BLs as MSB input. The weights are duplicated 3 times to the BLs which are sharing the same SSL, the summed up current at SL is the multiplied result of the 2-bit input and the 2-bit weight. Therefore, the input vector cycles are reduced by using the multi-bit input duplication which means the throughput can be improved. Depending on the kernel size of the layer, we could duplicate up to 255 times for the 8-bit input duplication.

We defined the max-bit input duplication as the n-bit input of a particular convolutional kernel (size $K=K_C \times K_W \times K_H$) is duplicated as much as possible, where $K\times(2^n-1)$ does not exceed the number of BLs in a designed subarray. In this work, the NAND array is designed to have 13824 BLs to duplicate the largest convolutional kernel of VGG-8 or ResNet (3×3×512) three times, e.g., 2-bit input duplication. The max-input duplication could adaptively adjust the copies of input vectors depending on the kernel size and the number of BLs. In the case of fully connected (FC) layers, we used 2-bit input duplication for all FC layers for simplicity.

3.2 Subarray Duplication

Owing to the high memory density of 3D NAND, one subarray already has 81 Mb cells. Our chip hierarchy defines the minimum number of subarrays per PE and the minimum number of PEs per tile as four, so the minimum tile capacity is 1296 Mb which is large enough to store the today's large DNNs such as VGG-16 (138MB with 8-bit weight). Our selected VGG-8 model for CIFAR-10 dataset contains 12.37MB weight, so we can duplicate the weights to the other subarrays, PEs, and tiles even after SSL duplication and multi-bit input duplication. How many times to duplicate the subarray (or PE, tile) depends on the targeted total chip size. We can compute the different input vectors in parallel with such duplication.

4 BENCHMARKING RESULTS

We defined the 3D NAND subarray dimension and on/off current as discussed in section 2, then the widely used open-source simulator DNN+ NeuroSim [16] version 1.1 was customized as a benchmarking framework to estimate the area, memory utilization, latency and energy consumption of the designed 3D NAND based CIM architecture.

Table 2 shows the input feature map and kernel size information of the VGG-8 network on CIFAR-10 dataset. The images of CIFAR-10 dataset have 32×32×3 input feature map (IFM). VGG-8 network has six convolution (Conv) layers and three 2×2 max-pooling layers

Table 2: VGG-8 network topology

Layer	Type	IFM	Kernel size
1	Conv	32×32×3	3×3×3, 128
2	Conv	32×32×128	3×3×128, 128
	Pool	32×32×128	2×2
3	Conv	16×16×128	3×3×128, 256
4	Conv	16×16×256	3×3×256, 256
	Pool	16×16×256	2×2
5	Conv	8×8×256	3×3×256, 512
6	Conv	8×8×512	3×3×512, 512
	Pool	8×8×512	2×2
7	FC	8192	8192×1024
8	FC	1024	1024×10

Table 3: Total chip area and peripheral circuit breakdown result with various peripheral logic technology nodes.

Technology node (LSTP)	**32nm**	**14nm**	**7nm**
Number of Tiles	4	9	12
Total Area [mm^2]	17.91	18.12	19.50
Cell Array [mm^2]	5.10	11.48	15.30
Cell Array Efficiency	28.5%	63.2%	78.4%
Chip Capacity	1.13 Gb	2.53 Gb	3.38 Gb
Interconnect [mm^2]	2.11	1.49	0.90
ADC [mm^2]	1.26	0.44	0.15
Accum [mm^2]	2.33	1.30	0.67
Other [mm^2]	7.08	3.39	2.48

where every two convolutional layers are followed by one max-pooling layer, and 2 FC layers at the end. The input and weight precisions are both 8-bit.

Table 3 shows the designed chip area and breakdown result with various logic transistor technology nodes. Because we limited the total chip size under 20mm^2 in this estimation, the number of tiles and total chip capacity vary with technology node. The 32WL 3D NAND process with 32 nm logic process has 28.5% cell array efficiency, while 7 nm logic process has 78.4%.

Basically, we mapped the weight of the one layer of DNN to the one WL of 3D NAND unless the number of layers is larger than the number of total WLs. Table 4 shows the layer-by-layer weight duplication strategy of VGG-8 network on the designed architecture with 32 nm technology node. The shallow layers can be duplicated many times because the kernel size of the layers is small. The layer 1 has 128 kernels which means 512 blocks (8 subarrays) are required to store 8-bit weight. Consequently, the layer 1 can be duplicated 8 times to the other subarrays in the 32nm technology node, so 8 times faster parallel computing is possible.

In terms of the max-bit input duplication, the layer 1 has only 27 of kernel size so the weights can be duplicated 255 times to other BLs for 8-bit input duplication. The 8-bit input can be computed in only one cycle. The layer 2 and 3 are duplicated 7 times (3-bit input, 3 cycle), and other layers are duplicated 3 times (2-bit input, 4 cycle). The shallow layers have large IFM size so that they can fully take the advantage of duplication in terms of latency improvement. Finally, total duplicated weights are stored in 110.25MB cells which is 8.9 times larger than the original size (12.37MB).

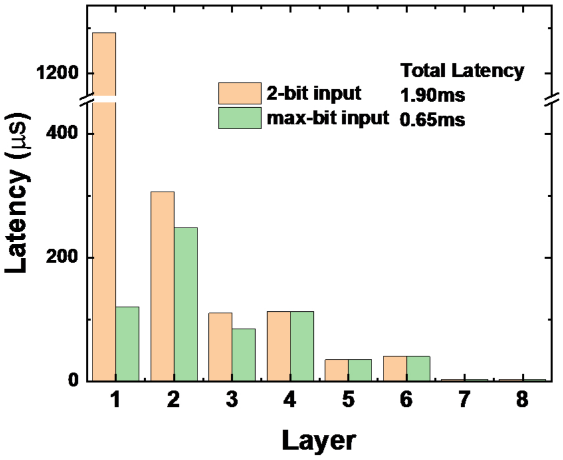

Figure 6: Simulated layer-by-layer latency result on VGG-8 network for CIFAR-10 dataset.

The memory utilization of each layer is also shown in Table 4. Since the subarray duplication method duplicates the weight to whole blocks of the chip, 100% of the blocks are utilized irrespective of the kernel size of the layer. On the other hand, the number of activated BLs always depends on the kernel size and multi-bit input duplication, so it is the key factor for the memory utilization. For instance, the layer 4 activates 6912 BLs in subarray of the entire chip, then the memory utilization is 50% as same as the number of activated BLs (6912) over the number of BLs in subarray (13824).

Figure 6 and Figure 7 show the simulated layer-by-layer latency and energy consumption on VGG-8 network for CIFAR-10 dataset with a uniform 2-bit input duplication and an adaptive max-bit input duplication. The shallow layers cause slow latency and large energy consumption because their large IFM size needs many times of computation. The max-bit input duplication can not only significantly reduce the latency of the first layer (8-bit input), but also decrease the energy consumption by 5%. The number of activated BLs is 85 times larger than that of 2-bit input duplication, so the energy consumed in single input cycle for charging the BLs and BL switch matrix increases as duplicated. However, the energy consumption in other modules does not change compared to the single input cycle, therefore the total energy rather decreases because the number of input cycles decreases 4 times.

Figure 8 shows the breakdown of the energy consumption of the max-bit input duplication case. Owing to the multi-bit input duplication which can effectively reduce the number of NAND array operations, NAND array only consumes 22.29% of total energy while 46.64% is consumed in interconnect buses in H-tree routing. It implies that input/output energy is also critical for CIM chip design, so further optimization of the interconnect bus and the input/output

Table 4: Weight duplication and speed up of VGG-8 on total chip (32nm) and memory utilization of each layer.

	Weight size(8-bit weight precision)	Subarray duplication	max-bit input duplication	MSB weight duplicate	1cell is duplicated	Required number of cells	Memory Utilization of each layer	TotalSpeed-up
Layer1	27 Byte × 128	8×	255× (8bit)	1.5×	3060×	10.09 MB	49.8%	128×
Layer2	1152 Byte × 128	8×	7× (3bit)		84×	11.81 MB	58.3%	48×
Layer3	1152 Byte × 256	4×	7× (3bit)		42×	11.81 MB	58.3%	24×
Layer4	2304 Byte × 256	4×	3× (2bit)		18×	10.12 MB	50.0%	16×
Layer5	2304 Byte × 512	2×	3× (2bit)		9×	10.12 MB	50.0%	8×
Layer6	4608 Byte × 512	2×	3× (2bit)		9×	20.25 MB	100.0%	8×
Layer7	8192 Byte × 1024	1×	3× (2bit)		4.5×	36 MB	88.9% (2WL)	4×
Layer8	1024 Byte × 10	1×	3× (2bit)		4.5×	0.044 MB	22.2%	4×
Total	12.37Mbyte					110.25 MB	62.94%	

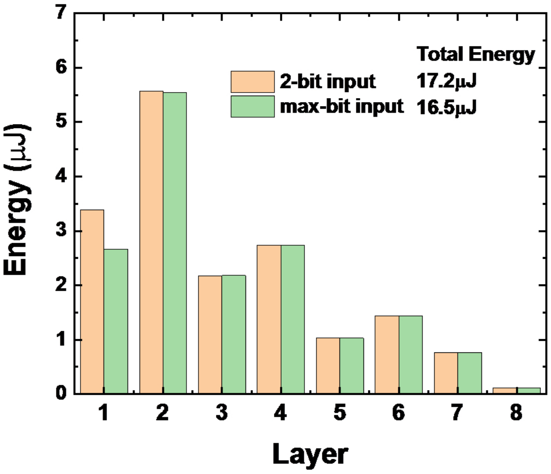

Figure 7: Simulated layer-by-layer energy consumption on VGG-8 network for CIFAR-10 dataset.

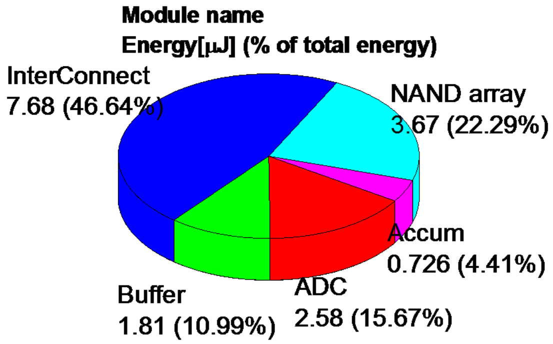

Figure 8: Energy consumption breakdown of 3D NAND based accelerator on VGG-8 network for CIFAR-10 dataset with max-bit duplication. Accum logic contains accumulation, max pooling and activation logic.

feature map dataflow is essential future work to achieve even higher energy efficiency.

Table 5 shows the benchmarking result comparison across the state-of-the-art device technologies. Evaluated SRAM and RRAM based CIMs have 128×128 cell array size [16]. Although all the duplication methods are applied on 3D NAND array, the total chip size is only 17.91 mm^2 which is significantly smaller than the CIM chips with other technologies (73.53 mm^2 for RRAM and over 100 mm^2 for SRAM). Thanks to the multiple duplication methods in 3D NAND array, 3D NAND Flash based accelerator has shown the 1545.6 frame per second (FPS) which is the highest throughput compared to the SRAM and RRAM technology at the same 32 nm technology node with the low-standby power (LSTP) library.

For SRAM, there are two designs, one is the conventional row-by-row read out with the digital adders at the edge of the array to perform near-memory compute, and the other is the SRAM-CIM mode where the peripheral circuits are modified to support parallel read-out in SRAM array, as recently demonstrated in several silicon prototypes [21-22]. Originated from its large R_{on} in the 3D NAND array, the large number of BL inputs results in the huge parallelism with 37.1 TOPS/W energy efficiency, outperforming other technologies at the same technology node. It is noted that the TOPS/W reported here is for 8-bit input by 8-bit weight MAC operations, and it is equivalent to 148.4 TOPS/W for 4-bit input by 4-bit weight operations if using the same precision definitions in some of the prior works. For example, the 28nm SRAM-CIM macro reported 68 TOPS/W [22]; and the 22nm RRAM-CIM macro reported 29 TOPS/W [23]. Our proposed 3D NAND-CIM design showed 2.2-5.1 times improvement in energy efficiency to state-of-the-art CIM designs.

5 DEEPER NETWORK BENCHMARKING FOR 3D NAND

As explained in the previous sections, we mapped each layer of the VGG-8 network to the single WL (except layer 7 which is the largest FC layer that does not fit into a single WL). Because multiple WLs

Table 5: Benchmark results of various DNN accelerators on VGG-8 for CIFAR-10 based on SRAM (both sequential and parallel read-out) and RRAM-CIM, 3D NAND-CIM at 32nm technology. 8-bit input and 8-bit weight precision were used.

Technology node	32nm (LSTP)			
Device	SRAM		RRAM [8]	3D NAND (max-bit input)
ADC Precision	Sequential: 1-bit	CIM: 4-bit	5-bit	7-bit
Cell Precision	1-bit		2-bit	1-bit
R_{on} (ohm)	-	-	6k	100M
On/Off Ratio	-	-	17	2000
Area (mm^2)	109.00	103.12	73.53	17.91
Memory Utilization (%)	98.73%	98.73%	96.86%	*62.94% (**17.70%)
Latency (ms)	1.25	0.76	1,262	0.65
Dynamic Energy (uJ)	147.64	42.70	30.27	16.5
Leakage power (mW)	2.61	2.25	0.58	0.12
Energy Efficiency (TOPS/W)	4.08	13.79	19.76	37.10
Throughput (FPS)	797.77	1318.77	792.3	1545.6

*This memory utilization is calculated at used WLs only.

**Percentage in whole WLs (32WLs).

cannot be read at the same time in the 3D NAND array, duplicating the weights to other WLs (WL10-32) does not have any advantage for the speed up, so the large number of WLs remain empty and have not been fully exploited on the relatively shallow DNNs.

Therefore, we have considered the deeper network such as the ResNet [18] families for ImageNet dataset to be mapped to the 3D NAND based CIM. First, we mapped our 32WL 3D NAND CIM on the various ResNet networks, from ResNet-18 to ResNet-152. The deeper the network is, the more WLs can be exploited, so we could achieve higher memory utilization of whole 3D NAND array as shown in Table 6. Because ResNet-34 to ResNet-152 have more than 32 layers, some (or whole) of the WLs store the weight of more than two layers. As can be seen in the memory utilization in Table 6, ResNet-18 to ResNet-50 have enough storage space to fully utilize max-bit input duplication. However, ResNet-101 and 152 cannot utilize the max-bit input duplication for whole layer so we optimized the number of input duplication layer by layer. The layers with large kernel size (3×3×512 and 1×1×1024) were not duplicated to other BLs for multi-bit input duplication.

Irrespective of the number of layers in the network, the energy efficiency on ResNet is relatively lower than that on VGG-8. The energy efficiency of individual layer is strongly related to the kernel size. Regardless of the kernel size, the whole 3D NAND array and the 7-bit ADC should operate once for one weighted sum computation without significant variation in energy consumption. Therefore, the small convolution kernels such as 3×3×64 in the shallow layers and 1×1 convolutional kernels (only in ResNet-50, ResNet-101 and ResNet-152) tend to have low energy efficiency. Not only the 3D NAND based design, other technology based designs also have lower energy efficiency on ResNet (Table 7) over VGG-8 with same reason.

Second, we evaluated the area and performance on ResNet-18 for ImageNet dataset with various device technologies the same as VGG-8 case, as shown in Table 7. Because 3D NAND based CIM could utilize more WLs and duplicate more than the case of VGG-8, the throughput is 3.4 and 5.5 times higher than that of SRAM and RRAM respectively (1.2 and 1.9 times higher for VGG-8). We could also achieve the energy efficiency which is 87% and 78% higher than that of SRAM and RRAM. Again, the 3D NAND chip area is just 17.91mm2, but the SRAM-CIM chip are exploded to over 100 mm2.

6 CONCLUSION

In this paper, we designed the architecture of 3D NAND Flash based CIM accelerator and evaluated the inference performance on representative DNN models using the modified DNN+NeuroSim framework. The high density of 3D NAND array is exploited by duplicating both the weights and the inputs for the throughput improvement. The outstanding energy efficiency has been achieved through the summing large input vectors at once for the highly parallel computing, showing tremendous chip area benefits than other device technologies. 3D NAND based CIM has also shown higher performance especially on the convolutional neural networks with larger kernel size, and suitable for mapping the very large-scale networks. Therefore, it is of great interests to explore the mapping of GB-scale models for speech recognition, language translation and recommendation system in the future work.

ACKNOWLEDGMENTS

We would like to thank to Hang-Ting Lue, Po-Kai Shu and Teng-Hao Yeh in Macronix International Co. for the technical discussions and providing the electrical parameters of 3D NAND array.

REFERENCES

[1] N. P. Jouppi, C. Young, N. Patil, D. Patterson G. Agrawal, R. Bajwa, S. Bates, *et al.*, "In-datacenter performance analysis of a tensor processing unit," *ACM/IEEE International Symposium on Computer Architecture (ISCA)*, 2017, pp. 1-12. DOI:https://doi.org/10.1145/3079856.3080246.

[2] S. Yu, "Neuro-inspired computing with emerging non-volatile memory," Proceeding of the IEEE, vol. 106, no. 2, pp. 260-285, 2018, DOI: 10.1109/JPROC.2018.2790840.

[3] M. Prezioso, F. Merrikh-Bayat, B. D. Hoskins, G. C. Adam, K. K. Likharev and D. B. Strukov, "Training and operation of an integrated neuromorphic network based

Table 6: Benchmarking result of 3D NAND-CIM architecture on ResNet-18 to ResNet-152 for ImageNet dataset. The chip area is fixed to 17.91mm^2.

	ResNet-18	ResNet-34	ResNet-50	ResNet-101	ResNet-152
ADC Precision			7-bit		
Cell Precision			1-bit		
Weight Parameter Size (8-bit precision)	11.5 MB	21.6 MB	22.7 MB	41.7 MB	48.6 MB
Memory Utilization	33.5%	63.5%	68.9%	98.6%	99.2%
Latency (ms)	5.07	7.93	21.7	40.3	61.7
Energy (uJ)	138	203	316	510	740
Leakage (mW)	0.12	0.12	0.12	0.12	0.12
Operation (GOPs)	1.80	3.64	3.50	7.21	10.9
TOPS/W	12.95	17.87	10.99	14.00	14.61
Throughput	197.24	126.10	46.08	24.81	16.21

Table 7: Benchmarking result of SRAM, RRAM, 3D NAND based DNN accelerators on ResNet-18 for ImageNet dataset. 8-bit input and 8-bit weight precision were used.

Technology node	32nm (LSTP)		
Device	SRAM-CIM	RRAM[8]	3D NAND
ADC Precision	4-bit	5-bit	7-bit
Cell Precision	1-bit	2-bit	1-bit
Ron (ohm)	-	6k	100M
On/Off Ratio	-	17	2000
Area (mm^2)	103.49	80.12	17.91
Memory Utilization (%)	94.38%	91.13%	33.50%
Latency (ms)	17.04	27.98	5.07
DynamicEnergy (μJ)	297.8	290.0	138
Leakage power (mW)	2.34	0.60	0.12
Energy Efficiency (TOPS/W)	6.89	7.27	12.95
Throughput (FPS)	58.68	35.74	197.24

on metal-oxide memristors," *Nature* 521, pp. 61–64, May. 2015, DOI: 10.1038/nature14441.

[4] J. Woo, K. Moon, J. Song, S. Lee, M. Kwak, J. Park, and H. Hwang, "Improved synaptic behavior under identical pulses using AlOx /HfO2 bilayer RRAM array for neuromorphic systems," *IEEE Electron Device Letters*, VOL. 37, NO. 8, pp. 994-997, Aug. 2016, DOI: 10.1109/LED.2016.2582859.

[5] F. Cai, J. M. Correll, S. H. Lee, Y. Lim, V. Bothra, Z. Zhang, M. P. Flynn, W.D. Lu M.A. Zidan, J. P. Strachan, and W. D. Lu, "A fully integrated reprogrammable memristor–CMOS system for efficient multiply–accumulate operations," *Nature Electronics* **2** (7), 290-299, DOI: 10.1038/s41928-019-0270-x.

[6] W. Wu, H. Wu, B. Gao, P. Yao, X. Zhang, X. Peng, S. Yu, H. Qian, "A methodology to improve linearity of analog RRAM for neuromorphic computing," *Symposium on VLSI Technology*, June, 2018, art. No. 8510690, pp. 103-104, DOI: 10.1109/VLSIT.2018.8510690.

[7] T. Gokmen, Y. Vlasov, "Acceleration of deep neural network training with resistive cross-point devices: design considerations," *Frontiers in Neuroscience*, 10, 333, 2016, DOI: 10.3389/fnins.2016.00333.

[8] P. Jain, U. Arslan, M. Sekhar, B. C. Lin, L. Wei, T. Sahu, J. Alzate-vinasco, A. Vangapaty, M. Meterelliyoz, N. Strutt, A. B. Chen, P. Hentges, P. A. Quintero, C. Connor, O. Golonzka, K. Fischer, F. Hamzaoglu, "A 3.6Mb 10.1Mb/mm2 Embedded Non-Volatile ReRAM Macro in 22nm FinFET Technology with Adaptive Forming/Set/Reset Schemes Yielding Down to 0.5V with Sensing Time of 5ns at 0.7V," *IEEE International Solid- State Circuits Conference* (*ISSCC*), San Francisco, CA, USA, 2019, pp. 212-214, DOI: 10.1109/ISSCC.2019.8662393.

[9] S. Ambrogio, P. Narayanan, , H. Tsai, R. M. Shelby, I. Boybat, C. Nolfo, S. Sidler, M. Giordano, M. Bodini, N. C. P. Farinha, B. Killeen, C. Cheng, Y. Jaoudi and G. W. Burr, "Equivalent-accuracy accelerated neural-network training using analogue memory," *Nature 558*, pp. 60–67, 2018, DOI: 10.1038/s41586-018-0180-5.

[10] W. Kim, R. L. Bruce, T. Masuda, G. W. Fraczak, N. Gong, P. Adusumilli, S. Ambrogio, H. Tsai, J. Bruley, J. -P. Han, M. Longstreet, F. Carta, K. Suu and M. BrightSky, "Confined PCM-based analog synaptic devices offering low resistance-drift and 1000 programmable states for deep learning, " *Symposium on VLSI Technology*, June, 2019, pp. T66-67, DOI: 10.23919/VLSIT.2019.8776551.

[11] X. Guo, F. Merrikh-Bayat, M. Bavandpour, M. Klachko, M. R. Mahmoodi, M. Prezioso, K. K. Likharev, D. B. Strukov, "Fast, energy-efficient, robust, and reproducible mixed-signal neuromorphic classifier based on embedded NOR flash memory technology," *IEEE International Electron Devices Meeting (IEDM)*,San Francisco, CA, 2017, pp. 6.5.1-6.5.4, DOI: 10.1109/IEDM.2017.8268341.

[12] P. Wang, F. Xu, B. Wang, B. Gao, H. Wu, H. Qian, S. Yu, "Three-dimensional NAND Flash for vector-matrix multiplication," *IEEE Trans. VLSI Systems*, vol. 27, no. 4, pp. 988-991, 2019, DOI: 10.1109/TVLSI.2018.2882194.

[13] H. -T. Lue, P. -K. Hsu, M. -L. Wei, T. -H. Yeh, P. -Y. Du, W. -C. Chen, K. -C. Wang and C. -Y. Lu, "Optimal design methods to transform 3D NAND Flash into a high-density, high-bandwidth and low-power nonvolatile computing in memory (nvCIM) accelerator for deep-learning neural networks (DNN)," *IEEE International Electron Devices Meeting (IEDM)*,San Francisco, CA, 2019, pp. 38.1.1-38.1.4, DOI: 10.1109/IEDM19573.2019.8993652.

[14] S. -T. Lee, H. Kim, J. -H. Bae, H. Yoo, N. Y. Choi, D. Kwon, S. Lim, B. -G. Park, J. -H. Lee, "High-Density and Highly-Reliable Binary Neural Networks Using NAND Flash Memory Cells as Synaptic Devices," *IEEE International Electron Devices Meeting (IEDM)*, San Francisco, CA, USA, 2019, pp. 38.4.1-38.4.4, DOI: 10.1109/IEDM19573.2019.8993478

[15] D. -H. Kim, H. Kim, S. Yun, Y. Song, J. Kim, S. Joe, K. Kang, J. Jang, H. Yoon, K. Lee, M. Kim, J. Kwon, J. Jo, S. Park, J. Park, J. Cho, S. Park, G. Kim, J. Bang, H. Kim, J. Park, D. Lee, S. Lee, H. Jang, H. Lee, D. Shin, J. Park, J. Kim, J. Kim, K. Jang, I. H. Park, S. H. Moon, M. Choi, P. Kwak, J. Park, Y. Choi, S. Kim, S. Lee, D. Kang, J. Lim, D. Byeon, K. Song, J. Choi, S. J. Hwang, J. Jeong, "A 1Tb 4b/cell NAND Flash Memory with tPROG=2ms, tR=110μs and 1.2Gb/s High-Speed IO Rate," *IEEE International Solid- State Circuits Conference (ISSCC)*, San Francisco, CA, USA, 2020, pp. 218-220, DOI: 10.1109/ISSCC19947.2020.9063053

[16] X. Peng, S. Huang, Y. Luo, X. Sun, S. Yu, "DNN+NeuroSim: An end-to-end benchmarking framework for compute-in-memory accelerators with versatile device technologies," *IEEE International Electron Devices Meeting (IEDM)*2019, San Francisco, USA, DOI: 10.1109/IEDM19573.2019.8993491. Open-source code available at: https://github.com/neurosim/DNN_NeuroSim_V1.1

[17] K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *International Conference on Learning Representations (ICLR)*, May. 2015, San Diego, USA.

[18] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, Jun. 2016, Las Vegas, USA, DOI: 10.1109/CVPR.2016.90.

[19] Techinsights, Samsung 32L 3D NAND teardown report.

[20] S. Yu, X. Sun, X. Peng, S. Huang, "Compute-in-memory with emerging nonvolatile-memories: challenges and prospects," *IEEE Custom Integrated Circuits Conference (CICC)*2020, Boston, USA, DOI: 10.1109/CICC48029.2020.9075887

[21] Q. Dong, M. E. Sinangil, B. Erbagci, D. Sun, W. Khwa, H. Liao, Y. Wang, J. Chang, "A 351TOPS/W and 372.4GOPS Compute-in-Memory SRAM Macro in 7nm FinFET CMOS for Machine-Learning Applications", *IEEE International Solid- State*

Circuits Conference (*ISSCC*), San Francisco, CA, USA, 2020, pp. 242-244, DOI: 10.1109/ISSCC19947.2020.9062985

[22] X. Si, Y. -N. Tu, W. -H. Huanq, J. -W. Su, P. -J. Lu, J. -H. Wang, T. -W. Liu, S. -Y. Wu, R. Liu, Y. -C. Chou, Z. Zhang, S. -H. Sie, W. -C. Wei, Y. -C. Lo, T. -H. Wen, T. -H. Hsu, Y. -K. Chen, W. Shih, C. -C. Lo, R. -S. Liu, C. -C. Hsieh, K. -T. Tang, N. -C. Lien, W. -C. Shih, Y. He, Q. Li, M. Chang, "A 28nm 64Kb 6T SRAM Computing-in-Memory Macro with 8b MAC Operation for AI Edge Chips," *IEEE International Solid- State Circuits Conference (ISSCC)*, San Francisco, CA, USA, 2020, pp. 246-248, DOI: 10.1109/ISSCC19947.2020.9062995.

[23] C. -X. Xue, T. -Y. Huang, J. -S. Liu, T. -W. Chang, H. -Y. Kao, J. -H. Wang, T. W. Liu, S. Y. Wei, S. P. Huang, W. -C. Wei, Y. -R. Chen, T. -H. Hsu, Y. -K. Chen, Y. -C. Lo, T. -H. Wen, C. -C. Lo, R. -S. Liu, C. -C. Hsieh, K. -T. Tang, M. F. Chang, "A 22nm 2Mb ReRAM Compute-in-Memory Macro with 121-28TOPS/W for Multibit MAC Computing for Tiny AI Edge Devices," *IEEE International Solid-State Circuits Conference (ISSCC)*, San Francisco, CA, USA, 2020, pp. 244-246, DOI: 10.1109/ISSCC19947.2020.9063078.