Extended Abstracts of the 2020 International Conference on Solid State Devices and Materials, VIRTUAL conference, 2020, pp95-96

A General Assessment of Computing-In-Memory (CIM) for Deep Neural Network (DNN) with Flash Memory Devices

Hang-Ting Lue, Po-Kai Hsu, Keh-Chung Wang, and Chih-Yuan Lu

Macronix International Co., Ltd., 16 Li-Hsin Road, Hsinchu Science Park, Hsinchu, Taiwan (e-mail: htlue@mxic.com.tw)

Abstract-. This paper provides a general overview of using Flash memory devices to realize the CIM for inference accelerator of DNN. Two examples of Flash memory devices are studied, 3D NAND and vertical split-gate NOR. Critical criteria for CIM are addressed. After practical engineering studies we noted that the intrinsic TOPS/W and TOPS/mm^2 of CIM for MAC computing may not far exceed digital solution. However, the major advantage of Flash-based CIM is to provide a high-density non-volatile memory to support heavy-weight DNN and save data (weight) movements. We also suggest that the near-memory digital computing with Flash is an alternative viable path to support the high-density neural network.

I. Introduction:

Computing-in-Memory (CIM) [1,2] is widely studied in recent years. The major motivation is that AI applications such as image recognition require huge data movements during computing, while the conventional Von-Neumann architecture has a substantial bottleneck in the memory bandwidth. The digital approach for AI accelerator generally requires high-density on-chip SRAM and high-bandwidth DRAM to optimize the system performances.

On the other hand, CIM aims to provide a fundamentally much efficient way to save data movements. There are quite diverse topics in CIM. In this work, we focus on the non-volatile CIM using Flash memory devices as an inference accelerator.

Figure 1(a) illustrates the conventional Von-Neumann architecture to use Flash memory device. The digital data (mostly weight) are stored in the Flash, and the computing mostly involves SRAM and LOGIC circuits. Using Flash memory directly in computing is very inefficient due to the mush slower read performances. However, when the weight number of DNN far exceeds the SRAM capacity, it is beneficial to design Flash CIM.

Figure 1(b) shows the Flash CIM, which aims to provide a direct MAC (multiply and accumulate) computing inside the array. We just have to move in and out the meta data (feature maps, input and output) without the need to move out the weight. If the DNN network is heavy-weight (such as >100Mb) with small repeated usage of weight, the CIM can automatically save lots of data movements.

II. Two examples of Flash memory devices for CIM

(a) 3D NAND CIM [2]:

Figure 2 summarizes the 3D NAND CIM. We use BL's as input, and the weights are the conductance of cell current (Icell). We suggest single-level weights and inputs for better reliability and design. The plural SSL's in 3D NAND provide a way to represent multi-bit resolution of weight. To produce 4-bit resolution network (4I4W), we can apply "shifter and adder design" to transform to 4bit, at the penalty of more memory usage. The major advantage of 3D NAND CIM is that it naturally has the highest density. Through device tuning we can get very small Icell of ~ 2nA, with large ON/OFF ratio of > 4 orders. The cell current is essentially the saturated current of the NAND string. The Icell distribution consists of array loading effect and has a finite distribution. 3D NAND CIM allows parallel computing with large number of inputs due to the small Icell and small leakage. The large parallelism compensates the slower access time, and is very efficient to carry out a heavy-weight (>1Gb) fully-connected (FC)-like network.

(b) Vertical Split-Gate NOR [3]:

Figure 3 summarizes the vertical split-gate Flash CIM. We make select gate (SG) as inputs (at low voltage ~0.5V), and the weights are the trans-conductance of cell current (Icell). The plural bitline transistor (BLT) in NOR-type array design readily provides the multi-bit resolution of weight. Vertical split-gate Flash can be treated as a scaled version of ordinary embedded Flash (eFlash) that provides higher memory density. The major merit of this device is that it has very large ON/OFF ratio >7 orders, and has a flexible Icell ranging from 150nA to 1.5uA, with tight distribution possible. It is totally read-disturb free since memory gate can be applied 0V during read.

III. General Assessment of Flash memory CIM

We take the image recognition with VGG7 neural network as an example for CIM simulation, as shown in **Fig. 4**. **Figure 5** summarizes the major design requirements and criteria of Flash memory devices:

(1) **Memory density**: Ranging from 100Mb to 10Gb to support heavy-weight DNNs.

(2) **Transistor ON/OFF ratio of Icell**: > 4 orders of magnitude of "1" and "0" to support parallel sum of >1000 inputs.

(3) **Input signal design**: Need to allow low-voltage (<1V) bias at inputs to avoid charge pumping circuits that retards the power and speed performances. Also need dense inputs in the array. We don't recommend complex analog signals to produce multi bits since usually WL/BL RC loading is large that causes signal distortion.

(4) **Summed MAC current and ADC design**: The analog-to-digital converter (ADC) is the most critical design for CIM. From various simulations it is identified that ADC needs to support 8-10bit resolution even for a 4-bit network. To develop low-power and high-speed ADC is challenging. Through various reference works [4] and general design discussions it is recommended to design the MAC current ranging from 0.5uA to 128uA with 256 levels (8-bit). The estimated Tread ~ 150ns with optimal design, and the average MAC current is ~30uA (sparsity included). The ADC is the dominant factor for CIM performance and power.

(5) **Icell for CIM**: Icell needs to be flexible and tunable to support various network. For an FC network with large inputs >10'000, it is better to have small Icell ~ nA like 3D NAND. The large number of summation helps to cancel the cell variation and produce MAC closer to the mean value, according to the "central limit theory". One the other hand, for the first few convolution layers, the smaller input number require larger Icell ~ uA to meet the ADC design range. For 3D NAND, since Icell is essentially the Idsat of string and not tunable, we can repeatedly use many cells to produce the MAC current in ADC range. For vertical split-gate NOR, the tunable Icell makes it easier to meet the design target.

(6) **Calibration for MAC:** Calibration of ADC for MAC current is critically important to ensure the accuracy. Calibration is to design a known MAC value and tune the ADC circuit parameters to match the target. The calibration requires even higher-resolution (~10bit) ADC with some digital computing (shift and multiply) to match the desired MAC value, and the calibration parameters needs additional cache to store the data. The calibration can be made in a "on-the fly" way to compensate possible reliability aging effects of Icell drifts [5].

(7) **Icell sigma and RTN**: Both Icell variation and RTN sigma are suggested to be within 10% to ensure the accuracy. Moreover, the device needs to have small read disturb for intensive read.

(8) **Icell mean shift tolerance**: Both positive and negative shift of Icell mean need to be controlled within +/-10% to ensure the accuracy.

Figure 6 summarizes the CIM performances for Flash memory devices. The optimal TOPS/W and TOPS/mm^2 are ~40 and 1-5, respectively. It is probably similar, or only slightly better than digital solution. We think that the major bottlenecks are ADC performances that take hundreds of nano-second to respond and it's much slower than digital circuits, while it also needs sizable MAC currents which restrict the power consumption. Fortunately CIM allows higher parallelism to compute more inputs to compensate the slower ADC performances.

After a thorough study considering many practical issues, we'd like to admit that CIM may not far exceed digital performances for MAC. However, the major merit should be mostly "**to save data movements of weight**", especially for a heavy-weight network. Meanwhile, to guarantee the accuracy with such analog MAC computing is quite complex and requires some design overhead.

We'd like to point out that the digital-mode near-memory computing [6] in **Fig. 7** is a viable, mid-way approach before analog CIM is successful. Near-memory computing does not have superior TOPS/W performances for MAC, but it already saves data movements for a heavy-weight network. Digital-mode near-memory computing does not suffer the analog device challenges and is much easier to design.

References: **[1]** M. F. Chang, VLSI 2019, Short Course. **[2]** H. T. Lue, et al, s38-1, IEDM 2019. **[3]** H. T. Lue, et al, pp. 347-348, VLSI 2020. **[4]** C. X. Xue, et al, ISSCC 2020, 15.4, pp.243-245. **[5]** P. K. Hsu, et al. ,pp. 1-4, IMW 2020. **[6]** P. Y. Du, et al, pp.1-4, IMW 2019.

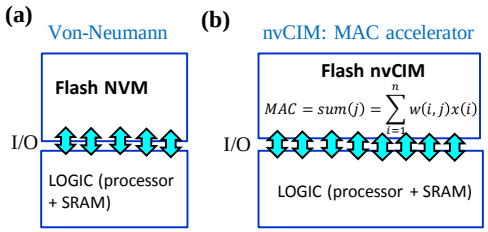


Fig. 1 (a) Conventional Von-Neumann approach to use Flash memory. Digital data is transferred to processor through I/O. The memory bandwidth limitation restricts the computing and power consumptions. **(b)** CIM approach to use Flash memory as a MAC accelerator. The Flash memory array stores the weight and can readily compute MAC without moving weights through I/O. This drastically reduces data movements and power consumptions.

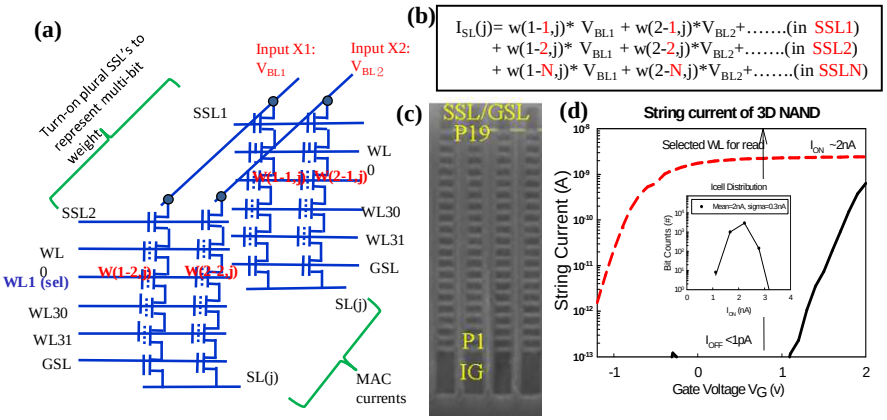


Fig. 2 A brief summary of using 3D NAND as CIM [2]. **(a)** Method of summing currents to represent MAC. BL's are used as inputs, and cell currents are summed at source line. Plural SSL's stands for multi-bit weight. **(b)** Equation for MAC. **(c)** The device structure of the SGVC 3D NAND, with extreme-thin body to get small Ion. **(d)** The IdVg curves. The inset shows the collected Icell distribution.

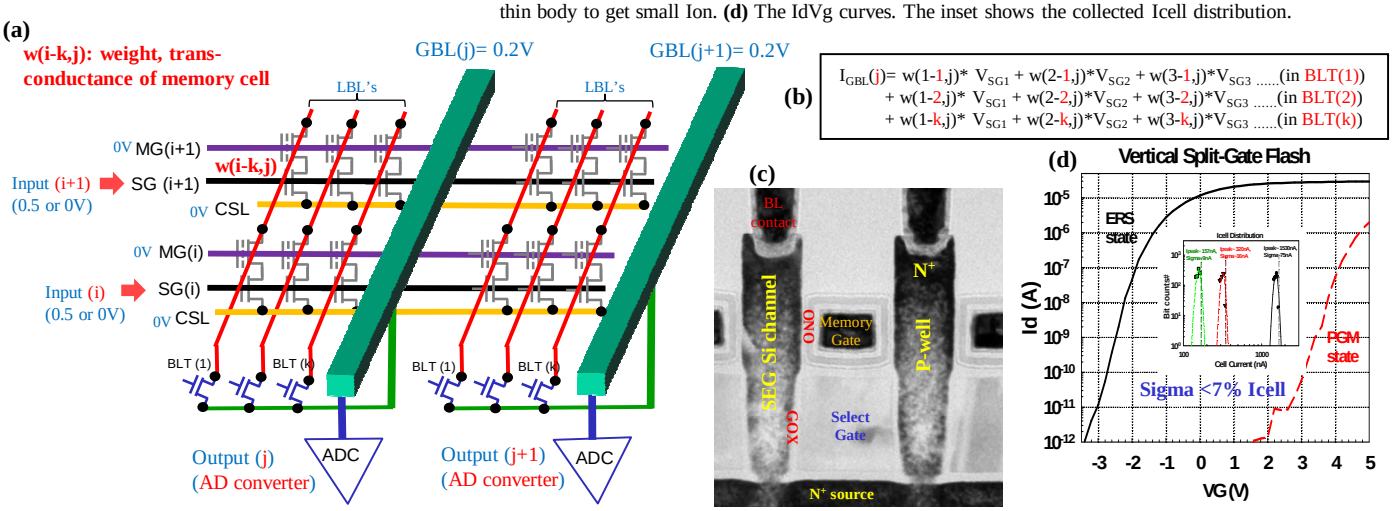


Fig. 3 A brief summary of using vertical split-gate Flash [3] as CIM. **(a)** Method of summing currents to represent MAC. SG's as inputs, and cell currents are summed at global BL. Plural BLT's stands for multi-bit weight. **(b)** Equation for MAC. **(c)** The device structure of the vertical split-gate NOR. **(d)** The IdVg curves. The inset shows the collected Icell distribution. The device can provide flexible Icell ranging from 150nA to 1.5uA. Tight distribution with sigma <7% of Icell mean value is produced.

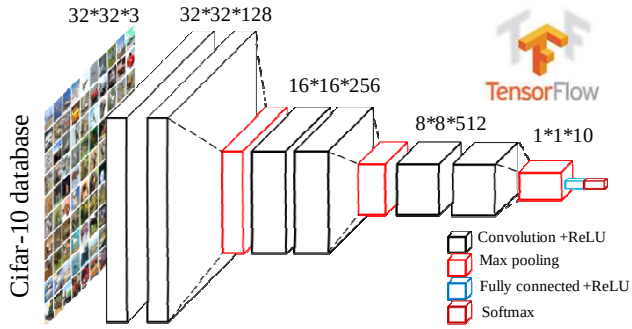
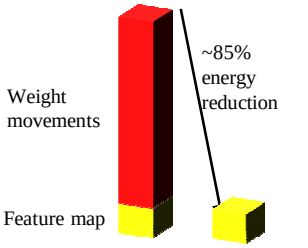


Fig. 4 Schematic of a VGG7 DNN network for CIM simulation.

Items for Flash Memory Device Requirements	Criteria/ Targets	Comments
Available memory density	Ranging from 100Mb to 10Gb for heavy-weight DNN	(a) High-density 3D Flash; (b) 3DIC chiplets to connect to CMOS circuit
Transistor ON/OFF ratio (for "1" and "0" of Icell)	>4 orders of magnitude to support MAC computing with >1000 inputs	To allow large number of summation
Input signal design	(1) Low-voltage (<1V) operation; (2) Small RC delay; (3) Dense inputs (>1000) in the array	To reduce the power consumption of extensive input signal
Summed MAC current range; ADC design	(1) MAC current: from 0.5uA to 128uA (2) 8-10bit resolution for ADC. Tread ~150ns.	For ADC design considerations
Icell range for CIM	Flexible range preferred. (a) ~nA for large inputs (~10'000); (b) ~uA for small inputs (<100)	To support a wide input # variations for CNN
Calibration of MAC	(1) To match a known MAC for calibration of ADC; (2) On-the-fly calibration	A necessary design overhead for Icell variations and drifts
Icell variation and RTN	(a) Sigma <10% of mean value; (b) RTN sigma < 10% of mean value	To maintain sufficient DNN accuracy
Icell mean shift tolerance	(a) Icell positive shift < 10% of target; (b) Icell negative shift < 10% of target	To maintain sufficient DNN accuracy

Fig. 5 A summary table to illustrate key factors for CIM using Flash devices.



	3D NAND (16OP SGVC)	Vertical Split-Gate Flash
Flash Memory density for VGG7 (4I4W)	1.03Gb (16 cores)	575Mb (16 cores)
CIM die size	~1.5 mm²	17.3 mm²
SRAM (for feature map)	64KB	64KB
Frame/sec	~14K	29K
TOPS/W	~40	~37
TOPS/mm²	~5	~1

Fig. 6 Summary of advantages in CIM for VGG7 (4-bit). (a) The data movement energy is saved by ~85%. (b) The detail performances.

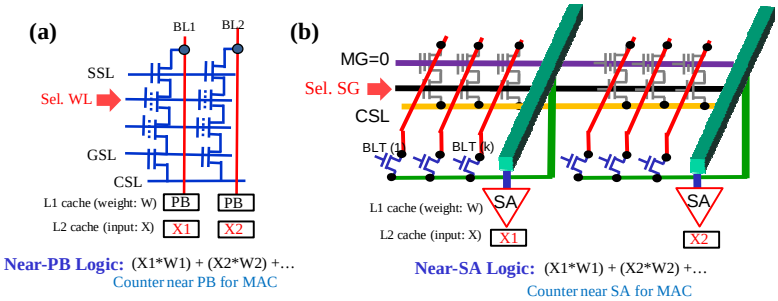


Fig. 7 Near-memory digital computing for (a) 3D NAND [5]; (b) Vertical split-gate NOR.