The Implications of Ferroelectric FET Device Models to the Design of Computing-in-Memory Architectures

Dayane Reis¹, Michael Niemier¹, and X. Sharon Hu¹

¹Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN, USA
e-mails: { dreis, mniemier, shu }@nd.edu

Abstract— Data transfer between a processor and memory frequently represents a bottleneck with respect to improving application-level performance. Computing-in-memory (CiM), where logic and arithmetic operations are performed in memory, could significantly reduce both energy consumption and computational overheads associated with data transfer. This work presents a revisited study of FeFET-CiM, a CiM architecture capable of performing Boolean ((N)AND, (N)OR, X(N)OR, INV) as well as arithmetic (ADD) operations between words in memory. In this study, we employ two types of FeFET-based memory cells in the CiM architecture. Namely, the 2T+1FeFET and the 1-FeFET memory cells. The use of these two types of memory cells in the FeFET-CiM architecture is enabled by two distinct models for FeFET devices. The FeFET-CiM architecture based on 2T+1FeFETs (1FeFETs) offers an average speedup of ∼2.5X (∼1.1X) and energy reduction of ∼1.7X (∼1.4X) when compared to a SRAM baseline across 12 benchmark programs. Despite smaller speedups and energy savings enabled by 1FeFET-CiM when compared to 2T+1FeFET-CiM, 1FeFET memory arrays may offer up to ∼5.3X density improvements when compared to conventional 6T-SRAM arrays. Furthermore, 1FeFET-CiM offers significant application-level improvements when compared to a counterpart STT-CiM architecture.

Index Terms— Computing-in-Memory; Non-traditional Computer Architectures; Ferroelectric FETs; FeFETs.

I. INTRODUCTION

With the rise of “Big Data”, the Internet of Things (IoT), and computing in the cloud, the so-called “data-intensive” applications (e.g., in the domains of machine learning, bioinformatics, financial, etc.) have become ever popular. Data-intensive applications deal with a large workload, which leads to concerns about the practicality of transferring high volumes of data between processor and main memory due to the usually far restricted processor-memory bandwidth (i.e., the “memory wall”). These demands of data-intensive applications have exacerbated the need for larger quantities of available on-chip memory (i.e., caches).

Meanwhile, as transistors shrink in the conventional complementary metal-oxide semiconductor (CMOS) technology, quantum effects and current leakage become very significant, which result in prohibitively high static power. Such high leakage associated with the low density of 2-D Static Random Access Memories (SRAMs) make it extremely challenging for CMOS to satisfy the requirements of data-intensive applications.

Computing-in-memory (CiM) has been proposed as a solution for overcoming the “memory wall”, as latency and energy required for data transfers can be significantly reduced [1]. With a CiM architecture, a subset of logic and memory operations associated with a given task are performed in-memory. CiM exploits large, internal bandwidth to achieve parallelism, which reduces latency and saves energy due to fewer external memory references. CiM can be implemented at the level of CPU caches or as a main memory [2, 3]. Regardless of where in the memory hierarchy CiM is implemented, it is possible to boost energy and latency gains from the CiM architecture by employing emerging technologies.

For instance, it is possible to design dense and low power CiM architectures based on resistive, non-volatile memories such as Spin-Transfer Torque Random Access Memory (STT-RAM) [4, 5], Resistive Random-Access Memory (ReRAM) [6, 7] and Phase-Change Memory (PCM) [8]. The aforementioned non-volatile memories are built from two terminal resistive devices, and have shared write and read bitlines as with SRAM. That said, there are considerable differences between currents for read and write accesses. High write currents can prevent read-disturbance failures in resistive memories. However, write energy and latency of STT-RAM, ReRAM and PCM cells are typically quite high when compared to conventional SRAM. Furthermore, the low $I_{on}/I_{off}$ ratios for read currents of resistive devices (on the order of $10^{7}$) may result in degraded sensing margins.

Transistors with integrated ferroelectrics (FE) offer unique possibilities for low power and dense memories, as they can be used just as regular transistors or also retain their logic state without power supply due to non-voltatility. Unlike SRAM, STT-RAM, ReRAM and PCM designs, Ferroelectric FET (FeFET)-based memory cells employ the FeFET’s three-terminal structure to achieve a write scheme based on the gate-source voltage — i.e., in contrast to other write schemes based on variable resistance tuned by write currents. These unique characteristics of the FeFET devices make them a very competitive alternative for the design of CiM architectures for various applications.

Reference [9] introduces a pioneer CiM architecture devised for 2T+1FeFET memory cells (FeFET-CiM). FeFET-CiM was originally designed and simulated by employing the single domain model described in Sec. II.B.1. Promising results are reported with respect to speedup and energy savings for single CiM operations (e.g., the in-memory addition of 32-bit words) when compared to ReRAM and STT-RAM CiM designs, as well as to a not in-memory approach based on SRAM. An application-level evaluation also demonstrated potential benefits (again in terms of speedup and energy improvements) compared to STT-RAM CiM and SRAM designs. Despite these encouraging results, the char-

Digital Object Identifier 10.29292/jics.v16i1.477
characteristics of FeFETs as captured by the simulation model employed in [9] are not very much in pair with FeFET devices that have been fabricated to date [10, 11].

In this paper, we revisit [9] by leveraging two different models for FeFETs that are described in Sec. II.B.1 and Sec. II.B.2. With these two different FeFET models, we implement two variations of the FeFET-CiM architecture—namely, the 2T+1FeFET-CiM and the 1FeFET-CiM. The 2T+1FeFET-CiM is a similar design to the one proposed in [9], which employs 2T+1FeFET memory cells based on the single-domain model for FeFETs [12]. The 1FeFET-CiM design, in turn, employs a similar structure for the CiM-enabled peripheral circuitry as described in [9]. However, 1FeFET memory cells based on the multi-domain model for FeFETs [13] are used in this version of the FeFET-CiM architecture.

The two variations of the FeFET-CiM architecture (with different memory cells) lead to different latency and energy consumption for reads, writes and in-memory computing accesses. The distinct performances of the two FeFET-based memory cells reflect in the improvements with respect to other technologies, e.g., STT-MRAM, and CMOS (SRAM). In this regard, this paper carries out the same type of array/application level evaluation as reported in [9], for both 2T+1FeFET-CiM and the 1FeFET-CiM designs. As summarized in Sec. IV., the FeFET-CiM architecture based on 2T+1FeFETs (1FeFETs) offers an average speedup of ∼2.5X (∼1.1X) and energy reduction of ∼1.7X (∼1.4X) when compared to a SRAM baseline across a wide range of benchmark programs from different domains. Despite smaller speedups and energy savings when compared to 2T+1FeFET-CiM, 1FeFET memory arrays may offer up to ∼5.3X density improvements when compared to conventional 6T-SRAM arrays [14, 15]. Furthermore, 1FeFET-CiM offers significant application-level improvements when compared to a counterpart STT-CiM architecture.

The rest of this paper is organized as follows. Sec. II. introduces the FeFET device, as well as the different simulation models, while Sec. III. presents the FeFET-CiM architecture, along with the two variations for FeFET-based memory cells (i.e., the 2T+1FeFET and the 1FeFET). Sec. IV. presents the results and discussion of our array/application level evaluation for the two variations of the FeFET-CiM architecture. Finally, Sec. V. concludes the paper.

II. BACKGROUND

A. The FeFET device

The structure of a FeFET resembles the one of a MOSFET, except a layer of FE oxide is deposited in the gate stack. Fig. 1(a) and Fig. 1(b) depict the equivalent circuit for a FeFET and the device’s 3-D view, respectively. In Fig. 1(a), the FE and CMOS capacitances are represented as $C_{FE}$ and $C_{CMOS}$. The coupling between these capacitances can lead to a hysteretic effect, conferring non-volatility to the device. Note that some FE materials (e.g. lead zirconium titanate (PZT) [12]) may be incompatible with CMOS processes. However, recent demonstrations of ferroelectricity in hafnium zirconium oxide (HZO) [18] (highly compatible with CMOS processes) have mitigated concerns regarding large-scale demonstrations of FE transistors that might impede industrial-scale realizations.

B. FeFET Models

B.1 LK-based model for FeFETs: A model based on the time-dependent Landau Khalatnikov (LK) equation [19] (Eq. (1)) has been used to describe the switching behavior of FeFETs in some logic-in-memory and CiM works, e.g., [16, 20, 21, 9].

$$E = \alpha P + \beta P^3 + \gamma P^5 + \rho P/dt$$  (1)

In Eq. (1), $E$ stands for electric-field; $\alpha$, $\beta$, and $\gamma$ are the static coefficients and $\rho$ is the kinetic coefficient associated with the FE material. The LK equation is a theoretical model that can describe ferroelectric behavior. The model
coefficients (listed above) can be fitted to different FE materials. Fig. 2 depicted simulated I-V characteristics illustrating how device hysteresis spans over positive and negative gate-source voltages ($V_{gs}$). When the LK-based model is employed for the simulation of FeFETs, write voltages are in the range of $\pm V_{DD} - \pm V_{DD}/2$. For instance, the $I_{ds}$ vs. $V_{gs}$ characteristics depicted in Fig. 2 indicate that write voltages of only 0.4V are possible. The device retains the $V_{th}$ state when $V_{gs}$=0.

Despite the low write voltages demonstrated by the LK-based model for FeFETs, FE switching dynamics are not easily incorporated in the LK equation, which assumes a single-domain FE material with a single coercive field for the whole FE thin film. Phenomena such as non-saturated hysteresis loops, history effects, and polarization switching dynamics cannot be captured by the LK model [19].

B.2 Preisach-based model for FeFETs: These issues are addressed by a compact model for FeFETs that is based on the Preisach theory [13]. The Preisach-based multi-domain model is derived from experimental data. Unlike a single-domain LK model, the multi-domain model more accurately reproduces characteristics of FE devices that have been fabricated to date. In more detail, the multi-domain model assumes a FE film with multiple independent single crystal domains with a distribution of coercive fields. Hence, the model can capture the behavior of both saturated and non-saturated hysteresis loops. Moreover, the multi-domain model tracks the FE history by employing an efficient turning-point tracking algorithm [22]. The model successfully reproduces the FE behavior by combining the aforementioned characteristics with a delay unit that can model polarization switching dynamics.

Fig. 3 depicts the $I_{ds}$ vs. $V_{gs}$ characteristics of a FeFET device obtained via both experimental measurements as well as simulations that employ a multi-domain Preisach-based model. The device is programmed/erased by applying $\pm 4V$ pulses to the gate. Measurement pulses, i.e., sweeps between -1V and +1V, are applied to read out the logic state of the FeFET, as illustrated in Fig. 3(a). A sufficiently wide memory window (MW) of $\sim 0.96V$, as well as $I_{on}/I_{off}$ ratios on the order of $10^4$ lead to good sensing margins, as depicted in Fig. 3(b). The threshold voltage ($V_{th}$) could be shifted through body biasing or gate metal work function engineering to meet requirements of particular designs, as illustrated in Fig. 3(c).

III. COMPUTING-IN-MEMORY WITH FeFETS

In this section, we revisit the FeFET-CiM architecture from [9] by employing the multi-domain model for FeFETs in the design of 1-FeFET memory cells. FeFET-CiM can perform Boolean (N)AND, (N)OR, X(N)OR and inversion operations, as well as the addition of any two words in memory. Sec. III.A. describes the FeFET-CiM architecture and the sense amplifier (SA) circuit operation. 2T+1FeFET and 1FeFET memory cells (based on single- and multi-domain models) are explained in Sec. III.B. Finally, Sec. III.C. details FeFET-CiM’s conventional RAM and the compute operation modes.

A. Architecture overview and SA operation

An overview of the FeFET-CiM architecture is shown in Fig. 4(a), where we represent the FeFET-based memory cells and the peripherals necessary to achieve two modes of operation: (i) Memory Mode (MM), and (ii) Compute Mode (CM). The column decoder, bitline and wordline drivers have the same standard functionality regardless of which mode is active. While in CM, the simultaneous activation of the two row decoders A and B enable the selection of two words in memory. The customized sense amplifier (SA), depicted in Fig. 4(b), is important to both MM and CM. The SA is designed so it can be either voltage and/or current-based in order to accomplish CiM operations more efficiently. Reads and Boolean (N)OR logic operations are performed entirely through a voltage-based sense scheme. The rest of the FeFET-CiM operations — (N)AND, X(N)OR and ADD — are performed based on a mix of voltage and current sense schemes.

The voltage-based sense scheme requires precharge of the bitline to $V_{PRE}$. The precharger circuit is highlighted by an orange dotted-dashed outline in Fig. 4(b). After precharging the bitline, we enable $BUF_{SENSE}$ through the signal $EN_{VS}$, that must remain active until one of the row decoders selects a word in memory to be read, and the $RD/OR$ output produces the result. Note that, during the described op-
operation, $\text{SENSE}$ and $E_{\text{NCS}}$ are not active (both stay at “0”). Hence, the reference current, current sense, and sum and carry circuits — depicted inside the green dotted outline (CM) in Fig. 4(b) — are turned off.

The current-based sense scheme requires the activation of $E_{\text{NCS}}$, the selection of any two wordlines in memory through row decoders A and B, and the selection of the SA current reference cell. When $\overline{\text{SENSE}} = 1$, the currents through both $\text{BLR}$ and $\text{BLR}_{\text{REFCELL}}$ are mirrored to the cross coupled inverter in the differential amplifier. The buffer stages connected to both ends of the SA ensure full voltage swing at the SA AND and XOR outputs. As explained in Fig. 8, we leverage an OR-enabled buffer at one of the differential outputs of the SA — labeled as $\text{BUF}_{\text{XOR}}$ in Fig. 4(b) — to turn Boolean NAND into a XOR. Finally, in-memory addition operations (ADDS) are achieved in a single memory access (to be explained in Sec. III.C.).

B. FeFET-based Memory Cells

In this section, we describe two memory cells that can be employed in the FeFET-CiM architecture.

B.1 The 2T+1FeFET memory cell: Proposed in [9], 2T+1FeFET memory cells were originally devised for (and simulated with) a single-domain model for FeFETs [12]. Although a multi-domain model for FeFETs [13] could also be employed for the design and simulation of 2T+1FeFET memory cells, the difference between the read and write voltages of FeFETs observed in real devices and captured by [13] enable the design of denser memory cells in a FeFET-CiM architecture (to be described in Sec. III.B.2).

Fig. 5 depicts the schematic of a 2T+1FeFET memory cell. In the figure, T1 and T2 are the access transistors for write and read respectively, and F1 is the FeFET storage device. While the 2T+1FeFET memory cells have some structural similarities to the FeFET-based memory cells proposed in [23], there are some important differences. 2T+1FeFET cells have (i) an extra access transistor (T2) that isolates the read bitline (BLR) of unselected cells, and (ii) different signal values ($V_{DD}$ or $GND$) at the read/write wordline (WLRW). Hence, the 2T+1FeFET memory cells enable a voltage-based sense scheme and eliminate the need for a negative power supply. Fig. 5(a)(b) describes the procedure to write either a logic “0” or logic “1” to a 2T+1FeFET cell. The different signal values at WLRW generate either the negative or positive $V_{gs}$ required to switch the FeFET polarization state. Reads can be performed by simply sensing the current path between the BLR and ground, as illustrated in Fig. 5(c). When compared to a 6T-SRAM, besides non-volatility, this memory cell offers the potential for low leakage, and ~2X savings in density due to reduced device count (assuming that both cells employ minimum size transistors).

B.2 The 1FeFET memory cell: 1-FeFET memory cells and arrays have been proposed and fabricated [11]. These memory cells do not employ access transistors, therefore they can achieve higher densities than 2T+1FeFET memory cells [14]. Fig. 6(a) illustrates the write of logic “0” to a 1-FeFET memory cell. A negative voltage ($-V_{W}$) is applied at the gate of F1 through the wordline/write bitline (WL/BLW), while the sourceline (ScL) is grounded. Similarly, writing logic “1” to a 1-FeFET memory cell requires the application of a positive voltage ($+V_{W}$) at the FeFET’s gate through WL/BLW, while the ScL is grounded. On the other hand, reading out the logic state stored in the 1FeFET memory cell requires a select voltage to be applied at the gate of F1 (also through the wordline/write bitline (WL/BLW)).

According to experimental work reported in [10, 14], it is appropriate to employ $\pm V_{W}=\pm 4V$ to enable the complete polarization switch of FE domains in the 1FeFET memory cell. Conversely, select voltages are in the much smaller range of 0.5V — 1.0V [10, 14]) to be applied at the wordline/write bitline (WL/BLW).

C. FeFET-CiM Operation Modes

Two modes of operation — i.e., the memory mode (MM) and the compute mode (CM) — are possible with FeFET-CiM. While in memory mode (MM), the architecture can serve as general purpose, non-volatile RAM memory that transfers data from and to external processing units (PUs). When in compute mode (CM), the row decoders will simultaneously activate two wordlines so logic and arithmetic operations between words can be performed in memory. Next, we detail the MM and CM operation modes.

C.1 MM operation: The MM enables reads/writes in FeFET-CiM. While in MM, changes in the voltage level of the BLRs can be immediately sensed by a buffer and an inverter, respectively labeled as $\text{BUF}_{\text{SENSE}}$ and $\text{INV}_{\text{SENSE}}$ in
Fig. 7: Bitline currents for different input combinations in Boolean AND operation. Current distributions are obtained by Monte Carlo simulations, with 3σ variations of 42% and 12% for Vth and W/L, respectively [24]. Reference current is 33µA. (Figure from [9]).

Fig. 4(b). The current-sensing part of the SA — highlighted by a green dotted outline (CM) in Fig. 4(b) — can be completely powered off during MM operation, leading to significant power savings. Per [9], experiments on a 1MB FeFET-CiM array indicate that static power is ∼100X less for MM when compared to CM as the current reference is turned off. With respect to latency, the high Ion of FeFETs and the compact size of the memory cells ensure a fast voltage drop on BLRs, which ultimately translates to fast (∼2.4 ns) reads [9].

C.2 CM operation: Here, we detail the several logic/arithmetic CiM operations supported by FeFET-CiM.

**Boolean (N)OR and (N)AND:** Due to the high Ion/Ioff ratios of the FeFETs and the way by which bitlines are connected in the FeFET-CiM memory array, (N)OR logic is achieved through voltage-based sense (with an inverter-based SA). Alternatively, for (N)AND operations, the SAs connected to the bitlines sense the resulting current and compare it to a reference. The choice of the reference boundary is critically important. To find the right value, [9] obtains bitline current distributions by performing 1000 runs of Monte Carlo simulations, with 3σ variations of 42% and 12% for Vth and W/L, respectively [24]. The logic values stored in the FeFET-based memory cells lead to distinct bitline current distributions for different input combinations, as depicted in Fig. 7. A properly sized FeFET dummy memory cell is used to generate the reference current of 33µA, such that the differential SA can perform (N)AND logic.

**Boolean X(N)OR:** X(N)OR operations require both voltage and current sense schemes. X(N)OR is implemented through an OR-based activation of the buffer stage on the branch of the differential SA that originally performs NAND operations. There is no need for additional CMOS logic gates (see details in Fig. 4(b)). Bufnxor is modified to perform a Boolean XOR function by a selective enabling scheme. Namely, Bufnxor is turned on only when the result of the voltage-based Boolean OR is “1” (see Fig. 8 for a complete truth table of XOR implemented through OR-enabled NAND).

**Arithmetic ADD:** The ADD operation is accomplished through additional sum and carry circuitry embedded into the SA. FeFET-CiM leverages the results of bitwise Boolean (OR, AND, X(N)OR) logic, as well as the carry from the previous stage (cin) to implement an in-memory ripple carry adder (RCA), which is highlighted by a dotted purple outline in Fig. 4(b). cout and cin are computed through a compact and fast dynamic circuit. The precharge phase for this circuit occurs when ENC = 0. When this signal switches to “1”, the evaluation phase of the carry circuit begins. At this moment, Cout = 1 if either (i) the result of the bitwise AND is “1” or (ii) Cin and the bitwise Boolean OR results are “1”. The described circuit takes ∼26.6 ps to output a carry. Sum, in turn, is produced by the aforementioned in-memory RCA through a pass transistor-based multiplex logic where Cin is the selector bit, and Xor XOR are the signals to be selected. Namely, when Cin = 0, the sum output can only be “1” if the inputs have distinct (“0” and “1”) values. Alternatively, if Cin = 1, the sum output is “1” if the inputs have equal (“0” and “0”) or (“1” and “1”) values. The described pass transistor-based logic takes ∼151.4 ps to generate the SUM output. To enable the in-memory addition of two N-bit words through the aforementioned RCA circuit, Cout of a (ith-1) bit is routed to Cin of a (ith) bit at the sense amplifier level. For example, for a word size of 32-bit and a memory array with 1024 columns, we can build up to 32 blocks of 32-bit in-memory RCAs that perform additions in parallel.

IV. Evaluation

In this section, we compare Figures-of-Merit (FoMs) such as delay, energy and area of FeFET-CiM based on 2T+1FeFET and 1FeFET memory cells, where single- and multi-domain models for FeFETs are employed, respectively. Our evaluation is carried out at both the array and application levels (with the same benchmark programs as [5]).

A. Experimental setup

To evaluate FeFET-CiM arrays, we must first characterize the (i) 2T+1FeFET and (ii) 1FeFET memory cells. For (i), we employ a single-domain SPICE model for FeFETs [12]. For (ii), we employ a multi-domain SPICE model for FeFETs [13]. Both FeFET models employ the underlying 45nm Predictive Technology (PTM) model for CMOS [25].

We measure write energy and Ron/Roff for a single cell, as well as for two parallel-connected cells. Note that single cell data is used for a standard memory read, while data for parallel-connected cells corresponds to Boolean and arithmetic operations, which require two words to be selected simultaneously in the same memory access. Furthermore, SPICE simulations are also used to measure the delay, energy and leakage power for our customized sense amplifier design described in Sec. IIIA. We then employ cell-level and sense amplifier data in a modified version of the DESTINY simulator [26] to account for FeFET-CiM read (and write) energy and latency at the array-level.
B. Array Level Evaluation

In our array level evaluation, we compare the energy and latency for an in-memory addition of 32-bit words, with the energy and latency of the memory accesses of a conventional (not in-memory) approach where data must first be fetched from SRAM. We use DESTINY to simulate read and write memory accesses on 1MB arrays for FeFET-CiM, as well as for a SRAM baseline. Data in Table I suggests that the read/compute FoMs of the two FeFET-based memory cells are in a comparable range (with a variation of less than 2×). However, there are significant differences for the write energy and delay with 2T+1FeFET-CiM and 1-FeFET-CiM architectures. Namely, writes with 1-FeFET-CiM require 2.31× more energy (and 4.96× more time) when compared to 2T+1FeFET-CiM that assumes a single-domain FeFET model. Such differences in the FoMs are caused by the high write voltages employed in 1-FeFET memory cell, associated with long write pulses (here, in the order of 10 ns) that are necessary for effectively switch the polarization domains in the FeFET device.

Alternatively, our results suggest that in-memory addition between 32-bit words in 2T+1FeFET-CiM (1FeFET-CiM) consumes ∼28.4% (∼13.1%) more energy than a read from SRAM. The additional energy required to perform computation in FeFET-CiM is due to the activation of the current-based circuits in the SA, necessary for (N)AND/X(N)OR/ADD operations (including the dummy memory cells that generate reference currents). That said, a conventional not-in-memory approach requires that two data operands be fetched from memory prior to CPU-based processing. Hence, with CiM, some memory reads would be avoided. Per [5], a CiM in-memory operation can effectively replace two reads. A subsequent store operation to write the result is needed in both in-memory/not-in-memory approaches. A quantitative analysis of the impact of CiM on energy and latency of benchmark programs due to fewer memory access is provided in Sec. IV.C.

C. Application Level Evaluation

Here, we provide an application level analysis of the two versions of FeFET-CiM, i.e., the 2T+1FeFET-CiM based on a single-domain model and the 1FeFET-CiM based on a multi-domain model for FeFETs. We compare the performance and energy consumptions of the FeFET-based designs to a STT-MRAM-based CiM (STT-CiM [5]). Furthermore, we compare the CiM approach to a not-in-memory SRAM baseline (where computation is done outside the memory).

Methodology: We leverage the methodology described in [5], that includes the implementation of custom instructions in the Nios II processor for running twelve benchmark applications from various domains. Each application has memory accesses categorized into convertible reads (CR), non-convertible reads (NCR) and writes (W). CRs refer to those memory accesses that can be reduced in a 2:1 ratio, i.e., two reads can be replaced by a single in-memory compute operation. NCR represents memory accesses that cannot be avoided or reduced by CiM. Finally, as the results of both in-memory and not-in-memory computations are always written to memory, we do not consider reductions in the number of writes.

Note that our comparison to a not-in-memory SRAM baseline accounts only for the time and energy of memory fetches. The computation overhead, i.e., the energy/time for processing, is not included in our analysis as it can vary widely depending on the central processing unit (CPU) model, the cache sizes, and the place in the memory hierarchy at which CiM is implemented (an analysis on the impact of placing CiM at different levels in the memory hierarchy, as well as the impact of CiM on the CPU energy/delay can be found in [27]).

Results and discussion: Table II shows a breakdown of the memory accesses across 12 benchmark applications [5], while Fig. 9(a,b) shows the comparison between energy and latency of 2T+1FeFET-CiM, 1FeFET-CiM, and STT-CiM using SRAM memory as a baseline. We use the time/energy of STT-CiM memory accesses as reported in [5].

Our results suggest that reduction in energy and delay resulting from the in-memory computing approach may vary according to the CiM design (and FeFET model) employed. For instance, the performance of 2T+1FeFET-CiM based on single-domain model is quite insensitive to the different memory accesses patterns in the benchmark programs. The low write energy and latency of a 2T+1FeFET memory cell makes the 2T+1FeFET-CiM design to outperform other CiM architectures (i.e., the 1-FeFET-CiM and the STT-CiM) for applications with a large number of memory write accesses (IMGSEG, RC4, EDIST and LCS). Overall, 2T+1FeFET-CiM offers an average speedup of ∼2.5X and energy reduction of ∼1.7X when compared to SRAM across the 12 benchmarks.

Fig. 9: (a) Energy consumption and (b) latency of the memory accesses during the execution of 12 benchmark programs with STT-CiM, 2T+1FeFET-CiM, 1-FeFET-CiM, and SRAM (not in-memory). Results are normalized to the SRAM approach.
In the case of STT-CiM and 1-FeFET-CiM (based on multi-domain model), the observed energy and latency improvements at the application level are highly sensitive to the memory access patterns of the different benchmark programs. For instance, benchmark programs that have a high percentage of writes (e.g., IMGSEG, EDIST, LCS, RC4) are expected to result in less benefits from the CiM architectures as the 1T+1MTJ and 1FeFET memory cells assumed here have longer write times (and higher energies) than the 2T+1FeFET memory cells [9]. Overall, 1FeFET-CiM offers an average speedup of 1.2X and energy reduction of 1.4X when compared to SRAM across the 12 benchmarks.

When comparing the two FeFET-based CiM designs devised for different FeFET models (i.e., 2T+1FeFET-CiM and 1-FeFET-CiM), we note that the speedups (and energy savings) offered by the 1FeFET-CiM design are 2.3X (1.2X) smaller when compared to 2T+1FeFET-CiM. This can be attributed to the more expensive write operation of the 1FeFET memory cells. Nevertheless, the 1FeFET-CiM architecture still yields improvements when compared to the STT-CiM architecture — i.e., when considering a wide range of benchmark programs, 1.4X (1.5X) speedups (energy savings) are possible by employing the 1FeFET-CiM architecture instead of a STT-CiM counterpart.

The high density of the 1FeFET memory cells is another advantage of the 1-FeFET-CiM architecture. As discussed in [14, 15], arrays based on 1FeFET memory cells may offer up to 5.3X density improvements when compared to conventional 6T-SRAM arrays. The high density and low leakage power (that arises from non-volatility) make 1FeFET memory cells great candidates for the design of low-leakage and dense CiM architectures.

### Table I. Dynamic Energy and Latency of SRAM and FeFET-CiM

| Memory Cell | Read | Compute | Write | Delay (ns) |
|-------------|------|---------|-------|------------|
| SRAM        | 58.97| NA      | 57.57 | 3.397      |
| 2T+1FeFET-CiM (w/ single-domain) | 45.65 | 75.72 | 45.02 | 2.402 |
| 1FeFET-CiM (w/ multi-domain) | 48.06 | 66.70 | 104.18 | 2.650 |

### Table II: Memory Accesses Breakdown of the Benchmark Programs Used in our Application Level Evaluation (From [5])

![Table II](attachment:table.png)

| Application | Writes | CNC-Reads | CC-Reads |
|-------------|--------|-----------|----------|
| BLIT        | 0.00%  | 0.00%     | 100.00%  |
| AHC         | 0.00%  | 51.19%    | 48.81%   |
| SVM         | 0.01%  | 0.01%     | 99.98%   |
| KMP         | 0.02%  | 12.46%    | 87.52%   |
| GLVQ        | 0.10%  | 0.10%     | 99.80%   |
| MLP         | 0.55%  | 0.66%     | 98.79%   |
| OCR         | 0.76%  | 6.51%     | 92.73%   |
| KMEANS      | 4.00%  | 0.89%     | 95.11%   |
| IMGSEG      | 8.19%  | 11.06%    | 80.75%   |
| EDIST       | 23.55% | 23.49%    | 52.96%   |
| LCS         | 29.66% | 29.57%    | 40.77%   |
| RC4         | 33.33% | 22.23%    | 44.44%   |

### V. Conclusion

We presented a revised study of FeFET-CiM, a computing-in-memory (CiM) architecture. In this study, we employ two types of FeFET-based memory cells in the CiM architecture, i.e., the 2T+1FeFET (as originally proposed in [9]), and the 1-FeFET [11]. The use of different memory cells in FeFET-CiM is enabled by the use of distinct models for FeFET devices [12, 13]. The FeFET-CiM architecture based on 2T+1FeFETs offers superior application-level benefits when compared to the same architecture based on 1FeFETs. Nevertheless, CiM architectures based on 1FeFET memory cells may offer up to 5.3X memory density improvements when compared to conventional 6T-SRAM architectures [15, 14]. Finally, 1FeFET-CiM still offers relevant application-level improvements when compared to a counterpart STT-CiM architecture.

### Acknowledgements

This work was supported in part by ASCENT, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA.

### References

[1] A. Sebastian, T. Tuna, N. Papandreou, M. Le Gallo, L. Kull, T. Parnell, and E. Eleftheriou, “Temporal correlation detection using computational phase-change memory,” Nature Communications, vol. 8, no. 1, p. 1115, 2017.

[2] S. Aga, S. Jeloka, A. Subramaniyan, S. Narayanasamy, D. Blaauw, and R. Das, “Compute caches,” in HPCA, Feb 2017, pp. 481–492.

[3] V. Seshadri, D. Lee, T. Mullins, H. Hassan, A. Boroumand, J. Kim, M. A. Kozuch, O. Mutlu, P. B. Gibbons, and T. C. Mowry, “Ambit: In-memory accelerator for bulk bitwise operations using commodity dram technology,” in 2017 50th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), 2017, pp. 273–287.

[4] W. Kang, H. Wang, Z. Wang, Y. Zhang, and W. Zhao, “In-memory processing paradigm for bitwise logic operations in stt-mram,” TMAG, vol. 53, no. 11, pp. 1–4, 2017.

[5] S. Jain, A. Ranjan, K. Roy, and A. Raghunathan, “Computing in memory with spin-transfer torque magnetic ram,” TVLSI, vol. PP, no. 99, pp. 1–14, 2017.

[6] N. Talati, S. Gupta, P. Mane, and S. Kvatinsky, “Logic Design Within Memristive Memories Using Memristor-Aided LoGIC (MAGIC),” TNANO, vol. 15, no. 4, pp. 635–650, 2016.

[7] V. Seshadri, D. Lee, T. Mullins, H. Hassan, A. Boroumand, J. Kim, M. A. Kozuch, O. Mutlu, P. B. Gibbons, and T. C. Mowry, “Buddy-ram: Improving the performance and efficiency of bulk bitwise operations using DRAM,” CoRR, vol. abs/1611.09988, 2016. [Online]. Available: http://arxiv.org/abs/1611.09988
[1] X. Chen, X. Sun, P. Wang, S. Datta, X. S. Hu, X. Yin, M. Jerry, S. Yu, A. F. Laguna, K. Ni, M. T. Niemier, and D. Reis, “The impact of ferroelectric FETs on digital and analog circuits and architectures,” *IEEE Design Test*, vol. 37, no. 1, pp. 79–99, 2020.

[2] M. H. Lee, P. G. Chen, C. Liu, K. Y. Chu, C. C. Cheng, M. J. Xie, S. N. Liu, J. W. Lee, S. J. Huang, M. H. Liao, M. Tang, K. S. Li, and M. C. Chen, “Prospects for ferroelectric HfZrOx FETs with experimentally CET=0.98nm, $SS_{for}=42$mV/dec, $SS_{rev}=28$mV/dec, switch-off <0.2V, and hysteresis-free strategies,” in *2015 IEEE International Electron Devices Meeting (IEDM)*, Dec 2015, pp. 22.5.1–22.5.4.

[3] T. Song, “Landau-khalatnikov simulations for ferroelectric switching in ferroelectric random access memory application,” *JKPS*, vol. 46, no. 1, pp. 5–9, 2005.

[4] X. Yin, M. Niemier, and X. S. Hu, “Design and benchmarking of ferroelectric FET based TCAM,” in *DATE*, 2017, pp. 1444–1449.

[5] A. Aziz, E. T. Broyer, A. Chen, X. Chen, S. Datta, S. K. Gupta, M. Hoffmann, X. S. Hu, A. Ionescu, M. Jerry, T. Mikolajick, H. Mulaosmanovic, K. Ni, M. Niemier, I. O’Connor, A. Saha, S. Slesazeck, S. K. Thurumala, and X. Yin, “Computing with ferroelectric fets: Devices, models, systems, and applications,” in *DATE*, March 2018, pp. 1289–1298.

[6] B. Jiang et al., “Computationally efficient ferroelectric capacitor model for circuit simulation,” in *VLSI Symposium*. IEEE, 1997, pp. 141–142.

[7] ITRS, “Design,” Tech. Rep., 2007.

[8] R. Vattikonda, W. Wang, and Y. Cao, “Modeling and minimization of pmos nbti effect for robust nanometer design,” in *DAC*, 2006.

[9] M. Poremba, S. Mittal, D. Li, J. S. Vetter, and Y. Xie, “Destiny: A tool for modeling emerging 3d nvm and edram caches,” in *DATE*, 2015.

[10] D. Gao, D. Reis, X. S. Hu, and C. Zhuo, “Eva-cim: A system-level performance and energy evaluation framework for computing-in-memory architectures,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2020.