A Practical Design-Space Analysis of Compute-in-Memory With SRAM

Samuel Spetalnick, *Graduate Student Member, IEEE*, and Arijit Raychowdhury, *Fellow, IEEE*

***Abstract*— Analog-domain compute-in-memory (CIM) is a technique that has emerged in part as a response to the memory-intensive vector-matrix-multiplications (VMMs) required to implement important emerging applications, notably machine learning inference. Implemented CIM systems have demonstrated good energy efficiency for lower-precision systems and/or with loosened compute-level accuracy requirements. *A priori* it is unclear exactly how the efficiency advantages of CIM emerge and therefore the generalizability of these advantages, beyond the specific demonstrated examples, is unclear. Noting that not all VMM-heavy workloads can tolerate imperfect accuracy and/or reduced precision, this work combines high-level models with circuit models and simulations to examine the efficiency gains and penalties associated with CIM in static random-access memory (SRAM) arrays. Extracted models which are needed to make assertive statements about CIM are developed and discussed. An energy comparison to standard SRAM is made, and the issues of accuracy loss and area are contextualized. Finally, a few example models comparing the energy efficiency of CIM to that of SRAM are shown to verify that CIM is most effective for error-tolerant, low-precision applications.**

***Index Terms*— Compute-in-memory, analog, modeling, SRAM, memory, multiply-accumulate, practicality.**

I. Introduction

THE vector-matrix multiplication (VMM, or a series of multiply-accumulates, MACs) is a fundamental computational building block which finds important use in many modern computing applications. Popular among these is machine learning inference which, as is typical of VMM-based tasks, combines this digital computing workload with a large memory access requirement. This challenging combination has encouraged system designers to propose novel digital hardware architectures which specifically address the need for large amounts of weight data to be accessed during compute [1]. Two additional observations can be made which encourage an *analog*, *compute-in-memory* (CIM) approach to VMMs for machine learning: a VMM amounts to a set of scaled sums of stored weights, and machine learning applications are relatively tolerant to computing error [2], [3].

In light of this apparent opportunity for performance gains and the plethora of fabricated and modeled designs which leverage CIM, a few analytical works have sought to model the costs (accuracy) and benefits (primarily energy but potentially throughput). Rekhi, A.S. *et al.* [2] analyze the energy-accuracy tradeoff via a specific computing task (ImageNet using ResNet-50). Accordingly, the authors create a simple energy-accuracy abstraction by assuming the analog MAC energy is analog-to-digital converter (ADC)-dominated. Gonugondla, S.K. *et al.* [4] focus on the theoretical derivation of signal-to-noise ratio (SNR) for various CIM architectures, and do not present a ground-up energy model for use in high-level analysis. Finally, Murmann [5] provides insight into the breadth of CIM design possibilities and their potential to achieve competitive energy efficiencies but, similarly, does not present present energy models targeted to implementation-specific circuits.

To supplement these prior works, this work focuses on a deliberately narrow subsection of the CIM design space to enable both the development of detailed models from circuit fundamentals and the construction of strong conclusions. While chip demonstrations have shown the potential for emerging technologies such as resistive random-access memory (RRAM) to implement energy-efficient CIM ([6]–[9]), resistive memory technologies have yet to reach maturity and still present practical issues including low endurance and high write voltages [10]. The scope of this work is therefore limited to standard CMOS-based static random-access memory (SRAM)-based CIM designs, which additionally enables modeling and simulation using a foundry process design kit (PDK). Limiting scope further, this work focuses on CIM designs whose readout is analogous to traditional SRAM in that the bitcells *directly* pull current from a precharged bitline without transferring their value onto another device (such as a capacitor). The purpose of Section II is to describe and motivate this second scope limitation.

The contributions of this work are as follows, in the order they are presented. This work will:

1) categorize CIM systems to enable more detailed discussion and modeling of energy and design challenges;
2) construct a comparative model of the digital-compute advantage of CIM relative to traditional systems;
3) present a circuit-based model of CIM energy use, in analogy to standard SRAM;

Manuscript received June 1, 2021; revised October 9, 2021 and November 24, 2021; accepted December 8, 2021. This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) and the Semiconductor Research Corporation through the Applications and Systems Driven Center for Energy-Efficient Integrated NanoTechnologies (ASCENT) and in part by the Center for Brain-Inspired Computing (C-BRIC). This article was recommended by Associate Editor M.-F. Chang. *(Corresponding author: Samuel Spetalnick.)*

The authors are with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: spetalnick@gatech.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSI.2021.3138057.

Digital Object Identifier 10.1109/TCSI.2021.3138057

TABLE I
CIM-SPECIFIC ABBREVIATIONS USED IN MODELS

Symbol	Definition
B_x	MAC input vector precision (in bits).
B_w	MAC weight vector precision (in bits).
P_x	# of input vector bits applied on a single WL per read cycle.
P_{WL}	# of parallel WLs allowed per read cycle.
P	# of states possible on each BL minus 1; $P_{WL} \times (2^{P_x} - 1)$.
p	Actual state, $p \in [0, P]$, represented on the BL.
M_s	# of equivalent SRAM single-cell reads required to emulate the memory accesses of the CIM system.
$S(P)$	Characteristic state separation of CIM system designed for P states represented on the BL.
K	Effective fraction of dynamic range used to separate states.
D	Ratio of SRAM design margin to CIM design margin.
P_{OUT}	# of states (conversion steps) in the BL ADC output.
$\beta(p)$	Relative BL swing to represent state p, such that $V_{SNS}\beta(p)$ is the BL swing, in Volts, to represent state p.

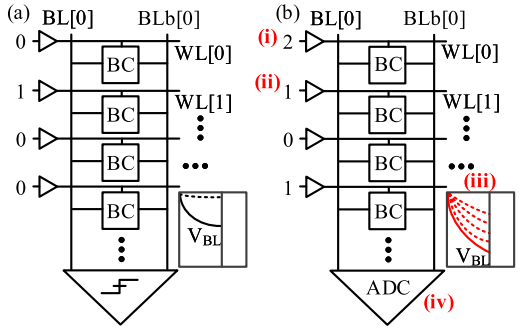

Fig. 1. (a) Traditional and (b) CIM memory systems shown with 1:1 column multiplexing (unlikely) for simplicity. Four important degrees of freedom are introduced in the CIM system: (i) there may be more than two allowed input (WL) states, (ii) multiple WLs may be enabled at once, (iii) there may be more than two allowed output (BL) states, and (iv) these multiple states may require a more complex ADC than a 1-bit sense amplifier. Feature (ii) is the hallmark feature of CIM. Bitcell is abbreviated as 'BC.'

4) estimate the severity of three challenges imposed by CIM by presenting analysis of read disturb, showing a novel analytical technique for assessing state separation for certain systems, and discussing accuracy issues;
5) summarize the results of these analyses by way of four illustrative example system models, and present brief design guidance based on the analysis.

Some CIM-specific abbreviations are introduced in this work, and a set of these is shown in Table I.

II. A BRIEF TAXONOMY

In this work, a CIM system is *any system which enables the values of multiple memory cells to be combined in the analog domain then read out, as a single value, into the digital domain*. Additionally, the number of output states may be increased by allowing more than two wordline (WL) states. To introduce CIM, four important differences between CIM and traditional memory systems are outlined in Figure 1.

An initial distinction must be drawn between systems which perform readout by *directly* measuring bitline (BL) voltage due to a pull-down (PD) operating step (during which a current proportional to the number on-state selected SRAM cells is pulled on the BL) and systems which measure the BL state *indirectly* by, briefly, using the BL to control a separate system which performs an analog computing step. These two methods are described below:

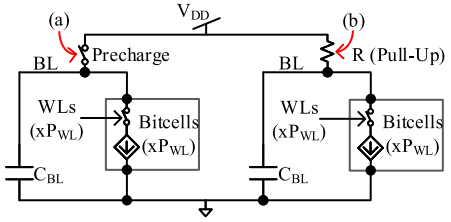

Fig. 2. Symbolic representations of the (a) resistive-capacitive pull-down (RC-PD) and (b) resistive-dividing (R-dividing) flavors of *direct* CIM systems.

A. The Direct Approach

The direct systems [11]–[14] operate analogously to traditional binary-BL SRAM: a single read-step produces the analog value on the BL which is then quantized. The read PD circuit formed in an SRAM cell when its wordline (WL) is selected may be viewed as a variable current source, and the read step may therefore be viewed generically as a current to voltage conversion. This conversion may take one of two forms, which further classifies the direct systems (Figure 2). In resistive-capacitive-pull-down (RC-PD) macros, which operate almost identically to traditional SRAM, the BL is first precharged (*e.g.* to a positive rail) after which a set of WLs is enabled. The on-state selected cells create a PD current which causes the BL to slew downward. After a period of time, the value of the BL is quantized. In resistive-dividing (R-dividing) macros, a pull-up (PU) device flows an active current on the BL while the selected cells are activated. In the simplest case, the relationship of the large-signal resistance of the PU device to that of the PD circuit formed by the on-state selected cells creates a steady-state output voltage on the BL.

B. The Indirect Approach

As will be demonstrated later in this work, the direct systems present a set of challenges including (1) reduced and variable state separation leading to restricted sensing margins and nontrivial reference generation, (2) inconsistent (noisy) state placement due to poorly matched SRAM PD circuits, and (3) limited and non-ideal mechanisms for multi-bit inputs and no native compatibility with analog-domain multi-bit weights.

These challenges may be avoided by shifting the analog computing operation away from the SRAM storage cells and onto a special-purpose summing apparatus. These special-purpose analog add-ons are characteristic of *indirect* systems [15] and are typified by well-matched metal-oxide-metal (MOM) capacitors [16], [17], shared transistors with improved characteristics [18]–[21], or time-domain computing [22]. The effect is to decouple the (usually, matching and linearity) behavior of the analog compute from the nonideal density-oriented storage cells. These systems operate in a weaker analogy, relative to direct systems, to traditional SRAM - area efficiency and memory frequency are expected to scale worse in these systems than in typical SRAM due to the increased hardware and timing complexity; the approximate trend toward lower array density in indirect systems is visible in Table II. Due to this practical advantage, we choose to analyze *direct* systems in Sections IV and V.

TABLE II

SORTING RECENT CIM SRAM SYSTEMS BY OPERATING SCHEME[1] (RC-PD AND R-DIVIDING SYSTEMS HIGHLIGHTED)

Ref.	Year	Cell.	Tech.	Cell Additions & Alterations	Scheme	WL/ADC[2]	TOPS/W[3]	In/Weight/Out[4]	Mcell/mm^{2}[5]
[13]	2017	6T	130nm	None	RC-PD	5/1	N/A	5/1/1	0.220
[15]	2018	10T	65nm	16 rows per BL.	Indirect	7/7	28.1	7/1/7	0.233*
[12]	2018	6T	65nm	BL is shorter than standard SRAM.	RC-PD	4/4	3.125	8/8/8	0.352
[14]	2018	6T	65nm	None	R-Div.	1/1	111.6	1/1/1	1.817
[22]	2019	8T	28nm	Novel time-based system; uses pulse-width modulation.	Indirect	4/8	46.6	8/8/8	N/A
[16]	2019	8T	65nm	One MOM per cell, an added "shorting" device is shared among a few (∼3) cells.	Indirect	1/1	866	1/1/1	0.228
[11]	2020	8T	65nm	Separate RBL/RBLb/RWL as in 10T cell with RWL on source of read device.	RC-PD	1/1-5	490-15.8	1/1/1-5	0.285
[17]	2020	8T	65nm	MOM per cell, similar to [16].	Indirect	1-8/1-8	400	1/1/1	0.195
[18]	2020	6T	28nm	Cells grouped into 16-cell units with a matched analog "multiplier" current source ("LCC").	Indirect	4/5	68.44	4/4/12	3.709
[20]	2020	6T	28nm	One two-way analog "multiplier" ("TWT-MC") per 16 cells.	Indirect	2/5	61.1	2/4/10	N/A
[23]	2020	8T	7nm	Small (64x64) array.	RC-PD	4/4	351	4/4/4	17.994
[19]	2021	6T	65nm	Similar to [18] with 8 cells per "LCC."	Indirect	2/2-4	370	2/1/32	0.037*
[21]	2021	6T	28nm	Similar to [18] with 32 cells per "LCC."	Indirect	2/5	94.3	4/4/12	3.887

C. Example Sorting

This taxonomy as it applies to a sample of CIM SRAM examples is presented in Table II. Low-precision inputs and outputs are preferred by designers, especially when reporting energy efficiency, which aligns with the trend that will be shown by examples in Section V.

III. COMPARATIVE APPROACH

The energy cost of modern MAC workloads consists of digital logic and memory access components, the latter having become more important in recent years [1], [24]. CIM has the potential to address both, as it offers:

1) *Higher effective memory bandwidth* for improved memory access energy, and
2) *Reduced digital logic workload* due to analog-domain computing.

To maintain generalizability across technology generations and applications (for which the compute/memory energy balance shifts), we develop separate comparisons for (a) energy cost per effective bit accessed and (b) overall digital workload under traditional vs. CIM regimes. Since the magnitude of CIM's advantages in both areas is proportional to the allowed parallelism (P, *i.e.* the number of output states allowed to be accumulated in analog), analysis in this work will relate energy penalties or the severity of area, robustness, and accuracy concerns to P. In this section, we will first frame memory-access energy before estimating the digital workload savings due to CIM.

A. Effective Memory Bandwidth

The memory-bandwidth CIM thesis holds that there is an advantage to be gained by reading out the sum of the values of a subset of memory cells rather than area- or time-serially reading the subset and performing the MAC entirely in digital. A single read operation is used to simultaneously read P_{WL} weight bits (on a single BL). The sum across element-wise products in a B_x- by B_w-bit precision MAC with P_{WL} elements is an implicit lossy compression operation in the weights; a group of P_{WL} same-binary-position weight bits need only be represented by at most $\log_2 P$ bits when added together.

The lossy compression applies a penalty in the form of input bit striping: multiplying the same weight against several inputs or input bits requires additional read operations. If analog multi-bit inputs are used, then the input-bit-striping penalty is divided by the precision available on each WL. In net, the number (M_s) of equivalent SRAM single-cell reads required to emulate the CIM system is shown in Equation 1, where P_x is the analog-hardware-implemented input precision (N.B. P_{WL} is linear and P_x is binary weighted).

$$M_s = \frac{P_{WL} \times P_x}{B_x} \tag{1}$$

Note that $P_x \leq B_x$. It is convenient to use a system parameter $P = P_{WL} \times (2^{P_x} - 1)$ to represent one less than the total number of states allowed on a BL (# of state boundaries). Equation 1 computes the per-memory-read increase in throughput of any CIM system. It remains to apply an analog multiplier to Equation 1 to account for changes in per-read energy (Section IV).

B. Digital Workload

The most straightforward benefit of performing addition in analog is to reduce the digital (multiplying and summing) workload. This advantage is softened by two factors: first,

[1]This table is not exhaustive and rather includes some representative examples from the past few years.

[2]This is the number of bits that can be applied simultaneously (within one read cycle/path) on the WL and the precision of the ADC.

[3]This is a tera-operations per watt (TOPS/W) reported in the cited work.

[4]This is the bit precisions describing the operation for which the nominal TOPS/W was reported in the cited work.

[5]This is the density of storage cells themselves, excluding peripherals. * Indicates that the array-only value was unavailable and an estimate or provided value was used which includes peripheral area.

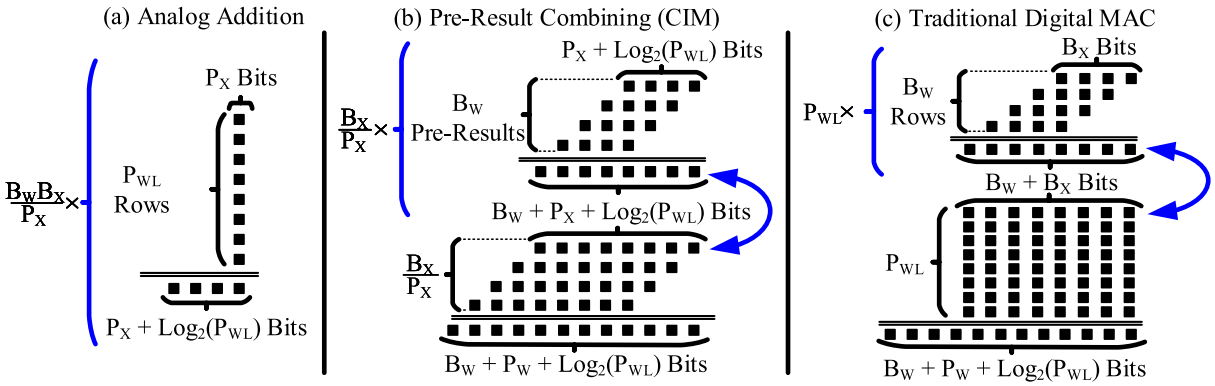

Fig. 3. Each bit shown as a square for (a) the analog addition performed on the BL in a CIM system, (b) the pre-result combining workload needed by a CIM system, and (c) the same MAC performed completely in digital. The shown dots correspond to an 8-wide MAC with 4-bit input and weight precisions.

implementation realities (*e.g.* pertaining to the ADC) may add back in some of the saved compute. For example, typical Flash ADCs require thermometer-to-binary conversion, whose implementation may approximately be viewed as a P-wide one-bit adder (the Flash ADC may also require bubble error correction, [25]). Second, the limited (analog-domain) input and stored weight precision of CIM systems may necessitate striped inputs and weights, which implies an added pre-result combining step (Figure 3 (b)) to produce the MAC output. The first issue is not explicitly accounted for in this work; instead, this added digital energy is included in the literature-derived constant ADC energy assumption motivated in Section IV. The second issue is the focus of the remainder of this section.

We are interested in the *fraction of saved compute* due to the analog-domain sums. Figure 3 breaks down the components of the CIM (a, b) and traditional (c) adding workloads along feasible lines. For example, (b) assumes the input is applied across all weight bits and that these pre-results are immediately combined, then barrel shifted and accumulated onto the sum corresponding to the bit-striped input. Two *ad hoc* methods will be used to compare the energy of the traditional system's digital workload (Figure 3(b)) to that of the CIM system's digital workload (Figure 3(c)). The simplest method is to consider the total digital MAC workload as consisting of Figure 3(a) followed by Figure 3(b). The fraction of saved compute can then be computed by dividing (a) by (a) + (b). To approximate the computing cost of these sums, we observe that both (a) and (b) correspond *in shape* to the partial product reduction of a standard multiplier; typically, an algorithmic method [26], [27] will be employed to construct adder trees to reduce these sums. Counting the bits to be combined therefore provides a compact, although approximate, representation of complexity. Omitting algebra, the ratio (ν) of this bit count for the saved CIM compute to that of the total compute is given by Equation 2:

$$\nu = \frac{P_{WL}}{P_{WL} + 1 + \frac{\log P_{WL}}{P_x} + \frac{1}{P_x} + \frac{1}{B_w} + \frac{\log P_{WL}}{P_x \times B_w}} \tag{2}$$

More detailed estimation is possible by modeling gate-level implementations of (b) and (c) and counting gates *each time they are used* to estimate energy. This implies further

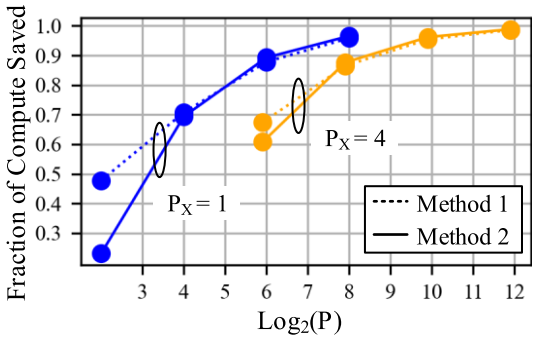

Fig. 4. The fraction of MAC computing workload saved by using CIM in the case of $P_x = 1$ (one-bit WLs, with up to P wordlines active at once) and $P_x = 4$ (four-bit WLs). For this example, the MAC precision is set to $B_x = B_w = 8$. These curves are not very sensitive to MAC precision.

assumptions: the adders are formatted as in a fast multiplier with some specific kind of partial product reduction step followed by a ripple-carry adder, noting that additional carry look-ahead logic may be important in a practical implementation. Registers are not modeled, and a simple greedy algorithm for forming adder trees is used where appropriate. The value of interest is ((c) - (b)) / (c).

The two methods for approximating the digital compute savings due to CIM are plotted in Figure 4 for $B_x = B_w = 8$ (example case) across increasing CIM parallelism. The results of this analysis are shown in Figure 4. The methods generally agree, and the models show that the analog-domain compute is a dominant portion of overall compute for large P. This also confirms that supporting more states on the WL is less favorable (than enabling more WLs) for reducing digital adding workload due to the linear benefit to the striping penalty at exponential cost in terms of states represented on the BL. A synthesized area and energy analysis of this digital workload is not appropriate in this work as it suffices to estimate the relative fraction of digital workload that is saved, and the preference is toward maintaining generality.

To recap, the first portion of analysis showed the effects of P_{WL}, P_X, and B_X on the *effective memory bandwidth*, leaving open the need for circuit-level models to combine with Equation 1. which will be presented next. The second portion

of analysis showed that for large P, the bulk of the *digital* energy can be saved via analog-domain compute (Figure 4).

IV. Circuit Analysis

This section will develop circuit models toward two distinct goals. First, a CIM energy model will be developed to provide circuit-level energy estimates to be combined with the analytical multiplier given by Equation 1. Second, circuit models will be developed to estimate the outlook for CIM with SRAM with respect to three major challenges: read stability, ADC area, and accuracy. Unless otherwise noted, simulations are performed using TSMC's 28nm PDK [28] and abbreviations are described in Table I.

A. Energy Accounting

Leakage energy is not analyzed in this work as it relates to technology and the amount and format of storage cells rather than to read peripherals and techniques. Active energy for a CIM array can be broken into components just as with a traditional SRAM array [29], as long as some components (configuration and sensing energy) are generalized and others (WL and BL energy, which relate to input and input/weight values, respectively) are converted to expected values. Single-BL read energy is shown in Equation 3:

$$E_{Rd} = \mathbf{E}\left(E_{WL}\right) + E_{CONF} + \frac{i}{m}E_{SENSE} + \frac{i}{m}\mathbf{E}\left(E_{BL}\right) \quad (3)$$

Each of these components will now be analyzed in the context of CIM. For the remainder of this section, the assumption is made that the systems under consideration are *direct* systems (Subsection II-A) which allows principled analysis. Notably, the coefficient of E_{BL} is $\frac{i}{m}$ rather than i since well-designed CIM systems, with their larger BL slews, may avoid wasting BL energy on non-selected columns. This benefit will be given to the SRAM system used for comparisons, as well.

1) Wordline Energy: In contrast with traditional SRAM, the number of WLs which are enabled in any CIM read-step depends on input value - this provides an area for potential overall energy savings. In multi-bit-input systems, the voltage on the WL may be amplitude modulated (pulse-width and pulse-count modulation are more typical), otherwise, the WL will be zero with some probability that is likely close to 50%. Generally,

$$\mathbf{E}\left(E_{WL}\right) = C_{WL}V_{DD}\sum_{j=1}^{P_{WL}}\mathbf{E}\left(V_{WLj}\right) \quad (4)$$

If the input bits are independent, identically distributed:

$$\mathbf{E}\left(E_{WL}\right) = P_{WL}C_{WL}V_{DD}\mathbf{E}\left(V_{WL}\right) \quad (5)$$

In the binary-input case, where WLs are set to 0 or V_{WL}:

$$\begin{aligned}\mathbf{E}\left(E_{WL}\right) &= P_{WL}C_{WL}V_{DD}V_{WL}\mathbf{P}\left(\text{Bit} = 1\right)\\ &\approx \tfrac{P_{WL}}{2}C_{WL}V_{DD}V_{WL}\end{aligned}$$

where the last approximation holds if input bits are binomial distributed with probability 50% (or, *e.g.*, if amplitude modulated inputs are used with a mean amplitude of $V_{WL}/2$). If multi-bit WLs are applied using a train of identical pulses ([23]), a multiplier of 2^{P_X} generally applies.

2) Configuration Energy: Expanding on the column-select energy (E_{SEL}) of traditional SRAM, the E_{CONF} term here represents the average reconfiguration energy incurred between each P-wide CIM read operation. This means:

$$E_{CONF} = C_{CONF}V_{DD}^2 \quad (6)$$

where C_{CONF} corresponds to the average capacitance of any reconfiguration switching nodes.

3) Sensing Energy: In binary-output systems, the analogy to traditional SRAM holds and, given some C_{SA} which characterizes the capacitance of switching nodes in the sense amplifier (SA):

$$E_{SA} = C_{SA}V_{DD}^2 \quad (7)$$

It is possible to hypothesize about sensing energy from design principles or to reference the energy efficiency of implemented ADCs. We use a hybrid approach: for lower precision outputs, we assume the CIM system has access to an SA (as a starting point for modeling a Flash ADC) *at least* as good as that in the reference traditional SRAM system. At some point, the Flash energy exceeds that of the trend seen in published ADC designs, and the model switches.

In a simple comparator (without offset cancellation), capacitance must track effective precision because reduced state separation requires larger input device scaling. This implies, extending Equation 7, that comparator energy must scale quadratically [30] as:

$$E_{SENSE} = \frac{S(P_{EQ})^2}{S(P)^2}C_{SA}V_{DD}^2 \quad (8)$$

where C_{SA} is the equivalent switching capacitance of a reference SRAM's SA and P_{EQ} is the value of P for which the SRAM's SA would provide sufficient matching in a CIM system. P_{EQ} captures differences in design margin (CIM systems might tolerate more readout errors) and the effect of state separation. It will be shown later that $S(P) \approx KV_{DD}/P$ for some constant K (K is the *effective fraction of available dynamic range used to separate states*) which depends on CIM system behavior, confirming that the overall sensing energy trend is indeed quadratic. Simplifying, the energy trend becomes:

$$E_{SENSE} \approx P_{OUT}\frac{P^2}{P_{EQ}^2}C_{SA}V_{DD}^2 \quad (9)$$

where, using $S(P) \approx KV_{DD}/P$:

$$P_{EQ} \approx \frac{KV_{DD} \times D}{V_{SNS}} \quad (10)$$

where the name V_{SNS} is maintained from [29] and refers to the typical BL swing, due to a single on-state cell, required by an SRAM to make an accurate read. If the reference SRAM uses differential sensing (typical) P_{EQ} is additionally divided by two. D is the matching design margin ratio of the SRAM comparator to the CIM comparators. For example, if the standard SRAM requires 6σ margin and the CIM system requires only 2σ between the reference voltage and the nearest state, then $D = 3$. The interaction of D, K, and P_{EQ} are shown for a typical case in Figure 5.

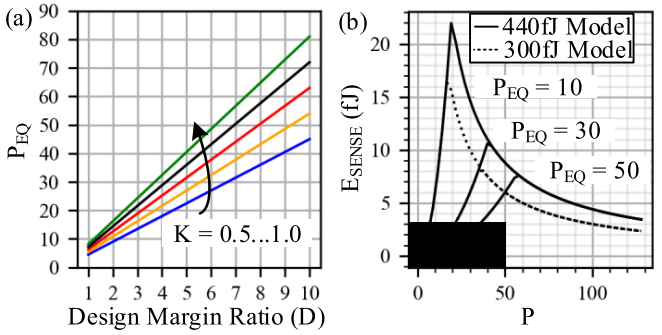

Fig. 5. Sensing energy example following the introduced model. (a) shows the fitting constant, P_{EQ}, for comparing sensing energy in CIM and SRAM. Computed as in Equation 10 with V_{DD} = 900mV and V_{SNS} = 100mV. (b) shows the per-output-state sensing energy as in Equation 8, where $P_{OUT} = P$. SA energy ($V_{DD}^2 C_{SA}$) and a rough blocked-region (where the SA energy trend is no longer observed) are extracted from schematic simulation and combined with the results in (a).

If $P_{OUT} \propto P$, naïve energy scaling is cubic in allowed output states. Avoiding this trend requires a topological change to, at least, reduce the matching requirement. ADC designs which offer superior energy performance at higher precision by way of a different operating mechanism will now be evaluated via a study of published designs. Analysis will be based on ADC survey data [31] as in [2]. Figure 6(a) shows two kinds of trend lines: the dashed black line shows an approximate 2b/decade trend for decreasing per-state energy as precision increases. This trend, however, *cannot* be a result of physical design restrictions, since it corresponds to decreasing *total* energy as precision increases, and must therefore indicate application trends (*e.g.* design effort and more modern processes being dedicated to higher-precision designs). An approximate envelope is drawn by the dotted grey (constant total energy) lines which project the total energy of a few performant higher-precision (ENOB $\approx$ 9) designs backwards assuming that total energy should not get worse as precision drops. The upper grey line gives an estimate which holds near to real implementations below about 10-12 bits and is given by Equation 11:

$$E_{SENSE} \approx 440\text{fJ} \tag{11}$$

This is close to the 0.3pJ estimated in [2] and both will be considered in this work. The interface between the Flash and literature-fit models is shown in Figure 5 (b). The P_{EQ} values, combined with SA energy, determine the per-state energy up to the point when the constant total ADC energy trend allows a sharp drop in per-state energy. There is a technology-dependent region along the bottom of Figure 5 (b), shown blocked, which is inaccessible. This is due to minimum device sizes and the fact that capacitive energy due to the large matching pair will eventually become a less-important component of total comparator energy (secondary, for example, to shoot-through current or wire parasitic capacitance).

4) Bitline Energy: The analysis of BL energy depends on how exactly the CIM readout scheme operates. The single-ended capacitive BL energy is now a function of output state

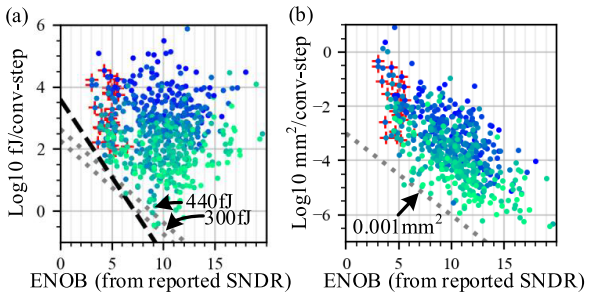

Fig. 6. ADC survey data [31]. Flash ADCs are highlighted (red cross), with year shown in color gradient. Dashed black line shows a hypothetical 2b/decade energy trend while lighter dotted lines show exemplary iso-total-energy trends in (a). Grey lighter dotted line shows an extracted iso-total-area trend line in (b).

and is generally modeled as an expected value:

$$\mathbf{E}(E_{BL}) = C_{BL} V_{DD} V_{SNS} \mathbf{E}(\beta(p)) \tag{12}$$

where $\beta(p)$ is introduced as a multiplier to extend this energy definition to CIM systems; $\beta(p)$ is such that $V_{SNS}\beta(p)$ is the voltage change on the BL required to represent output state p. The most direct top-down extension of SRAM BL energy to differential-mode CIM BL energy therefore occurs if $\beta(p) = p$, which implies identical any-error rate at iso-matching and noise. This is incompatible with bottom-up modeling due to the nonlinear state placement that straightforward (RC-PD, R-divider) direct operating modes yield and, furthermore, is incompatible with CIM systems for which $P > V_{DD}/V_{SNS}$ (due to a loosened any-error rate requirement or improved noise and matching). As a result, bottom-up BL energy models will now be proposed and discussed.

For generalized energy analysis with respect to 1-bit input RC-PD CIM systems (and later R-dividing systems), it suffices to model memory cell PD circuits as perfect resistors with some resistance R. This analysis roughly extends to multi-bit (amplitude modulated WL) inputs, although it is useful to model the 1-bit case as a representative example. It is helpful also to define a variable, $\gamma = t/RC_{BL}$, representing normalized PD time. It can then be shown that the maximum worst-case state separation occurs with:

$$\gamma_{opt} = \ln(P) - \ln(P-1) \tag{13}$$

this gives a (V_{DD}-normalized) BL swing of:

$$\frac{V_{SNS}\beta(p)}{V_{DD}} = 1 - \left(\frac{P-1}{P}\right)^p \tag{14}$$

where p is the number of cells pulling down on the BL. BL energy is therefore exponential in on-state cells and average BL energy is highly dependent on application statistics. As given by Equation 13, optimum PD time is super-linearly decreasing in P. As a result, BL energy for both differential and SE schemes tends to decrease as the optimal point is shifted for larger P. The interaction of Equation 14 and binomial distributed cells is shown in Figure 7. Two standout observations: the lowest energy for the differential scheme is the same as the maximum energy for the SE scheme, and average energy is likely to consistently be near the median.

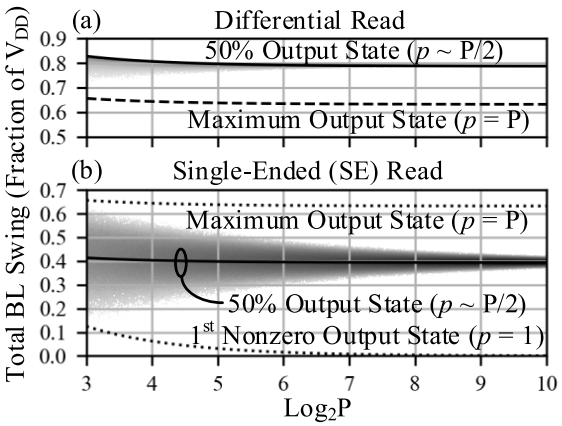

Fig. 7. RC-model BL voltage swing as a fraction of VDD for RC-PD systems due to Equation 14 at a few values of p. To show smooth trend lines, p is interpolated between discrete output values. A log-shaded 10^5-sample Monte Carlo experiment (using normal approximation to a binomial with probability = 50%) is used to show long-term energy behavior.

This model does not consider non-optimal PD times, although reduced PD times are unlikely to be a strong candidate for energy savings due to the relative insensitivity of overall power to PD time, compared to the strong dependence of state separation (and therefore sensing energy) on PD time.

Compared to the RC-PD systems just described, R-divider systems suffer from increased energy at a given BL slew due to the added rail-to-rail leakage path. They have nonetheless been demonstrated in hardware [14] and will be briefly discussed here. There are two important energy components: capacitive and feed-through. Capacitive energy is added on top of the feed-through energy and is modeled as before with Equation 12, whereas feed-through requires a new model:

$$E_{BLSC} = pT_{SNS}V_{DD}\frac{(V_{DD} - \beta(p)V_{SNS})}{R} \tag{15}$$

where T_{SNS} is the equivalent time spent with the BL voltage settled during the read operation and $\beta(p)$ is conceptually identical to the $\beta(p)$ in Equation 12. Unlike the RC-PD case, here ideal (V_{DD}/P) state separation is possible given an ideal pull-up transfer function (Equation 16, V_S is the state separation voltage). That said, it is unclear how such an inverted-parabola transfer function could be implemented. It is more likely that a system designed this way would implement a voltage regulator combined with current sense to achieve ideal state separation, although such designs are out of scope for this work.

$$I_{BL}(V_o) = \frac{1}{RV_S}(V_{DD}V_o - V_o^2) \tag{16}$$

If the pull-up device is modeled as a resistor, the energy can be compactly written as:

$$E_{BLSC} = T_{SNS}\frac{V_{DD}^2}{R_{PU} + R/p} \tag{17}$$

If R_{PU} is set to optimize minimum state separation given cell resistance R, then the following can be substituted for R_{PU}:

$$R_{PU} = \frac{R}{\sqrt{P^2 - P}} \approx \frac{R}{P} \tag{18}$$

Considering T_{SNS} may be a few $C_{BL}R_{PU}$ time constants, feed-through energy can be significant. This observation favors RC-PD systems.

B. CIM-Specific Challenges

1) Read Stability and Bitcell Implications: Read stability is an important consideration in any SRAM-based design [32] and the extended BL swings used in CIM with SRAM may be expected to cause stability issues in an array designed for traditional SRAM use. Here, simulation and statistical analysis will be used to address whether *larger (than standard 6T) bitcells are required to maintain read stability when CIM is used*. Such larger bitcells might feature dedicated read circuitry (7T, 8T, etc.) or devices sized for read stability.

Popular choices for PD chains are shown in Figure 8. This figure shows the SE case, in which the pass device on one half of the cell is supplemented with dedicated read devices. In the differential case, 8T or 10T are needed instead of 7T or 8T. In Figure 8 (a), the (Rd) label on the WL and BL indicates that the cell may be used in an SE mode where only PgR is used for read (enabled by a dedicated WLRd). This may improve read margin (RM). Figure 8 (c) shows the characteristic 2T PD circuit of the traditional 8T bitcell, while (b) shows how, as in [11], PgRd may be eliminated at the cost of requiring WLRd to drive a low-impedance node.

This section will use a definition of RM as in, *e.g.*, [33] and shown in Figure 9, which naturally extends to full-scale BL swings and lends itself directly to intuitive measurement and analysis. Using the same terminology as [33] the read-margin test uses SN1 = H, SN2 = L, and transitions BL(W) from V_{DD} to $-\infty$. The RM is defined as V_{DD} minus the BL(W) voltage at which SN2 transitions from L to H ("the amount of BL read slew allowed before the internal state of a victim cell flips"). Where write margin is needed, a similar metric, "combined WL margin" (CWLM), is used [34], [35].

To avoid transitioning to a different cell topology, insufficient 6T-cell RM may be addressed with two techniques: scaling Pd: Pg and reducing read WL voltage (V_{WLRd}). To show the trends clearly, both techniques are simulated using minimum-size nominal-Vt transistors and 900mV V_{DD} (although variation *must* be considered for robust CIM SRAM design with respect to read/write margin, even if accuracy loss is tolerable, to avoid bit flips and stuck cells).

Figure 10 shows the two trade-off spaces. Figure 10 (a) shows how sizing is generally a poor method for attempting to achieve read stability across very wide BL swings since the sensitivity of write margin (CWLM) to stronger Pd is large relative to that of RM. Not shown for brevity, scaling Pu: Pg is worse still and results in approximately twice the relative CWLM sensitivity. Further, while area is not shown in these plots, in layout experiments the final 10x Pd point required 235% cell area even relative to the large non-push-rule minimum bitcell used in this work. More promising is WL amplitude adjustment, the effects of which are shown

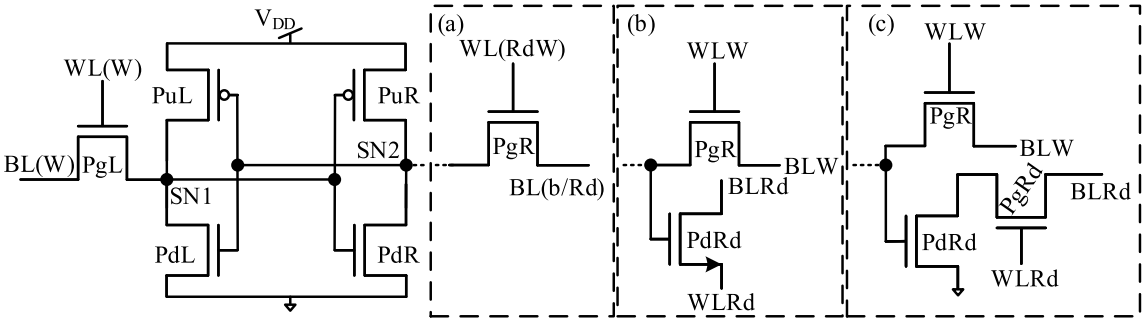

Fig. 8. SRAM bitcells showing the BL PD circuit in 6T(a), 7T(b), as in [11], and 8T(c) implementations. In (a), the BL net attached to the right side of PgR can be the complementary BLb in the typical SRAM scenario (and for write, in either case) or a designated SE read BL. In (b) and (c), this pass-device is exclusively used for write. In (b), a single transistor is used as the dedicated read PD circuit, whereas in (c) two series transistors are used (so that the WLRd drives a capacitive node).

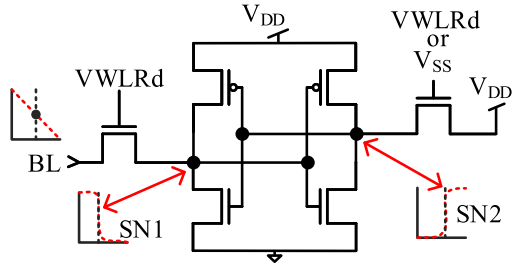

Fig. 9. The DC BL read margin (RM) test condition. The access device gates are set to VWLRd (in the case of single-ended read, the write-only access device, right side, is disabled). Internal nodes are configured, left to right, to represent '1' and '0.' The BL then slews downward (opposite BL is tied to the worst-case condition, V_{DD}) until the stored state flips. The amount of slew allowed ($V_{DD}-$ final V_{BL}) is the BL RM.

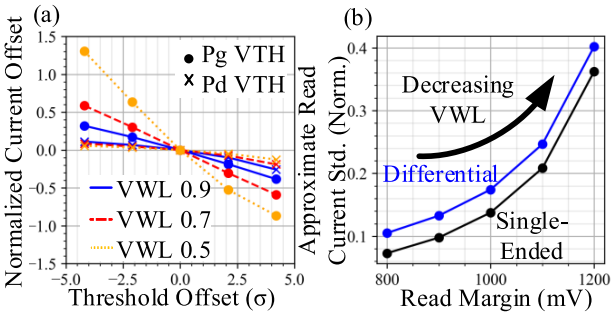

Fig. 11. The sensitivity of bitcell PD current to threshold offset in the two devices involved in PD is shown in (a), the resulting normalized standard deviation trend estimate is shown in (b). Same conditions as in Figure 10.

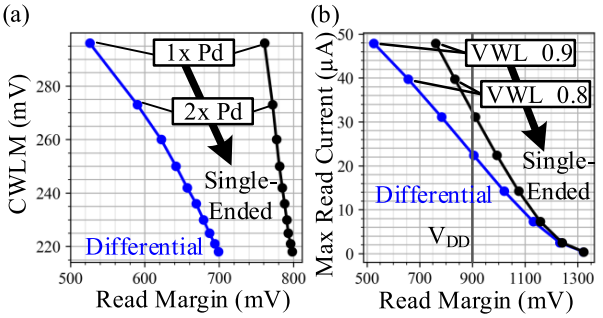

Fig. 10. Increasing RM in a 6T minimum-sized standard-Vt cell via (a) scaling Pd and (b) reducing V_{WL}. Schematic-level simulations with variation omitted for clarity (nominal corner, $V_{DD} = 900mV$, 25° C).

in Figure 10 (b). Here, the trade-off is read current, which is relatively benign from the perspective of robustness and energy. It is possible to achieve a full V_{DD} of RM at the cost of 'only' about 50% of read current. In either case, the greater robustness offered by the SE read is apparent. The SE and differential cases converge in 10 (b), although this only occurs as V_{WL} nears the threshold. Most importantly, *full BL swings can be allowed as long as V_{WL} is sufficiently low.*

On this concluding point, reducing V_{WL} increases relative bitcell PD current offset due to Vth variation. This introduces a more important tradeoff space for V_{WL} scaling than maximum read current: allowed BL swing (i.e. RM or state separation) vs. bitcell current error. The ramifications of this point are shown in Figure 11. A schematic-level sensitivity analysis is performed using local variation only ($A_{vt} = 2.86$mV/um as in [28]) and a segment of the results are shown in Figure 11 (a) across a 400mV range of V_{WL}. The complete results are used, with a linear approximation, to calculate the standard deviation of peak read current that results from local variation in the minimum, nominal-Vt cell. Note that below $V_{WL} \approx 500mV$ the current errors due to Vth offset become enormous and nonlinear due to the proximity to threshold voltage.

2) ADC Area Trends: The difficulties with estimating ADC area trends for CIM are the same as those for estimating energy trends: there is a many-dimensional design space and only two weak methods for estimating area (simple matching models under strong topological assumptions or analysis of existing work) are compelling. If area is determined by a matching-limited Flash ADC, then the situation for area is the same as that for energy and Equations 8 and 9 give the trend as long as area is appropriately substituted for energy.

Figure 6(b) shows the area trend in fabricated ADCs, with a constant-area 0.001mm^2 trend line superimposed, again using data from [31]. The plot is similar to that for energy although the alignment of the bulk of the samples with the constant-area slope is much clearer. As expected, however, the lack of modern examples showing even a constant-area trend for lower precision is clear with a large pocket above the trend line below 6 bits. The 0.001mm^2 trend is potentially a problem

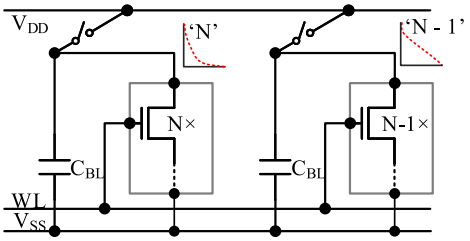

Fig. 12. The state-separation testbench. Note that C_{BL} must be much larger than active capacitances in switching cells for separation behavior to be reliably predicted.

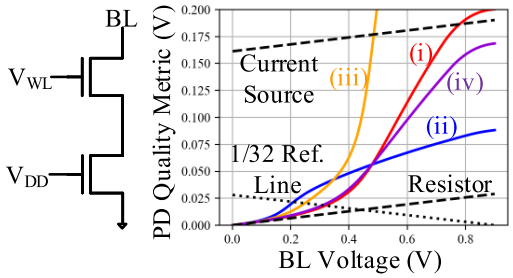

Fig. 13. 32-cell PD "quality" (allowed state separation during BL slew) for a few PD configurations (nominal-Vt, minimum size) are shown with a reference line drawn to show roughly which section of each curve is relevant for state separation: (i) is $V_{WL} = 0.9$, (ii) is $V_{WL} = 0.5$, (iii) is with V_{DD} and V_{WL} ports swapped and $V_{WL} = 0.5$, and (iv) is a single-T with $V_{WL} = 0.5$.

for advanced nodes: this is roughly 7.8kb or 37kb of SRAM cells in 28nm and 7nm, respectively [28], [36], and presumably several ADCs must be tiled per array to achieve satisfactory throughput. This encourages development of compact, energy-efficient ADCs for CIM.

3) State Separation: Modern scaled CMOS processes are limited in V_{DD} due to reliability issues with scaled devices at higher voltages and the requirement for low power in logic and SRAM. It is desirable to have embedded memories operate at or near logic voltages to simplify design and reduce energy. This implies that all of the possible CIM output states must reside in a limited dynamic range; at best:

$$V_{SEP} = \frac{V_{DD}}{P} \tag{19}$$

where V_{SEP} is the characteristic state-to-state separation and the single-ended sensing margin (here meaning the tolerable equivalent comparator input-pair offset before error occurs) is half this. This section will illustrate why Equation 19 gives an ideal estimate for the common *direct* CIM SRAM systems by showing theory, analysis, and simulation to predict state separation trends. RC-PD systems will be analyzed in the greatest depth while R-dividing systems will also be discussed.

In the case of RC-PD systems, Equation 19 applies to the ideal case where bitcells form ideal current sources with:

$$I_{Pd}T_{Pd} = \frac{C_{BL}V_{DD}}{P} \tag{20}$$

where I_{Pd} and T_{Pd} are the single-bitcell PD current and the BL development time. To state the issue compactly, transistor-based PD circuits can perform poorly for large P because of finite and voltage-dependent output resistance. In addition to finite saturated output resistance, the increasing resistance in the linear region as the BL slews downward causes state separation to compress. That is, the Vd/Id transistor transfer function provides feedback from BL voltage to the BL slew rate. This feedback relationship is key to predicting the PD "quality" of a transfer function and invites visual analysis - *a priori* it is unclear whether to prefer flatter output current, more time in saturation, *etc.*

The state-separation testbench is represented in Figure 12 and features a system with N on-state cells and another with $N-1$ on-state cells. State separation is zero at $t = 0$, increases, hits a peak, and starts to collapse (exponential convergence) approaching zero at $t = \infty$. State separation must start to collapse when the PD currents are equal - *i.e.* when the difference in BL voltage, translated by the transfer function, suffices to cause the $N-1$ system to start to catch up to the N system. Concisely,

$$(N-1)I_{Pd}(V_{BL,N-1}) = NI_{Pd}(V_{BL,N}) \tag{21}$$

where $I_{Pd}(V)$ is the I/V transfer function of interest. Therefore, a quality of import which determines achievable state separation is *the required increase in BL voltage for an* $(N-1)$*-on-cell system relative to the* N*-on-cell system to equalize the PD rate of the BLs.*

This quality ("allowed state separation at some point in the slew") is plotted with $N = 32/31$ and $V_{DD} = 0.9$ for a variety of PD circuits in Figure 13. A reference line with slope 1/32 is drawn from right to left showing ideal state separation, during BL slew, as a function of BL voltage. The intersection of this line with the quality metric curves estimates the state separation allowed by the transfer function at this value of N. The achieved separation will be lower than this, since the slope of a true state-separation line will decrease across the slew as the BLs converge. This predicts that the resistor model yields the worst state separation, since it lacks any current saturation, followed by the shown PD chain with simple biasing $V_{WL} = 0.9 = V_{DD}$. Reducing V_{WL} to 0.5 reduces current but allows the output device to remain saturated deeper into the curve, greatly improving state separation. Reversing the two transistors to form a more typical cascode causes the peak allowed separation to dramatically increase, but results in worse separation. A single-T PD circuit and a current source with large finite output resistance are also shown. The predictions of the model are confirmed with state-separation simulations (not shown). Concluding, it is important, for state separation, to maintain current-source behavior as deep into the BL swing as possible. This can supersede the importance of the value of output resistance *while saturated*.

Assessing state separation is further complicated since the time at which peak state separation occurs shifts lower as more cells are enabled - *e.g.* the optimal time when comparing 1 vs. 2 is much longer than that for comparing 31 vs. 32. Application statistics might inform the choice of PD time, but it can be shown that optimizing timing for minimum state separation causes an attractive near-constant set of state separations whereas optimizing for mean or median can result

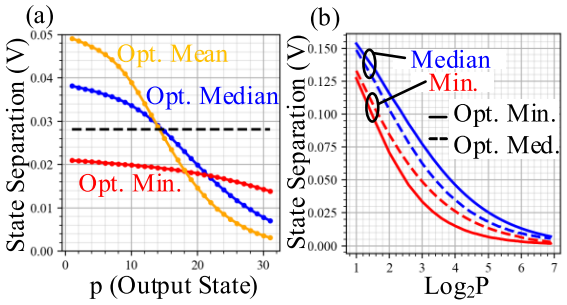

Fig. 14. As shown in (a), when PD time is chosen to optimize the minimum (P vs. P - 1) neighbor-state comparison, the result is a flatter distribution of state separations with potentially usable higher-value states. This contrasts with the case when mean or median state separation are chosen used to optimize PD time, where lower-value states use a disproportionate amount of the BL swing. The ideal state separation is overlaid with a dotted line. In (b), the minimum or median state separations are shown when PD time is chosen to optimize for one or the other (over a range of maximum allowed output state, P). Simulated using the PD circuit as in Figure 13(ii) with large C_{BL}.

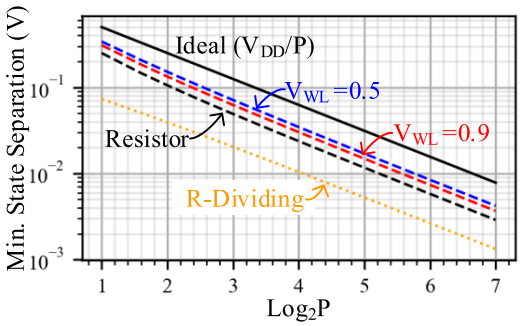

Fig. 15. Worst-case (highest-value state) state separations under the following conditions, from top to bottom: ideal current source, RC-PD with $V_{WL} = 0.5$ and 0.9, RC-PD with a resistor, and R-dividing with a PMOS pull-up sized as in Equation 18. These separations are optimistic and do not consider *any* problematic capacitances, *i.e.* simulations were performed numerically using extracted 25° C DC transfer functions to avoid these effects.

in good separation for low- and medium-value states but unusable high-value states (detailed in Figure 14).

Minimum state separations for a few readout methods are shown in Figure 15. As per the above discussion, the RC-PD system is biased as in Figure 13 with $V_{WL} \ll V_{DD}$ and the PD time is set to optimize the state separation of the highest output state. For the R-dividing system, recalling the discussion from Section IV-A, R_{PU} is chosen, as in Equation 18, as an appropriately-sized pull-up PMOS (simulation confirms that sizing PMOS for drive strength according to R/P is optimal for minimum state separation to within the 5nm grid step). The results are as expected with the naive R-dividing method performing substantially worse for worst-case state separation. For RC-PD, even the optimized transistors do not dramatically outperform the resistor. A useful observation is that the slope consistently corresponds to $1/P$, *e.g.* whereas the ideal state separation is defined by V_{DD}/P, the $V_{WL} = 0.5$ worst-case state separation under optimized conditions is defined by $K \times V_{DD}/P$ for $K \approx 2/3$.

4) Cell, Timing, and Noise-Induced Errors: In addition to increasing the chance of extrinsic errors by requiring the ADC to work with very small sensing margins, greater P increases the chance of intrinsic errors in which the BL voltage itself is incorrect due to cell current errors, timing errors, or noise. The most straightforward effect is accumulating cell current errors as the number of on-state cells is allowed to increase. Indirectly, however, system accommodations which may be required when increasing P can also increase the chance of error: as mentioned, reduced V_{WL} is desirable for increased state separation in RC-PD systems (and robustness in the 6T case), and reduced PD time is required to optimize state separation for higher-number states which increases sensitivity to PD timing errors. Furthermore, in extreme cases of small state separation the BL noise could become important.

These intrinsic error sources, cell current, timing, and noise, will now be briefly discussed. Before doing so, it is worthwhile to note that there is no clear way to discuss error in CIM results in an application-agnostic sense. In the most direct analogy to traditional bit-error rate (BER), it may be useful to consider any-error rate (AER). It should be noted, however, that errors due to the sources discussed here are typified by less severe compute consequences relative to an SRAM bit-error since, briefly, the error is always least-significant-bit (LSB) weighted with respect to the current binary position. For this reason, it is useful to model standard deviation (std. dev.) of BL voltage as a function of error source std. dev. as will be done in this section.

Ignoring systemic offsets, cell current errors arise due to random deviations in an individual cell's Id/Vd transfer function relative to the expected transfer function. Cell current variation vs. WL voltage has already been briefly discussed above (Figure 11), so this section will focus on two aspects: the relationship between random cell threshold offset-induced current error and BL voltage error, and the accumulation of BL voltage error when more cells are turned on. Modeling is simplified importantly if cell current error can be modeled as a BL-voltage-independent constant multiplier, that is:

$$I_{CELL}(V_{BL}) = (1 + \epsilon_{CELL}) I_{CELL,NOM}(V_{BL}) \qquad (22)$$

where $I_{CELL,NOM}$ is the nominal cell current and ϵ_{CELL} is the normalized and appropriately (*e.g.* normally) distributed error term. If Equation 22 is obeyed, then a BL-current induced error occurs in isolation when $\sum_i \epsilon_{CELLi} \geq 1$ for the set of on-state cells. For p (independent, identically distributed) on-state cells, BL current error has std. dev. $\sigma_{CELL}\sqrt{p}$ where σ_{CELL} is the std. dev. of ϵ_{CELL} under this model. While the fractional amount of sensing margin lost is roughly $\sum_i \epsilon_{CELLi}$, a more accurate model must incorporate the nonlinear mapping of *count of on state cells* onto *resulting BL voltage*, for example:

$$\sigma_{VBL,CELL}(p) \approx \sigma_{CELL}\sqrt{p}\frac{\partial V_{BL}(p)}{\partial p} \qquad (23)$$

where $V_{BL}(p)$ is the nominal mapping of cell count to BL voltage and $\sigma_{VBL,CELL}(p)$ describes the std. dev. of BL voltage error due to cell current error.

While cell current error applies to any direct system, WL-timing induced error is particular to RC-PD systems and is an artifact of static references used for a dynamic process

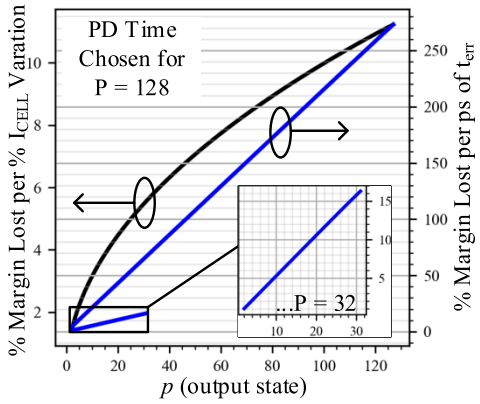

Fig. 16. Loss of sensing margin relative to cell current and WL timing errors. Timing errors assume $C_{BL} = 100fF$ and sensitivity scales inverse-linearly with BL time constant. Trends when PD time is optimized for both $P = 128$ and $P = 32$ (magnified) are shown.

(or mismatched timing between the true WL and the reference generator). The analysis is similar to the above and an effective model is:

$$\sigma_{VBL,t}(p) \approx \sigma_t \frac{\partial V_{BL}(p)}{\partial t} \tag{24}$$

Note that in either case V_{BL} is a function of many other system parameters, including P. The trends for the two coefficients of error are shown in Figure 16 with the coefficients normalized to state separation at p. After normalizing to state separation, cell current error sensitivity is just $\sqrt{p}$ as expected. Timing error sensitivity depends on the P chosen to set the PD time, and example lines for $P = 32$ and $P = 128$ are shown. Simulations show that under an example C_{BL} of 100fF, the sensitivity of RC-PD systems to a few picoseconds of timing error is a potentially large problem (as might be predicted due to the very small time constant) whereas the importance of bitcell current error is largely a function of process.

The final potential contributor to readout error which must be considered is noise. A full noise analysis is outside the scope of this work, but two comments can be made. First, noise on power rails, coupled through the substrate, due to adjacent BLs, and so on are more important in CIM than in standard SRAM due to the lower sensing margins and acceptable coupling noise immunity is unlikely without design changes relative to a normal SRAM. Second, total bitcell-induced BL noise is not necessarily a problem: flicker noise decreases with p (which increases the effective size of the PD device), and thermal noise is inversely a function of potentially large C_{BL}.

V. Energy Examples and Methodology

A. Methodology

Two principles enable strong conclusions based on the energy comparison examples that will be shown in this section. First, all analysis is stated in terms of the *ratio* of energy used by a CIM system to that used by the corresponding SRAM system. This enables generalized analysis of trends from circuit fundamentals (using models from Section IV)

TABLE III
Example CIM SRAM Systems and Reference SRAM

System	SRAM	(a)	(b)	(c)	(d)
Scheme	SE	SE Direct BL-PD			
Technology	28nm CMOS				
SRAM Cell	6T 0.32um^2 Non-Push Minimum 0.9V				
V_{WL}	0.5V				
Array Dim.	512x128 (Physically Square)				
BL Mux. Ratio	8:1			1:1	8:1
B_x	Same as CIM	8	4	4	1
P_x	1			4	1
P_{WL}	1	P_{WL}	P_{WL}	P_{WL}	P_{WL}
ADC States	2	$P+1$	64	16	2
ADC Margin	6σ	3σ	1σ	0.075σ	1σ

while ensuring that any comparison is intrinsically apples-to-apples: both systems use the same array with the same capacitances, have access to the same SA with the same scaling trend, and so on. The basis of this ratio-driven analysis is the choice to model total MAC energy as consisting of a digital-logic component (Section III) added to a memory-access component (Equation 1, Section IV), where the ratio of the former to the latter in a traditional implementation is a function of technology generation, application, and design choices. Second, only *direct*, RC-PD CIM systems are analyzed due to the practical advantages in area, energy, and state separation that such systems offer as described above.

The goal of this section is therefore not to compare *total* energy of the CIM and traditional systems but rather to compare memory access energy. CIM designs are compared to an SRAM design with the same dimensions which uses SE read. To this end, a standard-Vt 6T minimum-transistor 0.320um^2 SRAM cell layout was created to generate extracted SRAM array parasitics. A 128×128 array is used to extract capacitances while avoiding boundary effects, then scaled numerically. Transistors without push rules are used to align with the devices whose parameters have been simulated throughout this work. 512×128 is chosen for the presented results because it is a physically-square array, meaning the BL and WL capacitance are weighted equally. The model follows Equation 3 and subsequent analysis. WL energy is modeled as in Equation 5, and BL energy is modeled using Equations 12 and 14. The ADC energy model has two components: to provide more accuracy relative to Equation 9, SA (and Flash ADC) energy figures are modeled using schematic-level voltage comparator simulations following a standard design [37] across a range of input-pair sizes; as the number of output states is increased, Flash ADC energy eventually exceeds the constant-energy ADC models, and ADC energy is clipped at 440fJ (300fJ model shown in dotted line).

The relationship between BL capacitance and SA energy has an effect on the active-energy-optimal BL PD time in a standard SRAM array. That is, more PD time creates a larger sensing margin and allows for a smaller, lower-energy SA. Therefore, to strike a fair comparison between SRAM and CIM SRAM, the SA sizing is set so that the chosen SA model achieves minimum total energy while satisfying 6σ of design

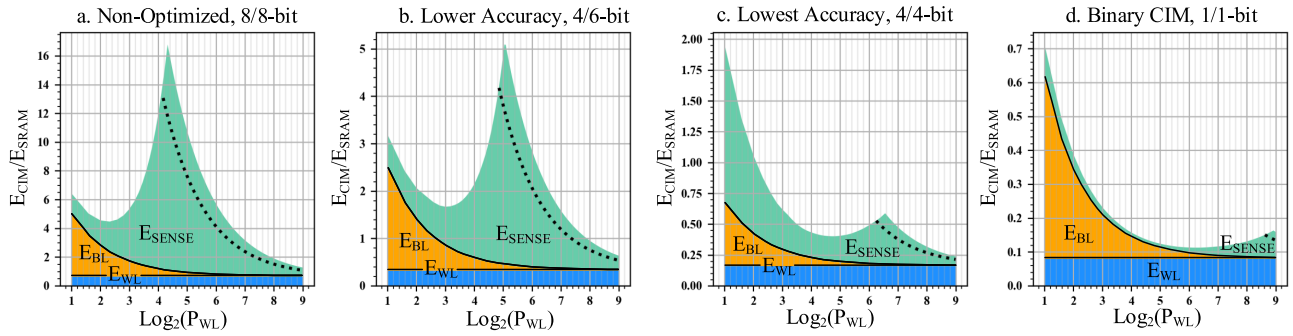

Fig. 17. Predicted CIM performance results using the extracted models. Parameters for (a) - (d) are defined in Table II. Array capacitances and SA energy are extracted from layout and schematic simulations, respectively. SAs are extracted across a range of matching requirements to extract a trend line for use across varying achieved state separation. Energy is normalized to an energy-optimized SE 6T SRAM array conforming to the same models and accomplishing the same memory task. Secondary energy sources such as multiplexer switching are ignored. These models use 440fJ as the upper-limit on sensing energy (see Equation 11 and discussion) with the 300fJ upper-limit [2] overlaid with a dotted line. For (d), this may be interpreted as a high-precision offset cancelled comparator (although such a comparator will likely, in fact, outperform 300fJ or 440fJ, this is tangential to the trend shown in here).

margin. Parameters for these demonstrative simulations are given in Table II.

A few assumptions simplify the modeling process. Bits and memory cells are assumed to be binomially distributed (50%) rather than following any specific application. Also, auxiliary energy costs (C_{CONF}) are not modeled and, importantly, data movement costs are not modeled (as if the MAC logic were directly abutted to the SRAM).

B. Results

Modeling results are presented in Figure 17 as ratios of memory access energy across increasing parallelism P_{WL}. As mentioned, these examples have omitted the effects of digital workload savings which were shown to be plausible by Figure 4 and associated discussion. All the shown examples demonstrate an energy peak at which the constant-energy ADC model outperforms the scaled Flash ADC. Figure 17 (a) illustrates the most pessimistic case for CIM: the MAC is computed in full 8-bit precision, and the ADC is designed to recover the full precision ($1+P$ bits) on the BL while satisfying 3σ of design margin. Loosening these three requirements yields improvements and the potential for energy-efficient designs. Figure 17 (b) demonstrates 4-bit inputs, which halves the energy ratio relative to (a), and only requires the ADC to achieve 6 bit precision with 1σ of design margin. The last change improves Flash ADC energy scaling and push the extrema toward higher P_{WL}, although CIM memory accesses remain more expensive than SRAM out to $P_{WL} \approx 2^8$.

One recent work ([23]) showed high efficiency (>351 TOPS/W) using RC-PD. It is challenging to contextualize the performance of a 7nm 4-bit MAC engine, but this work provides the framework for a higher-efficiency example. Three changes allow CIM to consistently outperform SRAM (Figure 17 (c)): the ADC precision is reduced to the input precision ($P_{OUT} = 15$), 4 bits are applied on each WL, and most importantly the design margin is aggressively reduced to 0.075σ.[6] This allows >50% reduction in memory access energy across a range of P_{WL}. As in [23], the 4-bit WLs have been modeled as successive (thermometer-coded) pulses applied on the WL, and reducing the BL MUX ratio for this example is critical to dilute the increased WL energy. Energy efficiency achieves its logical maximum with a binary CIM example (Figure 17 (d)): even if the ADC design margin is tightened to 1σ, and the BL MUX ratio is reduced back to 8:1, the lack of input striping penalty and quadratic (vs. cubic) Flash scaling allow very efficient memory accesses.

[6]Extracted from [23] for the case of 2^{10} states on the BL and 700mV dynamic range. [23] actually applies many more than 2^{10} states to the BLs, as binary weighted versions of 4 columns are combined before sensing.

VI. Discussion and Conclusion

A. Design Insights

The models in this work support certain guidelines about CIM system design. From Figure 15, state separation in RC-PD systems can easily be below 10mV when $P \geq 2^6$; this sets a stringent requirement for noise coupling onto the BL (not simulated in this work). Accurate readout is further limited by cell current variation, WL timing errors, and ADC limitations. These present complex interactions: since Flash energy is inversely quadratic in K, and noise-induced accuracy issues are reduced with larger K, wide voltage swings on the BL are desired. SRAM cells are not natively robust to full-swing reads, and sizing is largely ineffective at correcting read margin to the extent needed (Figure 10). Reducing V_{WL} improves read margin, reduces the effect of absolute timing errors (Figure 16) by increasing the BL pulldown time constant, and improves state separation (Figure 15). However, decreased V_{WL} linearly to super-linearly increases cell current offset (Figure 11). If V_{WL} is reduced as far as allowed by design accuracy requirements and read margin, channel resistance, and/or state separation requirements are not met, the logical step is to increase the length of the read pass-device which improves read margin, channel resistance, and cell current offset at the cost of write margin.

Generally, single-ended implementations are preferred since they offer better read margin and reduced BL energy. That said, for CIM, BL energy (Equation 12 and Figure 17) is less relevant for large P as per-state BL swing is necessarily reduced. This motivates taller arrays with more C_{BL} which improves density, increases the PD time constant (reduced

sensitivity to timing offsets), potentially reduces coupling noise, and offers a lower-noise sampling capacitor to benefit a high-precision ADC if one is used. This also may allow R-dividing systems (or more complex BL-regulating systems) to perform well, since they trade added BL energy for reduced sensitivity to absolute BL timing errors.

At a higher abstraction level, Equation 1 shows that CIM performance is proportional to P_{WL} and P_X and inversely proportional to B_X due to bit striping. P_{WL} is preferred to P_X since P_X exponentially increases the number of states on the BL. Similarly, P_X shows inferior trends to P_{WL} in Figure 4, as well, in terms of saved compute vs. states on BL. P_X is valuable if application or array parameters restrict P_{WL}; without specific reason otherwise, it is better to scale P_{WL}.

B. CIM Outlook

From Figure 17 and the motivating analysis, a few statements can similarly be made about how CIM systems can outperform SRAM. First, if a traditional system is heavily biased toward compute energy, especially at lower precisions, then CIM (analog compute) may improve efficiency by reducing digital computing workload (Figure 4) at similar memory access costs. Second, at higher (input) precisions, RC-PD systems struggle to overcome the linear penalty of striped input bits (Figure 17(a)). Third, if a CIM system is to directly achieve improved memory access costs, the following flow defines the energy savings: (1) the input-striping penalty applies a multiplier to energy that must be overcome; (2) increasing P through P_{WL} and potentially P_X eventually reduces per-memory-cell E_{BL} (Equation 12 and Figure 7), trading away sensing margin; (3) the sensing system (ADC) accommodates the reduced sensing margin without imposing an unmanageable energy cost or at reduced per-state memory cost. Since, per-state, Flash ADCs are characterized by super-linear costs and CIM ADC costs are only linearly amortized, achieving (3) *implies* reduced ADC design criteria (accuracy, precision) *or* a design which places enough states on the BL (P) to amortize a high-precision high-efficiency ADC.

Accuracy loss in these systems relates directly (and potentially super-linearly) with energy efficiency. The accuracy characteristics of the ADC, and overall system, are core to understanding the merit of a CIM implementation - accuracy notwithstanding, it is possible to place arbitrary numbers of states on the BL for arbitrary efficiency. Works like [2] and [4], which relate analog computing errors to viability for larger networks (or other such real-world tasks) are therefore important for designing and assessing this kind of CIM system.

Representing fewer states on the BL typically corresponds to better accuracy due to less accumulating error and less sensitivity to noise, and potentially enables better generality across applications (which might have lower-dimension vectors). Moving forward, we expect designers to continue to focus on CIM systems with ADC designs that are energy efficient across a lower precision range (*e.g.* 2-7 bits), like the medium-precision successive-approximation register (SAR) designs recently shown in some CIM SRAM works [17]–[19], [21]. To avoid poor array efficiency and a resulting potential increase in expensive off-chip data movement, successful designs will likely demonstrate novel ADC architectures with smaller footprints, the popular trend so-far having been not to report detailed peripheral area breakdowns.

REFERENCES

[1] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017.

[2] A. S. Rekhi *et al.*, "Analog/mixed-signal hardware error modeling for deep learning inference," in *Proc. 56th Annu. Design Autom. Conf.*, Jun. 2019, pp. 1–6.

[3] B. Reagen *et al.*, "Ares: A framework for quantifying the resilience of deep neural networks," in *Proc. 55th ACM/ESDA/IEEE Design Autom. Conf. (DAC)*, Jun. 2018, pp. 1–6.

[4] S. K. Gonugondla, C. Sakr, H. Dbouk, and N. R. Shanbhag, "Fundamental limits on the precision of in-memory architectures," in *Proc. 39th Int. Conf. Comput.-Aided Design*, Nov. 2020, pp. 1–9.

[5] B. Murmann, "Mixed-signal computing for deep neural network inference," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 29, no. 1, pp. 3–13, Jan. 2020.

[6] J.-H. Yoon, M. Chang, W.-S. Khwa, Y.-D. Chih, M.-F. Chang, and A. Raychowdhury, "29.1 A 40 nm 64Kb 56.67 TOPS/W read-disturb-tolerant compute-in-memory/digital RRAM macro with active-feedback-based read and *in-situ* write verification," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, vol. 64, Feb. 2021, pp. 404–406.

[7] C.-X. Xue *et al.*, "15.4 A 22 nm 2Mb ReRAM compute-in-memory macro with 121-28 TOPS/W for multibit MAC computing for tiny AI edge devices," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 244–246.

[8] C.-X. Xue *et al.*, "16.1 A 22 nm 4Mb 8b-precision ReRAM computing-in-memory macro with 11.91 to 195.7 TOPS/W for tiny AI edge devices," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, vol. 64, Feb. 2021, pp. 245–247.

[9] Q. Liu *et al.*, "33.2 A fully integrated analog ReRAM based 78.4 TOPS/W compute-in-memory chip with fully parallel MAC computing," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 500–502.

[10] B. Crafton, S. Spetalnick, Y. Fang, and A. Raychowdhury, "Merged logic and memory fabrics for accelerating machine learning workloads," *IEEE Des. Test. Comput.*, vol. 38, no. 1, pp. 39–68, Feb. 2021.

[11] C. Yu, T. Yoo, T. T.-H. Kim, K. C. T. Chuan, and B. Kim, "A 16K current-based 8T SRAM compute-in-memory macro with decoupled read/write and 1-5bit column ADC," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Mar. 2020, pp. 1–4.

[12] S. K. Gonugondla, M. Kang, and N. Shanbhag, "A 42pJ/decision 3.12 TOPS/W robust in-memory machine learning classifier with on-chip training," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2018, pp. 490–492.

[13] J. Zhang, Z. Wang, and N. Verma, "In-memory computation of a machine-learning classifier in a standard 6T SRAM array," *IEEE J. Solid-State Circuits*, vol. 52, no. 4, pp. 915–924, Apr. 2017.

[14] W.-S. Khwa *et al.*, "A 65 nm 4Kb algorithm-dependent computing-in-memory SRAM unit-macro with 2.3ns and 55.8 TOPS/W fully parallel product-sum operation for binary DNN edge processors," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2018, pp. 496–498.

[15] A. Biswas and A. P. Chandrakasan, "CONV-SRAM: An energy-efficient SRAM with in-memory dot-product computation for low-power convolutional neural networks," *IEEE J. Solid-State Circuits*, vol. 54, no. 1, pp. 217–230, Jan. 2019.

[16] H. Valavi, P. J. Ramadge, E. Nestler, and N. Verma, "A 64-tile 2.4-Mb in-memory-computing CNN accelerator employing charge-domain compute," *IEEE J. Solid-State Circuits*, vol. 54, no. 6, pp. 1789–1799, Jun. 2019.

[17] H. Jia, H. Valavi, Y. Tang, J. Zhang, and N. Verma, "A programmable heterogeneous microprocessor based on bit-scalable in-memory computing," *IEEE J. Solid-State Circuits*, vol. 55, no. 9, pp. 2609–2621, Sep. 2020.

[18] X. Si *et al.*, "15.5 A 28 nm 64Kb 6T SRAM computing-in-memory macro with 8b MAC operation for AI edge chips," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 246–248.

[19] J. Yue *et al.*, "15.2 A 2.75-to-75.9 TOPS/W computing-in-memory NN processor supporting set-associate block-wise zero skipping and ping-pong CIM with simultaneous computation and weight updating," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2021, pp. 238–240.
[20] J.-W. Su *et al.*, "15.2 A 28 nm 64Kb inference-training two-way transpose multibit 6T SRAM compute-in-memory macro for AI edge chips," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 240–242.
[21] J.-W. Su *et al.*, "16.3 A 28 nm 384kb 6T-SRAM computation-in-memory macro with 8b precision for AI edge chips," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2021, pp. 250–252.
[22] J. Yang *et al.*, "24.4 sandwich-RAM: An energy-efficient in-memory BWN architecture with pulse-width modulation," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2019, pp. 394–396.
[23] Q. Dong *et al.*, "15.3 A 351 TOPS/W and 372.4 GOPS compute-in-memory SRAM macro in 7 nm FinFET CMOS for machine-learning applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 242–244.
[24] M. Horowitz, "1.1 Computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 10–14.
[25] S. Tsukamoto, W. G. Schofield, and T. Endo, "A CMOS 6-b, 400-MSample/s ADC with error correction," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1939–1947, Dec. 1998.
[26] L. Dadda, "Some schemes for parallel multipliers," *Alta Frequenza*, vol. 34, no. 4, pp. 349–356, 1965.
[27] K. A. C. Bickerstaff, M. Schulte, and E. E. Swartzlander, "Reduced area multipliers," in *Proc. Int. Conf. Appl. Specific Array Processors (ASAP)*, 1993 pp. 478–489.
[28] S.-Y. Wu *et al.*, "A highly manufacturable 28 nm CMOS low power platform technology with fully functional 64 mb SRAM using dual/tripe gate oxide process," in *Proc. Symp. VLSI Technol.*, 2009, pp. 210–211.
[29] N. Verma, "Analysis towards minimization of total SRAM energy over active and idle operating modes," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 9, pp. 1695–1703, Aug. 2010.
[30] M. J. M. Pelgrom, H. P. Tuinhout, and M. Vertregt, "Transistor matching in analog CMOS applications," in *IEDM Tech. Dig.*, 1998, pp. 915–918.
[31] B. Murmann. *ADC Performance Survey 1997-2020*. Accessed: Nov. 24, 2021. [Online]. Available: https://web.stanford.edu/~murmann/adcsurvey.html
[32] E. Seevinck, F. J. List, and J. Lohstroh, "Static-noise margin analysis of MOS SRAM cells," *IEEE J. Solid-State Circuits*, vol. JSSC-22, no. 5, pp. 748–754, Oct. 1987.
[33] Y. Tsukamoto *et al.*, "Worst-case analysis to obtain stable read/write DC margin of high density 6T-SRAM-array with local Vth variability," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, Nov. 2005, pp. 398–405.
[34] N. Gierczynski, B. Borot, N. Planes, and H. Brut, "A new combined methodology for write-margin extraction of advanced SRAM," in *Proc. IEEE Int. Conf. Microelectronic Test Struct.*, Mar. 2007, pp. 97–100.
[35] K. Takeda, H. Ikeda, Y. Hagihara, M. Nomura, and H. Kobatake, "Redefinition of write margin for next-generation SRAM and write-margin monitoring circuit," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2006, pp. 2602–2611.
[36] S.-Y. Wu *et al.*, "A 7 nm CMOS platform technology featuring 4th generation FinFET transistors with a 0.027 μm^2 high density 6-T SRAM cell for mobile SoC applications," in *IEDM Tech. Dig.*, Dec. 2016, pp. 2–6.
[37] B. Wicht, T. Nirschl, and D. Schmitt-Landsiedel, "A yield-optimized latch-type SRAM sense amplifier," in *Proc. 29th Eur. Solid-State Circuits Conf. (ESSCIRC)*, 2003, pp. 409–412.

Samuel Spetalnick (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering and computer engineering from Johns Hopkins University, Baltimore, MD, USA, in 2018. He is currently pursuing the Ph.D. degree with the School of Electrical and Computer Engineering, Georgia Institute of Technology. His research interests include memory design for low power and emerging applications.

Arijit Raychowdhury (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from Purdue University in 2007. He joined the Georgia Institute of Technology (Georgia Tech) in January 2013. From 2013 to 2019, he was an Associate Professor and held the ON Semiconductor Junior Professorship at the Department. Prior to joining Georgia Tech, he held research positions at Intel Corporation for six years and at Texas Instruments for one and a half years. He is currently the Steve W. Chaddick Chair and a Professor with the School of Electrical and Computer Engineering, Georgia Tech. He holds more than 27 U.S. and international patents. He has published over 250 articles in journals and refereed conferences. His research interests include low power digital and mixed-signal circuit design, design of power converters, signal-processors, and exploring interactions of circuits with device technologies. He is currently a Distinguished Lecturer of the IEEE Solid State Circuits Society (SSCS) and a Mentor of the IEEE Young Professionals and the IEEE Women in Circuits and Systems. He serves on the Technical Program Committee of Key Circuits and Design Conferences, including ISSCC, VLSI Symposium, DAC, and CICC. He is the winner of several prestigious awards, including the SRC Technical Excellence Award in 2021, the Qualcomm Faculty Award in 2020, the IEEE/ACM Innovator under 40 Award, the NSF CISE Research Initiation Initiative Award (CRII) in 2015, the Intel Labs Technical Contribution Award in 2011, the Dimitris N. Chorafas Award for outstanding doctoral research and best thesis in 2007, and several fellowships. He and his students have won 14 best paper awards over the years.