CapCAM: A Multi-Level Capacitive Content Addressable Memory for High-Accuracy and High-Scalability Search and Compute Applications

Xiaoyang Ma, *Student Member, IEEE*, Hongtao Zhong, *Student Member, IEEE*, Nuo Xiu, *Student Member, IEEE*, Yiming Chen, *Student Member, IEEE*, Guodong Yin, *Student Member, IEEE*, Vijaykrishnan Narayanan, *Fellow, IEEE*, Yongpan Liu, *Senior Member, IEEE*, Kai Ni, *Member, IEEE*, Huazhong Yang, *Fellow, IEEE*, and Xueqing Li, *Senior Member, IEEE*

***Abstract*—As one type of associative memor addressable memory (CAM) has become a critical c several applications, including caches, routers, patte etc. Compared with the conventional CAM that deliver a "matched or not-matched" result, eme level CAM is capable of delivering "the degree of multi-level distance calculation. This feature has b in the applications that need beyond-Boolean matc However, existing multi-level CAM designs are lin bit-cell device discharging current mismatch and to the timing of sensing operations for distance calc inherent constraint makes it difficult to further accuracy and scalability towards higher-accuracy dimension matching. In this work, we propose CapC level Capacitive Content Addressable Memory. It c plemented based on either SRAM or emerging tech the ferroelectric field-effect transistor (FeFET). Cap provide linear and stable voltage drop scaled by the n and need no strict timing for result sensing, which e high-accuracy and high-scalability search. The inhe of CapCAM is the charge-domain computing mech paper will present the basic concept, operating mechanisms, detailed circuit designs and circuit-level simulations of CapCAM. Besides, we apply CapCAM to few-shot learning applications, and compare CapCAM with the current-domain TCAM designs. Results show 99.2% accuracy for a 5-way 5-shot classification task with our proposed CapCAM design, while considering 1-fF capacitors, 20-domain FeFETs, and 256 columns. In contrast, the prior work based on discharging dynamics requires strict timing controls and suffers from accuracy degradation under the same configuration, which demonstrates CapCAM's capability of low-power, accurate, and scalable multi-level CAM computing.**

Manuscript received May 13^{th}, 2022, revised July 16^{th}, 2022, accepted August 7^{th}, 2022;

This work is supported in part by the National Key R&D Program of China (No. 2019YFA0706100), in part by NSFC (U21B2030, 61874066, 61934005), in part by Tsinghua University – Daimler Greater China Ltd. Joint Institue for Sustainable Mobility, and in part by NSF (2008365, 2132918). Corresponding author: Xueqing Li.

X. Ma, H. Zhong, N. Xiu, Y. Chen, G. Yin, Y. Liu, H. Yang, and X. Li are with BNRist/ICFC, The Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: {maxy18, zht21, xiun19, cym21, ygd20}@mails.tsinghua.edu.cn; {ypliu, yanghz, xueqingli}@tsinghua.edu.cn).

V. Narayanan is with the Department of Computer Science and Engineering, Penn State University, University Park, PA 16802, USA (e-mail: vijay@cse.psu.edu).

K. Ni is with the Electrical and Microelectronic Engineering Department, Rochester Institute of Technology, Rochester, NY 14623, USA (e-mail: kai.ni@rit.edu).

Digital Object Identifier XXXX/XXXX.XXXX.

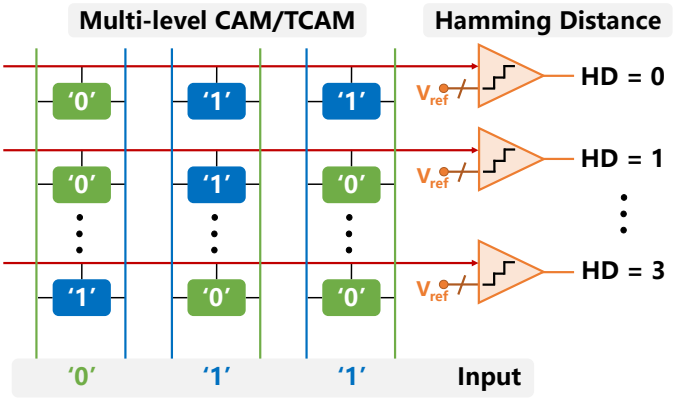


Fig. 1. The basic concept of ML-CAM and ML-TCAM.

***Index Terms*—Content-Addressable Memory, Multiple-Level CAM, Ferroelectric FET, Low-Power Design, Pattern Matching.**

I. Introduction

VARIOUS data-intensive applications require frequent parallel data search to figure out whether the data in a memory array match the input data stream [1]–[10]. These applications include caches, routers, networking, and other mapping-aware applications. Recent reports even show that it is also promising for emerging applications like deep learning [7], [8], DNA sequence alignment [9], [10], etc. In these scenarios, content-addressable memory (CAM) has been a critical component that could support intrinsic *in-situ* data search in parallel for all stored memory vector candidates in the rows of a memory array without the need of pouring out data to the external for matching computing. The result of searching is either *"match"* or *"mismatch"* for each vector comparison, and would be encoded by following peripherals. In addition to CAM, ternary CAM (TCAM) is often utilized to support the extra *"don't care"* or *'X'* search rule claimed by the *'X'* bits, which could be applied to bypass some certain bits. For each bit location, if either the input bit or the store bit is an *'X'* bit, the matching result of this bit location is always *"match"*, so the vector matching result is determined by other bit locations.

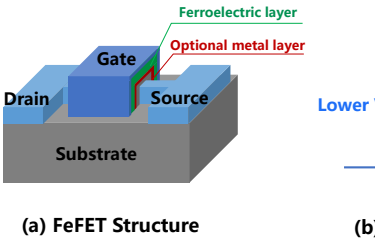


Fig. 2. FeFET device and typical I_D-V_G

There have been CAM and TC
CMOS SRAM and nonvolatile me
MTJ, RRAM, FeFET) or other e
These designs show the difference
scalability, reliability, etc. Moreov
tional CAM and TCAM designs
results between the input data s
patterns. In other words, the search result in each row is eit
"fully matched" or *"not fully matched"*. Recently, the rep
in [7] has shown a new CAM or TCAM category, defir
as multi-level CAM (ML-CAM) or ML-TCAM, as illustra
in Fig. 1. ML-CAM search output could deliver the "match
degree" feature for each row vector according to the hamming distance, and offer practical and efficient support to feature classifications, and further, few-shot learning applications [7].

However, the existing ML-CAMs or ML-TCAMs still face many challenges to calculate "match degree". The design in [7] track the settling slope of the matching results. However, this is not scalable, reliable, or accurate due to the high peripheral sensing overhead and intrinsic FeFET device variations. In this paper, we propose a new kind of ML-CAM and ML-TCAM designs based on capacitors, and define them as CapCAM: a multi-level **Cap**acitive **C**ontent **A**ddressable **M**emory. CapCAM could provide linear and stable voltage output scaled by the match degree, so the peripheral sensing overhead is reduced significantly. Besides, CapCAM handles the memory device variations by using capacitors with much less variations, especially cycle-to-cycle variations, thanks to the much more mature capacitor technology. This enhances the scalability and reliability of ML-CAMs and ML-TCAMs.

Itemized contributions of this work include:

- We propose a new operating theory of capacitive ML-CAM and ML-TCAM, i.e. the proposed CapCAM, for enhanced power efficiency, scalability, and reliability;
- We propose four different CapCAM designs, FeFET-based ML-CAM/ML-TCAM, and SRAM-based ML-CAM/ML-TCAMs;
- At the circuit level, we evaluate the proposed four ML-CAM and ML-TCAM designs in terms of functionality, energy consumption, latency, area, and also reliability;
- At the application level, we evaluate the CapCAM in few-shot learning applications with different mapping methods and compare with existing ML-TCAM designs.

In the rest of this paper, Section II reviews FeFET device basics, the existing CAM/TCAM designs and ML-CAM/ML-TCAM background. Section III shows the details of the proposed CapCAM based on CMOS SRAM and FeFET devices.

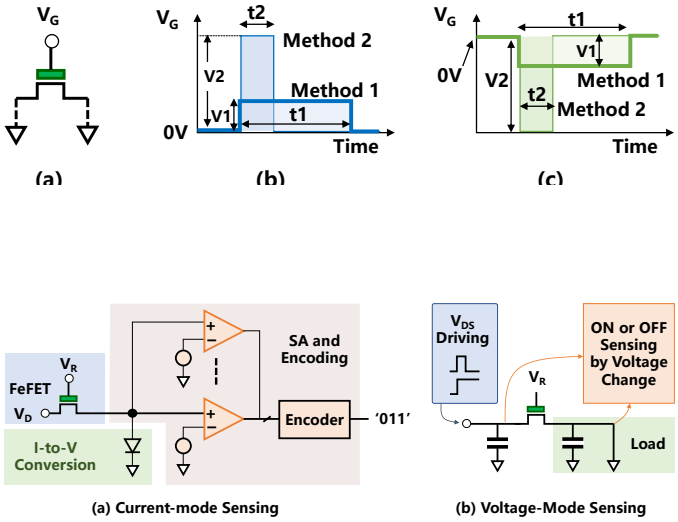


Fig. 4. Reading an FeFET in current or voltage modes.

Section IV evaluates the proposed designs at the circuit level. Section V evaluates the proposed designs at the application level. Section VI concludes this work.

II. Background

A. The FeFET Device Basics

FeFET is essentially a MOSFET (either a planar FET or a FinFET) with a ferroelectric (FE) material layer sandwiched in the gate stack, as illustrated in Fig. 2(a) [28]–[39]. While the FeFET devices have been proposed for many years [28], the recent fast progress has been achieved with the success of the material and structure improvement,which makes the FeFET devices embrace smaller dimensions, lower operating voltage, higher endurance, etc. Recent works have shown that the FeFET could be successfully fabricated on 20-nm-thick SOI technology [37], and exhibit a high ON/OFF ratio beyond 10^6, implying the capability of large memory arrays [32]. Another report has even shown an FeFET with 10 ns programming time at 1.8 V operating voltage and endurance of 10^{12} cycles [38]. Besides, the FeFET physical mechanism has been widely explored by many modeling works, which makes FeFETs more interpretable and further provides practical EDA tools [40]–[42]. In the future, more FeFET research is expected to be carried out for a smaller size, a lower operating voltage, higher endurance, less variations, better understanding of the physical mechanism, and more accurate modeling.

The key to understanding the difference between a MOSFET and an FeFET is the FE layer polarization behavior. FeFETs utilize the polarization direction in the FE layer to store the memory state. When applying a positive gate voltage pulse to the FE layer, an n-type FeFET accumulates electrons in the nucleation-dominated channel and the domains in the FE layer switch to the positive state probabilistically, which reduces the device threshold voltage (V_{th}). Similarly, when applying a negative voltage, the FeFET V_{th} increases [40], [41]. Fig. 2(b) shows the FeFET I_D-V_G curve. In detail, writing an FeFET is essentially a process of modulating the amplitude and the pulse width of the gate voltage across the

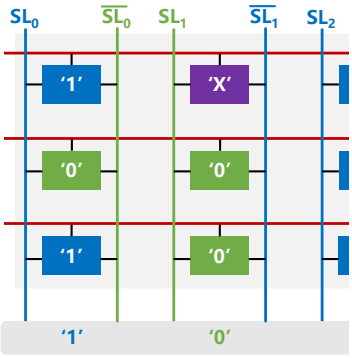


Fig. 5. General (T)CAM array structure

FE layer. Higher gate voltage am
both lead to more switched dom
a certain amount of switched
width decreases with higher ap
as illustrated in Fig. 3. Note that
level states through fine polariz

Reading an FeFET is to det
drain-source current I_{DS}. It co
mode and voltage mode. The
through a current-voltage conver
fiers (SA), as shown in Fig. 4(a)
out by pre-charging the sense
change. If the FeFET is in the high V_{th} state, the voltage change is negligible. Otherwise, the voltage change could be sufficient for SA to detect, as shown in Fig. 4(b).

B. CAM and TCAM: Operations and Existing Designs

The functionality of CAM and TCAM is shown in Fig. 5. A CAM or TCAM is usually organized as a 2D array. The stored vector data are placed row-wise (one row for one vector), and the input data stream is sent into the array through vertical searchlines SL and $\overline{SL}$. For both CAM and TCAM, during the search operation, the matchline ML is firstly pre-charged and then left floating. After that, each cell compares the bitline input with the stored data in a differential XNOR style. The ML will be discharged if any mismatch occurs. Therefore, the settling-down behavior of ML in each row indicates the search comparison result.

There have been many existing CAM and TCAM designs using CMOS technology and NVM devices. The widely used CAM and TCAM designs are 10T CMOS CAM and 16T CMOS TCAM, as illustrated in Fig. 6(a) and Fig. 6(b) respectively [11]. The SRAM-based designs are mature, stable, and fast, but occupy more area (due to more transistors) and more leakage power (due to the SRAM leakage currents). Many new SRAM-based designs have been proposed to improve area- and energy-efficiency. For example, [12] proposes a dense and energy-efficient reconfigurable SRAM/CAM/TCAM with push-rule SRAM cells at the cost of stability. Other designs that enhance SRAM-based CAM/TCAM flexibility, such as two-direction search and computing-in-memory reconfigurablity, have also been explored [13], [14]. Moreover, high-

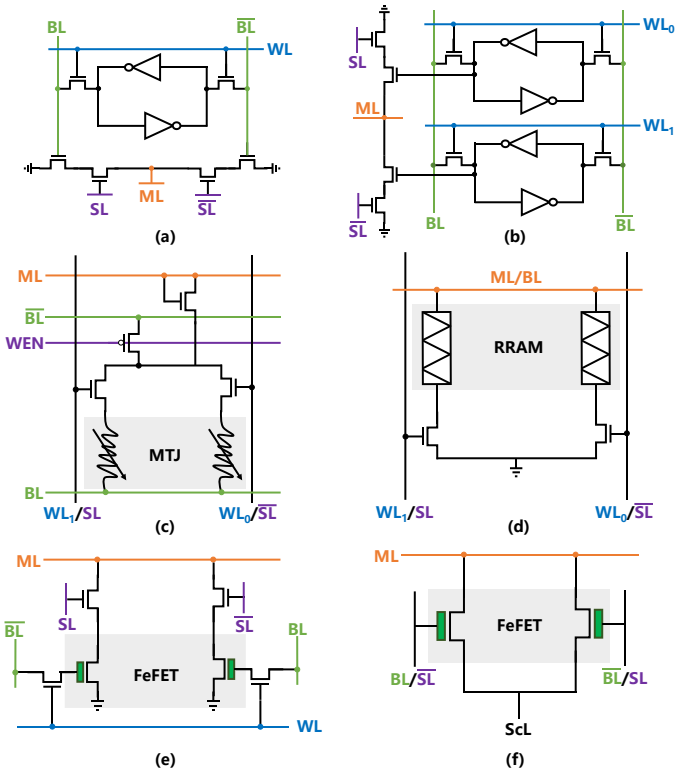


Fig. 6. Existing CAM and TCAM designs. (a) 10T CMOS CAM [11]; (b) 16T CMOS TCAM [11]; (c) 4T-2MTJ TCAM [18]; (d) 2T2R TCAM [21];

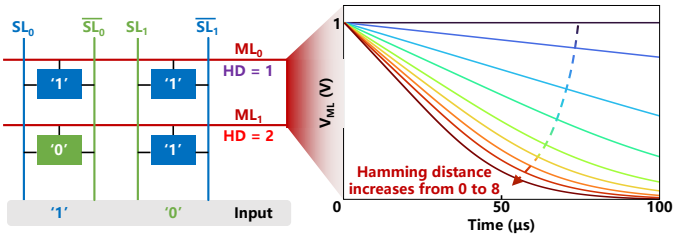


Fig. 7. Current-mode multi-level TCAM: an existing work [7].

performance FinFET technologies also provide SRAM-based CAM/TCAMs with a rich design space to explore [15], [16].

C. The ML-CAM and ML-TCAM Background

Recently, non-volatile TCAMs are receiving more attention. Some NVM-based CAM/TCAM designs include the 4-transistor-2-MTJ (4T-2MTJ) TCAM in Fig. 6(c) [18], the 2-transistor-2-RRAM (2T2R) TCAM in Fig. 6(d) [21], and the 4-transistor-2-FeFET (4T2F) TCAM in Fig. 6(e) [23], the 2FeFET TCAM in Fig. 6(f) [24]. Compared with the CMOS-based designs, the NVM-based designs are not only much more compact but also eliminate the standby leakage power consumption. Among them, the CAM/TCAM designs based on MTJ and RRAM have higher write energy and longer write latency in write operations due to the current-driven write mechanism. Besides, the low ON-OFF ratio of MTJ and RRAM also leads to high sensing complexity and costs. Last but not the least, MTJ and RRAM suffer from the device

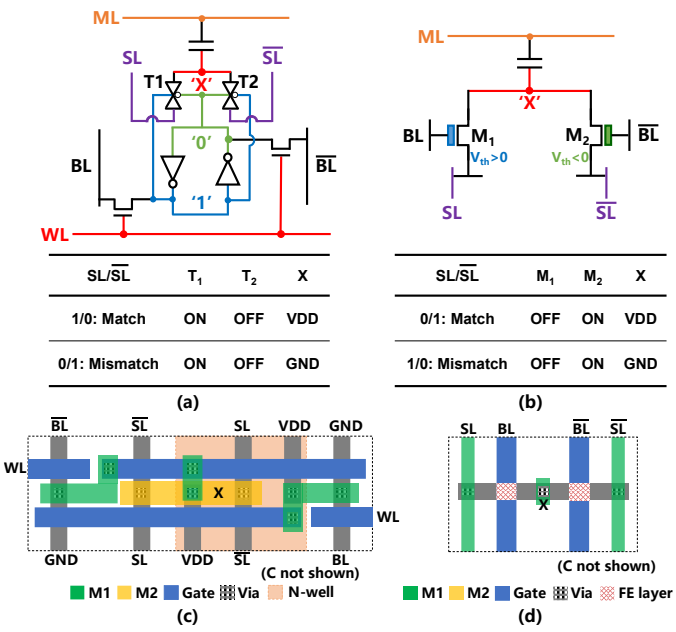


Fig. 8. The proposed ML-CAM schematics based on (a) SRAM and (b) FeFET, and the ML-CAM layouts based on (c) SRAM and (d) FeFET.

variations of ON-state and OFF-state currents, which limits th scalability and reliability significantly. By contrast, FeFET exhibit a high ON/OFF ratio (e.g. 10^6 [32]), and consum no DC power during write and search operations. The FeFET based design in Fig. 6(f) further shows FeFET as both memor and comparison device to achieve very high density.

ML-CAM and ML-TCAM could not only distinguis whether the input vector and the stored data match exactly but also output the number of the matched cells, or rather th match degree. An existing design in [7], as illustrated in Fig. 7 achieves the functionality in the current mode by sensin the discharging dynamics: ML is pre-charged firstly, and th output sense amplifier on the ML senses the decreasing slop of ML voltage after the input pattern is applied to $SL/\overline{SL}$ With more mismatched bits, ML is discharged more quickly This time-domain method needs careful and strict timing and is highly vulnerable to the device variations, especially when many comparisons mismatch, which results in weak scalability towards a large array. This challenge is overcome in this work by adopting the proposed capacitive search method, as to be further discussed.

D. ML-CAM and ML-TCAM Applications

Recent research has demonstrated the potential of utilizing high-performance CAM/TCAMs in meta-learning (learning to learn) applications. Meta-learning aims at learning new concepts with only a few samples, e.g., few-shot classification as a supervised version of meta-learning. CAM/TCAMs are capable of efficiently searching for an input query vector and updating their entries with new samples learned, where ML-CAM and ML-TCAMs could further implement custom distance metrics. Thus, these designs are useful in both metric-based and model-based meta-learning methods [7], [43]–[45].

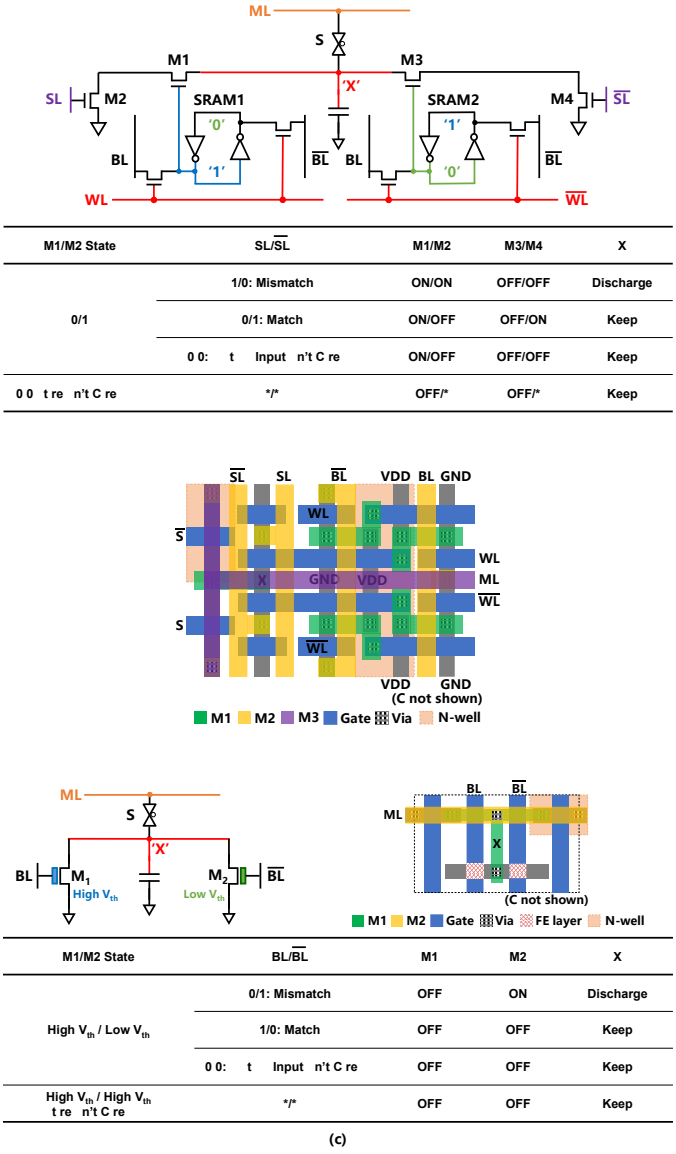


Fig. 9. (a) and (b): Schematic and layout of the proposed SRAM-based ML-TCAM; (c) Schematic and layout of the proposed FeFET-based ML-TCAM.

Metric learning algorithms that learn a distance metric over objects have been proved successful in few-shot learning applications [46], [47]. Such algorithms typically involve a nearest neighbor search (NNS) step to carry out the final decision, which can be solved by CAM/TCAMs with high performance and energy-efficiency. Conventional TCAMs could solve NNS by range encoding schemes, where a tradeoff exists between hot update flexibility and code compactness [48]–[51]. Some of these encoding schemes require exponential row expansion, and are impractical for real applications [48]–[50], while others are limited on short ranges despite the ability to solve multiple NNS using one TCAM search operation without row expansion [51]. A one-shot learning accelerator with aggressive quantization and range encoding has demonstrated high accuracy and energy-efficiency of this method [43], [44].

ML-CAM and ML-TCAM designs are more useful in another method for solving NNS, which is to perform an

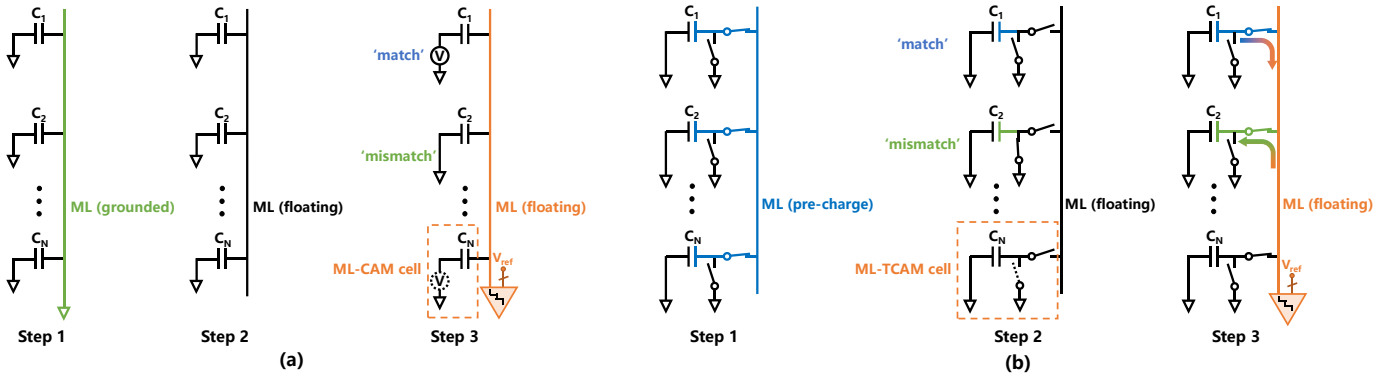


Fig. 10. Two CapCAM operation theories: (a) Capacitive coupling for ML-CAM; (b) Charge re-distribution for ML-TCAM.

approximate nearest neighbor search (ANNS) with a distance metric easier for query, e.g., Hamming distance. Locality sensitive hashing (LSH) is commonly used in ANNS to reduce the dimension and hash the problem into a metric space easier for computation [52], and the original NNS problem could then be easily solved with high probability. By extending the LSH function into a ternary locality sensitive hashing (TLSH) function, TCAMs are leveraged to perform NNS with higher performance [53]. Existing ML-TCAM design in [7] outputs match degree (equivalent to Hamming distance) and utilizes LSH to transfer real-valued feature vector into binary signatures. The work in [45] implements a customized distance metric that achieves high accuracy and has been experimentally demonstrated with a 2-bit FeFET multi-bit CAM array. Moreover, these designs could be used in memory-augmented neural networks (MANN) [54] with their ability to update the entries, which is useful in meta-learning when new samples are seen and learned. The proposed CapCAM designs could improve the scalability of these existing designs, mitigating the impacts of device variation and timing effects.

III. Proposed CapCAM

A. CapCAM Operation Theory

The proposed CapCAM supports two basic capacitive multi-level content searching schemes: capacitive coupling for ML-CAM and charge re-distribution for ML-TCAM. The ML-CAM and ML-TCAM schematics and layouts are demonstrated in Fig. 8 and Fig. 9 respectively, and will be further elaborated in Section III-B and III-C. The two operating theories are illustrated in Fig. 10.

Fig. 10(a) shows the capacitive coupling operation theory [55]. The capacitors in all cells within a row short their top plates through ML. Firstly, ML is discharged to GND, and each capacitor bottom plate is also initially grounded. After that, ML is left floating, and each capacitor bottom plate receives a voltage input determined by the XNOR matching result of each cell. A *"match"* result charges the bottom plate to VDD, while a *"mismatch"* result keeps the bottom plate at GND. Due to the capacitive coupling, the shorted top plate will converge to the average voltage of the inputs weighted by the capacitor size.

The charge re-distribution operation theory is illustrated in Fig. 10(b). The capacitor in each cell is charged to VDD through pre-charging ML. Then, the capacitor may be discharged to GND or stay unchanged subsequently by the pull-down path determined by the matching result of each cell. A *"match"* result keeps the capacitor at VDD, while a *"mismatch"* result discharges the capacitor to GND. At last, these capacitors (some may have been discharged) are shorted, resulting in the charge re-distribution towards a weighted voltage at the top plate.

B. Proposed ML-CAM in SRAM and FeFET

The ML-CAM could be achieved by both SRAM and FeFET. The SRAM-based ML-CAM is shown in Fig. 8(a). Two cross-coupled inverters in the SRAM cell generate '0' and '1' at two ends. If the input matches the stored SRAM data, the transmission gate (T_1 or T_2) on the same side of '1' will be turned on, and charge the node X to VDD. Otherwise, the transmission gate on the same side of '0' will be turned on and keep X at GND. Therefore, an XNOR operation is implemented by charging the capacitor bottom plate through two transmission gates controlled by the stored bit and the external search pattern bits.

The FeFET-based ML-CAM is shown in Fig. 8(b), and the two n-type FeFETs (M_1 and M_2) store complementary states (one positive V_{th} and one negative V_{th}). For the FeFET with negative V_{th} (representing the stored datum '1'), the device is turned-on, so the corresponding input voltage on SL or $\overline{SL}$ on the same side could arrive at X, and charge X to VDD with input '1' or keep X at GND with input '0'; for the FeFET with positive V_{th} (representing '0'), the device is turned-off, so the input on the same side would not affect the voltage at X. Therefore, the XNOR operation is implemented between the FeFET source inputs and the FeFET stored ON/OFF states.

For both SRAM-based and FeFET-based ML-CAM, the step-by-step operation is shown as below:

- Step 1: ML, SL and $\overline{SL}$ are all grounded, which leads to the capacitor bottom plate voltage reset to GND;
- Step 2: With ML floating, the inputs driven by SL and $\overline{SL}$ set the capacitor bottom plate voltage to either VDD or GND, depending on the matching XNOR results;
- Step 3: sense the ML voltage and compare it with pre-defined reference voltages to digitize the match degree.

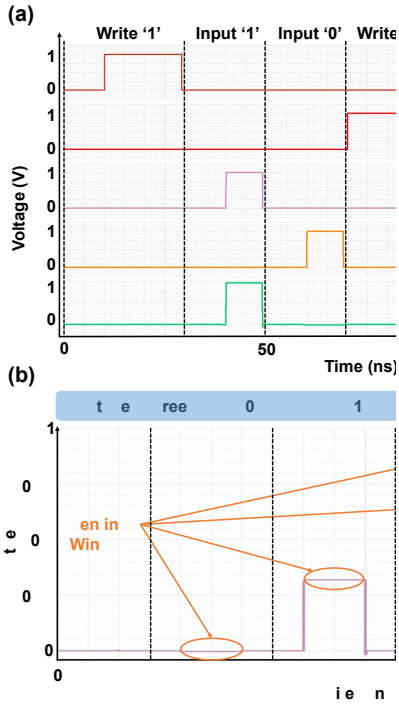


Fig. 11. A transient waveform snapshot of t CAM: (a) Single-cell simulation; (b) 3-colu degree" scenarios.

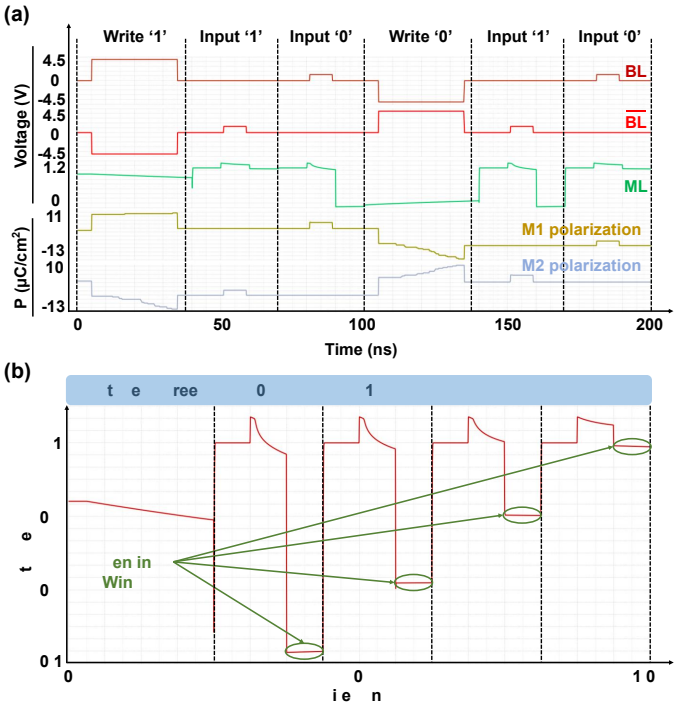


Fig. 12. A transient waveform snapshot of the proposed FeFET-based ML-TCAM: (a) Single-cell simulation; (b) 3-column simulation with 4 "match degree" scenarios.

Taking the SRAM-based ML-CAM as an example, Fig. 11 shows a transient waveform snapshot, where the "match degree" of 0, 1/3, 2/3, and 1 in a 3-column array is shown.

C. Proposed ML-TCAM in SRAM and FeFET

The proposed ML-CAM does not support storing the *'X'* state. In the ML-CAM structures, one of the two voltage inputs from SL or $\overline{SL}$ is sent to X, either through one transmission gate turned on in the SRAM-based structure, or one FeFET in the FeFET-based structure. To avoid a short circuit between SL and $\overline{SL}$, there must be at least one path open between X and the input (SL or $\overline{SL}$), so storing an *'X'* state could not be supported in the capacitive coupling scheme for ML-CAM. Thus, we propose ML-TCAM to support both stored *'X'* state and input *'X'* state that bypass certain bits at the corresponding bit location.

The proposed SRAM-based ML-TCAM is shown in Fig. 9(a). Two SRAM cells storing the data Q_1 and Q_2 are connected together through BL and $\overline{BL}$, and control two NMOS pass transistors (M_1 and M_3), respectively. The input SL and $\overline{SL}$ are connected to two search transistors M_2 and M_4, respectively. M_1/M_2 and M_3/M_4 provide two possible pull-down paths between X and ground. When a match occurs, both pull-down paths are OFF because there are at least one OFF-state transistors on each path. Otherwise, if a mismatch occurs, there is one ON pull-down path with two ON-state transistors, as illustrated in Fig. 9(a). To pre-charge the capacitor, one CMOS transmission gate (S) is adopted.

The step-by-step operation of the SRAM-based ML-TCAM is shown as follows:

- Step 1: S is turned on to pre-charge the capacitor through ML, while SL and $\overline{SL}$ are set to GND;
- Step 2: S is turned off to floating ML, and then SL and $\overline{SL}$ are driven by the corresponding input (VDD/GND for '1', GND/VDD for '0' and GND/GND for *'X'*). The capacitor may be discharged if Q_1/SL or Q_2/$\overline{SL}$ are both high; Otherwise, the capacitor is not discharged.
- Step 3: SL and $\overline{SL}$ are both grounded and then S is turned on for charge re-distribution; the settled ML voltage indicates the match degree.

The proposed FeFET-based ML-TCAM is shown in Fig. 9(b). Compared with the FeFET-based ML-CAM, one CMOS transmission gate (S) for pre-charging is added between ML and the capacitor, and two FeFET sources are grounded directly. It is noted that, besides the complementary bits, the two FeFETs could be configured to exhibit both high V_{th} or one high positive V_{th} plus one low positive V_{th} to represent the *'X'* state. The FeFET with low positive V_{th} is OFF at zero gate biasing and could be turned on at a proper $V_R > 0$, while the FeFET with high positive V_{th} is OFF at both zero gate biasing and the preset V_R (> 0).

The step-by-step operation of the FeFET-based ML-TCAM is slightly different from that of the SRAM-based ML-TCAM:

- Step 1: S is turned on, and BL and $\overline{BL}$ are set to GND to prevent an ON pull-down path between ML and ground;
- Step 2: S is turned off, and BL and $\overline{BL}$ are driven by the corresponding input (V_R/GND for '1', GND/V_R for '0' and GND/GND for *'X'*). The capacitor may be discharged if the FeFET with low positive V_{th} is turned on by the V_R input. Otherwise, the capacitor is not discharged. The operation scheme is summarized in Fig. 9(b);

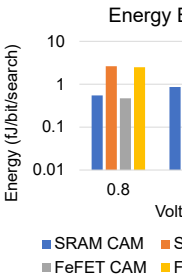


Fig. 13. Energy and
based ML-CAM/ML

- Step 3: San
operation s

Taking the Fe
shows a transien
of 0, 1/3, 2/3, a

Though the ch
TCAM supports
it has brought
and discharging
latency and ener
the cell capacitc
(VDD-V_{ML} or V
compared with
CAM could also
one SRAM cell
ML-TCAM. Th
CAM or ML-TC

D. Sensing Meth

In the existing
sense amplifier (
ing control to se
discussed above, this is not scalable due to device variations. Thanks to the new CapCAM designs that do not rely on the ML discharge dynamics but a settled static voltage, the sensing schemes could be significantly simplified.

While using a shared ADC is practical, typical multi-level CAM/TCAM applications do not need to output match degrees of all the rows, but to select the best matched row, similar to conventional CAM/TCAM. We exploit a sensing method that compares the ML voltage with a single-slope decreasing reference V_{REF}. The row with the highest match degree is found first when the first '1' appears. Moreover, in some applications, V_{REF} could be further optimized to reduce sensing latency. For example, V_{REF} could start with an initial voltage that maximizes the probability of finding the largest match degree at first guess, and then increases or decreases depending on the number of rows larger than V_{REF}. A binary search method could also be used to improve the search efficiency. In case multiple best-matched rows with same match degree emerge, an arbiter or priority encoder might be needed based on specific application requirements. We would discuss this sensing method further with application examples in Section V, showing that over 80% samples could be sensed at first guess of V_{REF} if application knowledge is considered, which reduces much sensing latency and power.

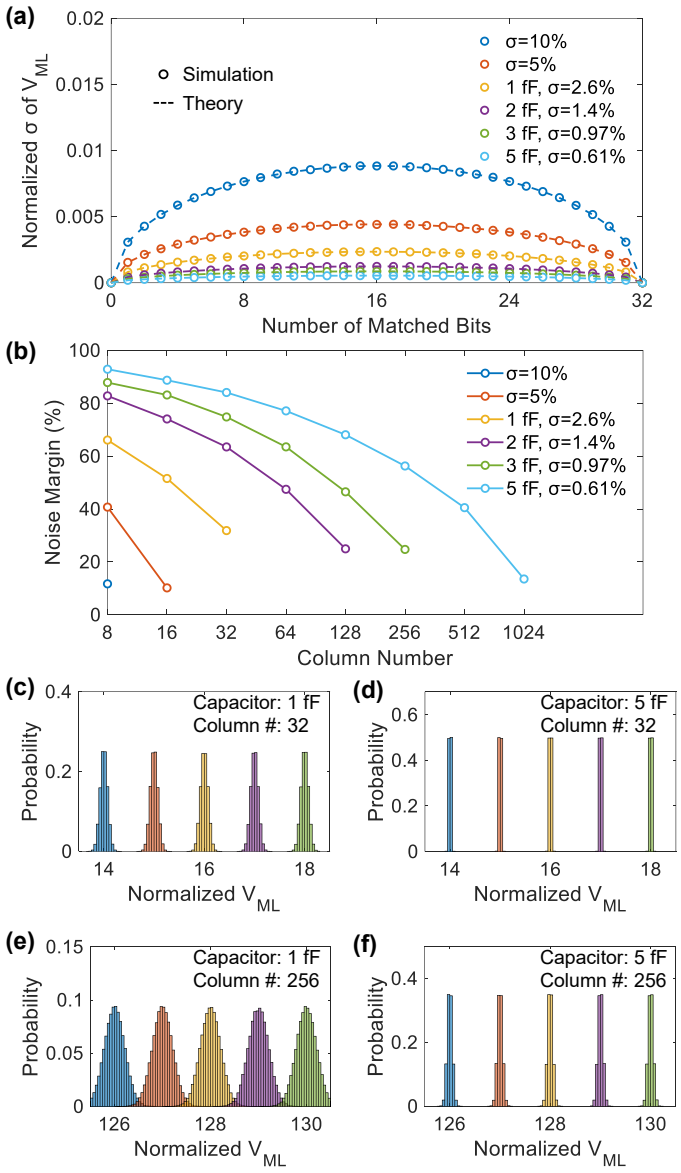


Fig. 14. Noise margin and variation Monte Carlo simulation results of CapCAM (100,000 samples) under different column widths and cell capacitances: (a) Standard deviation of ML voltage normalized over VDD; (b) Noise margin of CapCAM; (c)-(f): Simulated distribution of CapCAM ML voltage.

Voltage-mode winner-take-all (WTA) circuits [57]–[63] could also be used for sensing CapCAM. With MLs connected to WTA input ports, typical WTA circuits involve inner nodes each row to indicate by voltage whether the row has the largest V_{ML}, i.e. "winner". The non-winners have the inner node voltage at VDD or GND, while the winner has its voltage (might or might not depend on the match degree) different from all other rows. These inner nodes could be then connected to a conventional CAM/TCAM sense amplifier to select the best matched row. With high-speed, low-power WTA circuits, the overhead of this sensing method is relatively small. For example, with m rows, [59] in 0.18 µm CMOS technology could achieve latency lower than 10 ns and power $\sim$ 10 µW/row

by using $\sim m^2$ transistors. Meanwhile, [60], [61] in 0.35 or 0.5 µm CMOS technology involves $3m + 3$ transistors, and could operate with MHz frequency and $\sim$ 20 µW/row power. Moreover, there are also WTA circuit designs that support multiple most matched outputs [62]. However, this analog sensing method requires good transistor matching in WTA circuits, and circuit evaluation in advanced technologies nodes are desired in the future. WTA circuits also suffer from accuracy and latency degradation when the confidence of the application result is low, and multiple most matched rows have similar but relatively low match degrees.

The sensing complexity increases with array size in that a large number of rows would introduce significant sensing latency overhead and device matching requirements. Existing WTA design with 4k rows and 1 µs search latency has been proposed [63], exceeding typical row number requirements of few-shot learning applications (less than 100). The number of columns mainly impact search power and the desired resolution of sense amplifiers. Search power is proportional to the number of columns for our charge-domain CAM/TCAM designs, while all the mentioned WTA designs could distinguish $\sim$ mV voltage differences, which is sufficient for a reasonable column width of less than 1000.

IV. Circuit Evaluation

A. Benchmark Settings

The MOSFETs and capacitors in all the designs are modeled in a commercial 65 nm CMOS process. MIM capacitors are used for cell capacitors. The FeFET-based ML-CAM and ML-TCAM designs are simulated with the FeFET model from [41] that captures FeFET variation, with $W/L = 1$, 200 ferroelectric domains ($\sim$ 100nm $\times$ 100nm), 8nm ferroelectric layer thickness and 19 ns τ_0. The FeFET model has been calibrated with ferroelectric device samples from the foundry. The simulation is carried out for an array with 128 rows, and the bitline parasitic capacitance is modeled as 12.8 fF. All the four CapCAM cell structures adopt the same 2.0 fF capacitor unless mentioned otherwise.

B. Energy and Latency Evaluation

Fig. 13 shows the latency and energy comparison between different CapCAMs. We simulated the array at half-matched case with a different supply voltage VDD. In the energy benchmark, the multi-level CapCAMs are able to achieve parallel search with low energy consumption (<10 fJ). It can be observed that the energy is generally proportional to VDD^2. This is because CapCAMs operate at the charge-domain mode, and that the search operation only consumes dynamic power during charging and discharging the capacitors. ML-CAM needs no pre-charge operation and operates with lower energy than ML-TCAM.

In the latency benchmark, comparison between different designs is also illustrated in Fig. 13. The CapCAMs can achieve fast search operation (less than 1 ns). The ML-CAM CapCAMs with no pre-charge operation operate at a higher search speed than ML-TCAM. In addition, since the FeFET model operates at a low-voltage mode, the FeFET-based CapCAM can potentially achieve faster search operation, also can be further improved with an enhanced fabrication process. Comparison with existing TCAM designs is shown in Table I.

C. Multi-level Output Analysis

In the proposed CapCAM designs, the accuracy of match degree is dominated by the capacitor matching instead of FeFET or SRAM device variations. The match degree of a row is weighted by the capacitors if FeFET or SRAM variations are neglected, and ML voltage could be written as

$$V_{ML} = \frac{\sum_{i=1}^{n} C_i V_{Xi}}{\sum_{i=1}^{n} C_i} = \frac{\text{VDD} \sum_{V_{Xi}=\text{VDD}} C_i}{\sum_{V_{Xi}=\text{VDD}} C_i + \sum_{V_{Xi}=\text{GND}} C_i} \tag{1}$$

where n is the total number of columns, C_i is the capacitance of the i-th cell, and V_{Xi} is the voltage of the 'X' node in Fig. 8 and Fig. 9, theoretically VDD when matched and GND when mismatched. We denote $C_1 = \sum_{V_{Xi}=\text{VDD}} C_i$ and $C_2 = \sum_{V_{Xi}=\text{GND}} C_i$. If we assume that each C_i follows an independent and identically distributed normal distribution $\mathcal{N}(\mu, \sigma^2)$, while the variation of the total capacitance $\sum_{i=1}^{n} C_i = C_1 + C_2$ is relatively small compared with itself, the variance of V_{ML} could be estimated as

$$\begin{aligned} &\text{Var}(V_{ML}) = \text{Var}\left(\frac{\text{VDD} \cdot C_1}{C_1 + C_2}\right) \\ &\approx \text{Var}\left(\frac{\text{VDD} \cdot C_1}{C_1 + C_2} \middle| C_1 + C_2 = n\mu\right) = \frac{k(1-k)\sigma^2}{n\mu^2}\text{VDD}^2 \end{aligned} \tag{2}$$

where k is the match degree (the number of cells such that $V_{Xi} = \text{VDD}$ normalized by the total number of cells in a row). Hence, the variance follows a parabolic trend, where the largest variation $\text{Var}(V_{ML}) \approx \frac{\sigma^2}{4n\mu^2}\text{VDD}^2$ exists when half of the columns are matched, and the variance is zero when fully matched or mismatched. The largest supported number of columns could be then estimated by the desired accuracy and capacitance variance. For example, if unit capacitors of 2 fF with 1.4% σ are used, and $\pm 3\sigma$ of the output is required to be inside the ML least significant bit (LSB) voltage (0.3% bit error rate), we could solve from $\sqrt{\frac{\sigma^2}{4n\mu^2}\text{VDD}^2} \leq \frac{1}{6}\frac{\text{VDD}}{n}$ and get $n \leq 566$.

We evaluate the multi-level output accuracy in FeFET-based ML-TCAM with different cell capacitor sizes and column numbers as an example of four CapCAM designs. Both capacitance mismatch and FeFET device variations are taken into consideration and are evaluated by Monte Carlo simulations for 100,000 runs. In addition to Monte Carlo simulations with MIM capacitors from 1 to 5 fF, capacitors with standard deviation of 5% and 10% are also used for simulations to demonstrate noise margin and variation of CapCAMs under large capacitance mismatch. The FeFET is modeled with the 200-domain model mentioned above, and shows a V_{th} sigma value of 61.5 mV. With better matching accuracy of capacitors over FeFETs, and the high on/off ratio of FeFETs, the proposed designs could achieve an excellent noise margin. Fig. 14(a) shows the simulated standard deviation of ML voltage, which matches well with

TABLE I
COMPARISON OF DIFFERENT TCAM DESIGNS

	Latency	Energy	ML	Scalability	Density
FeFET-ML [7][1]	355 ps	0.40 fJ/b	Yes	Limited	Good
FeFET-based ML-TCAM[2]	**269 ps**	**3.89 fJ/b**	**Yes**	**Good**	**Good**
SRAM-TCAM [64]	582 ps	1.0 fJ/b	No	Good	Limited

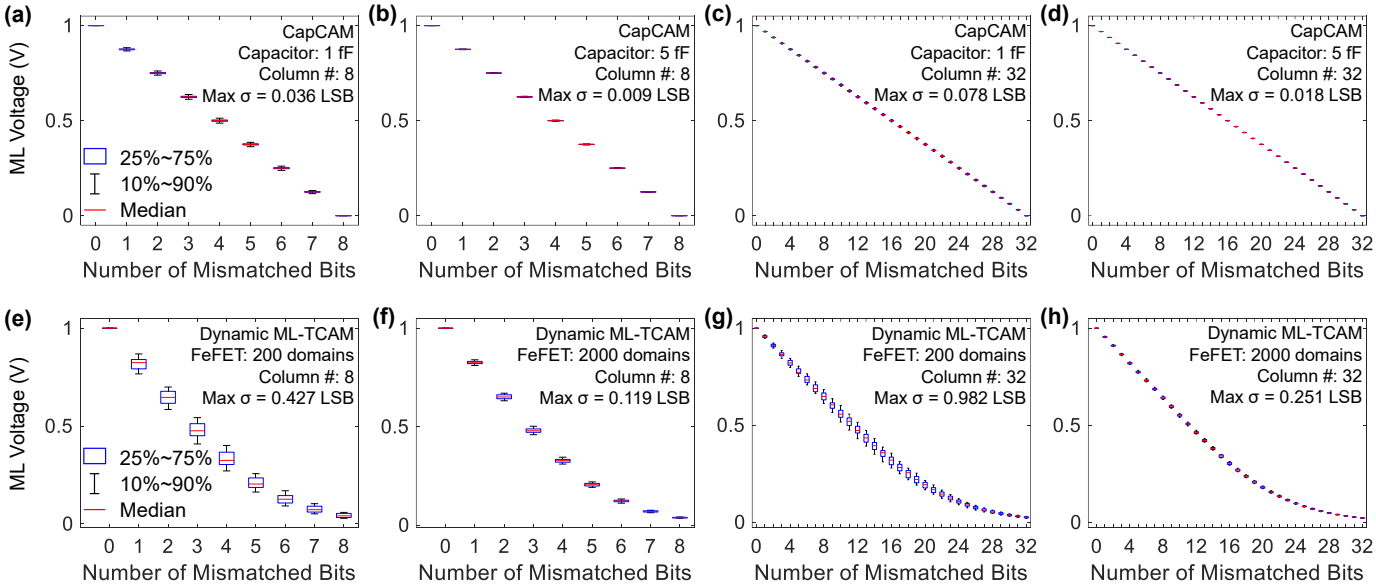


Fig. 15. Multi-level output linearity and noise margin Monte Carlo simulation of: (a)-(d) proposed FeFET-based ML-TCAM with different cell capacitance and column number; (e)-(h) existing dynamic ML-TCAM with different FeFET variation and column number, where 2000-domain FeFETs ($\sim$ 300nm $\times$ 300nm) have V_{th} sigma value of 16.7 mV, and 200-domain FeFETs ($\sim$ 100nm $\times$ 100nm) have V_{th} sigma value of 61.5 mV.

the theoretical analysis. Fig. 14(b) shows the noise margin of CapCAMs with different cell capacitance and column number. For all configurations in Fig. 14(b), all multi-level states could be reliably distinguished. Fig. 14(c)-(f) shows simulated distribution examples, each containing 5 consecutive match degree near half-matched, where the variation is the largest. In Fig. 14(e), the overlapping distributions show that, if the exact hamming distance result is desired, the proposed design with 256 columns, cell capacitance of 1 fF, and an ideal ADC would suffer from a 1.9% bit error rate when match degree is near half-matched. However, in few-shot learning applications, we only need to select a most matched row rather than computing every Hamming distance. We will show in Section V that the overlap would have almost no accuracy degradation compared with software implementation.

Moreover, the proposed designs could achieve better linearity and do not require strict timing controls since the settled ML voltage is static, rather than the dynamic changing case in [7]. In Fig. 15, comparison with the existing ML-TCAM design in [7] is shown under different FeFET device variations. The ML-TCAM design is evaluated with two different FeFET sizes: 2000-domain FeFETs ($\sim$ 300nm $\times$ 300nm) with simulated V_{th} sigma value of 16.7 mV, and 200-domain FeFET ($\sim$ 100nm $\times$ 100nm) with simulated V_{th} sigma value of 61.5 mV. As shown by the results, CapCAM has better scalability towards a larger number of columns and smaller devices. We could reliably distinguish the match degree states, while the existing dynamic ML-TCAM of a large-size array using deeply-scaled transistors faces challenge of distinguishing neighbor states. For example, a 256-column dynamic ML-TCAM array with 200-domain FeFETs has a maximum output voltage σ of 0.982$\times$ its LSB voltage (58% bit error rate), while the proposed FeFET-based CapCAM with same configuration and 1-fF unit capacitors has a σ only 0.078$\times$ its LSB voltage (1.5×10^{-10} bit error rate).

D. Area Overhead

Fig. 8(c), (d) and Fig. 9(b), (c) show the layouts for the proposed ML-CAM and ML-TCAM designs without the extra capacitor. We have evaluated the area in a 65 nm CMOS technology, where a 1-fF MIM capacitor is about $0.71\,\mu m^2$, and an SRAM cell is about $0.52\,\mu m^2$.

For SRAM-based ML-CAM and ML-TCAM designs, the adopted 1 fF MIM capacitors could be stacked on top of the

TABLE II
ACCURACY OF FEW-SHOT LEARNING TASKS

Model	TCAM Circuit	5-way Acc. 1-shot	5-way Acc. 5-shot	20-way Acc. 1-shot	20-way Acc. 5-shot
Baseline Prototypical Network [46]	N/A	98.4%	99.5%	94.6%	98.5%
256-bit LSH with BN	**CapCAM / Dynamic ML-TCAM** [7][1]	**96.8%**	**98.8%**	**90.6%**	**96.4%**
256-bit LSH w/o BN	CapCAM / Dynamic ML-TCAM [7][1]	93.0%	97.4%	82.4%	92.3%
256-bit LSH with BN	Dynamic ML-TCAM [7] with timing constraint[2]	43.7%	54.5%	52.6%	66.4%
256-bit LSH with BN	Dynamic ML-TCAM [7] with timing constraint[3]	20.6%	20.8%	7.2%	8.1%
256-bit LSH with BN	Deeply-scaled dynamic ML-TCAM [7][4]	94.1%	97.5%	86.8%	93.8%
256-bit LSH with BN	**Deeply-scaled CapCAM**[4]	**96.8%**	**98.8%**	**90.6%**	**96.4%**
4-bit quantized range encoding	**CapCAM / Dynamic ML-TCAM** [7][1]	**97.1%**	**99.2%**	**91.3%**	**97.1%**
4-bit quantized range encoding	Dynamic ML-TCAM [7] with timing constraint[2]	97.0%	99.1%	91.3%	97.1%
4-bit quantized range encoding	Dynamic ML-TCAM [7] with timing constraint[3]	39.3%	40.9%	49.8%	47.4%
4-bit quantized range encoding	Deeply-scaled dynamic ML-TCAM [7][4]	90.4%	96.0%	81.1%	90.6%
4-bit quantized range encoding	**Deeply-scaled CapCAM**[4]	**97.1%**	**99.2%**	**91.3%**	**97.1%**

[1] Assuming ideal timing control, FeFET-based ML-TCAM with 1-fF capacitors and 200-domain FeFETs, or dynamic ML-TCAM in [7] with 200-domain FeFETs shows no noticeable difference from software implementation. The ML-TCAM and dynamic ML-TCAM are evaluated separately with the same network structure.
[2] 200-domain FeFETs ($\sim$ 100nm $\times$ 100nm) with V_{th} sigma value of 61.5 mV are used for simulation, assuming only the 64 most matched states could be sensed with variation.
[3] 200-domain FeFETs are used for simulation, assuming only the 32 most matched states could be sensed with variation.
[4] 20-domain FeFETs ($\sim$ 30nm $\times$ 30nm) with V_{th} sigma value of 234 mV are used for simulation, assuming ideal timing control.

transistors within the cell footprint. The layout in Fig. 8(c) shows an ML-CAM area of $0.91\,\mu\mathrm{m}^2$ ($1.75\times$ of an SRAM cell), almost the same as a conventional 10T SRAM CAM cell. ML-TCAM in Fig. 9(b) has an area of $1.90\,\mu\mathrm{m}^2$ ($3.65\times$ an SRAM cell), which is about 25% larger than a conventional 16T SRAM TCAM cell due to the use of two extra transistors.

For FeFET-based ML-CAM and ML-TCAM designs, the impact of the extra capacitors could be more significant. The layouts in Fig. 8(d) and Fig. 9(c) show that, when neglecting the capacitor, an ML-CAM cell has an area similar to a 2FeFET TCAM (about $0.31\,\mu\mathrm{m}^2$, projected from the 45-nm FeFET TCAM in [24]), while an ML-TCAM cell has $2\times$ area of a 2FeFET TCAM. The overall area is thus determined by the MIM capacitor area ($0.71\,\mu\mathrm{m}^2$).

Therefore, when it comes to FeFET-based ML-CAM/ML-TCAM designs, or if a larger cell capacitance is adopted in SRAM-based designs for better matching accuracy, the extra capacitor dominates the overall cell area. This overhead could be significantly reduced by adopting high-κ capacitors [67]–[69] in more advanced technologies, which is commonly provided along with FeFET fabrication processes. More advanced capacitor technologies, such as pillar or 3D capacitors [70]–[72], could also further mitigate the area overhead.

V. APPLICATION BENCHMARK AND DISCUSSIONS

A. Benchmark Settings

We evaluate the FeFET-based ML-TCAM as an example of our proposed CapCAMs in few-shot learning applications on a commonly used dataset, the Omniglot dataset [73]. We use a prototypical network [46], an efficient model commonly used for few-shot learning applications, as a base model. Two CAM/TCAM-based methods are used to solve the NNS problem in the last layer: LSH or range encoding.

For the LSH method, we substitute an LSH function layer for the fully connection layer, same as [7] in the prototypical network. The width of LSH results is a hyperparameter that should balance accuracy and hardware performance. In addition, we also noticed that adding a batch normalization (BN) layer before the LSH layer could significantly improve the accuracy till near the prototypical network baseline, since normalizing the original Euclidean space could contribute to higher LSH probability of maintaining the distance metric, which is further discussed below. The complete model contains four convolutional blocks, each comprising a 64-filter 3×3 convolution, a batch normalization layer, a ReLU layer and a 2×2 max-pooling layer. A 64-dimensional feature space is then batch-normalized and sent into the LSH layer to generate binary signatures, and the signatures are compared in an $N \times M$ ML-CAM/ML-TCAM array, where N is the same as the N-way K-shot task that classifies the samples of N classes, and M is the same as the width of LSH output.

For the range encoding method, we take the encoding method from [43], [44], which utilizes ML-CAM for more compact coding. A same network with four convolutional blocks is used to extract the 64-dimensional feature space. Each 32-bit floating-point number in a feature vector is then quantized to 5 ranges based on their mean and variation, and further coded into a 4-bit thermometer code (range 1: "0000", range 2: "1000", range 3: "1100", range 4: "1110", range 5: "1111"), whose Hamming distance represents the range distance. Hence each 64-dimensional feature becomes a 256-bit binary code, and is programmed into the $N \times 256$ ML-CAM/ML-TCAM array.

The few-shot learning applications are described by N-way K-shot tasks. In an N-way K-shot task, a classifier learns from a support set containing N classes with K samples each, and it is then evaluated on a query set containing the same N classes with 1 sample each. Both the training stage and testing stage contain multiple tasks with non-overlapping N classes, so the model learns from a large dataset and is

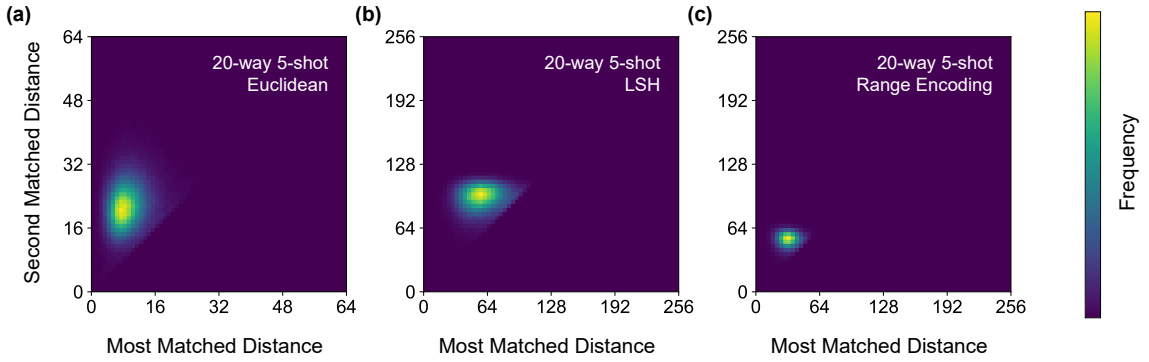


Fig. 16. Comparison of two most matched distances in 20-way 5-shot tasks: (a) Euclidean distance (used in software baseline); (b) 256-bit LSH; (c) 4-bit quantized range encoding. The distance is the complement of match degree.

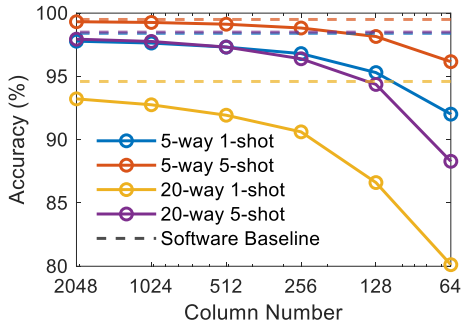


Fig. 17. Few-shot learning accuracy for different number of columns.

then tested many times on small unseen tasks to evaluate an overall performance. With a larger N and a smaller K, the task becomes more difficult to solve.

B. Accuracy and Noise Margin Analysis

Table II shows the few-shot learning accuracy under different circumstances. For both methods, we evaluate 256-column ML-TCAM arrays. It is shown that, with either a 256-bit LSH function and batch normalization or 4-bit quantized range encoding, the FeFET-based ML-TCAM could achieve high accuracy close to the prototypical network baseline. As we have demonstrated in section IV, the proposed CapCAM could reliably separate all the multi-level states with 1-fF capacitors, 200-domain FeFETs, and 256 columns.

The application benchmark results in Table II reveals that, in practice, the output variation control is not the most essential factor that affects the overall accuracy as long as the variation is not too high. Existing dynamic ML-TCAM in [7] using 200-domain FeFETs with a maximum σ 2.36$\times$ its LSB voltage (83% bit error rate) or the same CapCAM configuration as Fig. 14(e) with 1.9% bit error rate could still exhibit accuracy almost the same as software implementations. The major insight from the benchmark is that, the major limiting issue for dynamic ML-TCAM designs is the difficulty to sense the states far from fully matched. For dynamic ML-TCAM designs based on discharge rates, a small match degree indicates a large amount of cells discharging and a short discharge time on the ML parasitic capacitance. Such a critical timing constraint causes dynamic ML-TCAM to suffer from significant accuracy loss when only 64 most matched states could be distinguished.

The range encoding method is more resistant to the timing constraint and begins to suffer accuracy degradation till only 32 most matched states could be distinguished. Hence, the demonstrated capability to distinguish the 8 most matched states in [7] is not sufficient for such applications, and the timing control must be further improved for practical use.

We also evaluate how deeply-scaled devices would affect ML-CAM/ML-TCAMs. Dynamic ML-TCAMs begin to suffer accuracy degradation when the FeFETs are scaled to 20 domains ($\sim$ 30nm $\times$ 30nm) with V_{th} sigma value of 234 mV. In this case, the maximum output σ of dynamic ML-TCAM is 12$\times$ its LSB voltage, while the proposed CapCAM is almost not affected.

The batch normalization layer is also proved to be important in the LSH method, as shown in Table II. This is because LSH relies on the randomly generated hyperplanes to determine the binary signature, and that normalized features save the number of hyperplanes needed. The range encoding scheme could achieve higher accuracy and is easier for sensing compared with the LSH scheme, but it is also less resistant to FeFET device variation for dynamic ML-TCAMs.

Fig. 16 shows the most matched distance vs. the second matched distance for each sample in 20-way 5-shot tasks, which indicates the sensing difficulty. It is easier for samples near the upper-left corner, and more difficult for the ones near the diagonal, especially for those near the center (half-matched), where ML variation is the largest. Fortunately, the frequency near the diagonal or the center is relatively low, ensuring accuracy close to the software baseline. With range encoding, samples have less mismatch bits, and thus require less timing margin. The V_{REF} that maximizes the probability for distinguishing the states with first guess could also be optimized with Fig. 16. For example, in the range encoding scheme in Fig. 16(c), 81.5% samples could be correctly sensed with a V_{REF} corresponding to 43 mismatch cells.

C. Impact of Array Size

We evaluate the impact of array size on few-shot learning applications. While there are several encoding and quantization schemes, we evaluate the LSH method as an example.

Fig. 17 shows the accuracy of different few-shot learning tasks with different column numbers where the column number corresponds to the number of hyperplanes used to separate the feature space and generate LSH binary signatures. More column numbers could help improve the accuracy in distance metric mapping. It could be seen that a 256-bit or 512-bit LSH layer could achieve high accuracy comparable to the prototypical network baseline. Though conventional CAM arrays do not exceed 144-bit width for variation and sensing peripheral complexity control [11], the proposed ML-CAM/ML-TCAM arrays with higher variation tolerance and simplified sensing peripherals could provide a lower cost solution.

VI. Conclusion

In this paper, we propose CapCAM: a multi-level capacitive CAM that could provide static, scalable, and accurate match degree outputs beyond the Boolean outputs. It outperforms existing multi-level CAM designs based on dynamic match degree output sensing scheme by providing device-variation-mitigated and sensing-timing-error-tolerated static ouputs. Four different CapCAM designs, FeFET-base ML-CAM/ML-TCAM, and SRAM-based ML-CAM/ML-TCAM designs are proposed. The simulation results on circuit-level and application-level show excellent scalability and immunity against device variation. We have analyzed the noise margin and output variation of the proposed CapCAM circuits, and their impact on few-shot learning applications with different distance metric mapping methods. Overhead of extra capacitors and sensing peripherals are also discussed. The proposed CapCAM could achieve accuracy similar to software implementations in few-shot learning applications, and higher accuracy than existing dynamic ML-CAM-based designs considering the sensing timing constraints or deeply scaled memory devices.

References

[1] K. E. Batcher, "STARAN parallel processor system hardware," in Proceedings of the May 6-10, 1974, National Comput. Conf. and Exposition, 1974, pp. 405–410.

[2] A. J. McAuley et al., "Fast routing table lookup using CAMs," in Proc. IEEE INFOCOM Conf. Comput. Commun., 1993, pp. 1382–1391.

[3] K. Lakshminarayanan, A. Rangarajan, and S. Venkatachary, "Algorithms for advanced packet classification with ternary CAMs," ACM SIGCOMM Comput. Commun. Rev., vol. 35, no. 4, pp. 193–204, 2005.

[4] Y.-J. Chang, "A high-performance and energy-efficient TCAM design for IP-address lookup," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 56, no. 6, pp. 479–483, 2009.

[5] A. Anand et al., "Cheap and Large CAMs for High Performance Data-Intensive Networked Systems," in NSDI, 2010, vol. 10, p. 29.

[6] R. Karam, R. Puri, S. Ghosh, and S. Bhunia, "Emerging trends in design and applications of memory-based computing and content-addressable memories," Proc. IEEE, vol. 103, no. 8, pp. 1311–1330, 2015.

[7] K. Ni et al., "Ferroelectric ternary content-addressable memory for one-shot learning," Nat. Electron., vol. 2, no. 11, pp. 521–529, 2019.

[8] X. Xie et al., "SpaceA: Sparse Matrix Vector Multiplication on Processing-in-Memory Accelerator," in 2021 IEEE Int. Symp. High Perform. Comput. Archit. (HPCA), 2021, pp. 570–583.

[9] R. Kaplan, L. Yavits, R. Ginosar, and U. Weiser, "A resistive cam processing-in-storage architecture for DNA sequence alignment," IEEE Micro, vol. 37, no. 4, pp. 20–28, 2017.

[10] A. F. Laguna et al., "Seed-and-vote based in-memory accelerator for DNA read mapping," in 2020 IEEE/ACM Int. Conf. Comput. Aided Design (ICCAD), 2020, pp. 1–9.

[11] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (CAM) circuits and architectures: A tutorial and survey," IEEE J. Solid-State Circuits, vol. 41, no. 3, pp. 712–727, 2006.

[12] S. Jeloka, N. Akesh, D. Sylvester and D. Blaauw, "A configurable TCAM/BCAM/SRAM using 28nm push-rule 6T bit cell," in Symp. VLSI Circuits (VLSIC), 2015, pp. C272-C273.

[13] Z. Lin et al., "Two-Direction In-Memory Computing Based on 10T SRAM With Horizontal and Vertical Decoupled Read Ports," in IEEE J. Solid-State Circuits, vol. 56, no. 9, pp. 2832-2844, Sept. 2021.

[14] J. Chen et al., "A Reliable 8T SRAM for High-Speed Searching and Logic-in-Memory Operations," in IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 30, no. 6, pp. 769-780, June 2022.

[15] D. Bhattacharya et al., "Design of Efficient Content Addressable Memories in High-Performance FinFET Technology," in IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 23, no. 5, pp. 963-967, May 2015.

[16] Meng-Chou Chang et al., "Design of asymmetric TCAM (ternary content-addressable memory) cells using FinFET," 2014 IEEE 3rd Global Conf. Consumer Electronics (GCCE), 2014, pp. 358-359.

[17] V. Vinogradov, et al., "Dynamic ternary CAM for hardware search engine," Electron. Lett., vol. 50, no. 4, pp. 256–258, 2014.

[18] S. Matsunaga et al., "A 3.14 μm^2 4T-2MTJ-cell fully parallel TCAM based on nonvolatile logic-in-memory architecture," in Symp. VLSI Circuits (VLSIC), 2012, pp. 44–45.

[19] S. Matsunaga et al., "Design of a nine-transistor/two-magnetic-tunnel-junction-cell-based low-energy nonvolatile ternary content-addressable memory," Jpn. J. Appl. Phys., vol. 51, no. 2S, p. 02BM06, 2012.

[20] L. Xue et al., "ODESY: a novel 3T-3MTJ cell design with optimized area DEnsity, scalability and latencY," in 2016 IEEE/ACM Int. Conf. Comput. Aided Design (ICCAD), 2016, pp. 1–8.

[21] J. Li, R. K. Montoye, M. Ishii, and L. Chang, "1 Mb 0.41μm^2 2T-2R cell nonvolatile TCAM with two-bit encoding and clocked self-referenced sensing," IEEE J. Solid-State Circuits, vol. 49, no. 4, pp. 896–907, 2013.

[22] M.-F. Chang et al., "A 3T1R nonvolatile TCAM using MLC ReRAM for frequent-off instant-on filters in IoT and big-data processing," IEEE J. Solid-State Circuits, vol. 52, no. 6, pp. 1664–1679, 2017.

[23] X. Yin, M. Niemier, and X. S. Hu, "Design and benchmarking of ferroelectric FET based TCAM," in Design, Autom. Test Eur. Conf. Exhibit. (DATE), 2017, 2017, pp. 1444–1449.

[24] X. Yin, K. Ni, D. Reis, S. Datta et al., "An ultra-dense 2FeFET TCAM design based on a multi-domain FeFET model," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 66, no. 9, pp. 1577–1581, 2018.

[25] X. Yin et al., "FeCAM: A universal compact digital and analog content addressable memory using ferroelectric," IEEE Trans. Electron Devices, vol. 67, no. 7, pp. 2785–2792, 2020.

[26] H. Zhong, S. Cao, H. Yang and X. Li, "Dynamic Ternary Content-Addressable Memory Is Indeed Promising: Design and Benchmarking Using Nanoelectromechanical Relays," 2021 Design, Autom. Test Eur. Conf. Exhibit. (DATE), 2021, pp. 1100-1103.

[27] H. Zhong et al., "DyTAN: Dynamic Ternary Content Addressable Memory Using Nanoelectromechanical Relays," IEEE Trans. Very Large Scale Integr. Syst., vol. 29, no. 11, pp. 1981–1993, 2021.

[28] S.-Y. Wu, "A new ferroelectric memory device, metal-ferroelectric-semiconductor transistor," IEEE Trans. Electron Devices, vol. 21, no. 8, pp. 499–504, 1974.

[29] M. Trentzsch et al., "A 28nm HKMG super low power embedded NVM technology based on ferroelectric FETs," in IEDM Tech. Dig., 2016, pp. 11–15.

[30] S. Dünkel et al., "A FeFET based super-low-power ultra-fast embedded NVM technology for 22nm FDSOI and beyond," in IEDM Tech. Dig., 2017, pp. 17–19.

[31] K. Ni et al., "Write Disturb in Ferroelectric FETs and Its Implication for 1T-FeFET AND Memory Arrays," IEEE Electron Device Lett., vol. 39, no. 11, pp. 1656–1659, 2018.

[32] K. Ni et al., "Critical role of interlayer in Hf 0.5 Zr 0.5 O 2 ferroelectric FET nonvolatile memory performance," IEEE Trans. Electron Devices, vol. 65, no. 6, pp. 2461–2469, 2018.

[33] A. Aziz, S. Ghosh, S. Datta, and S. K. Gupta, "Physics-based circuit-compatible SPICE model for ferroelectric transistors," IEEE Electron Device Lett., vol. 37, no. 6, pp. 805–808, 2016.

[34] J. Wu et al., "A 3T/Cell Practical Embedded Nonvolatile Memory Supporting Symmetric Read and Write Access Based on Ferroelectric FETs," in Proc. 56th Annu. Design Autom. Conf. (DAC), 2019, pp. 1-6.

[35] X. Li et al., "Design of 2t/cell and 3t/cell nonvolatile memories with emerging ferroelectric fets," IEEE Des. Test, vol. 36, no. 3, pp. 39–45, 2019.

[36] M. Lee et al., "FeFET-based low-power bitwise logic-in-memory with direct write-back and data-adaptive dynamic sensing interface," in Proc. ACM/IEEE Int. Symp. Low Power Electron. and Design (ISLPED), 2020, pp. 127–132.

[37] J.-H. Bae et al., "Highly Scaled, High Endurance, Ω-Gate, Nanowire Ferroelectric FET Memory Transistors," IEEE Electron Device Lett., vol. 41, no. 11, pp. 1637–1640, 2020.

[38] A. A. Sharma et al., "High speed memory operation in channel-last, back-gated ferroelectric transistors," in IEDM Tech. Dig., 2020, pp. 15–18.

[39] A. Keshavarzi et al., "Ferroelectronics for edge intelligence," IEEE Micro, vol. 40, no. 6, pp. 33–48, 2020.

[40] K. Ni et al., "A Circuit Compatible Accurate Compact Model for Ferroelectric-FETs," in IEEE Symp. VLSI Technol., 2018, pp. 131–132.

[41] S. Deng et al., "A comprehensive model for ferroelectric FET capturing the key behaviors: Scalability, variation, stochasticity, and accumulation," in IEEE Symp. VLSI Technol., 2020, pp. 1–2.

[42] S. Deng et al., "Examination of the Interplay Between Polarization Switching and Charge Trapping in Ferroelectric FET," in IEDM Tech. Dig., 2020, pp. 4.4.1-4.4.4.

[43] H. Li et al., "One-Shot Learning with Memory-Augmented Neural Networks Using a 64-kbit, 118 GOPS/W RRAM-Based Non-Volatile Associative Memory," in IEEE Symp. VLSI Technol., 2021, pp. 1-2.

[44] H. Li et al., "SAPIENS: A 64-kb RRAM-Based Non-Volatile Associative Memory for One-Shot Learning and Inference at the Edge," in IEEE Trans. Electron Devices, vol. 68, no. 12, pp. 6637-6643, Dec. 2021.

[45] A. Kazemi et al., "FeFET Multi-Bit Content-Addressable Memories for In-Memory Nearest Neighbor Search," in IEEE Trans. Comput..

[46] J. Snell et al., "Prototypical Networks for Few-shot Learning," in Proc. Adv. Neural Inf. Process. Syst. 30, 2017, pp. 4077–4087.

[47] V. Oriol et al., "Matching networks for one shot learning," in Proc. Adv. Neural Inf. Process. Syst. 29, 2016, pp. 3637–3645.

[48] V. Srinivasan et al., "Fast and scalable layer four switching," in Proc. SIGCOMM, 1998, pp. 191–202.

[49] K. Lakshminarayanan et al., "Algorithms for advanced packet classification with ternary CAMs," in Proc. SIGCOMM, 2005, pp. 193–204.

[50] A. Bremler-Barr et al., "Space-Efficient TCAM-Based Classification Using Gray Coding," in IEEE Trans. Comput., vol. 61, no. 1, pp. 18-30, Jan. 2012.

[51] A. Bremler-Barr et al., "Encoding Short Ranges in TCAM Without Expansion: Efficient Algorithm and Applications," in IEEE/ACM Trans. Netw., vol. 26, no. 2, pp. 835-850, April 2018.

[52] D. Ravichandran et al., "Randomized algorithms and NLP: using locality sensitive hash function for high speed noun clustering," in Proc. 43rd Annu. Meeting Assoc. Comput. Linguistics (ACL), 2005, pp. 622–629.

[53] R. Shinde et al., "Similarity search and locality sensitive hashing using ternary content addressable memories," in Proc. 2010 Int. Conf. Management of data (SIGMOD '10), 2010, pp. 375-386.

[54] A. Santoro et al., "Meta-learning with memory-augmented neural networks," in Proc. 33rd Int. Conf. Mach. Learn. - Vol. 48 (ICML '16). 2016, pp. 1842–1850.

[55] G. Yin et al., "Enabling Lower-Power Charge-Domain Nonvolatile In-Memory Computing With Ferroelectric FETs," in IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 68, no. 7, pp. 2262-2266, July 2021.

[56] I. Arsovski et al., "Self-referenced sense amplifier for across-chip-variation immune sensing in high-performance Content-Addressable Memories," IEEE Cust. Integr. Circuits Conf., 2006, pp. 453-456.

[57] J. Lazzaro, S. Ryckebusch, M. A. Mahowald, and C. A. Mead, "Winner-take-all networks of O(N) complexity," in Proc. Adv. Neural Inform. Process. Syst. I, (NIPS'88), 1988, pp. 703-711.

[58] J. Choi and B. J. Sheu, "A high-precision VLSI winner-take-all circuit for self-organizing neural networks," in IEEE J. Solid-State Circuits, vol. 28, no. 5, pp. 576-584, May 1993.

[59] P. R. Surkanti, V. Siripurapu and P. M. Furth, "A high precision and high speed voltage-mode loser/winner-take-all circuit," IEEE 58th Int. Midwest Symp. Circuits Syst. (MWSCAS), 2015, pp. 1-4.

[60] M. Soleimani et al., "Voltage-mode loser/winner-take-all circuits," IEEE 54th Int. Midwest Symp. Circuits Syst. (MWSCAS), 2011, pp. 1-4.

[61] J. Ramirez-Angulo et al., "Low-voltage high-performance voltage-mode and current-mode WTA circuits based on flipped voltage followers," in IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 52, no. 7, pp. 420-423, July 2005.

[62] S. Ramakrishnan and J. Hasler, "Vector-Matrix Multiply and Winner-Take-All as an Analog Classifier," in IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 22, no. 2, pp. 353-361, Feb. 2014.

[63] F. Pardo et al., "A 4K-Input High-Speed Winner-Take-All (WTA) Circuit with Single-Winner Selection for Change-Driven Vision Sensors," Sensors, vol. 19, no. 2, p. 437, Jan. 2019.

[64] A. T. Do et al., "Design of a power-efficient CAM using automated background checking scheme for small match line swing," in 2013 Proc. ESSCIRC (ESSCIRC), 2013, pp. 209-212.

[65] B. Song et al., "A 10T-4MTJ Nonvolatile Ternary CAM Cell for Reliable Search Operation and a Compact Area," in IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 64, no. 6, pp. 700-704, June 2017.

[66] C. -C. Lin et al., "7.4 A 256b-wordlength ReRAM-based TCAM with 1ns search-time and 14× improvement in wordlength-energyefficiency-density product using 2.5T1R cell," in 2016 IEEE Int. Solid-State Circuits Conf. (ISSCC), 2016, pp. 136-137.

[67] Y. Bai et al., "High dielectric-constant ceramic-powder polymer composites", Appl. Phys. Lett., vol. 76, pp. 3804-3806, 2000.

[68] J. Lu and C. P. Wong, "Recent advances in high-k nanocomposite materials for embedded capacitor applications," in IEEE Trans. Dielect. Electr. Insul., vol. 15, no. 5, pp. 1322-1328, October 2008.

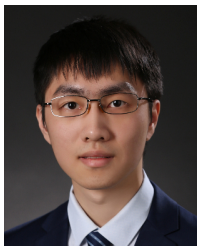
[69] S. Beyer et al., "FeFET: A versatile CMOS compatible device with game-changing potential," 2020 IEEE Int. Memory Workshop (IMW), 2020, pp. 1-4.

[70] H. Sunami, "Development of three-dimensional MOS structures from trench-capacitor DRAM cell to pillar-type transistor," 2008 9th Int. Conf. Solid-State and Integrated-Circuit Technology, 2008, pp. 853-856.

[71] M. Popovici et al., "High-performance (EOT $<$0.4 nm, Jg˜10^{-7}A/cm^2) ALD-deposited Ru\SrTiO$_3$ stack for next generations DRAM pillar capacitor," in IEDM Tech. Dig., 2018, pp. 2.7.1-2.7.4.

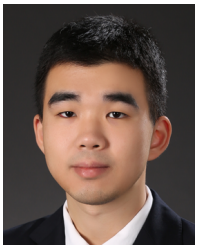
[72] S. Jeannot et al., "Toward next high performances MIM generation: up to 30fF/µm2 with 3D architecture and high-k materials," in IEDM Tech. Dig., 2007, pp. 997-1000.

[73] B. M. Lake et al., "Human-level concept learning through probabilistic program induction," Science, vol. 350, no. 6266, pp. 1332–1338, Dec. 2015.



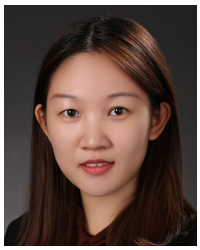
Xiaoyang Ma (Student Member, IEEE) is currently working toward the B.S. degree in electronic engineering at Tsinghua University, Beijing, China.

His current research interests include emerging memory devices and energy-efficient circuits.



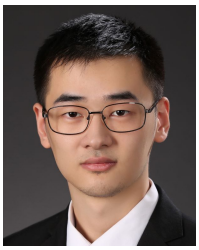
Hongtao Zhong (Student Member, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 2021, where he is currently pursuing the Ph.D. degree with the Department of Electronic Engineering.

His current research interests include low-power circuit design, emerging memory design with beyond-CMOS technologies.



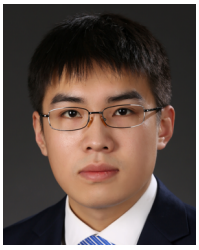
Nuo Xiu (Student Member, IEEE) received the B.S. degree in electrical engineering from University of Electronic Science and Technology of China, Chengdu, China, in 2017. She is currently pursuing the M.S. degree with the Department of Electronic Engineering, Tsinghua University, Beijing, China.

Her current research interests include computing in memory, design of circuits and architectures based on beyond-CMOS technologies.



Yiming Chen (Student Member, IEEE) received the B.S. in electronic engineering from Tsinghua University, Beijing, in China, in 2021. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Tsinghua University, Beijing, China.

His current research interests include computing-in-memory architecture and co-optimization on artificial intelligence.



Guodong Yin (Student Member, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 2020. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Tsinghua University, Beijing, China.

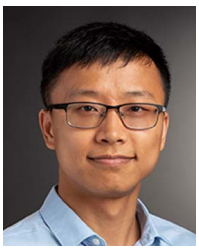
His current research interests include circuit design of SRAM, memory, and computation-in-memory circuit designs.

Vijaykrishnan Narayanan (Fellow, IEEE) received the B.S. degree in computer science and engineering from the University of Madras, Chennai, India, in 1993, and the Ph.D. degree in computer science and engineering from the University of South Florida, Tampa, FL, USA, in 1998.

He is currently the Robert Noll Chair Professor of Computer Science and Engineering and Electrical Engineering at Pennsylvania State University, University Park, PA, USA. He is also the Co-Director of the Microsystems Design Laboratory. His current research interests include power-aware and reliable systems, embedded systems, nanoscale devices, and interactions with system architectures, reconfigurable systems, computer architectures, network-on-chips, and domain-specific computing.

Yongpan Liu (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees from the Department of Electronic Engineering, Tsinghua University, Beijing, China, 1999, 2002, and 2007, respectively.

He was a Visiting Scholar with Pennsylvania State University, State College, PA, USA, and the City University of Hong Kong, Hong Kong. He is currently a Professor with the Department of Electronic Engineering, Tsinghua University. He has published over 200 peer-reviewed conference and journal papers and developed several fast sleep/wakeup nonvolatile processors using emerging memory and artificial intelligent accelerators using algorithm-architecture co-optimization. His main research interests include energy-effcient circuits and systems for artificial intelligence, emerging memory devices, and Internet-of-Things (IoT) applications.



Kai Ni (Member, IEEE) received the B.S. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 2011, and the Ph.D. degree in electrical engineering from Vanderbilt University, Nashville, TN, USA, in 2016, with a focus on characterization, modeling, and reliability of III-V MOSFETs.

Since then, he became a Post-Doctoral Associate with the University of Notre Dame, working on ferroelectric devices for nonvolatile memory and novel computing paradigms. He is currently an Assistant Professor in electrical and microelectronic engineering with the Rochester Institute of Technology. He has 80 publications in top journals and conference proceedings, including nature electronics, IEDM, VLSI Symposium, IRPS, EDL, etc. His current interests lie in nanoelectronic devices empowering unconventional computing, AI accelerator, and 3-D memory technology.

Huazhong Yang (Fellow, IEEE) received the B.S. degree in microelectronics and the M.S. and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 1989, 1993, and 1998, respectively.

In 1993, he joined the Department of Electronic Engineering, Tsinghua University, where he has been a Professor since 1998. He has been in charge of several projects, including projects sponsored by the National Science and Technology Major Project, the 863 Program, NSFC, and several international research cooperations. He has authored or coauthored over 500 technical articles, seven books, and over 180 granted Chinese patents. His research interests include wireless sensor networks, data converters, energy-harvesting circuits, nonvolatile processors, and brain-inspired computing.

Xueqing Li (Senior Member, IEEE) received the B.S. and Ph.D. degrees from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2007 and 2013, respectively.

He is currently an Associate Professor with the Department of Electronic Engineering, Tsinghua University. From 2013 to 2017, he was a Postdoctoral Researcher with the Department of Computer Science and Engineering, Penn State University, University Park, PA, USA. He joined the Department of Electronic Engineering, Tsinghua University, as an Assistant Professor, in 2018. He has more than 100 publications and holds over 20 China and U.S. patents. His research interests include low-power circuit design, emerging memory and memory-oriented computing with beyond-CMOS technologies, and high-performance data converter circuit design.