Capacitive Content-Addressable Memory: A Highly Reliable and Scalable Approach to Energy-Efficient Parallel Pattern Matching Applications

Nuo Xiu[1], Yiming Chen[1], Guodong Yin[1], Xiaoyang Ma[1], Huazhong Yang[1], Sumitha George[2], Xueqing Li[1]
[1] BNRist/ICFC, EE Dept., Tsinghua University, Beijing, China; [2] ECE Dept., North Dakota State University, Fargo, USA
Contact Email: xueqingli@tsinghua.edu.cn

ABSTRACT

Content-addressable memory (CAM) has been a critical component in pattern matching and also machine-learning applications. Recently emerged CAM that is capable of delivering multi-level distance calculation is promising for applications that need matching results beyond Boolean results of "matched" and "not matched". However, existing multi-level CAM designs are constrained by the bit-cell device discharging current mismatch and the strict timing of sensing operations for distance calculation. This fact results in the challenge of further improving the accuracy and scalability towards higher-resolution and higher-dimension matching. This work presents a multi-level CAM design that is capable of delivering high-accuracy and high-scalability search, which is immune to the discharging device mismatch and needs no strict timing for result sensing. The inherent enabler is the charge-domain computing mechanism. This work will present the operating mechanisms, the circuit simulation, and content-matching evaluation results, showing the promise towards high reliability, high energy efficiency, and high scalability.

CCS CONCEPTS

• Hardware → Static memory; None-volatile memory; Arithmetic and datapath circuits; • Theory of computation → Pattern matching

KEYWORDS

Ternary Content-Addressable Memory, Multiple-Level CAM, Ferroelectric FET, Low-Power Design, Pattern Matching

ACM Reference format:
Nuo Xiu, Yiming Chen, Guodong Yin, Xiaoyang Ma, Huazhong Yang, Sumitha George and Xueqing Li. 2021. Capacitive Content-Addressable Memory: A Highly Reliable and Scalable Approach to Energy-Efficient Parallel Pattern Matching Applications. In *Great Lakes Symposium on VLSI 2021 (GLSVLSI'21), June 22-25, 2021, Virtual Event.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3453688.3461744

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
GLSVLSI '21, June 22–25, 2021, Virtual Event, USA.
© 2021 Association of Computing Machinery.
ACM ISBN 978-1-4503-8393-6/21/06...$15.00. https://doi.org/10.1145/3453688.3461744

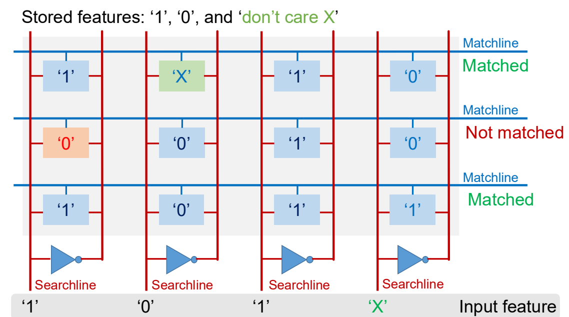


Figure 1. TCAM for pattern search applications.

1 Introduction

Various data-intensive applications require parallel data search to figure out whether the input data stream matches the stored vector data. This could be found in database, routers, and also deep learning edge computing tasks in recent efforts [1-4][8]. In these scenarios, content-addressable memory (CAM) has been a critical component that could carry out the search operation in parallel for all stored memory vector candidates in the rows of a memory array, leading to a Boolean result of either 'matched' or 'not-matched' for each vector comparison. In addition to CAM, there is also ternary CAM (TCAM) [5][6], in which a 'don't care' search rule claimed by the 'stored bit' or the 'input bit' could be applied to bypass certain bits. For one bit location, if either the input bit or the store bit is a 'don't care' bit, the matching result of this bit location is considered as always 'matched', and the vector matching result depends on the other bit locations. **Figure 1** illustrates these concepts.

There have been CAM and TCAM designs in both CMOS SRAM and emerging nonvolatile memory (NVM) devices, showing different density, power, latency, scalability, and reliability [7][8][9]. More importantly, it is noted that conventional CAM and TCAM designs yield *Boolean* matching results between the inputs data stream and the stored data patterns. In other words, the search result in each row vector is either 'fully matched' or 'not matched'. This is sufficient for many aforementioned applications. More interestingly, as reported in [8], if the CAM or TCAM search output could deliver the 'degree-of-match' feature for each row vector, in a form of, e.g., the Hamming Distance, we could have a new CAM or TCAM category, which is named multi-level CAM (ML-CAM) or ML-TCAM in this paper, as illustrated in

Figure 2. An exciting outcome of ML-CAM is that it becomes practical and also efficient to support feature classifications, and further, one-shot learning applications [8].

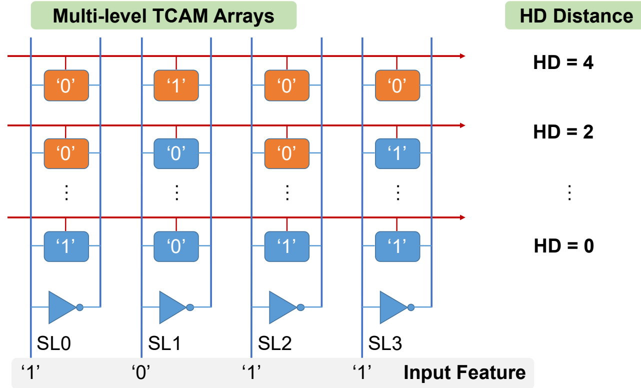


Figure 2. The concept of multi-level (T)CAM.

However, as to be further revealed, the 'degree-of-match' feature is not well supported in existing CAM and TCAM designs. The design in [8] is based on tracking the settling slope of the matching results, and is not scalable, reliable, or accurate because of high peripheral sensing costs and intrinsic FeFET device variations. In this work, we exploit a new method of capacitive ML-CAM and ML-TCAM design, which overcomes the sensing-peripheral-oriented power challenge and also the device-variation-oriented scalability and reliability issues. Detailed contribution includes:

(i) We propose a new operating theory of capacitive ML-CAM and ML-TCAM for enhanced power efficiency, scalability, and reliability;

(ii) We propose four capacitive ML-CAM/ML-TCAM designs, namely FeFET-based ML-CAM and ML-TCAM, and SRAM-based ML-CAM and ML-TCAMs;

(iii) We evaluate the proposed four ML (T)CAM designs in terms of functionality, energy performance, speed, and also reliability, showing the promise for both conventional content-addressing applications and the emerging patten-matching-based machine-learning applications.

In the rest of this paper, **Section 2** reviews the FeFET device background and existing CAM/TCAM designs; **Section 3** presents the proposed capacitive ML-CAM and TCAMs based on FeFET and CMOS SRAM; **Section 4** evaluates the proposed designs and **Section 5** concludes this work.

2 Background

A. The FeFET Device Basics

FeFETs are essentially MOSFETs with a ferroelectric layer integrated at their gate stack, as shown in **Figure 3** [11][13]. Interaction between the FE layer and the MOSFET gate oxides leads to unique FeFET characteristics different from a conventional MOSFET. FeFETs store the polarization direction in the FE layer as the memory state [12]. The direction of the polarization tunes the FeFET threshold voltage V_{TH} from low or high [8][14]. It is noted that the lower-end V_{TH} could be set to either negative or positive so as to provide low or high channel resistance with zero V_G.

Writing FeFETs can be done by applying a voltage across the ferroelectric layer beyond the coercive voltage for a certain period of time [16][17]. **Figure 3** shows the FeFET I_{DS}-V_{GS} curves with multiple states. In general, a positive (or negative) voltage applied to the gate of an n-type FeFET tends to reduce (increase) the device V_{TH}. The polarization switching could be modulated by tuning the write voltage pulse amplitude (V_{write}) or duration (T_{write}) applied to the gate. Reading FeFET could be done by detecting I_{DS} or V_{TH}, with an applied gate voltage lower than V_{write} to avoid read disturb.

Hafnium-based FeFETs are highly scalable even in advanced FinFET technologies [17]. Reports show that FeFETs could exhibit a high on/off ratio beyond 10^6, implying the capability of large memory arrays [13]. FeFETs also exhibit high switching speed, moderate endurance, and moderate write voltage. Currently, the device variations could be significant due to the polarization switching dynamics, possible memory access disturbs, and leaky gate stack. Reports of FeFET-based circuits are also emerging [7-9][12][14-16][21-26].

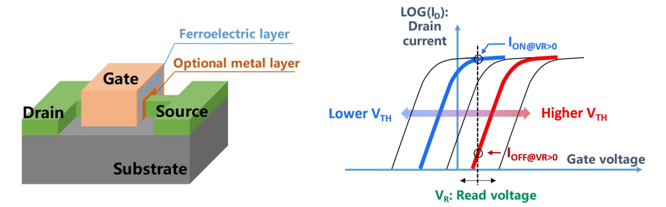


Figure 3. FeFET device and I_D-V_G curve [11][13].

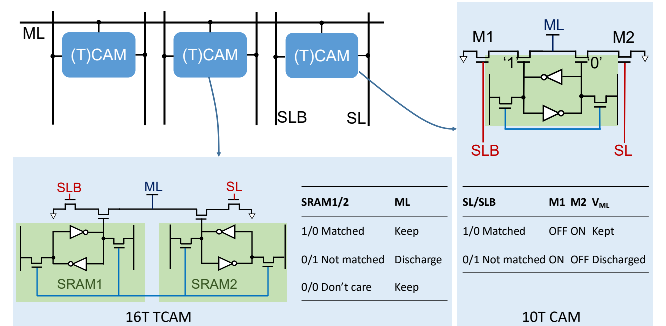


Figure 4: CAM and TCAM operation example in one row.

B. CAM and TCAM: Existing Designs and Operations

The CAM and TCAM functions have been illustrated in **Figure 1** (as a black box). This sub-section introduces typical circuit implementation and the operation theory. A CAM or TCAM is usually organized as a 2D array, as illustrated in **Figure 4,** in which SRAM is used as an example to show the basic operating mechanism. The stored vector bits are placed row-wise (one row for one vector) in the TCAM array. The input vector bit stream is sent into the array through vertical searchlines SL/SLB. For a CAM array, the matchline ML is precharged and then left floating. After that, the corresponding input bit and stored bit in each CAM cell

perform an XNOR-style operation, which will discharge ML if the two bits do not match (or leave ML floating otherwise). Therefore, if one or more cells do not match, ML will be discharged; otherwise, ML will remain high and deliver a "matched" output.

The ternary CAM, or TCAM, is slightly different from CAM, as TCAM enables an extra 'don't care' bit state beyond '0' and '1'. As illustrated in **Figure 4**, this is implemented by providing an option of always turning off the discharging path in the TCAM cell (usually by turning off the corresponding switches controlled by the 'don't care' bits). As a result, a TCAM cell needs 3-state memory: '0/1', '1/0', '0/0', and is usually implemented with 2 bits (the '1/1' is redundant and not used).

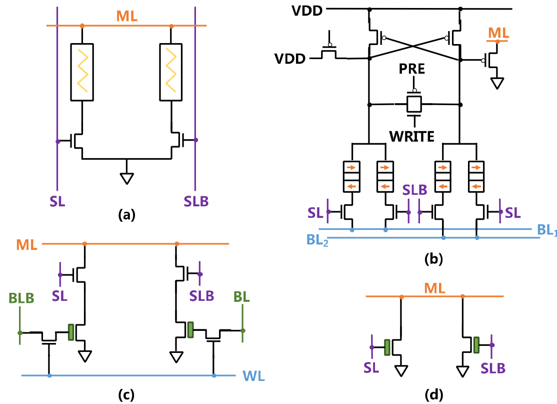


Figure 5. Existing NVM-based TCAM cell designs that deliver binary search results: (a) RRAM-based TCAM; (b) MRAM-based TCAM; (c-d) FeFET-based TCAM.

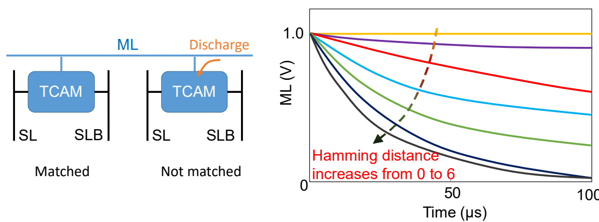


Figure 6. Multi-level TCAM: an existing work [8].

The widely used SRAM-based 10T CAM and 16T TCAM in **Figure 4** are mature, stable, and fast, but consume more area (due to large transistor count) and static power (due to the SRAM leakage currents) [20]. There have also been designs using emerging NVM technologies, as summarized in **Figure 5**. The RRAM-based 2T2R TCAM in **Figure. 5(a),** the MRAM-based TCAM in **Figure 5(b),** and the FeFET-based TCAM in **Figure 5(c)(d)** are much more compact [7]. Among them, the CAMs based on RRAM and MRAM usually consume higher power in write operations due to their device state switching mechanisms. Furthermore, the low on-off ratio of RRAM and MRAM also results in high sensing complexity and costs, as a few off-state NVM-based CAM cells may sum up significant discharging current close to the on-state currents. Last but not the least, these emerging devices suffer from significant device variations of on-state and off-state currents, which limits the scalability and reliability. FeFETs have a high on/off ratio, which is good for scalability. FeFETs have no DC power consumption during write and search operations, which also leads to high power efficiency [11][17]. The FeFET TCAM design in **Figure 5(d)** further exploits FeFET as both memory and comparison device, leading to very high density TCAM.

As illustrated in **Figure 2**, when multi-level search and comparison is needed for delivering the "degree of match", there is already a design in **Figure 6** from [8], which achieves this goal by sensing the discharging dynamics: ML is precharged first, and the ML output sense amplifier senses the decreasing slope of ML voltage after the input pattern is applied to SL/SLB. With more not-matched bits, ML is discharged more quickly. This time-domain comparison needs careful timing and is highly vulnerable to the device variations of on-state currents, resulting in weak scalability towards a large TCAM array. This issue is solved in this work by adopting the proposed capacitive search method, as to be further discussed.

3 Proposed Capacitive ML-CAM and ML-TCAM

A. Capacitive Coupling, Charge Distribution

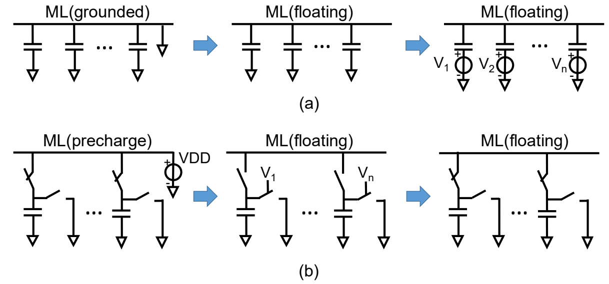


Figure 7. Adopted operation theories: (a) capacitive coupling for ML-CAM; (b) charge re-distribution for ML-TCAM.

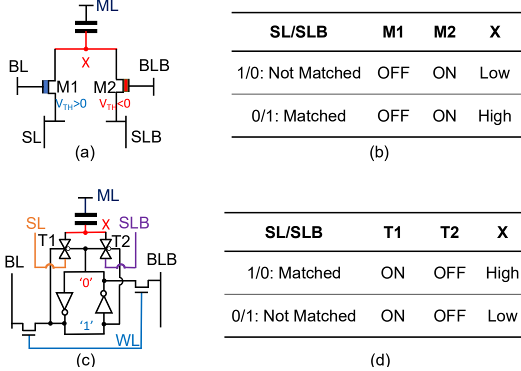
There are two basic capacitive multi-level content searching schemes, as illustrated in **Figure 7**. In **Figure 7(a)**, a few capacitors short their top plates with an initial voltage, say GND as an example. Each bottom plate is also initially grounded. After leaving the top plate floating, each bottom plate could accept a voltage input. Thanks to the capacitive coupling, the shorted top plate will settle down to the average voltage of the bottom plate inputs weighted by the capacitor size. This will be adopted for the proposed ML-CAM designs.

Another scheme is based on charge distribution. As illustrated in **Figure 7(b)**, each cell precharges the capacitor, and each capacitor may be discharged subsequently by an XNOR switch path. At last, these capacitors (some may be discharged) could be shorted together, leading to the charge re-distribution towards a weighted voltage at the top plate. This will be adopted for the proposed ML-TCAM designs.

B. Proposed Capacitive ML-CAM in FeFET and CMOS SRAM

The capacitive coupling from the capacitor bottom plates to the shorted top plate is adopted for the design of ML-CAM [15], by

adding an XNOR charging cell to provide the bottom plate input. For the FeFET-based ML-CAM design in **Figure 8(a)**, the XNOR operation is implemented between the FeFET source inputs (SL and SLB) and the FeFET stored on/off states, as summarized in **Figure 8(b)**.



SL/SLB	M1	M2	X
1/0: Not Matched	OFF	ON	Low
0/1: Matched	OFF	ON	High

SL/SLB	T1	T2	X
1/0: Matched	ON	OFF	High
0/1: Not Matched	ON	OFF	Low

Figure 8. Proposed Capacitive FeFETs and CMOS ML-CAM.

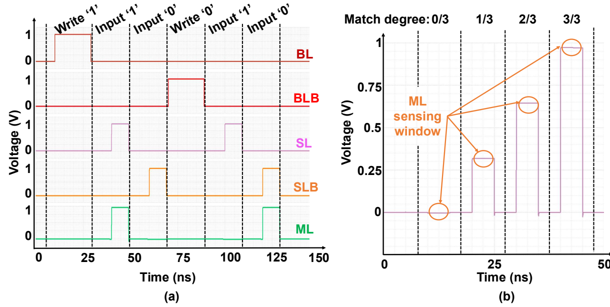


Figure 9. A transient waveform snapshot of the proposed capacitive CMOS SRAM ML-CAM: (a) Single-cell simulation; (b) 3-column simulation with 4 matching degree scenarios.

For the SRAM-based ML-CAM design in **Figure 8(c)**, the XNOR is implemented by charging the capacitor bottom plate through two NMOS switches connected in series, with the NMOS gates controlled by the stored bit and the external search pattern bit. The XNOR table is shown in **Figure 8(d).**

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

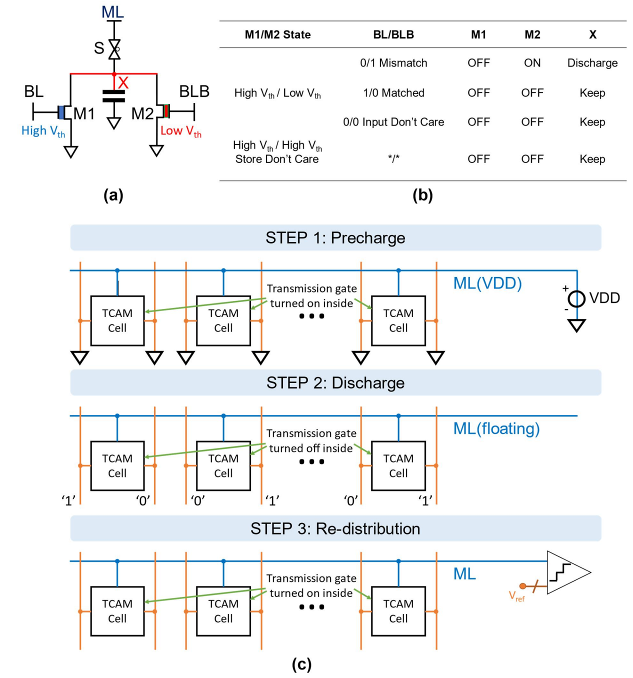
(i) Step 1: ML, SL, and SLB are grounded, which also leads to the capacitor bottom plate voltage reset to GND;

(ii) Step 2: leave ML floating, and then activate the input SL and SLB, which sets the capacitor bottom plate voltage to either GND or VDD (depending on the matching XNOR results);

(iii) Step 3: sense the ML voltage and compare it with predefined references to digitize the "degree-of-match".

As an example, **Figure 9** shows a transient simulation snapshot, in which the degree-of-match of 0, 1/3, 2/3, and 1 in a 3-column CAM array is shown. The simulation settings, along with the device models are the same as the default settings to be provided in **Section 4**.

C. Proposed Capacitive ML-TCAM in FeFET and CMOS SRAM



M1/M2 State	BL/BLB	M1	M2	X
High V_{th} / Low V_{th}	0/1 Mismatch	OFF	ON	Discharge
	1/0 Matched	OFF	OFF	Keep
	0/0 Input Don't Care	OFF	OFF	Keep
High V_{th} / High V_{th} Store Don't Care	*/*	OFF	OFF	Keep

Figure 10. Proposed Capacitive FeFET ML-TCAM: (a) Cell structure; (b) State mapping table; (c) Step-by-step operation scheme.

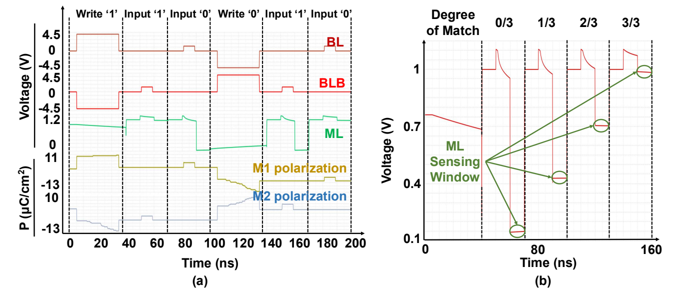


Figure 11. A transient waveform snapshot of the proposed capacitive FeFET-based ML-TCAM: (a) Single-cell simulation; (b) 3-column simulation with 4 matching degree scenarios.

The SRAM-based ML-TCAM needs to support both storing and receiving the 'don't care' state (indicating no discharging of the capacitor). Direct reuse of the ML-CAM circuit results in incapability of supporting 'don't care' state storage, because there is always one CMOS transmission gate turned on to set the capacitor bottom plate as the search input (SL or SLB).

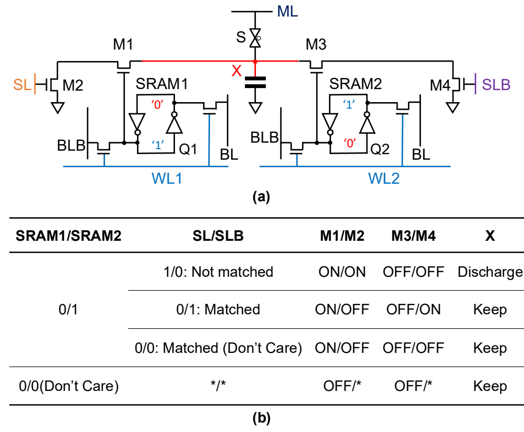
As presented above, the charge re-distribution theory could be adopted for the design of FeFET-based and SRAM-based ML-TCAM. **Figure 10** shows the cell structure and operation table of FeFET-based ML-TCAM. It is noted that the two FeFETs are configured to exhibit two high V_{TH} (for 'don't care' state) or one high positive V_{TH} plus one low positive V_{TH}. The FeFET with a low positive V_{TH} is off at zero gate biasing and on at a proper V_R>0;

The FeFET with a high positive V_{TH} is off at zero gate biasing and also at the preset V_R (>0). **Figure 10** also illustrates the step-by-step operation scheme:

(i) Step 1: The CMOS transmission gate between ML and the capacitor is turned on to precharge the capacitor through ML, while SL and SLB (shared with BL and BLB) are set to GND to ensure both FeFETs are off (to prevent a short path between ML and GND);

(ii) Step 2: The CMOS transmission gate between ML and the capacitor is turned off to leave ML floating, and then SL/SLB is activated to be V_R/GND or GND/GND ('don't care' input); The capacitor may be discharged if a V_R input at SL/SLB is applied to an FeFET with low positive V_{TH}, as summarized by the table in Figure 10; Otherwise, the capacitor is not discharged;

(iii) Step 3: SL/SLB is grounded and then the CMOS transmission gate is turned on for charge re-distribution; the settled ML indicates the degree of matching, with GND for fully not matched and VDD for fully matched.



SRAM1/SRAM2	SL/SLB	M1/M2	M3/M4	X
0/1	1/0: Not matched	ON/ON	OFF/OFF	Discharge
	0/1: Matched	ON/OFF	OFF/ON	Keep
	0/0: Matched (Don't Care)	ON/OFF	OFF/OFF	Keep
0/0(Don't Care)	*/*	OFF/*	OFF/*	Keep

(b)

Figure 12. Proposed capacitive SRAM-based ML-TCAM: (a) Cell structure; (b) State mapping table.

Figure 11 shows a transient simulation snapshot of FeFET ML-TCAM, in which the degree-of-match of 0, 1/3, 2/3, and 1 in a 3-column TCAM array is shown. The simulation settings, along with the device models are the same as the default settings to be provided in **Section 4**.

Figure 12 shows the structure and operation state mapping table of SRAM-based ML-TCAM. The step-by-step operation scheme is slightly different from that of FeFET ML-TCAM:

(i) Step 1: The CMOS transmission gate between ML and the capacitor is turned on to precharge the capacitor through ML, while SL and SLB are set to GND;

(ii) Step 2: The CMOS transmission is turned off to leave ML floating, and then SL/SLB is activated to be VDD/GND or GND/GND ('don't care' input); The capacitor may be discharged if SL and Q1 are both high or SLB and Q2 are both high; Otherwise, the capacitor is not discharged;

(iii) Step 3: same as Step 3 in the FeFET ML-TCAM scheme.

The transient waveforms of the SRAM-based ML-TCAM is similar to **Figure 11**, and not included due to page length.

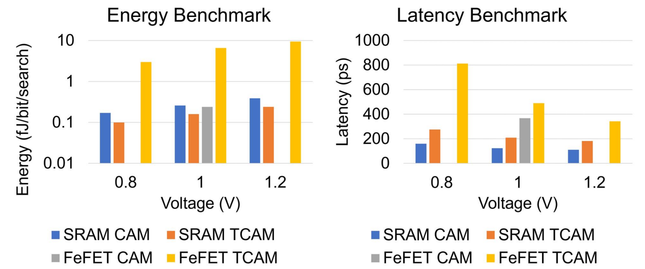
4 EVALUATION AND DISCUSSION

A. Benchmark Settings

The MOSFETs in all the designs are modeled in a commercial 65nm CMOS process. The FeFET-based ML-CAM design is simulated with the FeFET model from [18], with 9nm ferroelectric layer thickness, 0.11 $\boldsymbol{\rho}$, and 1 fin per FeFET. For FeFET-based ML-TCAM design simulation, the model from [19] is adopted to support low positive V_{TH} and high positive V_{TH}, with 200 ferroelectric domains, 8nm ferroelectric layer thickness and 19n τ_O. Both FeFET models have been calibrated with ferroelectric device samples from the foundry. All of the four cell structures adopt the same 2.0fF capacitor. The simulation is carried out for an array with 128 rows, and the bitline parasitic capacitance is modeled as 12.8fF.

B. Energy and latency evaluation

Figure 13 shows the latency comparison between the proposed designs. While this work did not consider the ML sensing peripheral overheads, the multi-level search is inherently fast (less than 1ns latency). In addition, CAMs are faster than TCAMs due to no need of precharging the capacitors.



* FeFET CAM lacks data for 0.8V and 1.2V due to the model operation range.

Figure 13. Energy and latency simulation results.

However, the adopted FeFET model operates at a low-voltage mode and its latency could be potentially lowered. In general, the operation speed is fast and the speed could be even improved with mature FeFETs.

Figure 13 also shows the worst-case search energy for different designs at different supply voltage VDD. It can be observed that the energy is generally proportional with VDD^2. This is because the search operations for both SRAM- and FeFET-based designs consume only dynamic power in charging the capacitors on the ML and the parasitic capacitors.

C. Multi-level Output Analysis

In the proposed designs, multiple cells with embedded capacitors are connected through ML. While this capacitor matching is much better than the FeFET matching accuracy, the matching degree result of this work is weighted by the capacitors in each column. Therefore, we evaluate the impact of the capacitor mismatch. Thanks to the high FeFET on/off ratio, **Figure 14**

shows that, under different capacitor and column sizes, the proposed design is capable of operating with 255 columns and only 1fF unit capacitor per cell, showing an excellent noise margin of 47% of the total voltage range.

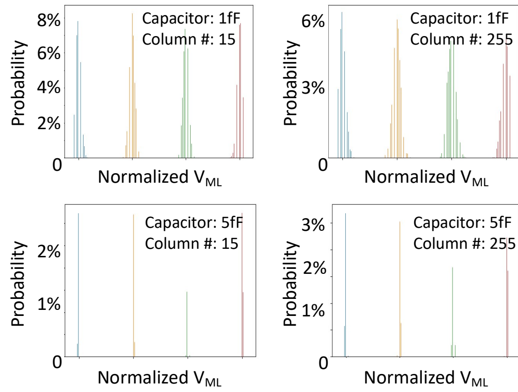


Figure 14. Scalability vs capacitance size and column size.

Table I. Benchmarking different TCAM designs

	Latency	Energy	ML	Scalability	Density
FeFET-ML [8]	355 ps	0.40 fJ/b *	Yes	Limited	Good
FeFET capacitive-ML#	**342** ps	**6.58** fJ/b	**Yes**	**Good**	Good
SRAM-TCAM [10]	582 ps	1.0 fJ/b	No	Good	Limited
SRAM capacitive-ML#	**182** ps	**0.16** fJ/b	**Yes**	**Good**	Limited
MTJ-TCAM [27]	1000 ps	40.5 fJ/b	No	Good	Good
RRAM-TCAM [28]	155 ps	0.71 fJ/b	No	Good	Good

*[8]: No sensing overhead or device variation included.
#This work: overhead of sensing not included.

The overall benchmarking is listed in **Table I**. Compared with existing designs, the proposed capacitive designs are the only one to have good scalability multi-level search operations, while still showing excellent energy and latency performance. Future work is of significance to quantify the sensing interface costs together with the impact of memory device variations.

5 CONCLUSION

This paper has presented capacitive multi-level CAM and TCAM designs for pattern matching and deep-learning application based on FeFET and CMOS SRAM. These designs enable multiple levels of matching degree beyond the Boolean levels of existing works. The evaluation has shown high energy efficiency and excellent scalability and immunity against nonvolatile memory device variations.

ACKNOWLEDGMENTS

This work is supported in part by National Key R&D Program of China (#2019YFA0706100) and NSFC (#61874066, #61720106013). The authors thank Prof. Vijaykrishnan Narayanan for helpful discussions. Authors N. Xiu and Y. Chen contributed equally to this work.

REFERENCES

[1] A. J. McAuley and P. Francis, “Fast routing table lookup using CAMs,” in *IEEE INFOCOM ’93 The Conference on Computer Communications, Proceedings, 1993*, pp. 1382–1391 vol.3.

[2] Y.-J. Chang, “A High-Performance and Energy-Efficient TCAM Design for IP-Address Lookup,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56, no. 6, pp. 479–483, 2009.

[3] R. McGeer and P. Yalagandula, “Minimizing Rulesets for TCAM Implementation,” in *IEEE INFOCOM 2009*, 2009, pp. 1314–1322.

[4] R. Karam, R. Puri, S. Ghosh, and S. Bhunia, “Emerging Trends in Design and Applications of Memory-Based Computing and Content-Addressable Memories,” *Proceedings of the IEEE*, vol. 103, no. 8, pp. 1311–1330, 2015.

[5] H. Mahmood, Z. Ullah, O. Mujahid, I. Ullah, and A. Hafeez, “Beyond the Limits of Typical Strategies: Resources Efficient FPGA-Based TCAM,” *IEEE Embedded Systems Letters*, vol. 11, no. 3, pp. 89–92, 2019.

[6] B. Rajendran et al., “Demonstration of CAM and TCAM Using Phase Change Devices,” in *2011 3rd IMW*, 2011pp. 1–4.

[7] X. Yin, D. Reis, M. Niemier, and X. S. Hu, “Ferroelectric FET Based TCAM Designs for Energy Efficient Computing,” in *2019 ISVLSI*, 2019, pp. 437–442.

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

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

[10] A. T. Do et al, “Design of a power-efficient CAM using automated background checking scheme for small match line swing,” in *2013 ESSCIRC*, 2013pp. 209–212.

[11] A. I. Khan, C. W. Yeung, C. Hu, and S. Salahuddin, “Ferroelectric negative capacitance MOSFET: Capacitance tuning antiferroelectric operation,” in *2011 International Electron Devices Meeting*, 2011, pp. 11.3.1-11.3.4.

[12] S. George et al., “Nonvolatile Memory Design Based on Ferroelectric FETs,” in *2016 53nd DAC*, 2016, pp. 1–6.

[13] K. Ni et al., “Critical Role of Interlayer in Hf0.5Zr0.5O2 Ferroelectric FET Nonvolatile Memory Performance,” *IEEE Transactions on Electron Devices*, vol. 65, no. 6, pp. 2461–2469, 2018.

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

[15] G. Yin et al., “Enabling Lower-Power Charge-Domain Nonvolatile In-Memory Computing with Ferroelectric FETs,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, pp. 1–1, 2021.

[16] K. Ni, S. Dutta, and S. Datta, “Ferroelectrics: From Memory to Computing,” in *2020 25th ASP-DAC*, 2020, pp. 401–406.

[17] J. Müller et al., “Ferroelectricity in HfO2 enables nonvolatile data storage in 28 nm HKMG,” in *2012 VLSIT*, 2012, pp. 25–26.

[18] A. Aziz et al, “Physics-Based Circuit-Compatible SPICE Model for Ferroelectric Transistors,” *IEEE Electron Device Letters*, vol. 37, no. 6, pp. 805–808, 2016.

[19] S. Deng et al., “A Comprehensive Model for Ferroelectric FET Capturing the Key Behaviors: Scalability, Variation, Stochasticity, and Accumulation,” in *2020 IEEE Symposium on VLSI Technology*, 2020, pp. 1–2.

[20] K. Pagiamtzis and A. Sheikholeslami, “Content-addressable memory (CAM) circuits and architectures: a tutorial and survey,” *IEEE Journal of Solid-State Circuits*, vol. 41, no. 3, pp. 712–727, 2006.

[21] M. Lee et al., “FeFET-based low-power bitwise logic-in-memory with direct write-back and data-adaptive dynamic sensing interface,” in *Proceedings of the ACM/IEEE International Symposium on Low Power Electronics and Design*, 2020.

[22] J. Wu et al., “Adaptive Circuit Approaches to Low-Power Multi-Level/Cell FeFET Memory,” in *2020 25th ASP-DAC*, 2020, pp. 407–413.

[23] X. Li et al., “Design of 2T/Cell and 3T/Cell Nonvolatile Memories with Emerging Ferroelectric FETs,” *IEEE Design Test*, vol. 36, no. 3, pp. 39–45, 2019.

[24] J. Wu et al., “A 3T/cell practical embedded nonvolatile memory supporting symmetric read and write access based on ferroelectric FETs,” in *Proceedings of the 56th Annual Design Automation Conference 2019*, 2019.

[25] X. Li and L. Lai, “Nonvolatile Memory and Computing Using Emerging Ferroelectric Transistors,” in *2018 ISVLSI*, 2018, pp. 750–755.

[26] X. Li et al., “Enabling Energy-Efficient Nonvolatile Computing With Negative Capacitance FET,” IEEE Transactions on Electron Devices, vol. 64, no. 8, pp. 3452–3458, 2017.

[27] B. Song, T. Na, J. P. Kim, S. H. Kang, and S.-O. Jung, “A 10T-4MTJ nonvolatile ternary CAM cell for reliable search operation and a compact area,” *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 64, no. 6, pp. 700–704, 2017.

[28] 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 ISSCC*.