IEICE TRANS. ELECTRON., VOL.E107–C, NO.10 OCTOBER 2024

BRIEF PAPER *Special Section on Analog Circuits and Their Application Technologies*

3D Parallel ReRAM Computation-in-Memory for Hyperdimensional Computing

Fuyuki KIHARA[†a)], **Chihiro MATSUI**[†], ***Nonmembers***, ***and*** **Ken TAKEUCHI**[†], ***Member***

SUMMARY In this work, we propose a 1T1R ReRAM CiM architecture for Hyperdimensional Computing (HDC). The number of Source Lines and Bit Lines is reduced by introducing memory cells that are connected in series, which is especially advantageous when using a 3D implementation. The results of CiM operations contain errors, but HDC is robust against them, so that even if the XNOR operation has an error of 25%, the inference accuracy remains above 90%.

key words: *Computation-in-Memory, ReRAM, chain-cell memory, hyperdimensional computing*

1. Introduction

Hyperdimensional Computing (HDC) is an emerging neuromorphic computing paradigm [1]–[5]. HDC has similar characteristics to neurons such as hyper-dimensionality, lower bit width, randomness, and robustness. The most important feature of HDC is that all data are represented in Hypervector (HV). Typically, a HV consists of 1 bit × 10,000 dimensions. Almost all operations are bitwise operations, therefore HDC is suitable for parallel processing and Computation-in-Memory (CiM) [5]–[11]. CiM is one of the many parallel processing methods. CiM performs simple calculations (e.g., multiply-accumulate (MAC) operation) simultaneously with the memory readout. For large-scale algorithms such as Machine Learning, the cost of data transfer is a major barrier, known as the von Neumann bottleneck, and CiM is seen as a promising way to solve this problem. In addition, using ReRAM, one of the emerging Non-Volatile Memories (NVMs), grants some advantages such as energy efficiency [9]–[13].

In this work, we propose a ReRAM CiM architecture suitable for *N*-gram Encoder in HDC.

2. HDC and Language Classification

Language Classification task classifies texts consisting of 26 letters of the alphabet and space into 21 European languages [4]. As shown in Fig. 1 (a), in the training phase, Prototype HVs (PHV) are made from texts written in the respective language by the Encode module and stored in Associative Memory (AM). In the inference phase, a Query HV (QHV) is made from the text written in a European language by the Encode module and taken similarity with each PHV. Figure 1 (b) shows the structure of an Encoding mod-

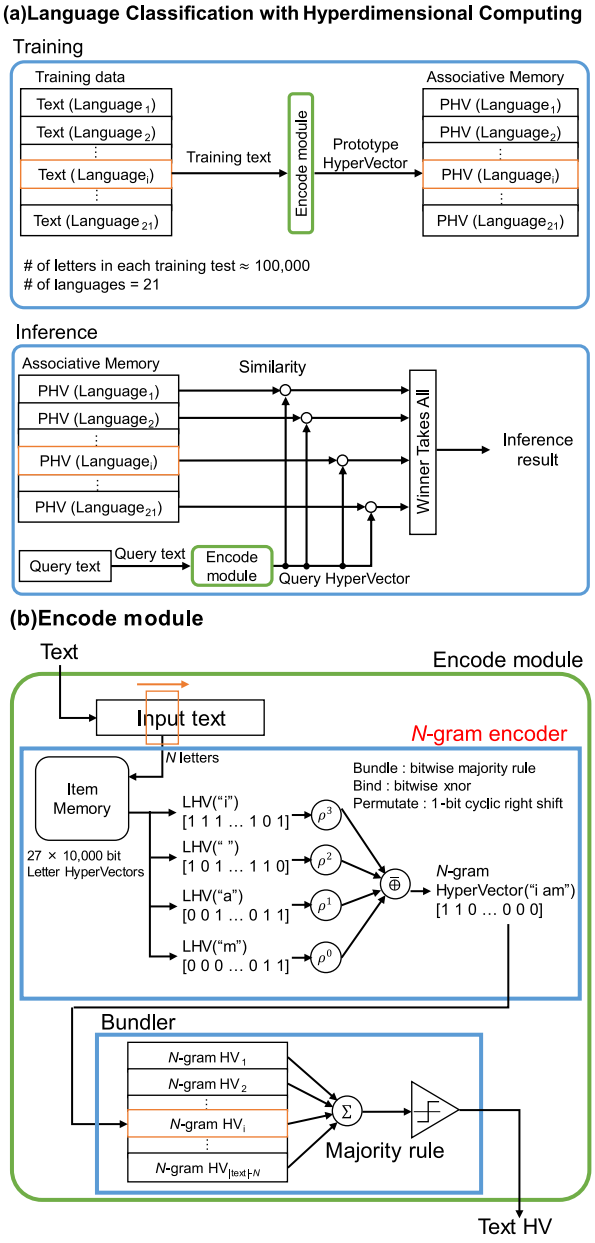


Fig. 1 Summary of Language Classification process. (a) Language Classification flow by Hyperdimensional Computing (HDC). (b) Structure of Encode module in (a).

Manuscript received November 2, 2023.
Manuscript revised February 26, 2024.
Manuscript publicized April 16, 2024.

[†]Department of Electrical Engineering and Information Systems, Graduate School of Engineering, The University of Tokyo, Tokyo, 113–8656 Japan.
a) E-mail: kihara@co-design.t.u-tokyo.ac.jp

DOI: 10.1587/transele.2023CTS0001

ule. Item Memory (IM) stores Letter HVs (LHV) that are independently assigned to each letter. All N-grams in the input text are converted into respective N-gram HV. LHVs of former letters are permutated by a permutation matrix (ρ) to keep series information. N-gram HV is obtained by taking the bitwise XNOR of all rows and permutated LHVs, and the Text HV is obtained by taking the bitwise major rule operation of all the N-gram HV.

3. Proposed ReRAM CiM

Figure 2 (a) shows the proposed ReRAM CiM array suitable for HDC. It can take the sum and XNOR by accumulating resistances of ReRAM cells when read out operation. The SUM operation is the number of ReRAM devices in the High Resistance State (HRS), and XNOR is calculated as HRS = 1, Low Resistance State (LRS) = 0. As shown in Fig. 2 (b), this block is arranged in a row and HVs are mapped to implement an N-gram encoder. Since the results of the SUM and XNOR operations have the relationship shown in Fig. 2 (c), the XNOR result can be calculated with this CiM array. Figure 3 (a) shows the ReRAM device and its HRS and LRS characteristics. ReRAM has a current distribution as shown in Fig. 3 (b).

Figure 4 shows the schematic and operations of proposed ReRAM CiM. In this architecture, one memory cell is composed of one FET and one ReRAM device connected in parallel. Several numbers of memory cells and selecting FETs are connected serially like NAND gate. An FET in a memory cell operates as a transmission gate and selects between an FET or a ReRAM device as the current path. In operations that use only one cell (e.g., R_2), the Word Line (WL) connected to the selected cell is at V_{OFF} and the other WLs are at V_{ON}, then the current flows through the ReRAM in the selected cell and the FETs in the unselected cells. When set/reset/read pulse voltages are applied to the Bit Line, almost the same voltages are applied to both ends of the selected cell. The CiM operation is shown in Fig. 4 (d). When the CiM operation, select multiple or all WLs instead of one WL. In this case, the current flows some ReRAM cells, then combined resistance can be measured.

This architecture reduces the number of Source Lines (SL) and Bit Lines (BL) [14], [15]. Then, each of these Lines can be manufactured thicker to suppress the resistance. SL and BL parasitic resistances cause IR drop and make device characteristics worse. In addition, cell size is expected to be smaller. According to [14], the chain cell structure reduces the area per cell from $8F^2$ to $4F^2$. We assume that a similar effect might be expected with ReRAM. Furthermore, in 3D integration, the construction of SL and BL on BEOL metallic layers as shown in Fig. 5 provides significant benefit [9], [15]. Implementation of the N-gram encoder by the ReRAM CiM is shown in Fig. 6. ReRAM CiM stores all raw and permutated LHVs as IM and outputs N-gram HV at the same time as read out. N WLs corresponding to each letter of an N-gram are activated, and bitwise XNOR of the LHVs is calculated, then N-gram HVs are obtained [4].

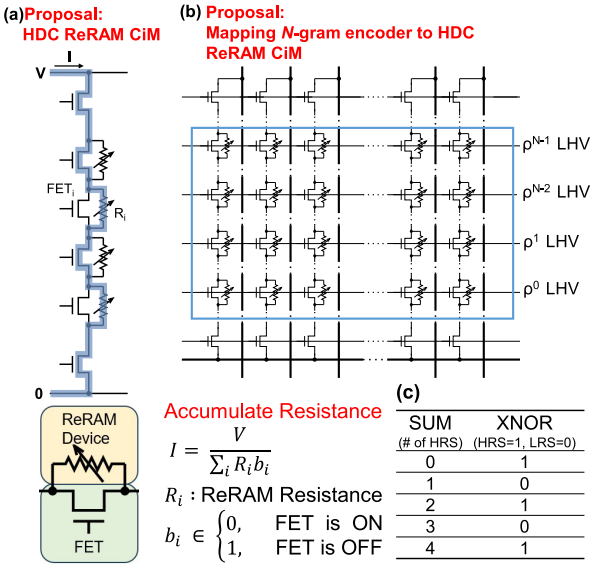


Fig. 2 (a) Schematic of proposed ReRAM CiM array. (b) Mapping N-gram encoder to HDC ReRAM CiM. (c) Relationship between bitwise SUM and XNOR (4 bit).

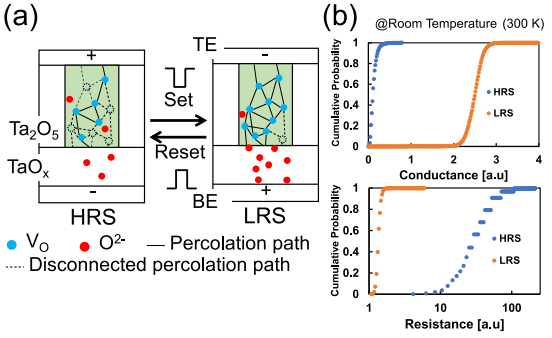


Fig. 3 (a) Switching mechanism of ReRAM [13]. (b) Conductance and Resistance distribution of ReRAM device (without FET).

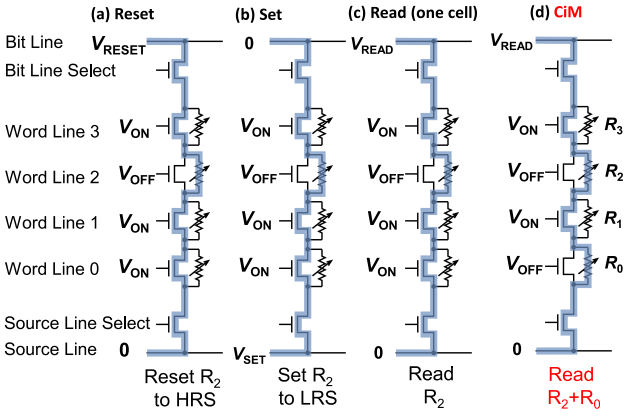


Fig. 4 Operations of ReRAM CiM array. (a) Reset. (b) Set. (c) Read (One cell). (d) Read (CiM).

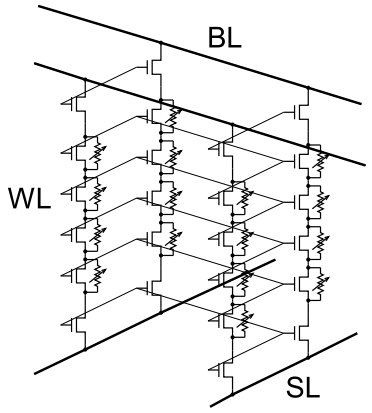


Fig. 5 Equivalent circuit of 3D integration.

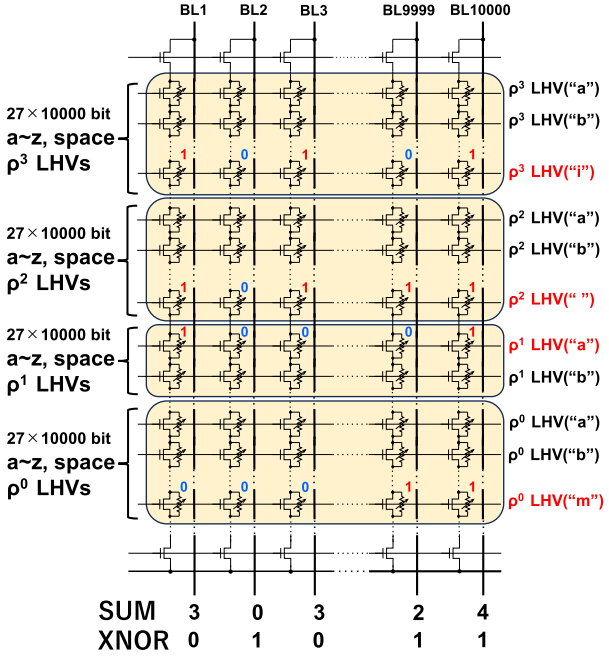


Fig. 6 Proposed N-gram encoder. 4-gram encoder mapped to ReRAM CiM array with 108 WLs. Operation example when Input text is "I am".

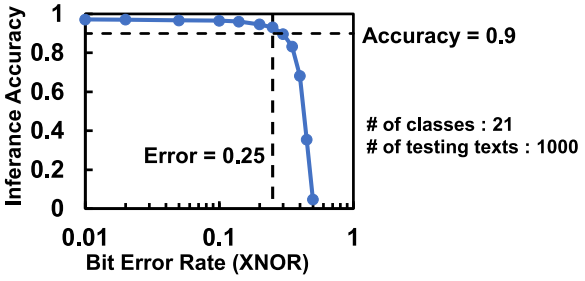


Fig. 7 Inference accuracy of error injection into N-gram encoder.

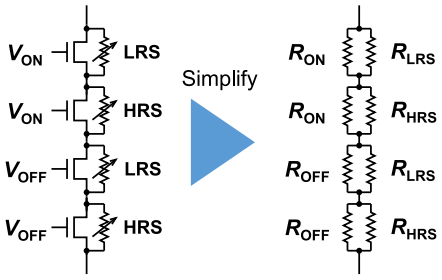
4. Evaluation

(i) Language Classification with Error

Figure 7 shows the evaluation result of Language Classification described in Sect. 2 with bit inversion error in XNOR operation of N-gram encoder in inference phase. For each language, there is one training text of about 100k characters and 1000 testing texts. 4-gram encoder with bit inversion error is used to generate PHV from the training text and QHV from the testing text. The inference accuracy remains higher than 0.9 for up to a bit error rate as high as 0.25.

(ii) Proposed ReRAM CiM

Figure 8 shows the simplified circuit that consists of fixed ohmic resisters used by the simulations. Resisters of resistance R_{ON} or R_{OFF} are replacement for the ON and OFF states of the FETs. Resistance of ReRAM devices, R_{LRS} and R_{HRS}, are chosen randomly from measured resistance of ReRAM (Fig. 3 (b)).



$R_{\mathrm{ON}} = 0.04 \times \overline{R_{\mathrm{LRS}}}$
R_{OFF} : Variable
$R_{\mathrm{LRS}} \in \{measured\ value\ of\ LRS\}$
$R_{\mathrm{HRS}} \in \{measured\ value\ of\ HRS\}$

Fig. 8 Simplified circuit for simulation.

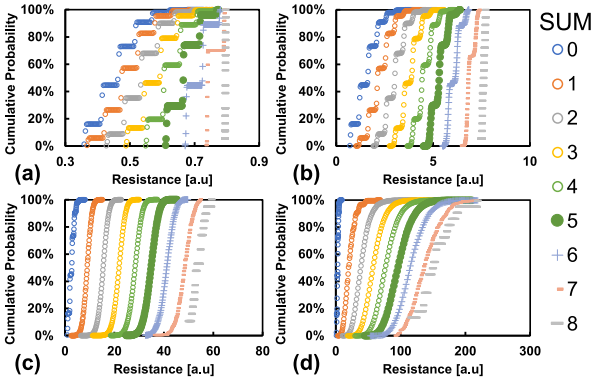


Fig. 9 Simulated combined resistance of CiM operation of 8-bit ReRAM array (for general purpose). R_{OFF} = (a) 0.1 (b) 1 (c) 10 (d) $100 \times \overline{R_{\mathrm{LRS}}}$.

Figure 9 shows simulated resistances of 8-bit ReRAM CiM. Overlaps are minimized when R_{OFF} equals $10 \times \overline{R_{\mathrm{LRS}}}$. The results of a 108-bit ReRAM CiM are shown in Fig. 10. To evaluate the behavior as an N-gram encoder ($N = 4$), in this simulation, the number of selected WLs is restricted to 4. A one shift in the result of the SUM operation means that the result of the XNOR operation is inverted. In this case, overlaps are minimized when R_{OFF} equals $1 \times \overline{R_{\mathrm{LRS}}}$. As shown in Fig. 10 (b), (c), by setting the thresholds of ADC (red line) appropriately, the read error and the error rate of the XNOR operation can be kept below 25% when R_{OFF} equals

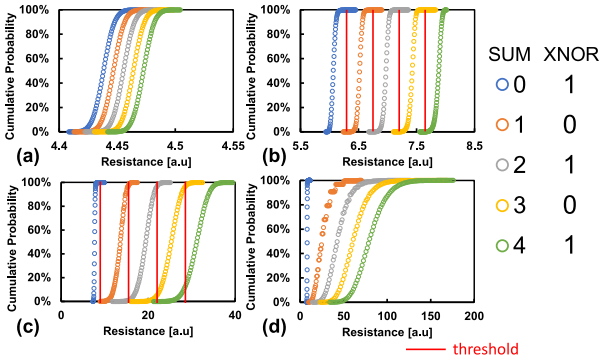


Fig. 10 Simulated combined resistance of CiM operation of 108-bit (27 letters × 4-gram) ReRAM array (for N-gram encoder). 4 WLs are selected. R_{OFF} = (a) 0.1 (b) 1 (c) 10 (d) $100 \times \overline{R_{LRS}}$. In (b) and (c), examples of thresholds (red line) where the read error is below 25% are included.

$1 \times \overline{R_{LRS}}$ or $10 \times \overline{R_{LRS}}$. These overlaps get worse because of change in the distribution of R_{HRS}, with increase in R_{OFF}. When R_{OFF} is not quite larger than R_{HRS}, the distribution of $R_{OFF}//R_{HRS}$ is suppressed. The change in $R_{OFF}//R_{HRS}$ when R_{HRS} changes by ΔR_{HRS} can be calculated as follows:

$$R_{OFF}//(R_{HRS} + \Delta R_{HRS}) - R_{OFF}//R_{HRS} = \frac{{R_{OFF}}^2 \Delta R_{HRS}}{(R_{OFF} + R_{HRS} + \Delta R_{HRS})(R_{OFF} + R_{HRS})}$$

By adjusting R_{OFF}, the error rate can be reduced.

5. Conclusion

This work proposes the ReRAM CiM architecture suitable for HDC. The size of memory cell block and IR drop in BL and SL can be reduced by introducing cell with FET and ReRAM connected in parallel. This CiM method includes errors in the calculation results, but Language Classification using HDC can maintain 90% inference accuracy even when the XNOR operation contains 25% errors. This result is not greater than the loss of inference accuracy due to other errors [4], [5]. Therefore, the reduction in inference accuracy due to the implementation of the proposed CiM into HDC can be tolerated. Reducing the R_{OFF} variation is important to reduce errors in the computation and a future challenge [16].

References

[1] P. Kanerva, "Hyperdimensional Computing: An introduction to computing in distributed representation with high-dimensional random vectors," Cognitive Computation, vol.1, no.2, pp.139–159, 2009. DOI: 10.1007/s12559-009-9009-8

[2] D. Kleyko, D.A. Rachkovskij, E. Osipov, and A. Rahimi, "A survey on hyperdimensional computing aka vector symbolic architectures, Part I: Models and data transformations," ACM Computing Surveys, vol.55, no.6, Article No.130, 2022. DOI: 10.1145/3538531

[3] E. Hassan, Y. Halawani, B. Mohammad, and H. Saleh, "Hyper-dimensional computing challenges and opportunities for AI applications," IEEE Access, vol.10, pp.97651–97664, 2022. DOI: 10.1109/ACCESS.2021.3059762

[4] C. Matsui, E. Kobayashi, N. Misawa, and K. Takeuchi, "Comprehensive analysis on error-robustness of FeFET computation-in-memory for hyperdimensional computing," Japanese Journal of Applied Physics (JJAP), vol.62, no.SC, pp.SC1053-1–SC1053-13, Feb. 2023.

[5] H. Li, T.F. Wu, A. Rahimi, K.-S. Li, M. Rusch, C.-H. Lin, J.-L. Hsu, M.M. Sabry, S.B. Eryilmaz, J. Sohn, W.-C. Chiu, M.-C. Chen, T.-T. Wu, J.-M. Shieh, W.-K. Yeh, J.M. Rabaey, S. Mitra, and H.-S.P. Wong, "Hyperdimensional computing with 3D VRRAM in-memory kernels: Device-architecture co-design for energy-efficient, error-resilient language recognition," 2016 IEEE International Electron Devices Meeting (IEDM), San Francisco, CA, USA, pp.16.1.1–16.1.4, 2016. DOI: 10.1109/IEDM.2016.7838428

[6] G. Karunaratne, M. Le Gallo, G. Cherubini, L. Benini, A. Rahimi, and A. Sebastian, "In-memory hyperdimensional computing," Nature Electronics, vol.3, pp.327–337, 2020. DOI: 10.1038/s41928-020-0410-3

[7] A. Kazemi, M.M. Sharifi, Z. Zou, M. Niemier, X.S. Hu, and M. Imani, "MIMHD: Accurate and efficient hyperdimensional inference using multi-bit in-memory computing," 2021 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED), Boston, MA, USA, pp.1–6, 2021.

[8] P.-K. Hsu and S. Yu, "In-memory 3D NAND flash hyperdimensional computing engine for energy-efficient SARS-CoV-2 genome sequencing," 2022 IEEE International Memory Workshop (IMW), Dresden, Germany, pp.1–4, 2022.

[9] T. Dubreuil, P. Amari, S. Barraud, J. Lacord, E. Esmanhotto, V. Meli, S. Martin, N. Castellani, B. Previtali, and F. Andrieu, "A novel 3D 1T1R RRAM architecture for memory-centric hyperdimensional computing," 2022 IEEE International Memory Workshop (IMW), Dresden, Germany, pp.1–4, 2022.

[10] T. Dubreuil, S. Barraud, B. Previtali, S. Martinie, J. Lacord, S. Martin, N. Castellani, A. Anotta, and F. Andrieu, "Fabrication of low-power RRAM for stateful hyperdimensional computing," 2023 International VLSI Symposium on Technology, Systems and Applications (VLSI-TSA/VLSI-DAT), HsinChu, Taiwan, pp.1–2, 2023. DOI: 10.1109/VLSI-TSA/VLSI-DAT57221.2023.10134182

[11] S. Mittal, "A survey of ReRAM-based architectures for processing-in-memory and neural networks," Machine Learning and Knowledge Extraction, vol.1, no.1, pp.75–114, 2019. DOI: 10.3390/make1010005

[12] Y. Chen, "ReRAM: History, status, and future," IEEE Trans. Electron Devices, vol.67, no.4, pp.1420–1433, April 2020. DOI: 10.1109/TED.2019.2961505

[13] A. Yamada, N. Misawa, C. Matsui, and K. Takeuchi, "ReRAM CiM fluctuation pattern classification by CNN trained on artificially created dataset," 2023 IEEE International Reliability Physics Symposium (IRPS), Monterey, CA, USA, pp.1–6, 2023.

[14] D. Takashima, I. Kunishima, M. Noguchi, and S. Takagi, "High-density chain ferroelectric random-access memory (CFRAM)," Symposium on VLSI Circuit, pp.83–84, 1997.

[15] M. Kinoshita, Y. Sasago, H. Minemura, Y. Anzai, M. Tai, Y. Fujisaki, S. Kusaba, T. Morimoto, T. Takahama, T. Mine, A. Shima, Y. Yonamoto, and T. Kobayashi, "Scalable 3-D vertical chain-cell-type phase-change memory with 4F2 poly-Si diodes," 2012 Symposium on VLSI Technology, pp.35–36, 2012.

[16] K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand, "Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits," Proc. IEEE, vol.91, no.2, pp.305–327, Feb. 2003. DOI: 10.1109/JPROC.2002.808156