Crossbar-Compatible Stateful Logic Using Phase Change Memory

Barak Hoffer

Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering, Technion

Nicolás Wainstein

Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering, Technion

Christopher M. Neumann

Stanford University

Eric Pop

Stanford University

Eilam Yalon

Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering, Technion

Shahar Kvatinsky (✉ shahar@ee.technion.ac.il)

Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering, Technion

Research Article

Keywords: digital, memory, logic

Posted Date: November 16th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1061047/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Crossbar-Compatible Stateful Logic using Phase Change Memory

Barak Hoffer¹, Nicolás Wainstein¹, Christopher M. Neumann², Eric Pop², Eilam Yalon¹, and Shahar Kvatinsky^{1,*}

¹Andrew and Erna Viterbi Faculty of Electrical and Computer Engineering, Technion, Haifa 3200003, Israel

²Department of Electrical Engineering, Stanford University, Stanford, CA 94305, USA

*Corresponding author. E-mail: shahar@ee.technion.ac.il

ABSTRACT

Stateful logic is a digital processing-in-memory technique that could address von Neumann memory bottleneck challenges while maintaining backward compatibility with von Neumann architectures. In stateful logic, memory cells are used to perform the logic operations without reading or moving any data outside the memory array. This has been previously demonstrated using several resistive memory types, but not with commercially available phase-change memory (PCM). Here we present the first implementation of stateful logic using PCM. We experimentally demonstrate four logic gate types (NOR, IMPLY, OR, NIMP) using commonly used PCM materials and crossbar-compatible structures. Our stateful logic gates form a functionally complete set, which enables sequential execution of any logic function within the memory and paves the way to PCM-based digital processing-in-memory systems.

Introduction

For the last 75 years, computers have been typically designed in the von Neumann architecture, which separates the memory from the processing units (Fig. 1a). While their programming model is simple, incessant data movement limits system performance because memory access time is often substantially longer than the computing time. This bottleneck has worsened over the years since CPU speed has improved more than memory speed and bandwidth (the so-called ‘memory wall’)¹. One attractive approach to deal with this problem is processing-in-memory (PIM), which suggests adding computation capabilities to the memory. PIM reduces the need for costly (in terms of processing-speed, bandwidth, and energy) chip-to-chip transfers, thus yielding higher performance and energy efficiency².

Stateful logic^{3,4} is a processing-in-memory technique based on memristive memory technologies (Fig. 1b), *e.g.*, resistive random-access memory (RRAM) or conductive-bridge RAM (CBRAM). In stateful logic, the stored resistive data is used as input and the result is written to an output memory cell without reading the input cells beforehand or moving any data outside the memory array. Stateful logic enables PIM architectures such as the *memristive memory processing unit* (mMPU, Fig. 1b)⁵ that offer massive intrinsic parallelism, high-performance, and energy-efficient processing, while maintaining backward compatibility with von Neumann architectures.

Prior studies on stateful logic mostly focused on bipolar RRAM devices^{6–13}, which still suffer from reliability and variability issues^{14,15} and are unavailable commercially in large scale. Phase-change memory (PCM) is a more mature resistive technology that offers fast operation speed, low power, good reliability and high density integration, while being already used commercially^{16–19}, *e.g.*, in the Intel Optane technology²⁰. However, previous studies of computation using PCM mainly focused on analog neuromorphic computation^{21–27} or non-stateful binary logic operations^{28–31}. Since the switching mechanism of PCM is completely different than RRAM, to achieve stateful logic using PCM devices, new circuit topologies and voltage schemes are needed. The few stateful logic methods previously proposed for PCM³² have limited results and a cumbersome voltage scheme.

In this article, we present and experimentally demonstrate a novel method to perform stateful logic operations using PCM. We demonstrate four different logic gates (NOR, IMPLY, OR, and NIMP) with robust and repeatable results. Our logical set is functionally complete, enabling sequential execution of any logic function in-memory. The gates are compatible with the memory crossbar structure and can be applied in parallel on multiple rows. Additionally, previous applications suggested for stateful logic using RRAM are applicable to our PCM-based method (Supplementary Text 1).

Results

Phase-change memory devices.

Phase-change memory exploits the behavior of certain chalcogenide materials that can be switched rapidly and repeatedly between amorphous and crystalline phases. These materials are typically compounds of Germanium, Antimony, and Tellurium ($\text{Ge}_x\text{Sb}_y\text{Te}_z$, GST). The amorphous phase presents a high electrical resistivity while the crystalline phase exhibits low resistivity. A PCM device consists of a certain volume of this phase change material sandwiched between two electrodes (Fig. 2). Applying pulses to a PCM device results in Joule heating, which alters the phase (state) of the material. A reset pulse is used to melt a significant portion of the phase change material. When the pulse is stopped abruptly, the molten material quenches into the amorphous phase. Following the reset pulse, the device will be in a high resistive state (HRS). When a slower set pulse, with an amplitude above a threshold voltage (V_{th})³³, is applied to a PCM device in the HRS, a part of the amorphous region crystallizes. After the SET pulse, the device will be in a low resistive state (LRS). The resistance state achieved after the application of reset or set pulses can be read by biasing the device with a small read voltage that does not change the phase configuration.

Our setup (Fig. 3) includes three PCM cells with a shared bottom electrode, and enables programming and reading each cell as well as performing the logic operations. A write-verify scheme is used to probe the maximum cycle count of a single device and characterize its switching behavior. The current, voltage, and power required to set and reset the devices are depicted in Fig. 4a-c. Our endurance test shows that a device can maintain a 10x resistance window for almost 10^4 cycles, with some degradation in the resistance distribution after several hundred cycles (Fig. 4d). The current and voltage waveforms across the PCM cell during a typical set operation are shown in Fig. 4e. Finally, the current-voltage (I-V) transition of the device from the amorphous state to the crystalline state is shown in Fig. 4f.

Phase-change memory stateful logic.

The proposed PCM-based stateful logic gate consists of three PCM cells where two cells serve as inputs and the third device as the output (Fig. 3). The output cell may also serve as an additional input at the cost of losing its stored data, *i.e.*, a destructive operation. A grounded fixed resistor (10 k Ω in our configuration) can be connected to the shared node as well (similar to material implication in RRAM⁴). A logic operation is achieved by applying voltage pulses to the top electrode (TE) of the input cells, causing a conditional output switching, depending on the resistive states of the inputs. The switching mechanism is based on the Ovonic threshold switching phenomenon³³ that occurs if the voltage across the output cell is above its threshold voltage, V_{th} , followed by the crystallization of the output cell. Furthermore, this design principle is compatible with the crossbar memory structure commonly used for PCM^{27,34}. Here, we propose four different logic functions (*i.e.*, NOR, IMPLY, NOR, NIMP) based on this principle. Table 1 summarizes the voltages and configurations used to realize the different logic gates.

A NOR operation is realized by initializing the output cell to the HRS and grounding the shared bottom electrode (BE) using the fixed resistor. Applying $\sim \frac{1}{2}V_{th}$ to TE_{IN1} , TE_{IN2} , and applying $\sim V_{th}$ to TE_{OUT} causes the voltage across OUT to be approximately V_{th} only if both inputs are in HRS. Additionally, the state of the inputs remains unchanged because the maximum voltage across each input cell is $\sim \frac{1}{2}V_{th}$. Similarly, a destructive implication (IMPLY) operation is realized. Here, TE_{IN2} is kept floating, OUT serves as an input as well, and the voltage across OUT is $\sim V_{th}$ only if IN1 is in HRS. An OR operation is realized by keeping the shared BE floating, grounding TE_{IN1} , TE_{IN2} , and applying V_{th} to TE_{OUT} . This causes the voltage across the output cell to be approximately V_{th} only if at least one input is in the LRS. Similarly, a not implication (NIMP) operation is realized by keeping the shared BE floating, grounding TE_{OUT} , and applying $\sim V_{th}$ to TE_{IN1} and $\sim \frac{1}{3}V_{th}$ to TE_{IN2} . Here, the voltage across the output cell is approximately V_{th} only if IN1 is in LRS and IN2 is in HRS. Note that an XOR gate can be performed in two cycles by running the NIMP operation twice on the same output with alternating inputs¹³. See Supplementary Fig. 1 for more information.

Experimental results of the proposed logic gates.

We measured the functionality and robustness of the proposed gates on the fabricated devices. In each test cycle, we examine the four input combinations for the tested gate. Each experiment includes: a) a write-verify procedure to initialize the inputs and output to the desired states, b) applying the voltage pulses required to evaluate the logic function, and c) reading cycles to examine the output result and to verify the stability of the inputs (see Supplementary Fig. 3). A stateful operation is evaluated not only by the correct logical result, but also the stability of the inputs. This is not always trivial, as reported by previous RRAM works^{12,13,15,35}. Results of 50 cycles of: (a) NOR, (b) IMPLY, (c) OR, and (d) NIMP logic gates are shown in Fig. 5. The results show successful logic operation for all cycles on all gates. Additionally, the inputs remain stable, without any meaningful change in their resistance.

The applied voltage pulses to implement the gates and the measured voltage at the shared BE, marked as BE_{ALL} , for each input case are depicted in Fig. 6. In the NOR and IMPLY tests, if one of the inputs is in LRS, the voltage at BE_{ALL} follows the constant voltage dictated by the inputs, thus keeping the voltage across the output below the set threshold. Otherwise, the voltage at BE_{ALL} remains at 0 V, and the output is switched once the voltage on its TE is above V_{th} . In the OR and NIMP tests,

the voltage at BE_{ALL} follows a constant trend for all non-switching cases, keeping the voltage across the output cell below V_{th} . In the cases where the output is switched, a meaningful change in the voltage at BE_{ALL} is noticeable, caused by the resistance change of the output.

Discussion

An increasing number of applications from high-performance computing (HPC) to databases, data analytics and deep neural networks require higher memory capacity to meet the needs of workloads with large data sets. DRAM scaling has slowed down in the last years, and it has become a Herculean task to improve its capabilities further^{36,37}. Thus, new technologies, such as RRAM, CBRAM, and PCM, are being explored³⁸. These technologies offer increased memory capacity and add non-volatility, enabling persistent memory. These types of persistent memories are usually referred to as Storage Class Memory (SCM)^{14,17}, as they combine both storage and memory characteristics. Although RRAM is considered an attractive memory technology for SCM, to date it is mostly used for small embedded nonvolatile memory, and has not been commercialized yet in large scale (e.g., in gigabytes scale). The Intel Optane Persistent Memory is already commercially available in dual in-line memory modules (DIMM) form with up to 512GB capacity, larger than DRAM DIMM, that typically range from 4GB to 128GB³⁹.

With the PCM-based Intel Optane, applications stand to benefit from the availability of large-capacity memory, but the performance will still be limited by the incessant data movement between the CPU and memory. To tackle this issue, we propose adding computation capabilities to PCM technology, inspired by previously proposed stateful logic operations for RRAM. We experimentally demonstrate a novel method to perform four stateful logic gates using PCM (NOR, IMPLY, OR, and NIMP). The measured results show correct and robust logic operation. The proposed gates are crossbar compatible, functionally complete and can be executed simultaneously on multiple rows, paving the path towards PCM-based digital processing-in-memory architectures.

Methods

Device fabrication.

PCM devices are fabricated⁴⁰, starting with the evaporation and etching of tungsten (W) to form the bottom electrodes (BEs). Next, we deposit SiO_x with plasma-enhanced chemical vapor deposition (PECVD) and pattern the confined vias using e-beam lithography. In-situ Ar sputtering is used to make sure the top of the BE isn't oxidized. Then, sputtering and liftoff are used to pattern the $Ge_2Sb_2Te_5$ (GST) layer with *in situ* TiN capping, and the final TiN/Pt top electrodes (TEs) and contact pads.

Electrical measurements.

The measurement setup is shown in Fig. 3; it includes three PCM cells and enables programming and reading each cell, as well as performing the logic operations. Electrical measurements are performed on-wafer using a Keysight B1500A with four B1530 WGFMU channels and a Keysight MSOX3104T oscilloscope. For set and reset, we use 30/500/500 ns and 30/50/30 ns rise/width/fall pulses, respectively. The resistance is measured with a 0.2 V read pulse.

Experimental demonstration.

We apply the voltage pulses in two stages or set a relatively long rise time to deal with the long RC delay in the shared node, caused by the large parasitic capacitance of the pads and probes. This long rise time is required since otherwise the RC delay causes an unwanted voltage state at the shared node, which might cause unwanted switching. In an integrated setup, the RC delay is considerably smaller, which will eliminate the need for the long rise time. Additionally, to reduce unintended switching of the inputs and/or output we chose to use exactly V_{th} with a longer pulse and not a higher voltage. See Supplementary Text 2 for more details. In the NOR gate test, we apply 0.6 V for 3 μs on all TEs, then we increase the voltage on TE_{OUT} to 1.2 V for 1 μs , while keeping the voltage on TE_{IN1} and TE_{IN2} at 0.6 V. We apply the same voltage scheme for the IMPLY gate test, but TE_{IN2} is kept floating. In the OR gate test, we keep the shared BE floating, ground TE_{IN1} and TE_{IN2} , and apply 1.2 V on TE_{OUT} with rise time of 70 μs and pulse length of 1 μs . Similarly, for the NIMP gate we keep the shared BE floating, ground TE_{OUT} and apply 1.2 V on TE_{IN1} and 0.35 V on TE_{IN2} with rise time of 70 μs and pulse length of 1 μs .

References

1. Hennessy, J. L. & Patterson, D. A. *Computer Architecture: a Quantitative Approach* (Morgan Kaufmann, 2017), 6 edn.
2. Mutlu, O., Ghose, S., Gómez-Luna, J. & Ausavarungnirun, R. A modern primer on processing in memory (2020). <https://arxiv.org/abs/2012.03112>.
3. Kvatinsky, S. *et al.* MAGIC—memristor-aided logic. *IEEE Transactions on Circuits Syst. II: Express Briefs* **61**, 895–899, DOI: <https://doi.org/10.1109/TCSII.2014.2357292> (2014).

4. Borghetti, J. *et al.* Memristive switches enable stateful logic operations via material implication. *Nature* **464**, 873–876, DOI: <https://doi.org/10.1038/nature08940> (2010).
5. Haj-Ali, A., Ben-Hur, R., Wald, N., Ronen, R. & Kvatinsky, S. Not in name alone: A memristive memory processing unit for real in-memory processing. *IEEE Micro* **38**, 13–21, DOI: <https://doi.org/10.1109/MM.2018.053631137> (2018).
6. Jang, B. C. *et al.* Zero-static-power nonvolatile logic-in-memory circuits for flexible electronics. *Nano Res.* **10**, 2459–2470, DOI: <https://doi.org/10.1007/s12274-017-1449-y> (2017).
7. Bae, H. *et al.* Functional Circuitry on Commercial Fabric via Textile-Compatible Nanoscale Film Coating Process for Fibertronics. *Nano Lett.* **17**, 6443–6452, DOI: <https://doi.org/10.1021/acs.nanolett.7b03435> (2017).
8. Jang, B. C. *et al.* Memristive Logic-in-Memory Integrated Circuits for Energy-Efficient Flexible Electronics. *Adv. Funct. Mater.* **28**, 1704725, DOI: <https://doi.org/10.1002/adfm.201704725> (2018).
9. Sun, Z., Ambrosi, E., Bricalli, A. & Ielmini, D. Logic computing with stateful neural networks of resistive switches. *Adv. Mater.* **30**, 1802554, DOI: <https://doi.org/10.1002/adma.201802554> (2018).
10. Shen, W. *et al.* Stateful Logic Operations in One-Transistor-One-Resistor Resistive Random Access Memory Array. *IEEE Electron Device Lett.* **40**, 1538–1541, DOI: <https://doi.org/10.1109/led.2019.2931947> (2019).
11. Xu, N. *et al.* A stateful logic family based on a new logic primitive circuit composed of two antiparallel bipolar memristors. *Adv. Intell. Syst.* **2**, 1900082, DOI: <https://doi.org/10.1002/aisy.201900082> (2020).
12. Kim, Y. S. *et al.* Stateful in-memory logic system and its practical implementation in a taox-based bipolar-type memristive crossbar array. *Adv. Intell. Syst.* **2**, 1900156, DOI: <https://doi.org/10.1002/aisy.201900156> (2020).
13. Hoffer, B., Rana, V., Menzel, S., Waser, R. & Kvatinsky, S. Experimental Demonstration of Memristor-Aided Logic (MAGIC) Using Valence Change Memory (VCM). *IEEE Transactions on Electron Devices* **67**, 3115–3122, DOI: <https://doi.org/10.1109/ted.2020.3001247> (2020).
14. Chen, Y. ReRAM: History, Status, and Future. *IEEE Transactions on Electron Devices* **67**, 1420–1433, DOI: <https://doi.org/10.1109/TED.2019.2961505> (2020).
15. In, J. H. *et al.* A universal error correction method for memristive stateful logic devices for practical near-memory computing. *Adv. Intell. Syst.* **2**, 2000081, DOI: <https://doi.org/10.1002/aisy.202000081> (2020).
16. Cheng, H. Y. *et al.* Novel fast-switching and high-data retention phase-change memory based on new Ga-Sb-Ge material. *IEDM 3.5.1–3.5.4*, DOI: <https://doi.org/10.1109/IEDM.2015.7409620> (2015).
17. Fong, S. W., Neumann, C. M. & Wong, H.-S. P. Phase-Change Memory - Towards a Storage-Class Memory. *IEEE Transactions on Electron Devices* **64**, 4374–4385, DOI: <https://doi.org/10.1109/TED.2017.2746342> (2017).
18. Cheng, H. Y. *et al.* Si incorporation into assege chalcogenides for high thermal stability, high endurance and extremely low V_{th} drift 3d stackable cross-point memory : Ibm/macronix pcam joint project. In *IEEE Symp. on VLSI Tech.*, 1–2, DOI: <https://doi.org/10.1109/VLSITechnology18217.2020.9265039> (2020).
19. Gong, N. *et al.* A no-verification multi-level-cell (mlc) operation in cross-point ots-pcm : Ibm/macronix phase change memory joint project. In *IEEE Symposium on VLSI Technology*, 1–2, DOI: <https://doi.org/10.1109/VLSITechnology18217.2020.9265020> (IEEE, 2020).
20. Choe, J. Intel 3D XPoint Memory Die Removed from Intel Optane™ PCM <https://www.techinsights.com/blog/intel-3d-xpoint-memory-die-removed-intel-optanetm-pcm-phase-change-memory>. *Tech Insights* (2017).
21. Suri, M. *et al.* Phase change memory as synapse for ultra-dense neuromorphic systems: Application to complex visual pattern extraction. In *International Electron Devices Meeting*, 4.4.1–4.4.4, DOI: <https://doi.org/10.1109/IEDM.2011.6131488> (IEEE, 2011).
22. Burr, G. W. *et al.* Experimental Demonstration and Tolerancing of a Large-Scale Neural Network (165 000 Synapses) Using Phase-Change Memory as the Synaptic Weight Element. *IEEE Transactions on Electron Devices* **62**, 3498–3507, DOI: <https://doi.org/10.1109/TED.2015.2439635> (2015).
23. Sebastian, A. *et al.* Temporal correlation detection using computational phase-change memory. *Nat. Commun.* **8**, 1–10, DOI: <https://doi.org/10.1038/s41467-017-01481-9> (2017).
24. Boybat, I. *et al.* Neuromorphic computing with multi-memristive synapses. *Nat. Commun.* **9**, 1–12, DOI: <https://doi.org/10.1038/s41467-018-04933-y> (2018).
25. Joshi, V. *et al.* Accurate deep neural network inference using computational phase-change memory. *Nat. Commun.* **11**, 1–13, DOI: <https://doi.org/10.1038/s41467-020-16108-9> (2020).

26. Nandakumar, S. R. *et al.* Mixed-Precision Deep Learning Based on Computational Memory. *Front. Neurosci.* **14**, 406, DOI: <https://doi.org/10.3389/fnins.2020.00406> (2020).
27. Sebastian, A., Gallo, M. L. & Eleftheriou, E. Computational phase-change memory: beyond von Neumann computing. *J. Phys. D: Appl. Phys.* **52**, 443002, DOI: <https://doi.org/10.1088/1361-6463/ab37b6> (2019).
28. Li, Y. *et al.* Nonvolatile "AND," "OR," and "NOT" Boolean logic gates based on phase-change memory. *J. Appl. Phys.* **114**, 234503, DOI: <https://doi.org/10.1063/1.4852995> (2013).
29. Lu, B., Cheng, X., Feng, J., Guan, X. & Miao, X. Logic gates realized by nonvolatile GeTe/Sb₂Te₃ super lattice phase-change memory with a magnetic field input. *Appl. Phys. Lett.* **109**, 023506, DOI: <https://doi.org/10.1063/1.4958708> (2016).
30. Loke, D. *et al.* Ultrafast phase-change logic device driven by melting processes. *Proc. Natl. Acad. Sci.* **111**, 13272–13277, DOI: <https://doi.org/10.1073/pnas.1407633111> (2014).
31. Giannopoulos, I. *et al.* In-memory database query. *Adv. Intell. Syst.* **2**, 2000141, DOI: <https://doi.org/10.1002/aisy.202000141> (2020).
32. Cassinerio, M., Ciocchini, N. & Ielmini, D. Logic Computation in Phase Change Materials by Threshold and Memory Switching. *Adv. Mater.* **25**, 5975–5980, DOI: <https://doi.org/10.1002/adma.201301940> (2013).
33. Ovshinsky, S. R. Reversible electrical switching phenomena in disordered structures. *Phys. Rev. Lett.* **21**, 1450–1453, DOI: <https://doi.org/10.1103/PhysRevLett.21.1450> (1968).
34. Ielmini, D. & Lacaíta, A. L. Phase change materials in non-volatile storage. *Mater. Today* **14**, 600 – 607, DOI: [https://doi.org/10.1016/S1369-7021\(11\)70301-7](https://doi.org/10.1016/S1369-7021(11)70301-7) (2011).
35. Siemon, A., Wouters, D., Hamdioui, S. & Menzel, S. Memristive device modeling and circuit design exploration for computation-in-memory. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, 1–5, DOI: <https://doi.org/10.1109/ISCAS.2019.8702600> (IEEE, 2019).
36. Mandelman, J. A. *et al.* Challenges and future directions for the scaling of dynamic random-access memory (dram). *IBM J. Res. Dev.* **46**, 187–212, DOI: <https://doi.org/10.1147/rd.462.0187> (2002).
37. Shiratake, S. Scaling and performance challenges of future dram. In *IEEE International Memory Workshop (IMW)*, 1–3, DOI: <https://doi.org/10.1109/IMW48823.2020.9108122> (2020).
38. Yu, S. & Chen, P. Emerging memory technologies: Recent trends and prospects. *IEEE Solid-State Circuits Mag.* **8**, 43–56, DOI: <https://doi.org/10.1109/MSSC.2016.2546199> (2016).
39. Fazio, A. Advanced technology and systems of cross point memory. In *IEEE International Electron Devices Meeting (IEDM)*, 24.1.1–24.1.4, DOI: <https://doi.org/10.1109/IEDM13553.2020.9371976> (2020).
40. Neumann, C. M., Wong, H.-S. P., Pop, E. & Saraswat, K. The effect of interfaces on phase change memory switching. <http://purl.stanford.edu/rq703nq8223> (2019).

Acknowledgements

Fabrication was performed at the Stanford Nanofabrication Facility (SNF) and the Technion Micro-Nano Fabrication Unit (MNFU). This work was supported in part by the European Research Council under the European Union's Horizon 2020 Research and Innovation Programme under Agreement 757259 and in part by the Israel Science Foundation under Grant 1514/17.

Author contributions statement

B.H., N.W., E.Y. and S.K. conceived this work. B.H. designed the experiments. C.M.N., E.P., N.W. and E.Y. designed and performed the device fabrication. B.H. conducted the electrical measurements. All authors contributed to the discussion and analysis of results. B.H. and S.K. wrote the manuscript with input from all coauthors.

Additional information

Competing interests The authors declare no competing interests.

Figure 1. Structure of (a) a von-Neumann architecture and (b) the memristive memory processing unit (mMPU) architecture. In addition to standard memory operations, the mMPU can run logic operations in-memory.

Figure 2. Confined PCM cell used in this work. (a) Cross-section schematic, (b) Cross-section scanning electron microscopy (SEM), (c)-(d) 3D cartoon of the crystalline and amorphous states. Evaporation and etching of tungsten (W) is used to form the bottom electrode (BE). Sputtering and liftoff are used to pattern the Ge₂Sb₂Te₅ (GST) layer, the TiN/Pt top electrode (TE) and contact pads. GST is patterned into a confined area with diameter $D \sim 125$ nm.

Figure 3. Experimental setup. (a) Schematic and (b) optical top-view. Three cells are connected to a shared bottom electrode. Three waveform generators are connected to the top electrode of each cell. The bottom electrode can be switched between: floating, grounded, and grounded using a 10 kΩ resistor.

Figure 4. PCM device characteristics. DC read resistance during set and reset versus (a) programming current, (b) applied voltage, and (c) power. (d) Endurance data for a representative PCM device. The device was cycled using a write-verify scheme. The voltages for set and reset are 1.2 V and 3.0 V, respectively. (e) Current and voltage across the PCM cell during set operation. Threshold switching occurs within ~ 100 ns and the voltage is kept high for the remaining time to complete the crystallization process. (f) I-V transition of the device from the amorphous state (HRS) to the crystalline state (LRS) showing its threshold switching voltage, $V_{th} = 1.2$ V. The I-V was measured using a triangular voltage ramp with 1 μ s rise and fall times.

Figure 5. Experimental results of the proposed logic gates. 50 cycles of the (a) NOR, (b) IMPLY, (c) OR, and (d) NIMP gates. The x-axis is the operation or read cell, the y-axis is the measured resistance plotted as a scatter and a median box. For the IMPLY gate, OUT is used also as IN2. All cycles of all the gates show correct logic operation and exhibit input stability.

Figure 6. Voltages applied to the TE of the input/output cells to evaluate the (a) NOR, (b) IMPLY, (c) OR, and (d) NIMP gates. The measured voltage on the BE shared node is plotted for each input case. (a) NOR gate; if at least one of the inputs is in LRS, the voltage at the shared node follows the constant voltage dictated by the inputs, thus keeping the voltage across OUT below the set threshold. (b) IMPLY gate; the voltage at the shared node changes according to the resistive state of the inputs. (c) OR gate; the voltage across OUT is higher than the set threshold if at least one input is in LRS, causing a switch event noticeable by the voltage change at the shared node. (d) NIMP gate; the voltage across OUT is higher than the set threshold if IN1 is in LRS and IN2 is in HRS, causing a switch event noticeable by the voltage change at the shared node.

Gate	Voltage/Configuration			
	TE _{IN1}	TE _{IN2}	TE _{OUT}	BE _{ALL}
NOR	$\sim \frac{1}{2}V_{th}$	$\sim \frac{1}{2}V_{th}$	$\sim V_{th}$	R
IMPLY	$\sim \frac{1}{2}V_{th}$	F	$\sim V_{th}$	R
OR	0 V	0 V	$\sim V_{th}$	F
NIMP	$\sim V_{th}$	$\sim \frac{1}{3}V_{th}$	0 V	F

Table 1. Summary of the applied voltages and configurations at each node to realize the logic gates. 'R' marks connection to the grounded fixed resistor, 'F' marks a node left floating.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [supp.pdf](#)
- [suppvideo.mp4](#)