HotReRAM: A Performance-Power-Thermal Simulation Framework for ReRAM based Caches

Shounak Chakraborty[§], *Senior Member, IEEE*, Thanasin Bunnam[§], *Member, IEEE*, Jedsada Arunruerk, Sukarn Agarwal, *Member, IEEE*, Shengqi Yu, *Member, IEEE*, Rishad Shafik, *Senior Member, IEEE*, Magnus Själander, *Senior Member, IEEE*

***Abstract*—This paper proposes a comprehensive thermal modeling and simulation framework, *HotReRAM*, for ReRAM-based caches that models cache temperature at the smallest possible granularity level by exploiting power traces, whereas the thermal model is verified with the VTEAM circuit model of memristor at the same granularity level. As the existing diversity in access counts across the cache area generates distinct power profiles within a single cache bank, the operational cache temperature also varies. *HotReRAM* models power at a fine grain level and generates temperature traces for different cache regions along with detailed analyses of the thermal stability, retention time and write latency. Temporal and spatial modeling of these important characteristics of ReRAM by combining *HotReRAM* with an existing full system simulator and a power simulator for ReRAM will enable the designers and architects to analyse uneven cache characteristics within a single cache bank and to take necessary measures in mitigating the thermal induced issues while designing ReRAM caches. Our simulation results for an 8MiB ReRAM cache also show that the spatial thermal variance can have a maximum magnitude of $7\,^{\circ}\mathrm{C}$ for a single cache bank, whereas the temporal thermal variance is noticed more than $40\,^{\circ}\mathrm{C}$. Such temperature variances vary retention time with a standard deviation of 3.9 to 10.2 for a set of benchmark applications, where the write latency can increase up to 14.5%.**

***Index Terms*—ReRAM, Memristor, Thermal Characteristics, Retention Time, Write Latency, Thermal Stability, Read/Write Energy, Simulation, VTEAM**

I. Introduction

HIGH leakage power consumption along with the low cell density of the conventional MOSFET based SRAM steers architects and researchers towards exploration of the alternative memory technologies for the caches. Among a large set of emerging memory technologies, latest Non-Volatile Memories (NVMs), such as Spin-Transfer Torque RAM (STT-RAM) [1], Resistive RAM (ReRAM) [2], Phase Change Memory (PCM) [3], flash memory [4], etc. are the most promising alternatives. Out of all these NVMs, ReRAM or memristor has the potential to replace SRAM due to its low leakage power [5], high endurance [6], long retention [7], comparable access times [6], [7] and better compatibility with the conventional transistors [8]. Such ReRAM devices can also be employed to design logic circuitry, as their resistance varies with previous as well as present supply voltage [9]. But, compared to SRAM, the latest version of ReRAM has a higher write energy that can significantly increase circuit temperature, leading to performance aggravation and permanent circuit failure. Hence, to tackle memory intensive workloads, ReRAM caches must be designed by comprehensively analyzing its underlying temperature induced characteristics.

Prior arts illustrated the effects of higher temperature in the ReRAM devices [10]–[13]. Basically, higher temperature in ReRAM devices reduces the conductance of the ReRAM cell, which drastically trims the functional correctness of the circuitry [13]. As the storage in ReRAM is driven by the conductance capability of the device, higher temperature can incur severe disturbance on the stored data. Additionally, higher temperature increases write latency of the ReRAM and can significantly affect the functional correctness of the write operation [12]. Moreover, most of the modern memory intensive workloads can increase cache temperature and can generate hotspots at some localized cache regions caused by locality of reference. Researchers tried to overcome higher temperature at the ReRAM caches; however, a complete simulation framework that simulates performance-power-temperature and its impacts on cache behavior yet to be developed.

In *HotReRAM*, we offer a comprehensive thermal model for the ReRAM caches based on an existing thermal simulator, HotSpot [14], that is not only able to simulate temperature by exploiting the power traces, but is also able to provide detailed temperature induced behaviors, which can lead architects and designers to study in-depth analysis of the ReRAM caches for a variety of design choices. In fact, our model can be further extended to support other NVMs and the contemporary technologies, like FinFET or GAAFET based designs. Basically, we attached *HotReRAM* with the existing simulation framework to simulate the thermal properties of the ReRAM cache. We periodically generate cache access counts at the apt granularity level from gem5 [15] and derive periodic power traces for ReRAM caches by employing the NVSim [16] simulator. The power traces will further be used as an input to the *HotReRAM*, which will next generate the thermal status of the ReRAM based cache. The generated temperature traces are next used to produce the thermal stability, retention

S. Chakraborty and M. Själander are with the Department of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway 7491. E-mail: shounak.chakraborty@ntnu.no and magnus.sjalander@ntnu.no.

T. Bunnam and J. Arunruerk are with Department of Computer Engineering, Rajamangala University of Technology Thanyaburi, Thailand. E-mail: thanasin.b@en.rmutt.ac.th and jedsada.a@en.rmutt.ac.th.

S. Agarwal is with School of Computing and Electrical Engineering, Indian Institute of Technology Mandi, India. E-mail: sukarn@iitmandi.ac.in.

S. Yu and R. Shafik are with Engineering School, Chongqing University of Posts and Telecommunications, 400065, CN, and Newcastle University, NE4 7RU, UK. E-mail: yusq@cqupt.edu.cn and rishad.shafik@ncl.ac.uk

Manuscript received April 00, 2023; revised August 00, 2023.

[§]Equal contribution

time, and write latency of the ReRAM cache. ReRAM's thermal resistance is significantly higher than the conventional MOSFET, potentially restricting lateral heat transfer with its peering on-chip components.

Towards developing a robust thermal model of the ReRAM caches, we first prepare a cell library that can support the ReRAM cells. To improve accuracy, we further developed a framework that can model temperature at the subarray level granularity, configuration details of which are discussed in Sec. III. Our detailed thermal model at the subarray granularity level is also verified with circuit level simulators like Cadence Spectre and VTEAM model [9]. To the best of our knowledge, *HotReRAM* is the first thermal model for the ReRAM caches that generates temperature values and several thermal characteristics for different LLC regions at the deeper granularity, which will enable researchers and architects to study the diverse thermal behaviors within a single LLC bank. The major contributions of *HotReRAM* can be listed as follows: *HotReRAM* analyzes the circuit level thermal characteristics of ReRAM/Memristor based caches at the subarray level granularity, which has been next exploited to build up a novel architecture level thermal model with the following characteristics-

- *HotReRAM* is able to analyze transient and steady-state temperatures of a single cache bank at the subarray level granularity, which will help in understanding spatial thermal variance within a single LLC bank.
- by exploiting the derived temperature values, *HotReRAM* next determines the cache's thermal stability, retention time, and write latency.

Based upon a prior thermal model, HotSpot [14], we developed *HotReRAM*, a thermal simulation framework that can model temperature and all the relevant temperature dependent characteristics of the ReRAM-based caches. Our simulation results for an 8MiB ReRAM cache show that the spatial thermal variance can have a maximum magnitude of 7 °C for a single cache bank. In contrast, the temporal thermal variance is noticed more than $40\,^{\circ}\mathrm{C}$. Such temperature variances vary retention time with a standard deviation of 3.9 to 10.2 for a set of benchmark applications, where the write latency can increase up to 14.5%. To the best of our knowledge, *HotReRAM*[1] is the first thermal simulation model for simulating NVM along with its detailed temperature induced power/performance behaviors.

II. ReRAM Background: Thermal Aspect

A representative link between flux and charge is theoretically presented for the first time as memristor [17], whereas the first practical device was built as a sandwich of TiO_2 and TiO_{2-x} [18]. These ReRAMs or memristors have been proposed as a promising alternative to the conventional SRAM or DRAM devices due to their ability to perform ALU operations in addition to storage capability [12]. In fact, ReRAM is more favorable than the other NVMs for their higher cell density, lower write energy, and reduced read latency [19], [20]. However, elevated temperature significantly affects basic ReRAM properties like retention time, switching speed, and reset current [21], [22]. Even the impact of increased temperature can potentially lead to storage failure while switching from low to high resistance state due to noticeable change in retention time. Moreover, endurance, in terms of write counts, of these ReRAM devices is drastically reduced at the higher temperature. Prior arts considered thermal impacts on the ReRAM for a limited number of circuit level properties, but a detailed thermal model that is able to generate temperature traces while accounting all important thermal induced properties is yet to develop.

The memristance of ReRAM can be modeled by employing the fundamental resistance expression, $R = \rho \times \frac{L}{A}$, where ρ, L and A are resistivity, length and cross-sectional area of the device, respectively. Out of these parameters, ρ only depends on the temperature: $\rho = \rho_0 \cdot e^{\frac{E_a}{k_B T}}$, where values of ρ_0 and activation energy E_a are determined empirically [23], [24]. Endurance of ReRAM is modeled as $Endurance \approx \frac{t_w}{t_0}^{\frac{U_F}{U_S}-1}$, where t_w is write latency, t_0 is a device related constant, U_F represents activation energy for the failure mechanism, and U_S is the activation energy for the switching mechanism [12]. Prior work reported that, the values of $\frac{U_F}{U_S}$ is in the range of $2-4$ for the NVMs, whereas $\frac{t_w}{t_0} > 1$ [22], [25]. From the above expressions, we can conclude that, t_w directly impacts the cell endurance of the ReRAM. However, t_w has a dependency on the temperature, which can be represented as [12]-

$$t_w = \frac{D^2}{\mu_I(T)\nu_w}\left(\frac{r_1 - 1}{2}({x_0}^2 - {x_f}^2) + (r_1 + r_2)(x_f - x_0)\right) \quad (1)$$

where D is the film thickness, $\mu_I(T)$ is the ion mobility at temperature T, ν_w is assumed as a constant write voltage, $r_1 = R_{on}/R_{off}$, $r_2 = R_{pd\_col}/R_{on}$, and x_0 and x_f are the initial and final states, respectively. R_{on}, R_{off} and R_{pd_col} are on resistance, off resistance and column pull down resistance, respectively. The temperature dependency of μ_I can be represented by Nerst-Einstein relation based on ion diffusion coefficient at low electric field [26]-

$$\mu_{I(T)} = \frac{q_I f a^2 exp(-\frac{E_a}{k_B T})}{k_B T} \quad (2)$$

where, q_I, f, a, and E_a are the ion charge, ion jump frequency, ion jump distance and ion activation energy, respectively, and k_B is Boltzman's constant. We further use the conventional heat transfer model that computes horizontal heat flow between adjacent components [14] to determine the steady-state temperature of the ReRAM based caches, where the values of respective thermal resistances are taken from prior work [12].

ReRAM in Multicore: Employing ReRAM to fabricate LLC can be a promising option, as ReRAMs exhibit higher write endurance, lower access time than the other NVMs. Additionally, these ReRAM cells can potentially be used for some specific chip design that caters applications like neuromorphic computing. However, the functional correctness of ReRAM devices can only be achieved at some safe thermal status. Prior work shows that [12], write endurance and write latency are heavily affected at 340K or more, which might even lead to permanent damage to the circuitry. In fact, an

[1]Link to HotReRAM: https://github.com/shounakchakraborty/HotReRAM

operational temperature more than 340K drastically reduces the $\frac{R_{OFF}}{R_{ON}}$, which incorporates the correctness issues for the stored data [11]. Moreover, the existing diversity in cache access patterns leads to significant temporal as well as spatial thermal variances across the single LLC bank, which can potentially entail uneven cache behaviors across a single LLC bank. Hence, using ReRAM to fabricate LLC needs intense prior thermal analysis by employing accurate thermal modeling of these devices before their integration with the cores. The prime focus of *HotReRAM* is to develop such an accurate thermal model that the designers and architects can use before deploying practical ReRAM caches.

III. HotReRAM: Proposed Simulation Model

In this section, we first describe the ReRAM based cache model of *HotReRAM*. Then we discuss how *HotReRAM* models the thermal stability, retention time and write latency of the ReRAM cache. Finally, our power and temperature model will be discussed.

A. ReRAM based Cache Design: at circuit level

Figure 1 illustrates the basic architecture of a ReRAM based cache, which consists of an input multiplexer, a 4 KiB ReRAM crossbar (32-bit data and 20-bit tag), an output decoder, a comparator bank, and an output multiplexer. RW signal to the input multiplexer is used to select between read or write operations yet to be done in the crossbar. For a data read, RW is set to 1, and $Vread$ is connected to the bitline input (BLi). The current will pass through the 1T1R cell, which is selected by the wordline signal (WL), and the output current (BLo) will next reach Rs where the voltage divider is formed. The Rs voltage will subsequently be decoded by comparators [27], resulting in binary values. At this point, the decoded tag will be sent to compare to the input tag to indicate cache hit or miss, and the decoded data will be sent to the output multiplexer, where the offset selects a 1-byte output. Now, for write operation, RW is set to 0, and based on the input data, either $Vwrite$ or $-Vwrite$ will be connected to BLi. If the data is 0, $Vwrite$ will be supplied to switch the memristor to LOW conductance (high resistance state), otherwise BLi will be connected with $-Vwrite$, resulting in the memristor switching toward $HIGH$ conductance (low resistance state). However, in both cases, BLo will be grounded, resulting in the positive/negative voltage drop across the target 1T1R cell, leading to the state switching. Keeping the voltage at one end of the memristor terminals under its threshold can prevent unintended tuning [28].

B. Modeling ReRAM based Cache: at micro-architecture level

Our ReRAM based LLC is physically distributed yet logically shared among the cores. Each of the equal sized physical partitions of the LLC is called a bank, where each bank is further partitioned into sub-banks. Individual sub-banks are partitioned into multiple mats, and each mat is partitioned into subarrays, each of which is further a collection of cache lines, called cell-array. The cell-array within a subarray is

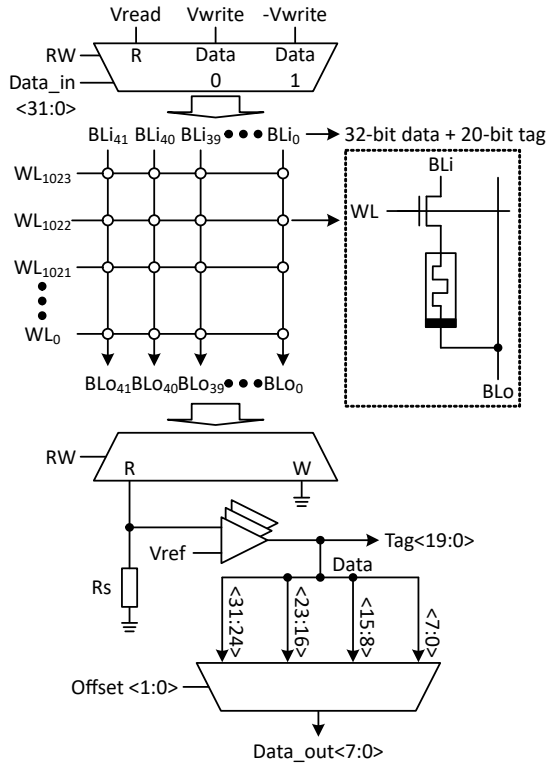

Fig. 1. ReRAM based Cache Design: Representation of a sub-array of LLC.

equipped with peripheral circuitry like row-decoder, wordline-driver, column mux, sense-amplifier and output driver. For example, we model a 1 MiB 16-way set associative LLC with the described structure illustrated in Figure 2.

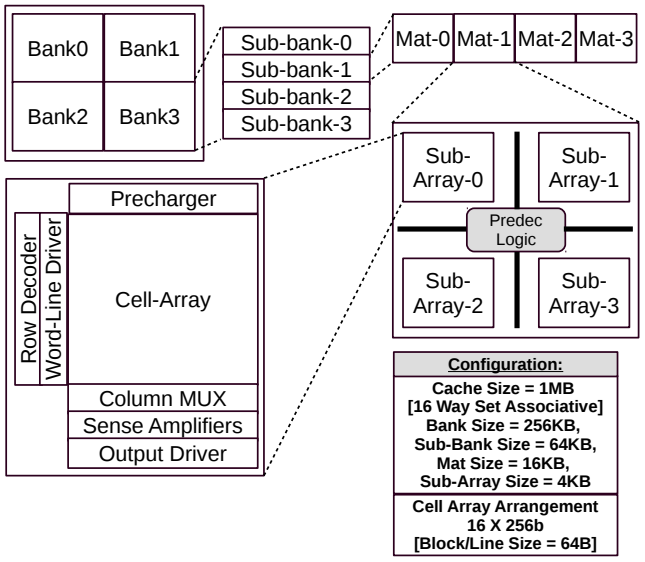

Fig. 2. LLC layout used in *HotReRAM* [29], where circuit design of each sub-array is illustrated in Figure 1.

In *HotReRAM*, we model the power and temperature of the LLC at the subarray level granularity, where each subarray has a size of 4 KiB. Note that, for larger caches, we replicate the structure of 1 MiB LLC as shown in Figure 2, and design the floorplan of the LLC accordingly. Such granularity incurs

higher accuracy for modeling temperature of the memory cells, hence more accurate analysis of the temperature dependent ReRAM properties. By employing NVSim [16], we periodically generate the power traces at this smaller granularity level, used by *HotReRAM* to generate the temperature traces. The prior thermal simulator, HotSpot [14], models the entire chunk of LLC as a single abstract component where the model is based on MOSFET based SRAM. *HotReRAM* models ReRAM based LLC, power and thermal behaviors of which are different from the SRAM based counterpart, which needs more in-depth analysis of the power consumption. As cache accesses are unevenly distributed across the LLC, power and thermal profiles of the cache locations are also diverse. However, *HotReRAM* precisely models the LLC power at the granularity of 4 KiB, which incurs enough accuracy in our thermal model.

Power and thermal properties of the ReRAM based LLC can be analyzed at various granularity levels. Higher level abstraction might underestimate the detailed characteristics of the cache, whereas deeper level comprehensive analysis might incorporate higher computational cost. However, a certain level of granularity mostly resembles the thermal properties of the deeper levels [14]. Hence, in this work, we considered subarray level granularity.

C. Power vs. Temperature

Temperature of any on-chip component depends upon three prime factors: **(1)** the component's own power consumption, **(2)** heat transfer between the adjacent components, and **(3)** heat abduction by the ambient [30]. Out of these three, the component's own power consumption plays a pivotal role in increasing temperature. Moreover, the higher thermal resistance of the ReRAM device limits the lateral heat transfer [12]. Basically, Joule heating formula establishes the relationship between power consumption and temperature of individual components [31]: $T = T_0 + P \cdot R_{th}$, where T represents the current temperature, and T_0 is the initial temperature of the component; P implies the power consumption, and R_{th} is the thermal resistance of the component. As vertical thermal resistance is almost $20\times$ higher than the lateral thermal resistance, in this work, we will mostly consider R_{th} as the vertical thermal resistance.

D. Thermal Stability, Retention Time and Write Latency

Fluctuation in temperature influences the operational behaviors of the memristor or ReRAM devices. Hence, a temperature model is to be built for the tracking of change in the memristance and correcting decoded data. Most of the reported memristor models have not included the impact of temperature [18], [32]–[38]. Although the variation in memristance programming speed is modeled as a function of temperature in some prior arts [39], [40], the impact of temperature on memristance is still missing. *HotReRAM* can bridge this research gap by proposing a suitable architectural level thermal simulation framework to simulate performance-power-temperature of the ReRAM based architectures. Towards studying the thermal behaviors of the ReRAM, first, we explore thermal stability of the ReRAM (Δ), which can be represented as: $\Delta = \frac{E_a}{k_B \cdot T}$, where E_a is the activation energy, k_B is the Boltzmann constant, and T implies the temperature in Kelvin. Basically, higher temperature reduces the thermal stability, which can potentially incorporate significant changes in the conductivity (σ) and retention time (τ) [41] of the ReRAM, which can be represented as follows:$\sigma = \sigma_0 \cdot exp(-\Delta)$ and $\tau = \tau_0 \cdot exp(\Delta)$, where σ_0 and τ_0 are initial conductivity and initial retention time, respectively. By employing Equation 1, we also derive the write latency (t_w) of each subarray.

IV. Simulating power-performance-temperature with *HotReRAM*

Our simulation model takes periodic performance traces from gem5 [15] full system simulator. Next, the performance traces are fed to McPAT [42] and NVSim [16] to simulate the power consumption of the individual on-chip components. The power traces will be next sent to *HotReRAM* thermal model for simulating the temperature of the individual on-chip components. While simulating ReRAM cache, *HotReRAM* will first produce the transient and steady state temperature of the LLC by considering power traces at the subarray level granularity having a size of 4 KiB. Once the temperature values for individual subarray are derived, *HotReRAM* will calculate the thermal stability, deviation in retention time, and write latency. The standard deviation of the steady state temperature values across the LLC subarrays is also derived to get the spatial thermal variance at the LLC, which can help one to take necessary measures to overcome thermal induced issues, if any. In this section, we will detail our thermal model, and how we have validated the proposed thermal model in VTEAM simulation framework by modeling the LLC circuitry at subarray level granularity.

A. Generating Performance-Power Traces

Toward generating the periodic performance traces, we execute a set of PARSEC benchmark applications [43] in gem5 simulator in full system mode [15]. Each periodic output of gem5 provides the values of the performance monitoring counters for the LLC, e.g., access counts, LLC hits and misses, number of reads and writes, etc., at the subarray level granularity. All the relevant stats for the LLC are collected and sent to the NVSim's inputs to simulate power consumption of each individual subarrays of the ReRAM-based LLC. From NVSim, we collect the power consumption at the granularity of the subarray of size 4 KiB. Such smaller granularity might increase the computational complexity of the simulation process in case of larger cache sizes. Hence, this size of the subarray is kept tunable with the overall size of the LLC. However, the power traces are then fed to the *HotReRAM* for generating the temperature traces, which includes both transient and steady-state temperature values. Modeling temperature needs floorplan details at the same granularity level for which power traces are generated. Towards that, we use NVSim to generate the area of each subarray and next use the HotFloorPlan simulator from HotSpot [14] to get the optimal floorplan of the whole LLC.

B. HotReRAM: Proposed Thermal Model

Our thermal model, *HotReRAM*, considers the floorplan of the LLC designed at some specific technology node. For each such technology node, thermal capacitance and resistance values for both memristor and CMOS devices are collected, and we update the configuration. We use the default configuration of the cooling system embedded in HotSpot 6.0. Figure 3 shows how *HotReRAM* is integrated with the existing power or performance simulators like NVSim and gem5. We have also modeled the circuitry in VTEAM simulation framework at byte level granularity to validate how temperature impacts energy/bit, access latency, leakage power and R_{off}.

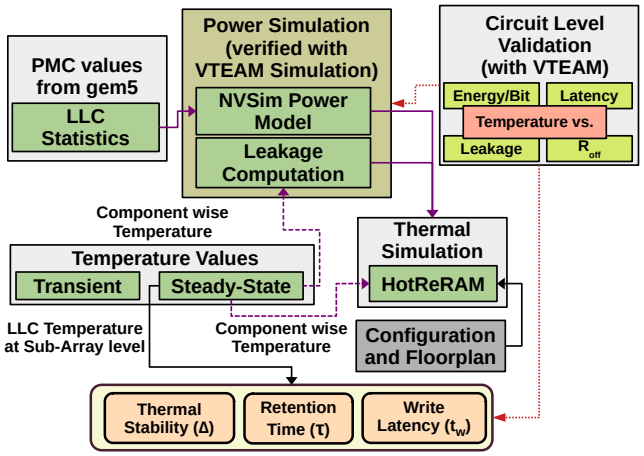

Fig. 3. Proposed Thermal Model.

From gem5, we collect the following performance monitoring counter (PMC) values for the LLC: read count (#LLC_Rd), write count (#LLC_Wr), LLC Miss count (#LLC_Miss) and number of allocations (#LLC_Allc). Unlike conventional MOSFET based SRAM, read and write dynamic energy consumption is not same in case of the ReRAM. Hence, to get the total energy consumption, we need to calculate read and write dynamic energy ($Dyn\_En_{Rd}^{Total}$ and $Dyn\_En_{Wr}^{Total}$) individually. Moreover, an allocation of the new cache block incurs energy for tag write ($Dyn\_En_{Wr}^{Tag}$) in addition to the writing of the data block ($Dyn\_En_{Wr}^{Data}$), whereas normal write operation only consumes data block write energy in addition with the energy usage for tag read ($Dyn\_En_{Rd}^{Tag}$). We also need to consider the dynamic energy for LLC misses ($Dyn\_En_{Miss}^{Total}$), which can also be collected from NVSim's output. As tag write is also costlier than the tag read operation in ReRAM caches, we considered them individually and calculated total dynamic energy for allocation ($Dyn\_En_{Allc}^{Total}$) separately. Once these performance traces are collected, we use the NVSim's dynamic energy consumption for tag and data arrays in the following way:

$$Dyn\_En_{Rd}^{Total} = (Dyn\_En_{Rd}^{Tag} + Dyn\_En_{Rd}^{Data}) \times \#LLC\_Rd \quad (3)$$

$$Dyn\_En_{Wr}^{Total} = (Dyn\_En_{Rd}^{Tag} + Dyn\_En_{Wr}^{Data}) \times \#LLC\_Wr \quad (4)$$

$$Dyn\_En_{Miss}^{Total} = Dyn\_En^{LLC\_Miss} \times \#LLC\_Miss \quad (5)$$

$$Dyn\_En_{Allc}^{Total} = (Dyn\_En_{Wr}^{Tag} + Dyn\_En_{Wr}^{Data}) \times \#LLC\_Allc \quad (6)$$

From these energy values, we derive the total energy consumption, that considers conventional sequential accesses of tag and data arrays for LLCs, as follows:

$$\begin{aligned} Dyn\_En^{Total} = {} & Dyn\_En_{Rd}^{Total} + Dyn\_En_{Wr}^{Total} \\ & + Dyn\_En_{Miss}^{Total} + Dyn\_En_{Allc}^{Total} \end{aligned} \quad (7)$$

Note that, we collect output periodically at the end of each $1M$ clock cycles, and the traces are collected at the subarray level. So, these dynamic energy values imply the energy usage at the individual subarray, which is next converted to dynamic power consumption at the subarray level for the last period. As each LLC allocation is followed by an LLC miss, in our implementation, the dynamic energy for allocation includes the dynamic miss energy.

Next, we employ a leakage computation framework for computing leakage for the individual subarrays. We develop a leakage computation framework by considering subarray level leakage consumption derived from NVSim. The leakage model of NVSim is based upon the leakage model of a prior cache simulator, CACTI [29], that can simulate leakage power only for some specific temperature values, having a range of 300 to 400 K, and the leakage can only be computed for the values multiples of 10. To overcome this issue, we consider the last temperature of each subarray generated by *HotReRAM*, and compute the leakage power for individual subarrays. We employ piece-wise linear approximation technique [30], [44] for generating leakage power for any temperature values. For each subarray, we first generate leakage power for all possible temperature values in NVSim. Next, for each of the 10K temperature ranges, we assume that leakage is increased linearly. Such linear approximation marginally overestimates the leakage consumption, which is compensated by the lower computational cost of the leakage model. To justify the correctness of our power modeling at the architecture level, we verified power values in our VTEAM simulation framework.

Once both leakage and dynamic power for each of these subarrays are derived, we next feed these power traces to the *HotReRAM*. Along with these power values, *HotReRAM* considers the floorplan of the LLC, and (steady-state) temperature of the last period, and generates transient and steady-state temperature values for each of the subarrays. Note that, for computing leakage, we consider the steady-state temperature derived at the end of the last period for the individual subarrays. We also use the steady-state temperature to derive the thermal stability, retention time, and write latency of the ReRAM by employing the respective equations discussed in Sec. III-D.

V. Simulation Results

In this section, we will analyze the results regarding circuit level validation and architectural model of *HotReRAM*.

A. Circuit Level Validation of HotReRAM

The circuit level simulation was done to validate the functionality of *HotReRAM* in Figure 1 using UMC 65nm technology with VTEAM model [45]. We obtained the model parameter set from a recent memristor model designed by Garda and Galias [46], which is extracted from self-directed channel (SDC) memristor [47]. To ensure that the memristor does not switch during a read operation, the reading voltage $Vread$ is set at 200mV. The writing voltage $Vwrite$ of 900mV and $-Vwrite$ of -300mV are selected asymmetrically, because the memristors' thresholds are asymmetric [46].

The simulation results in Figure 4 show all memristors of $word\#1$ are initialized to logic 0 ($LOW\ G_m$). At 3.2ns, $word\#1$ was selected (WL <1>=0) for writing logic 1 ($HIGH\ G_m$). Hence, the $memristor\#32$, that represents $bit\#0$ of $word\#1$, is switched from low to high conductance as shown in the bottom graph. Then, RW is set to logic 1, indicating read operation and the $Output$ <0>=1 signal at 4.7ns confirms logic 1 is successfully stored in $word\#1$. Thus, these results claim the functional correctness.

The temperature effects on the performance are depicted in Figure 5. The energy/bit of writing 0 is higher than writing 1 even logic 0 is represented by low conductance because writing 0 needs a higher voltage switching, from $Vread$ to $Vwrite$ (200mV to 900mV). The situation resembles for reading operation, the larger voltage switching between $Vread$ and $Vwrite$ causes a higher dynamic power dissipation, resulting in a higher energy/bit for reading 0. For the temperature effect, the energy/bit of all operations slightly increases with the temperature because the transistor can conduct more current and leakage when the temperature increases.

The latency of the circuit operation is illustrated in Figure 6. Overall, the latency of writing 1 is higher than writing 0 as described by the memristor model. Furthermore, latency for writing 1 increases non-linearly with temperature because the lower OFF memristance at the higher temperature (discussed in Figure 8) causes the lower voltage drop across the memristor. Consequently, this voltage reduces the switching speed. The reading latency includes the comparator latency that changes in the opposite direction to the temperature because the transistor operation is faster at higher temperature.

The graphs in Figure 7 and Figure 8 show the changes in leakage power and OFF resistance with temperature. As mentioned earlier, the transistor's current is larger at a higher temperature. Hence, the leakage power also increases. The OFF resistance decreases non-linearly when the temperature rises following our experimental-based model [23]. Note that, the thermal effect on the ON resistance is negligible [23], [48].

B. Configuration and Benchmarks

We simulated a homogeneous CMP with four x86 cores in the gem5 full system simulator [15]. Each of these cores has a private 32KiB data and instruction L1 caches [49], [50]. A single 8MiB bank of the ReRAM based L2 cache acts as our shared LLC. The thermal model of *HotReRAM* is agnostic to the number of cache levels, however, simulation time limits the type of workloads that can be evaluated, and

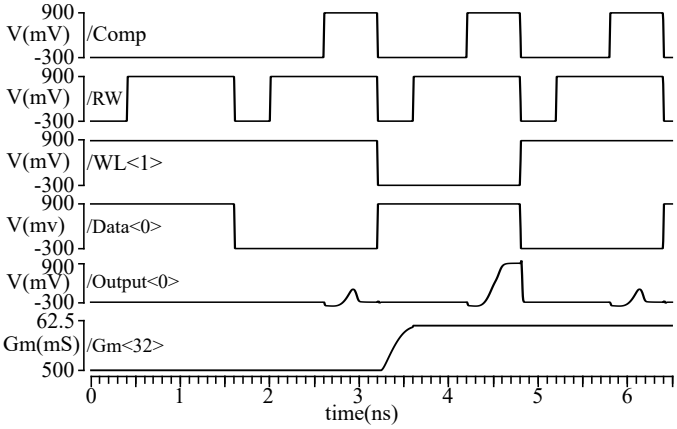

Fig. 4. Simulation waveform. At 3.2ns, WL <1> signal selects word#1 (active low) for writing logic 1. Then, $-Vwrite$ will be connected to 1T1R cells causing conductance (Gm) switching from LOW conductance to $HIGH$ conductance. The $Output$ signal confirms logic 1 is stored and read correctly.

TABLE I
TIME, ENERGY, AND AREA VALUES FOR SRAM (ISO-AREA (A)/ISO-CAPACITY (C)) AND RERAM CACHES ($4/8MiB$, $64B$ BLOCK, 16-WAY)

Memory Device	SRAM		Re-RAM
Cache Size	256KiB	8MiB	8MiB
Feature Size (C/A)	65nm-CMOS(A)	65nm-CMOS(C)	65nm
Wr Energy (nJ) (per access/bit)	0.011	0.082	0.103
Rd Energy (nJ) (per access/bit)	0.011	0.082	0.031
Leakage Power (at 350K)	556 mW	10,826 mW	2,065 mW
Rd Latency (cycles)	1	2	3
Wr Latency (cycles)	1	2	20
Area	19.07 mm^2	208.5 mm^2	17.98 mm^2

a deeper cache hierarchy might need applications with larger working sets to create a realistic workload for the LLC. We used the Ruby module of gem5 for simulating our cache hierarchy. Table II details the system configuration used in our all simulations. We modeled different retention times and energy parameters as mentioned in Table I which are obtained from NVSim [16]. We also compared against two SRAM configurations (SRAM_C (iso-capacity) and SRAM_A (iso-area)) and an Re-RAM configuration (Re-RAM_B) with different retention times for our LLC.

We integrated McPAT [42], NVSim [16] and *HotReRAM* with gem5 [15] into a complete performance-power-thermal simulation framework, as shown in Figure 3. Periodic *performance traces* are collected from gem5 that are fed to power model of NVSim. Considering prior thermal analyses [30], [44], we set the periodic interval as 0.33μs, during which the temperature across the CMP is assumed stable. Dynamic power for individual on-chip components are provided by McPAT, except for the Re-RAM LLC, for which NVSim is used. The leakage estimation in McPAT and NVSim assume uniform on-chip temperature. We use HotFloorPlan module of the HotSpot simulator to generate the floorplan of the CMP once at the beginning by considering the component-wise area-estimation from McPAT and NVSim. The floorplan of the considered homogeneous CMP is shown in Figure 9. The shared ReRAM L2 cache is placed at the center and we place

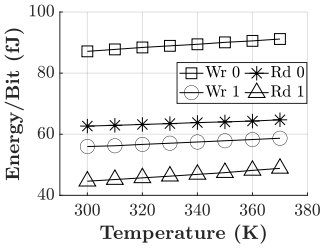

Fig. 5. Temperature vs. Energy/Bit

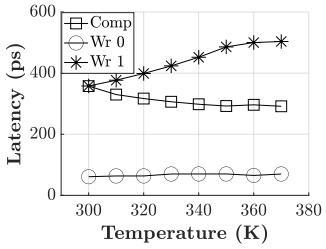

Fig. 6. Temperature vs. Latency

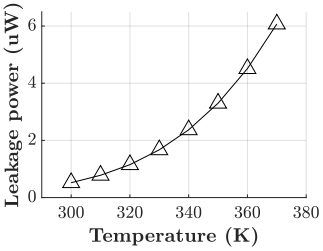

Fig. 7. Temperature vs. Leakage

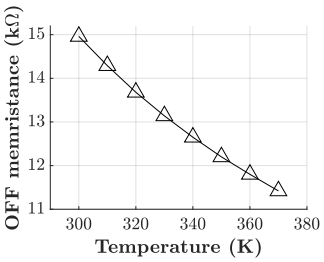

Fig. 8. Temperature vs. R_{off}

TABLE II
HotReRAM: SYSTEM CONFIGURATION

System Component	Configuration
CPU	x86 quad-core, @2GHz
L1 SRAM I/D-Cache	32KiB, 64B cache block, 4-way, LRU, MESI
L2 Re-RAM Cache	8MiB, 64B cache block, 16-way, Non MRU, MESI (with cache sets 0-4095, 4096-8191, respectively)
Main Memory	8GB DRAM

TABLE III
BENCHMARK APPLICATIONS (R: LARGE READ MPKI, W: LARGE WRITE MPKI AND M: COMPARABLE READ AND WRITE MPKI)

Benchmark-Suites	Applications
PARSEC [43]	Blackscholes (M), Bodytrack (M), Canneal (W), Dedup (R), Fluidanimate (R), Freqmine (M), Streamcluster (M), Swaptions (R), X264 (W)

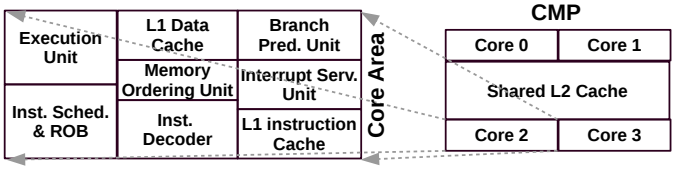

Fig. 9. Floorplan of the four core based CMP

the cores at the periphery of the chip. Each of these core areas is equipped with execution unit, instruction scheduler, reorder buffer, branch prediction, memory ordering units, interrupt handler and the L1 caches. Table IV details the parameters used for our simulations.

We have used read (R), write (W), and mixed (M) intensive multi-threaded PARSEC [43] (with large-sized input) benchmark suite, listed in Table III. We use four threads for each of the considered PARSEC applications, where each core executes one thread each. We have evaluated the CMP for the highest possible workload, for which, all of our cores are always active.

C. Periodic Status of the ReRAM-LLC

The change in cache accesses over time induced temporal thermal variation for the individual cache mats. We randomly selected a mat (numbered 25 among 0 to 63), and captured read ($\#RD\_Count\_L2\_25$), write ($\#WR\_Count\_L2\_25$) and write back ($\#WB\_Count\_L2\_25$) counts over a 100M clock cycles with a sampling granularity of 1M clock cycles for *Blackscholes (Black)*, a mixed workload (Table III). The change in accesses are plotted in Figure 10. As write operations are the main driving factor that increases mat temperature, spikes in $\#WR\_Count\_L2\_25$ or $\#WB\_Count\_L2\_25$ raise up the power consumption, so the temperature, of the mat, which are shown in Figure 11. The poor thermal conductance and higher heat capacitance of the ReRAM does not let the mat to be cooled down soon even when the access counts reduce. The spike $\#WR\_Count\_L2\_25$ at time-stamp 55 increases the power consumption and temperature. Once the access count is reduced after that, the temperature is still consistent.

TABLE IV
PARAMETERS FOR THERMAL MODELING IN *HotReRAM* [51]

Layer	Thermal Conductance (W/mK)	Heat Capacitance (J/m^3K)	Depth (μm)
Heat Sink	400.00	3.55×10^6	6,900
Heat spreader	400.00	3.55×10^6	1,000
TIM	4.0	4.00×10^6	20
Core/SRAM cache	100.0	1.75×10^6	150
ReRAM cache	5.0	1.92×10^5	200

By implementing the formula discussed in Sec. II and III, we derived the important ReRAM properties and their changes over time. We derived the temporal changes in thermal stability (Δ), retention time (τ) and write latency (t_w) for mat 25 while executing *Black*, and plotted the results in Figure 12. Both retention time (τ) and Δ are inversely proportional to the current temperature of the mat, which is also reflected in the results shown in Figure 12. This implies, the higher temperature curtails Δ while also reducing τ, which further increases the write latency (t_w) of the mat. The result shows that t_w increases at the higher temperature, which can potentially aggravate the performance of the cache. Note that, as τ is directly proportional to Δ, we have represented both with a common graph.

Temporal changes in accesses can however vary the temperature of a single mat, however, diversities in accesses at different mats can also be noticed if temperature of all the mats can be considered at a particular time-stamp. We further determine the temperature of all the mats over time and derive the maximum difference between the highest and the lowest temperature of the LLC, which is called maximum spatial thermal variance. The maximum thermal variance varies over

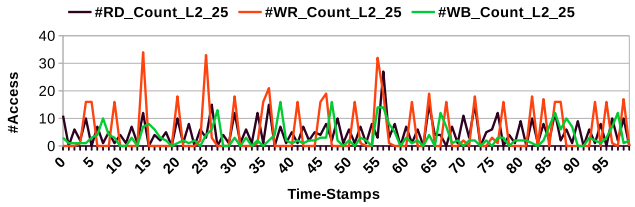

Fig. 10. Read, Write Allocate and Write Back counts for Mat 25 (Black)

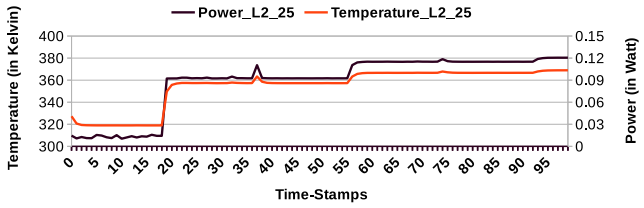

Fig. 11. Temporal Variation in Power and Temperature for Mat 25 (Black)

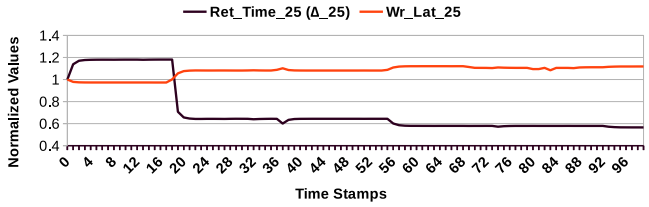

Fig. 12. Temporal Variation in thermal stability (Δ) and Write latency (t_w) for Mat 25 (Black)

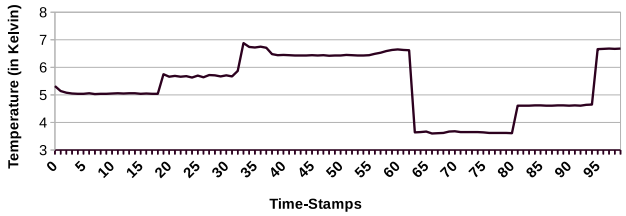

Fig. 13. Spatial Thermal Variance across all Mats (Black)

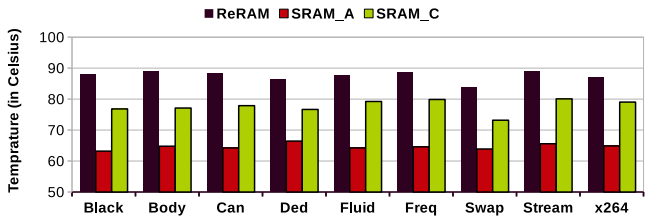

Fig. 14. Maximum Mat Temperature

time, and for *Black*, the value lies within a range of 3-7 K. We plotted the temporal changes in maximum spatial thermal variance in Figure 13.

D. Maximum Temperature and Spatial Thermal Variance

By employing *HotReRAM*, we can capture the maximum temperature of the LLC, i.e. the mat having highest temperature among all. The hottest mat can also be traced periodically and the existing diversity in mat access pattern (both temporally and spatially) changes the hottest mat at different time-stamps during execution. We capture the temperature of the hottest mat for our ReRAM based LLC during execution, and the highest temperature values for all of our nine PARSEC benchmarks have been plotted in Figure 14. The highest temperature for all applications lies within a range between 85-89 °C. We further compare the temperature of ReRAM based LLC with iso-capacity and iso-area SRAM based LLC, configurations of which are given in Table II. For iso-area LLC, the cache capacity is significantly smaller i.e. 256KiB than the ReRAM based LLC of size 8MiB. This small sized SRAM cache has lower power consumption, that has kept its maximum temperature within a range of 61-65 °C for all applications. On the other hand, iso-capacity SRAM has higher leakage power consumption, which increases the cache temperature within a range of 73-80 °C for all benchmarks. Specifically, lower heat conductance and higher thermal capacitance of ReRAM LLC over SRAM LLC results into lower heat dissipation which leads to higher maximum temperature for the ReRAM LLC over the iso-capacity SRAM counterpart.

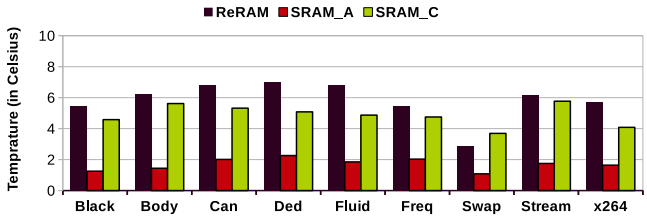

Fig. 15. Average Spatial Thermal Variance

We also captured the average spatial thermal variance of the LLC for different benchmarks, which are plotted in Figure 15. Dynamic power, which depends upon the cache accesses, is the main factor that decides the thermal status of different mats of the ReRAM cache. Hence, the mat having maximum (write) accesses has the higher temperature than the others. On the other hand, leakage power shares the significant portion of the total cache power for SRAM caches, which does not depend on the access counts, rather depends on how long the cache is kept turned on. But, SRAM cache mats adjacent to the processor cores have higher temperatures due to significant heat transfer, which increases the spatial thermal variance in case of iso-capacity SRAM LLC. However, ReRAM LLC still have higher average spatial thermal variances among the three cache configurations, for all benchmarks.

E. Change in Retention Time and Write Latency

As temperature directly impacts τ and t_w in ReRAM caches, we also plotted the maximum values for standard deviation of τ and maximum percentage increase in t_w. The maximum standard deviation in τ across all the benchmarks lies within a range of 3.9 - 10.2, which is shown in Figure 16. The applications having comparatively higher access counts with higher temporal locality leads to higher maximum standard deviation in τ. The higher cache temperature also

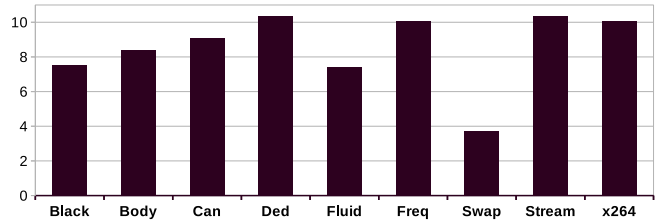

Fig. 16. Maximum Standard Deviation in Retention Time for ReRAM LLC

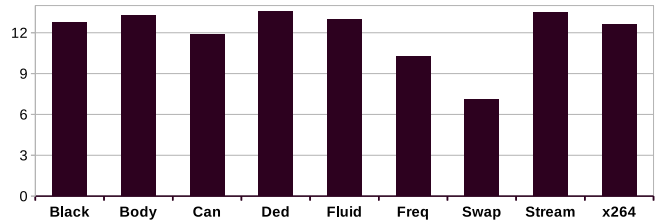

Fig. 17. Maximum percentage increase in Write Latency for ReRAM LLC

increases the write latency (t_w), which lies within a range of 7.1 - 13.9% for all nine benchmarks, as shown in Figure 17.

VI. Prior Work

Thermal simulators for MPSoC started appearing with the gradual reduction in channel length of the transistors, as thermal management for the CMPs built in lower technology nodes has become a prime design concern for the designers and architects. HotSpot [14] was the very first available thermal simulators that has been adopted by the architects and designers due to its ease of integration in an MPSoC simulation workflow. The HotSpot simulator has been built on the basic RC thermal model of the underlying circuitry while considering conventional CMOS transistors as the basic building block. Later, ISAC simulator [52] introduced an adaptive spatial grid with an adaptive simulation time step. However, the capabilities of ISAC was similar to HotSpot, but the simulator performance has been improved above $10\times$ as claimed by the authors. LUTSim [53], another thermal simulator that also attempted to boost up the simulation speed by adopting a look-up table approach. Next, 3D-ICE [54] extended the capabilities of the existing MPSoC thermal simulators by adding support for on-chip cooling mechanisms.

However, none of these simulators considered emerging memory devices, performance and lifetime of ReRAM which are also severely affected by higher temperature [10]–[13]. Additionally, the usefulness of these prior thermal models are mostly limited by generating transient and steady-state temperatures, which drives designers to put extra effort to derive several thermal induced properties of the circuitry. A plethora of prior arts [17]–[19], [21], [23] also showed how temperature affects the functionalities of the ReRAM circuitry. These prior analysis identified that, thermal stability (Δ) is the prime thermal induced basic parameter that needs to be kept stable for consistent performance of the ReRAM devices. Basically, the change in Δ affects the other fundamental properties like retention time and write latency of the ReRAM. Additionally, shrinking in process technology enables designers to integrate larger LLCs on-chip. The existing spatial and temporal diversities in LLC accesses lead to significant variation in power density, hence temperature, across the LLC area [30], [44]. None of the prior simulation framework is capable of capturing spatial and temporal thermal variance across the LLC area.

To the best of our knowledge, *HotReRAM* is the first thermal model for the ReRAM LLC based MPSoC, based on the RC-Thermal model of the HotSpot [14], that not only captures the spatial and temporal thermal variance across the different portions of the large LLCs, but also derives the changes in three fundamental properties of the ReRAM LLCs at the mat level granularity [29]- thermal stability, retention time and write latency. Our thermal model exploits the NVSim for the area and power analysis, which is further used in designing the floorplan of the CMP [14]. The power simulation is done by considering the periodic performance traces, derived from gem5 full system simulator [15]. This complete performance-power-thermal simulation framework will enable the architects and designers towards study a broad design space exploration in the domain of emerging memories.

VII. Conclusion

In this paper, we propose a comprehensive thermal modeling and simulation framework, *HotReRAM*, for the ReRAM based caches, that models cache temperature at the smallest possible granularity level by considering power traces and other thermal properties of the ReRAM. The existing diversity in access counts across the cache area generates distinct power profiles within a single cache bank, which leads to significant spatial thermal variance in operational cache temperature. By considering fundamental properties of ReRAM and power traces, *HotReRAM* generates temperature traces at the fine grain level along with detailed analysis of the thermal stability, retention time and write latency. This in depth analysis will enable future architects and designers to analyze detailed and uneven cache characteristics across a single cache bank, which will further assist one to take necessary measures to overcome the thermal induced issues at ReRAM based caches. Our simulation results for an 8MiB ReRAM cache also show that the spatial thermal variance can have a maximum magnitude of $7\,^{\circ}\mathrm{C}$ for a single cache bank, whereas the temporal thermal variance is noticed more than $40\,^{\circ}\mathrm{C}$. Such temperature variances vary retention time with a standard deviation of 3.9 to 10.2 for a set of benchmark applications, where the write latency can increase up to 14.5%.

References

[1] D. Apalkov *et al.*, "Spin-transfer torque magnetic random access memory (STT-MRAM)," *ACM JETC*, 2013.
[2] H. Y. Lee *et al.*, "Low power and high speed bipolar switching with a thin reactive Ti buffer layer in robust HfO2 based RRAM," in *IEEE IEDM*, 2008.
[3] M. K. Qureshi *et al.*, "Phase change memory: From devices to systems," *Synthesis Lectures on Computer Architecture*, 2011.
[4] R. Bez *et al.*, "Introduction to flash memory," *Proceedings of the IEEE*, 2003.
[5] R. Shafik *et al.*, "Real-Power Computing," *IEEE Trans. Computers*, 2018.

[6] H. Abunahla and B. Mohammad, "Memristor Device Overview," in *Memristor Technology: Synthesis and Modeling for Sensing and Security Applications*. Springer International Publishing, 2018.

[7] I. Vourkas and G. C. Sirakoulis, "Memristive Crossbar-Based Nonvolatile Memory," in *Memristor-Based Nanoelectronic Computing Circuits and Architectures*. Cham: Springer International Publishing, 2016.

[8] S. Maheshwari *et al.*, "Hybrid CMOS/Memristor Circuit Design Methodology," *arXiv e-prints*, 2020.

[9] S. Kvatinsky *et al.*, "Magic—memristor-aided logic," *IEEE TCAS II: Express Briefs*, 2014.

[10] H. Shin *et al.*, "A thermal-aware optimization framework for ReRAM-based deep neural network acceleration," in *ICCAD*, 2020.

[11] M. V. Beigi and G. Memik, "Thermal-aware optimizations of ReRAM-based neuromorphic computing systems," in *DAC*, 2018.

[12] M. V. Beigi and G. Memik, "THOR: thermal-aware optimizations for extending ReRAM lifetime," in *IPDPS*, 2018.

[13] X. Liu *et al.*, "HR3AM: A heat resilient design for RRAM-based neuromorphic computing," in *ISLPED*, 2019.

[14] R. Zhang *et al.*, "HotSpot 6.0: Validation, acceleration and extension." in *University of Virginia, Tech. Report CS-2015-04*, 2015.

[15] N. Binkert *et al.*, "The Gem5 simulator," *SIGARCH Comput. Archit. News*, 2011.

[16] X. Dong *et al.*, "NVSim: A Circuit-Level Performance, Energy, and Area Model for Emerging Nonvolatile Memory," *IEEE TCAD*, 2012.

[17] L. Chua, "Memristor-the missing circuit element," *IEEE Trans. Circuit Theory*, 1971.

[18] D. B. Strukov *et al.*, "The missing memristor found," *Nature*, 2008.

[19] P. Chi *et al.*, "PRIME: a novel processing-in-memory architecture for neural network computation in ReRAM-based main memory," in *ISCA*, 2016.

[20] C. Xu *et al.*, "Overcoming the challenges of crossbar resistive memory architectures," in *HPCA*, 2015.

[21] P. Sun *et al.*, "Thermal crosstalk in 3-dimensional RRAM crossbar array," in *Scientific Reports*, 2015.

[22] D. B. Strukov, "Endurance-write-speed tradeoffs in nonvolatile memories," in *Applied Physics A*, 2016.

[23] T. Bunnam *et al.*, "Empirical temperature model of self-directed channel memristor," in *IEEE SENSORS*, 2020.

[24] T. Bunnam, "Memristor-based design solutions for mitigating parametric variations in iot applications," Ph.D. dissertation, School of Engineering, 2021. [Online]. Available: https://theses.ncl.ac.uk/jspui/handle/10443/5398

[25] L. Zhang *et al.*, "Mellow Writes: Extending lifetime in resistive memories through selective slow write backs," in *ISCA*, 2016.

[26] U. Weinert and E. A. Mason, "Generalized nernst-einstein relations for nonlinear transport coefficients," *Phys. Rev. A*, 1980.

[27] S. Chevella *et al.*, "A low-power 1-v supply dynamic comparator," *IEEE Solid-State Circuits Letters*, 2020.

[28] T. Bunnam *et al.*, "Pulse controlled memristor-based delay element," in *PATMOS*, 2017.

[29] S. Thoziyoor *et al.*, "CACTI 5.1," in *Tech. Report, HP Lab.*, 2008.

[30] S. Chakraborty and H. K. Kapoor, "Exploring the role of large centralised caches in thermal efficient chip design," *ACM TODAES*, 2019.

[31] Y. Koo *et al.*, "Accelerated retention test method by controlling ion migration barrier of resistive random access memory," *IEEE Electron Device Letters*, 2015.

[32] S. Kvatinsky *et al.*, "VTEAM: a general model for voltage-controlled memristors," *IEEE TCAS II: Express Briefs*, 2015.

[33] I. Messaris *et al.*, "A Data-Driven Verilog-A ReRAM Model," *IEEE TCAD*, 2018.

[34] R. E. Pino *et al.*, "Compact method for modeling and simulation of memristor devices: Ion conductor chalcogenide-based memristor devices," in *NANOARCH*, 2010.

[35] S. Kvatinsky *et al.*, "TEAM: ThrEshold Adaptive Memristor Model," *IEEE TCAS I: Regular Papers*, 2013.

[36] C. Yakopcic *et al.*, "Generalized memristive device spice model and its application in circuit design," *IEEE TCAD*, 2013.

[37] E. Lehtonen and M. Laiho, "CNN using memristors for neighborhood connections," in *CNNA*, 2010.

[38] R. Berdan *et al.*, "A Memristor SPICE Model Accounting for Volatile Characteristics of Practical ReRAM," *IEEE EDL*, 2014.

[39] C. E. Merkel and D. Kudithipudi, "Temperature Sensing RRAM Architecture for 3-D ICs," *IEEE TVLSI*, 2014.

[40] Z. Jiang *et al.*, "A compact model for metal–oxide resistive random access memory with experiment verification," *IEEE Trans. Electron Devices*, 2016.

[41] K. Kim *et al.*, "Nanoscale resistive memory with intrinsic diode characteristics and long endurance," *Appl. Phys. Lett.*, 2010.

[42] S. Li *et al.*, "McPAT: an integrated power, area, and timing modeling framework for multicore and manycore architectures," in *MICRO*, 2009.

[43] C. Bienia *et al.*, "The PARSEC benchmark suite: Characterization and architectural implications," in *PACT*, 2008.

[44] S. Chakraborty and M. Själander, "WaFFLe: Gated Cache-Ways with Per-Core Fine-Grained DVFS for Reduced On-Chip Temperature and Leakage Consumption," *ACM TACO*, 2021.

[45] S. Kvatinsky *et al.*, "VTEAM: A general model for voltage-controlled memristors," *IEEE TCAS II: Express Briefs*, 2015.

[46] B. Garda and Z. Galias, "Modeling Sinusoidally Driven Self-Directed Channel Memristors," in *Int. Conf. Signals and Electronic Systems*, 2018.

[47] K. A. Campbell, "Self-directed channel memristor for high temperature operation," *Microelectronics J.*, 2017.

[48] N. Wald and S. Kvatinsky, "Understanding the influence of device, circuit and environmental variations on real processing in memristive memory using memristor aided logic," *Microelectronics J.*, 2019.

[49] Accessed: 2022-10-08. [Online]. Available: https://edc.intel.com/content/www/us/en/design/ipla/software-development-platforms/client/platforms/alder-lake-desktop/12th-generation-intel-core-processors-datasheet-volume-1-of-2/009/

[50] Accessed: 2022-10-08. [Online]. Available: https://developer.arm.com/documentation/100236/0100/functional-description/l1-memory-system/about-the-l1-memory-system

[51] M. V. Beigi and G. Memik, "TAPAS: temperature-aware adaptive placement for 3D stacked hybrid caches," in *MEMSYS*, 2016.

[52] Y. Yang *et al.*, "Adaptive multi-domain thermal modeling and analysis for integrated circuit synthesis and design," in *ICCAD*, 2006.

[53] C. Pan *et al.*, "I-LUTSim: An iterative look-up table based thermal simulator for 3-D ICs," in *ASP-DAC*, 2013.

[54] A. Sridhar *et al.*, "3D-ICE: Fast compact transient thermal modeling for 3D ICs with inter-tier liquid cooling," in *ICCAD*, 2010.