IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 August 2023, accepted 31 August 2023, date of publication 6 September 2023, date of current version 13 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3312637

RESEARCH ARTICLE

High Bandwidth and Highly Available Packet Buffer Design Using Multi-Retention Time MRAM

YONGWOON SONG, MUNHYUNG LEE, AND HYUKJUN LEE, (Member, IEEE)
Department of Computer Science and Engineering, Sogang University, Seoul 04107, South Korea

Corresponding author: Hyukjun Lee (hyukjunl@sogang.ac.kr)

ABSTRACT Significant challenges are posed in the design of routers and switches by the explosive growth of internet traffic and the stringent requirements for high availability in the research area of computer networks. Ensuring both high performance and system availability is crucial. To achieve this, recent advancements have turned to the utilization of non-volatile memories, such as magnetic RAM (MRAM) and phase-change memory (PCM), in routing lookup tables and packet buffers of routers and switches. However, the use of non-volatile memories show limitations in scaling with respect to bandwidth and capacity. With the increasing clock speed of the IO bus, high-capacity memories like PCM exhibit limited scalability in performance since accessing the cells in the array does not show significant improvement. Meanwhile, attempts to enhance the capacity of low-access-latency non-volatile memories like spin-transfer torque MRAM (STT-MRAM) through smaller cell sizes result in an adverse impact on the write time. The goal of this study is to design a packet buffer that can provide high bandwidth and ensure high availability. To achieve this, a multi-retention time MRAM-based packet buffer is presented, along with a packet mapping method, which aims to overcome the scalability challenge while ensuring high availability. The two-tier packet buffer (TT-PB) structure implements a small/fast MRAM combined with a large/slow MRAM, which outperforms the baseline MRAM/PCM hybrid memory-based packet buffer by up to 16% and 58% for 1.6 and 3.2 GHz IO bus clocks. For input, internet-mix packet traffic is utilized to depict realistic internet traffic. Moreover, the proposed latency-aware multi-retention time MRAM-based packet buffer structure (MR-PB) consists of short and long retention time MRAM partitions. It identifies the buffering latency demands of various packets and writes the packets into partitions that have sufficient retention time to accommodate the required buffering latencies, thereby achieving an optimal write latency for each packet. With this scheme, an additional speedup of up to 5.27% is achieved over TT-PB.

INDEX TERMS IP router, packet buffer, spin-transfer torque magnetic RAM (STT-MRAM), multi-retention time.

I. INTRODUCTION

The rapid surge in internet traffic is being driven by the widespread adoption of mobile devices and the proliferation of internet-based services, such as information search, streaming video, and video conferencing. Consequently, there is an increasing need for router/switch designs that can effectively handle this exponential growth. Additionally, many network applications require routers/switches to have high availability [1], [2], [3]. In financial enterprises, the target availability has already reached seven 9s (99.99999% availability) [4], which translates into only 3.16 seconds of acceptable downtime per year. Financial equity transactions demand extremely high availability and short packet latency ranging from several microseconds to a few hundred milliseconds [5].

The packet buffer plays a crucial role in routers/switches. When internet packets enter routers/switches through the input interface, they undergo packet processing and classification by the packet processor before being sent to the appropriate output interface for routing to their final destinations. During this process, the packets are stored in the

The associate editor coordinating the review of this manuscript and approving it for publication was Salekul Islam.

packet buffer, typically composed of multiple DRAM channels. Designing such a buffer becomes challenging due to the need to satisfy both high performance, with memory bandwidth in the range of hundreds or thousands of Gbps [6], and high availability.

Previous studies have explored methods to maximize bank-level parallelism and row-buffer locality of DRAM access to meet the increasing demand for high memory bandwidth [7]. Other approaches have employed SRAM buffers capable of absorbing transient congestion caused by traffic bursts that cannot be handled by slow DRAM-based packet buffers [8], [9], [10]. However, these approaches have limitations, as the performance of DRAM memories is degraded for several fundamental reasons. As the number of memory channels and banks of DRAM memory chips increases, the row buffer locality of the DRAM chips, which captures the spatial locality of packet data, decreases. This degradation adversely affects bandwidth utilization. Furthermore, the performance of DRAM cell arrays lags behind the increasing speed of IO bus clocks.

To ensure both high performance and availability, a hybrid memory-based packet buffer using magnetic RAM (MRAM) and phase change RAM (PCM) has been proposed in recent literature [11]. The MRAM/PCM-based packet buffer demonstrates superior performance compared to a traditional DRAM-based packet buffer while offering high availability in the event of a power reset, thanks to its non-volatile nature. This solution employs a combination of a small and fast MRAM memory alongside a larger PCM memory to construct the packet buffer. The fast MRAM memory surpasses the performance of DRAM, effectively handling packet traffic that could potentially deteriorate the row buffer locality of the slower PCM section. Consequently, this configuration enhances the overall performance of the packet buffer.

However, PCM is not scalable in terms of bandwidth, as the gap between IO clock speed and array cell access speed increases. As the IO clock speed increases, the memory bandwidth utilization decreases due to PCM becoming a performance bottleneck. This degradation worsens with increasing queue numbers. As the number of output queues supported in routers/switches increases, data from packets destined for different queues are spread over multiple memory rows, resulting in reduced row buffer locality. Low row buffer locality exacerbates the problem caused by the increasing gap between IO bus speed and cell array access speed.

To address the limitation of PCM, the design of the packet buffer could be exclusively based on MRAM memory. Recent reports indicate that MRAM integration has reached gigabit levels [12], [13], [14]. However, a major issue with MRAM-based packet buffers is that the write latency of MRAM memories increases as their capacity increases. This behavior is influenced by several contributing factors. First, as the memory capacity increases, the cells are partitioned into numerous sub-arrays, leading to increased access latency as signals have to traverse H-tree networks. Second, smaller cell sizes bring neighboring cells closer, leading to increased magnetic coupling [15]. To counteract this effect, the write pulse needs to be elongated to provide sufficient current for flipping the state of MRAM cells. This is why new high-capacity MRAM exhibits much higher write latency, ranging from 30 ns to 50 ns [14], [16], [17], compared to small embedded MRAM used for cache applications.

To attain a high-capacity MRAM memory similar to DRAM, deliberate reduction of the thermal stability factor could be implemented, thereby leading to reduced cell size and power consumption for accesses. However, this adversely affects the retention time of the memory. Recent studies reduce the retention time MRAM for last-level cache (LLC) and main memory [18], [19] in general-purpose computing. The reduced retention time MRAM-based LLC employs expensive counters to keep track of data retention time [19]. The MRAM-based main memory reduces the thermal stability factor to achieve small cell sizes but at the cost of reduced retention time. Data in such memory must be scraped and refreshed, similar to DRAM-based main memory, resulting in additional overheads [18].

To address the bandwidth scalability problem and provide high availability in packet buffers, we propose the multi-retention time MRAM-based packet buffer, which consists of multi-retention time MRAM chips. This approach is based on the observation that all packets are buffered in the packet buffer of routers/switches for only a limited time, and the MRAM's memory retention time for packet data only needs to satisfy the worst-case buffering time.

The contributions made in this study can be summarized as follows:

- The relationship among the thermal stability factor, retention time, write current, and write latency of STT-MRAM is derived. By identifying the maximum buffering period of packets in the packet buffer given network flows, we propose designing the packet buffer to retain packet data only for the buffering period. The reduced retention time helps achieve both large capacity due to a smaller cell size and high performance due to a shorter write latency.
- In a two-tier packet buffer structure, a packet buffer consists of embedded MRAM (small and fast) in the buffer device and stand-alone MRAM (large and slow). Typically, transmission control protocol (TCP) buffering requires buffering packets during the round trip time (RTT) (e.g., 100 msec) upon severe congestion. Thus, the large MRAM portion only needs to retain the packet data during this buffering time. This reduces the retention requirement of data written into the packet buffer and improves the write latency (or power consumption/cell size). Furthermore, it allows for a reduction in cell size to increase the memory capacity.
- In internet routers/switches, network configurations are often fixed. For instance, the bandwidth and latency requirements (QoS requirements) for supported network flows are predetermined. In the packet latency-aware multi-retention time MRAM-based packet buffer

structure, multi-retention time MRAM partitions are provisioned to accommodate the diverse buffering requirements of network flows. The MRAM memory partitions with different retention times have different thermal stability factors. They are written with different write latencies. Unlike general-purpose applications, it is not necessary to track how long packet data will stay in the packet buffer, as the packet latency requirement guarantees that the written data will be read before the retention time expires. The packet buffer controller is informed about the per-packet buffering requirement and the packet data is written into the proper partition, which guarantees the specified retention time. Latency-aware packet mapping improves the bandwidth utilization by optimizing write latency.

The two-tier packet buffer structure implements a small/fast embedded MRAM combined with a large/slow MRAM (with a 100 msec retention time), which outperforms the baseline MRAM/PCM-based packet buffer by up to 16% and 58% for 1.6 and 3.2 GHz IO bus clock, respectively. Furthermore, the latency-aware multi-retention time MRAM-based packet buffer structure identifies the latency requirement of different packets and writes the packets by guaranteeing a predetermined minimum retention time. With this scheme, an additional speedup of up to 5.27% is achieved over our proposed two-tier partitioned packet buffer.

The subsequent sections of this paper are structured as follows. In Related Work (Section II), the background information concerning packet processors and packet buffers is presented, accompanied by a discussion on the fundamental principles of Magnetic Random Access Memory (MRAM). The Methodology (Section III) elucidates the proposed packet buffer, which is based on Multi-Retention Time MRAM. In Results and Discussion (Section IV), the key findings and significant outcomes are presented. Lastly, in the concluding section (Section V), a summary of the paper and avenues for future research are provided.

II. RELATED WORK

This section presents the baseline architectures of the packet processor and packet buffer along with previous works. Furthermore, it provides an overview of MRAM fundamentals, encompassing its structure, key parameters, and operations.

A. PACKET PROCESSOR AND PACKET BUFFER

An internet router/switch consists of linecards, which serve as the main packet processing engines [6], [20]. A linecard typically includes a packet processor, a lookup table, and a packet buffer, as shown in Fig. 1. The packet processor is a many-core processor used in internet routers/switches that processes and classifies incoming packets. During this process, the packets are stored in the packet buffer. The packet buffer, implemented with DRAM memories, provides buffering and supports traffic at speeds of hundreds/thousands of Gbps. The amount of buffering is typically determined by TCP protocols, as they account for the majority of internet traffic [9], [21]. A packet buffer should temporarily store packets for a duration of RTT × interface bandwidth (e.g., 100 ms × 640 Gbps = 8 Gbytes).

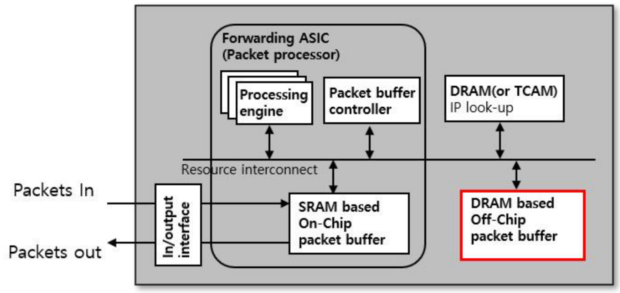

FIGURE 1. Linecard with a packet processor and a packet buffer.

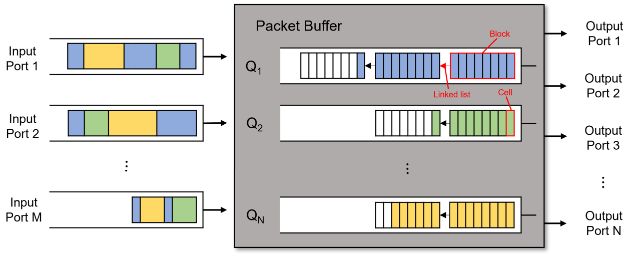

FIGURE 2. Packet data structure mapped on packet buffer.

In the packet buffer, the packet buffer controller (memory controller) writes incoming packet traffic to the packet buffer. Simultaneously, an output queue scheduler schedules a queue based on a scheduling algorithm. Packets stored in the scheduled queue are read from the packet buffer. To maximize the memory bandwidth utilization of a packet buffer, data structures for packets stored in the output queues are carefully designed to exploit the bank-level parallelism and row buffer locality of memory chips, as shown in Figure 2.

For instance, an output queue is constructed as a linked list of blocks. Each block consists of multiple cells, which represent fixed chunks, such as 64 bytes, of packet payload data. In the figure, a block consists of 8 cells. Typically, different cells belonging to the same packet are mapped to multiple banks or a single row, allowing the reading and writing of a single packet to exploit bank-level parallelism or row buffer locality.

To design scalable high bandwidth packet buffer for routers/switches, various DRAM based memory architectures with SRAM buffers have been proposed [7], [8], [9], [10], [22]. All these works basically differ in how to design SRAM input/output buffers in the memory controller to support the DRAM-based memory (packet buffer). These SRAM input/output buffers absorb temporary overload due to congestion from bank conflicts and improve read/write operations to DRAM.

SRAM buffers in [9] and [10] maintain per-queue queueing structures which are costly with increasing number of output queues. Instead, Lee et al. use a per-bank queueing structure and reorder bank scheduling to reduce the cost of increasing

TABLE 1. Packet buffer designs.

Methods	DRAM with SRAM buffer [7] [8] [9] [10]	SRAM-DRAM hybrid [22]	PCM-MRAM hybrid [11]	MR-MRAM (Proposed)
Bandwidth Scalability	Medium	High	Medium	High
Data Retention on Power Loss (Availability)	No	No	Yes	Yes

buffer size [7]. Mutter also uses a per-bank queueing structure, per-flow round robin dispatcher, requester, and minimum buffer delay scheduling to maximize the parallel access of DRAM modules [8]. However, the DRAM-based packet buffer suffers from degrading row buffer locality and worsening gap between IO bus speed and cell access latency. And the small SRAM buffers in the memory controller are only buffering temporary congestion and not prevent interference-inducing traffic from thrashing the DRAM's row buffer as the data in the buffer has to be written to DRAM ultimately through the DRAM's row buffer.

To address this issue, a method using SRAM and DRAM as parallel memories in the packet buffer is proposed [22]. In the method, packets degrading row buffer locality are mapped to the SRAM portion to enhance the scalability. However, all existing methods using volatile memory suffer from data loss upon power loss and provide limited availability.

To ensure both high bandwidth scalability and availability, a hybrid memory-based packet buffer consisting of MRAM and PCM memory has been introduced [11]. However, PCM is not scalable in terms of bandwidth, as the gap between IO clock speed and array cell access speed increases. Previous works are summarized in the Table 1.

B. MRAM BASICS

- **STT-MRAM:** An STT-MRAM cell consists of an access transistor and a magnetic tunnel junction (MTJ). The MTJ comprises two magnetic layers (free layer and reference layer) separated by an energy barrier. The direction of magnetism and the stored value in the free layer are changed by the current flowing through the MTJ. The characteristics of MRAMs are defined by important parameters, including the thermal stability factor, retention time, bit error rate, write current, read/write latency, and endurance. Formulas for some of these crucial parameters are provided below, extracted from [15].
- **Thermal stability factor (TSF):** In STT-MRAM, the thermal stability factor (TSF) refers to the stability of the magnetism direction in the MTJ. Its equation is shown in Eqn.(1). The TSF can be controlled by design parameters such as the size, shape, and material type of the MTJ. Controlling the TSF affects retention time, read/write current, read/write latency, cell size, and read/write energy. A larger TSF value increases the data retention time but also increases the write latency by requiring a higher amount of current to change the magnetism. Conversely, reducing the TSF can decrease the write latency at the expense of shorter retention time. To build an MRAM-based main memory, researchers reduce the size of cells, which reduces the write energy and retention time [18].

$$\Delta(\mathrm{TSF}) = \frac{E_b}{k_B T} = \frac{H_K M_S V}{2k_B T} \tag{1}$$

where:
E_b = energy barrier
k_B = Boltzmann constant
T = temperature
H_K = anisotropy field term
M_S = saturation magnetization
V = MTJ volume

- **Expected Retention Time:** The expected retention time is the anticipated time for a bit-flip to occur in the free layer of the MTJ. It is a function of the TSF, as shown in Eqn.(2). When the TSF increases or decreases, the expected retention time also increases or decreases exponentially.

$$\tau = \tau_0 \cdot \exp(\frac{E_b}{k_B T}) = \tau_0 \cdot \exp(\Delta) \tag{2}$$

where:
τ = relaxation constant (or expected retention time)
τ_0 = operating frequency

- **Retention Probability:** The retention probability is determined by Eqn.(3). It calculates the probability of a bit-flip occurring during the target retention time, t, given an expected retention time, τ.

$$P_{ret} = 1 - \exp(-\frac{t}{\tau}) \tag{3}$$

where:
P_{ret} = probability of a bit-flip
t = target retention time

- **Critical Current:** The critical current (J_{c0}^{PP}) in Eqn.(4) is the minimum current required to switch the content of a cell and is proportional to the TSF. When the TSF decreases, the critical current also decreases. A lower critical current reduces the time needed to write a cell, thereby reducing the write latency.

$$J_{c0}^{PP} = \frac{4\alpha\gamma e k_B T}{\mu_B g}\Delta \tag{4}$$

where:
α = Gilbert damping constant
γ = gyromagnetic constant
e = electron charge
μ_B = Bohr magneton constant
g = spin transfer efficiency

1) MRAM OPERATIONS

MRAM has asymmetric read and write latency. The write time in STT-MRAM is typically longer than the read time because the write process involves not just sensing the state of a memory cell, but also applying a current to switch the magnetic moment in the free layer. This current must be of sufficient magnitude to overcome the thermal energy of the system and align the magnetic moment in the desired direction. Typically, write latency is related to the TSF value.

On the other hand, the read time in STT-MRAM is relatively fast because it involves only sensing the resistance of the magnetic tunnel junction, which is dependent on the orientation of the magnetic moments in the two layers. This resistance can be measured using a small voltage, without the need for a large current.

III. METHODOLOGY

This section delves into the discussion of the advantages of multi-retention time MRAM by analyzing the trade-offs between retention time and write latency. Furthermore, two proposed packet buffer structures that incorporate multi-retention time MRAM are introduced.

A. MULTI-RETENTION TIME MRAM (MR-MRAM)

- **Retention Time and Bit Error Rate:** The expected retention time and the probability of a bit flip within a given time limit are shown in Eqn.(2) and (3). The expected retention time is exponentially proportional to the thermal stability factor (TSF). A typical MRAM chip is designed to guarantee 10-20 years of retention time for the bit error rate of 10^{-6}. In packet buffer applications, the expected buffering time for packets is usually in the range of hundreds of milliseconds in the worst case, as packets with larger delays are dropped and discarded. Therefore, it is possible to reduce the requirement for the expected retention time of stored packet data, which allows a reduction in TSF. The reduced TSF can be used to decrease the cell size, write current, or both.
- **Write Latency:** The write latency (t_p) in Eqn.(5) is determined by the current applied to the cell (J_c^{PP}) and the critical current (J_{c0}^{PP}). By reducing J_{c0}^{PP} through a smaller TSF, the write latency (t_p) can be reduced while maintaining the same current (J^{PP}) for cell switching. Alternatively, keeping the same t_p allows a reduction in J_c^{PP}, resulting in lower write energy consumption.

$$J_c^{PP} = J_{c0}^{PP}(1 + \frac{\epsilon}{t_p} ln \frac{\pi}{2\theta_0}) \tag{5}$$

where:

ϵ	= characteristic relaxation time constant
t_p	= write pulse width
θ_0	= initial angle variance

- **Trade-off between Retention Time and Write Latency:** Retention time is controlled by the TSF. When the target retention time and retention probability are given, the expected retention time (τ) can be calculated using Eqn.(3). Once τ is determined, TSF (Δ) can be calculated using Eqn.(2). If the target retention time can be reduced, the TSF can be decreased as well. A reduced TSF can be used to decrease the critical current using Eqn.(4).

  The primary discovery of this study is that the diminished critical current, as described by Eqn. (5), can be employed to decrease the write latency assuming a fixed applied current. Stated differently, there exists a trade-off between retention time and write latency. By diminishing the retention time, at the expense of reducing TSF, it is possible to decrease the write latency. This serves as one of the principal motivations for our paper, in addition to the advantages of enhanced capacity and reduced energy consumption associated with a smaller TSF.

  Table 2 shows the technology node, size, retention time, and performance of the latest MRAM chips. Recently, typical write latency for MRAM chips guaranteeing a 10-year retention time is around 30 ns to 50 ns. Embedded MRAMs can achieve a write time as low as 4 ns by reducing the retention time to less than 1 minute [23].

TABLE 2. Publicised MRAM specifications.

Company	Samsung [16]	TSMC [17]	SKhynix [14]	Everspin [13]	IBM [23]
Node (nm)	28 FD-SOI	22	–	28	14
Size	8Mb	4Mbx8	4Gb	1Gb	2Mb
Read latency (ns)	30-50	10	30	10-40	–
Write latency (ns)	30-50	30	30	10-13	4
Retention time	few minutes	10 yr	–	20 yr	< 1 minute

B. MULTI-RETENTION TIME MRAM-BASED PACKET BUFFER

1) OVERALL SCHEME

The proposed packet buffer is shown in Fig. 3 and Fig. 4. A single channel of a packet buffer consists of a buffer device and multiple MRAM chips, similar to a DRAM-based dual in-line memory module (DIMM). The buffer device includes an embedded MRAM, which ranges from hundreds of kilobytes to a few megabytes and has fast access time. The capacity-optimized MRAM is incorporated in the large MRAM portion. This arrangement, known as a two-tier packet buffer structure, encompasses both small and fast embedded MRAM and large and slow standalone MRAM, as detailed in section III-B2.

In the multi-retention time MRAM-based packet buffer, the large MRAM portion is implemented with multi-retention time MRAM parts. The large MRAM portion is constructed using depth expansion in DIMM, utilizing both short-retention time and long-retention time MRAM parts. To meet the latency requirements of individual packets, the proposed latency-aware packet mapping method assigns

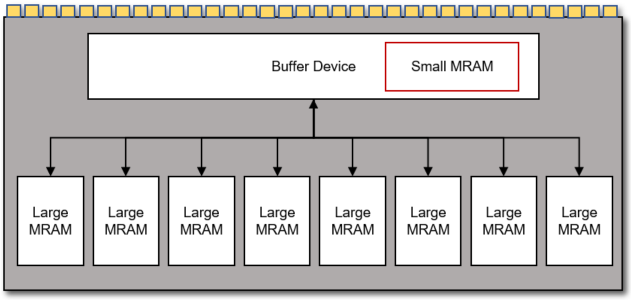

FIGURE 3. Two-tier MRAM packet buffer (TT-PB) consisting of small/fast embedded MRAM in the buffer device and large/slow standalone MRAM chips.

packets to specific parts with different retention times. It is important to note that different retention time parts exhibit distinct write latencies, as elaborated upon in section III-B3.

2) TWO-TIER PACKET BUFFER STRUCTURE (TT-PB)

The proposed two-tier MRAM-based memory packet buffer (TT-PB) consists of the small embedded MRAM and the large standalone MRAM portion (shown in Fig. 3), similar to [11]. Both portions share the I/O pins of the dual in-line memory module (DIMM). The embedded MRAM is integrated into the buffer device of DIMM, which is similar to [24]. The construction of high-capacity and high-performance MRAM, surpassing the capabilities of DRAM, poses a significant challenge. Reasonably small MRAM is known to perform comparably to SRAM, except for the write delay.

In the proposed packet buffer structure, the embedded MRAM portion in the buffer device is small compared to the large standalone MRAM portion. The embedded MRAM portion is a few megabytes and exhibits fast access time [23], while the standalone MRAM portion is large and slow. The target retention time for commercial MRAM is typically 10-20 years. However, in packet buffers, packets are written and read from the packet buffer within the round-trip time (RTT) of network flows, which is around 100 milliseconds. Therefore, the retention time requirement is relaxed in order to decrease TSF, resulting in increased capacity (attributable to smaller cells) and reduced write latency for the larger MRAM section.

Tiered MRAM memories provide two major functions. First, the embedded MRAM portion captures the working set of packets when queues are not congested. In typical scenarios, most packets enter and leave routers/switches with very little delay, except for a few congested interfaces (queues). This working set captured by the embedded MRAM can reduce the writes to the large/slow MRAM portion. The large MRAM portion is mostly used when extensive congestion occurs. Second, embedded MRAM can be used to absorb the packets that degrade the row buffer locality of the large MRAM portion. Writing small packets (e.g., 40-byte TCP acknowledgment packets) causes frequent row buffer misses in the large/slow MRAM portion. Thus, the embedded MRAM portion enhances the performance of the large MRAM portion by absorbing traffic that causes row buffer misses. Packets are stored in either the embedded MRAM or the large MRAM portion.

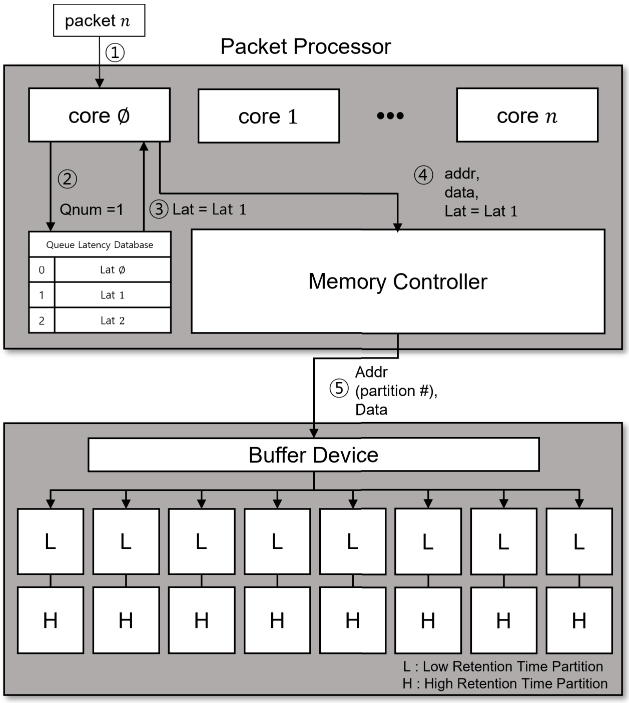

FIGURE 4. Operations of latency-aware multi-retention time MRAM based packet buffer (MR-PB).

3) LATENCY-AWARE PACKET MAPPING WITH MULTI-RETENTION TIME MRAM-BASED PACKET BUFFER (MR-PB)

Typical routers/switches support hundreds to millions of output queues. Each network flow (e.g., TCP/UDP connections) is mapped to an output queue to guarantee quality of service (e.g., bandwidth or latency). According to the service level agreement, routers/switches provide differentiated services to different queues. For instance, video traffic for interactive video communication applications should guarantee tens of milliseconds of end-to-end delay. Assuming that these packets go through multiple hops (routers/switches), each router/switch in the network path ends up processing packets in less than a millisecond.

Consequently, packets are written to and read from the packet buffer within the specified latency inside routers/switches. If the usage of routers/switches (including the number of flows and their QoS parameters) is known, the maximum buffering capacity for different flows with diverse buffering latency requirements (i.e., buffering time) can be determined.

To maximize the utilization of the aforementioned unique memory usage in routers/switches, the standalone portion (large MRAM) of the packet buffer is partitioned into multiple partitions. Each partition can be implemented with MRAM chips with different target retention times. The depth expansion method in DIMM can be used to build it. Figure 4 depicts the overall scheme of the partitioned packet buffer with an example.

In the packet processor, when a packet arrives, the processor core classifies the output queue of the packet ①. From the associated queue number (Q1), its latency requirement information (Lat1) is retrieved from the database ②. This database already exists in routers/switches as they need to keep track of latency to guarantee quality of service. The retrieved latency is used when the packet processor determines the proper partition to ensure the retention time determined by the latency requirement ③. After this, the processor core sends the address, data, and latency information (e.g. Lat1) to the packet buffer memory controller ④. The packet buffer memory controller sends address and command information to DIMMs, which select the corresponding partition that satisfies the retention time requirement ⑤. The size of partitions can be determined according to the usage plan of routers/switches.

IV. RESULTS AND DISCUSSION

In this section, the setup for the simulation environment will be discussed, encompassing the packet memory system simulator and traffic generator. Subsequently, the simulation results will be presented for the two-tier packet buffer structure and the latency-aware multi-retention time MRAM-based packet buffer, accompanied by a discussion of the associated costs for each scheme.

A. ENVIRONMENT

A simulator was implemented to assess the proposed scheme, comprising an in-house memory traffic generator and a memory system simulator. The memory system simulator is based on DRAMSim2 [25] as we need to compare the proposed scheme with a conventional DRAM-based one and the MRAM/PCM-based packet buffer (baseline) [11]. Table 3 shows the parameter values for the various packet buffer systems, including the DRAM-based packet buffer, MRAM/PCM-based packet buffer, and the proposed MR-MRAM-based packet buffer, for comparison.

The router/switch receives internet traffic in bursts as the TCP mechanism relies on traffic control to avoid severe congestion in routes. To emulate this behavior, the memory traffic generator stresses the packet buffer system by introducing periodic congestion, referred to as the congestion period (CP) [26]. The congestion period, during which the input traffic is periodically overloaded, is set from 1 to 10 ms to stress the packet buffer. However, for brevity, we only report the results for the worst case (10 ms).

B. PERFORMANCE RESULTS

1) TWO-TIER PACKET BUFFER STRUCTURE (TT-PB)

In this test, the two-tier packet buffer is utilized without incorporating the multi-retention time MRAM parts. The achieved memory bandwidth is measured with respect to the IO bus clock speed and the number of output queues. The write latency for the embedded MRAM portion is 5 ns since its size is small and its required retention time is also small [23]. The write latency for the large MRAM portion is fixed at 62.5 ns, obtained from NVSim [28], to accommodate an 8-Gbyte packet buffer (8 channels).

TABLE 3. System configuration.

Memory controller	DDR4 1066/1600 MHz or DDR5 3200 MHz; per-bank command queues; FIFO scheduling; open row policy
DIMM	8 channel; 1 rank; 8 chips on a DIMM
DRAM	8 Gbytes; 16 banks per chip; 2-Kbyte row buffer per bank; tRC = 48.75 ns
MRAM/PCM (baseline) [11]	MRAM: 512 Kbytes – 2 Mbytes per channel; tWR = 5 ns [23] PCM: 8 Gbytes; 16 banks per chip; 2-Kbyte row buffer per bank; single-level cell (SLC); tRCD = 40 ns; tWR = 220 ns [27]
MR-MRAM	embedded MRAM: 512 Kbytes – 2 Mbytes per channel; tWR = 5 ns [23] stand-alone MRAM: 8 Gbytes; 16 banks per chip; 512-byte row buffer per bank; tWR = 12.5 ns – 62.5 ns

The baseline measurement is taken with a 1.066 GHz bus IO clock. Two additional bus IO clocks are used: 1.6 GHz and 3.2 GHz. The embedded MRAM size is set to 2 megabytes, which is the optimal size reported in [11]. The number of output queues ranges from 1,000 to 100,000.

In Fig. 5, it is observed that as the bus IO clock frequency increases, a greater level of speedup is attained. This is because the slow memory array of PCM in the baseline MRAM/PCM-based packet buffer [11] cannot keep up with the increasing IO clock speed. Speedups of up to 16% and 58% are achieved for 1.6 GHz and 3.2 GHz IO bus clocks, respectively. Furthermore, as the number of queues increases, the packet traffic is spread across multiple rows of cell arrays, reducing row buffer locality and making the slow PCM write latency more noticeable.

2) LATENCY-AWARE MAPPING WITH MULTI-RETENTION TIME MRAM-BASED PACKET BUFFER (MR-PB)

Typically, different packets destined for different queues have associated latency requirements. In general, packets may stay in the routers/switches for a much shorter time than 100 msec. For instance, high-priority packets or video traffic packets stay in the routers for less than 1 msec. With a 1 msec retention time, the write latency for the MRAM can be further reduced.

For this test, the large MRAM portion is divided into two parts: a short retention time MRAM partition (write latency = 12.5 ns) and a long retention time MRAM partition (write latency = 62.5 ns). The size of the two partitions is based on the configuration of routers/switches. For example, if low-latency traffic (1 msec packet buffer delay) and high-latency traffic (100 msec packet buffer delay) account for 50% each of the total traffic, the two partitions are sized accordingly. In this experiment, the low retention time MRAM partition (with a 1 msec packet buffer delay) is varied to encompass 10% to 90% of the total traffic.

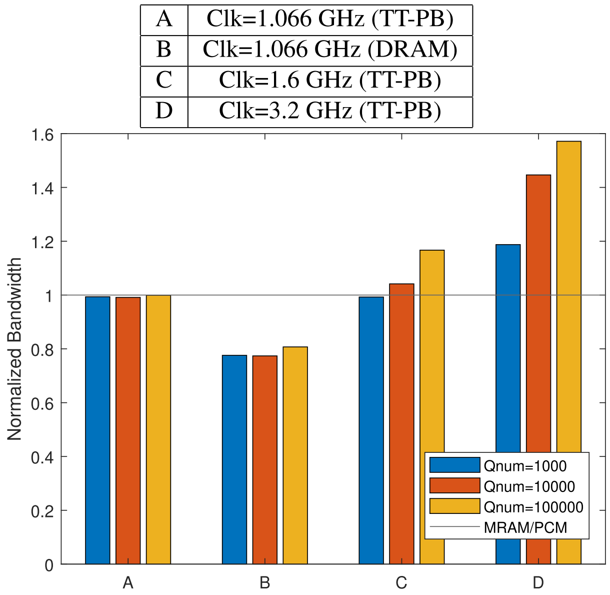

FIGURE 5. Normalized bandwidth of two-tier packet buffer with respect to Q numbers for different bus IO clock speeds (embedded MRAM size = 2 Mbytes, congestion period = 10 msec).

The results are shown in Fig. 6. The X-axis represents the number of queues ranging from 1,000 to 100,000. The different colors in the legend represent the weight of low-latency packet traffic ranging from 10% to 90%. The results are normalized to the MRAM/PCM-based packet buffer [11] with the same embedded MRAM size (2 Mbytes).

For a small number of queues, the high row buffer locality makes the performance insensitive to the write latency of the large MRAM portion. However, for a large number of queues (100,000), the speedup gradually increases with respect to the amount of low-latency packets. The speedup over the baseline (MRAM/PCM-based packet buffer) is as much as 18.53%, 49.66%, and 60.04% for queue numbers of 1,000, 10,000, and 100,000 respectively. When the low-latency packets account for 90%, MR-PB performs -0.57%, 5.27%, and 4.42% better than TT-PB for queue numbers of 1,000, 10,000, and 100,000 respectively.

C. COST EVALUATION

The area cost is compared among the multi-retention time MRAM-based packet buffer, the MRAM/PCM-based packet buffer [11], and the DRAM-based packet buffer. The results are reported in Table. 4. We use NVSim [28] to estimate the area for PCM and MRAM memories and Cacti [29] for DRAM memory, using a 22 nm process technology.

The MRAM/PCM-based packet buffer consists of an MRAM and PCM portion, with the area for the MRAM portion shown inside the parenthesis. MRAM/PCM has the least area cost due to the high cell density of PCM chips. The area cost of MR-MRAM is not significantly worse than that of the DRAM-based packet buffer. An MRAM cell size of 54 F^2 is used in NVSim. According to [18], reducing the cell size to 10 F^2 is feasible, which would allow for further area reduction. Both MR-MRAM and MRAM/PCM report

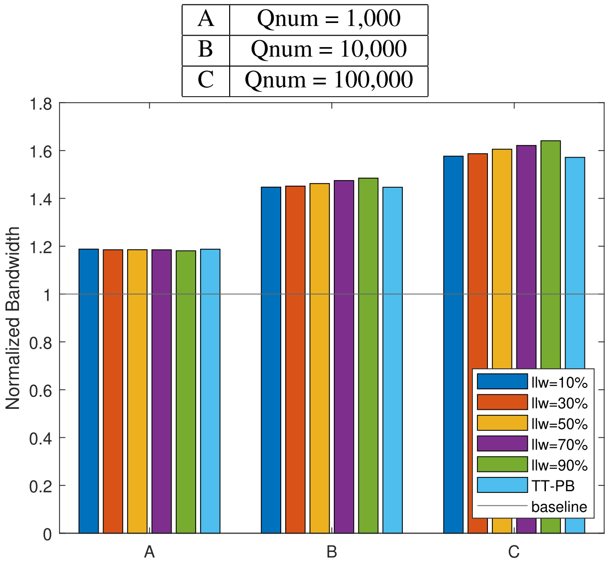

FIGURE 6. Normalized bandwidth of latency-aware multi-retention time MRAM-based packet buffer with respect to Q numbers while varying the weight for low latency packets (embedded MRAM size = 2 Mbytes, congestion period = 10 msec, IO bus clock = 3.2 GHz).

TABLE 4. Area cost per channel.

MR-MRAM	MRAM/PCM	DRAM
234.12(0.64) mm^2	55.28(0.64) mm^2	174.56 mm^2

the same area cost for the embedded MRAM portion, as both designs use 2 Mbytes.

V. CONCLUSION

In this paper, a review is conducted on various existing works that aim to implement high-bandwidth and highly available packet buffers for internet routers/switches, along with their limitations. To overcome these limitations, a proposal is made for multi-retention time MRAM-based packet buffer architectures. The structure of the two-tier packet buffer (TT-PB) involves the integration of a small/fast embedded MRAM with a large/slow MRAM. The structure of the multi-retention time MRAM-based packet buffer (MR-PB) utilizes multiple MRAM parts with different retention times. The MR-PB identifies the latency requirements of different packets and ensures predetermined minimum data retention times by mapping the packets to the appropriate partitions.

Both TT-PB and MR-PB demonstrate significant performance gains over existing packet buffers using non-volatile and volatile memories. TT-PB outperforms the baseline MRAM/PCM-based packet buffer by up to 16% and 58% for 1.6 and 3.2 GHz IO bus clock, respectively. Furthermore, with MR-PB, an additional speedup of up to 5.27% is achieved over TT-PB.

As a potential area for future research, the main idea of this study can be applied to various streaming data applications characterized by bounded latency between write and read operations. This approach enables the utilization of the suitable retention time of MRAM, thereby facilitating higher memory performance.

REFERENCES

[1] R. Govindan, I. Minei, M. Kallahalla, B. Koley, and A. Vahdat, ''Evolve or die: High-availability design principles drawn from Googles network infrastructure,'' in *Proc. ACM SIGCOMM Conf.*, Florianópolis, Brazil, Aug. 2016, pp. 58–72.

[2] C. Oggerino, ''High availability network fundamentals,'' in *A Practical Guide to Modeling and Designing Reliable Networks*, 1st ed. San Rafel, CA, USA: Cisco Press, 2001.

[3] C. She, Z. Chen, C. Yang, T. Q. S. Quek, Y. Li, and B. Vucetic, ''Improving network availability of ultra-reliable and low-latency communications with multi-connectivity,'' *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5482–5496, Nov. 2018, doi: 10.1109/TCOMM.2018.2851244.

[4] K. M. J. Sonderegger, O. Blomberg, and S. Palislamovic, *JUNOS High Availability: Best Practices for High Network Uptime*. Newton, MA, USA: O'Reilly Media, 2009.

[5] IXIA. (2012). *Measuring Latency in Equity Transactions*. Accessed: Apr. 5, 2023. [Online]. Available: https://support.ixiacom.com/sites/default/files/resources/whitepaper/lowlatencywhitepaperbooklet.pdf

[6] Cisco Systems. (2023). *Cisco Network Convergence System 6008 Single-Chassis System Data Sheet*. Accessed: Apr. 5, 2023. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/routers/network-convergence-system-6000-series-routers/data_sheet_c78-728048.html

[7] H.-J. Lee and E.-Y. Chung, ''Scalable QoS-aware memory controller for high-bandwidth packet memory,'' *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 16, no. 3, pp. 289–301, Mar. 2008, doi: 10.1109/TVLSI.2007.915367.

[8] A. Mutter, ''A novel hybrid memory architecture for high-speed packet buffers in network nodes,'' Ph.D. dissertation, Inst. Commun. Netw. Comput. Eng., Univ. Stuttgart, Stuttgart, Germany, 2012.

[9] S. Iyer, R. R. Kompella, and N. McKeown, ''Designing packet buffers for router linecards,'' *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 705–717, Jun. 2008, doi: 10.1109/TNET.2008.923720.

[10] J. Garcia-Vidal, M. March, L. Cerda, J. Corbal, and M. Valero, ''A DRAM/SRAM memory scheme for fast packet buffers,'' *IEEE Trans. Comput.*, vol. 55, no. 5, pp. 588–602, May 2006, doi: 10.1109/TC.2006.63.

[11] Y. Song, J. Hwang, I. Jo, and H. Lee, ''Highly available packet buffer design with hybrid nonvolatile memory,'' *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 29, no. 11, pp. 2008–2012, Nov. 2021, doi: 10.1109/TVLSI.2021.3116272.

[12] S. Aggarwal, H. Almasi, M. DeHerrera, B. Hughes, S. Ikegawa, J. Janesky, H. K. Lee, H. Lu, F. B. Mancoff, K. Nagel, G. Shimon, J. J. Sun, T. Andre, and S. M. Alam, ''Demonstration of a reliable 1 Gb standalone spin-transfer torque MRAM for industrial applications,'' in *IEDM Tech. Dig.*, San Francisco, CA, USA, Dec. 2019, pp. 2.1.1–2.1.4, doi: 10.1109/IEDM19573.2019.8993516.

[13] J. J. Sun, M. DeHerrera, B. Hughes, S. Ikegawa, H. K. Lee, F. B. Mancoff, K. Nagel, G. Shimon, S. M. Alam, D. Houssameddine, and S. Aggarwal, ''Commercialization of 1 Gb standalone spin-transfer torque MRAM,'' in *Proc. IEEE Int. Memory Workshop (IMW)*, Dresden, Germany, May 2021, pp. 1–4, doi: 10.1109/IMW51353.2021.9439616.

[14] S.-W. Chung, T. Kishi, J. W. Park, M. Yoshikawa, K. S. Park, T. Nagase, K. Sunouchi, H. Kanaya, G. C. Kim, K. Noma, M. S. Lee, A. Yamamoto, K. M. Rho, K. Tsuchida, S. J. Chung, J. Y. Yi, H. S. Kim, Y. S. Chun, H. Oyamatsu, and S. J. Hong, ''4Gbit density STT-MRAM using perpendicular MTJ realized with compact cell structure,'' in *IEDM Tech. Dig.*, San Francisco, CA, USA, Dec. 2016, pp. 27.1.1–27.1.4, doi: 10.1109/IEDM.2016.7838490.

[15] L. Wu, M. Taouil, S. Rao, E. Jan Marinissen, and S. Hamdioui, ''Survey on STT-MRAM testing: Failure mechanisms, fault models, and tests,'' 2020, *arXiv:2001.05463*.

[16] S. H. Han et al., ''28-nm 0.08 mm^2/Mb embedded MRAM for frame buffer memory,'' in *IEDM Tech. Dig.*, San Francisco, CA, USA, Dec. 2020, pp. 11.2.1–11.2.4, doi: 10.1109/IEDM13553.2020.9372040.

[17] Y.-D. Chih, Y.-C. Shih, C.-F. Lee, Y.-A. Chang, P.-H. Lee, H.-J. Lin, Y.-L. Chen, C.-P. Lo, M.-C. Shih, K.-H. Shen, H. Chuang, and T.-Y. J. Chang, ''A 22 nm 32 Mb embedded STT-MRAM with 10 ns read speed, 1M cycle write endurance, 10 years retention at 150 °C and high immunity to magnetic field interference,'' in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2020, pp. 222–224, doi: 10.1109/ISSCC19947.2020.9062955.

[18] M. Shihab, J. Zhang, S. Gao, J. Sloan, and M. Jung, ''Couture: Tailoring STT-MRAM for persistent main memory,'' in *Proc. 4th Workshop Interact. NVM/Flash Operating Syst. Workloads (INFLOW)*, Savannah, GA, USA, Nov. 2016, pp. 1–6.

[19] J. Park, M. Lee, S. Kim, M. Ju, and J. Hong, ''MH cache: A multi-retention STT-RAM-based low-power last-level cache for mobile hardware rendering systems,'' *ACM Trans. Archit. Code Optim.*, vol. 16, no. 3, pp. 1–26, Jul. 2019, doi: 10.1145/3328520.

[20] *NCS 5500 Modular Platform Architecture*, Cisco, San Jose, CA, USA, 2020.

[21] S. Hassayoun and D. Ros, ''Loss synchronization, router buffer sizing and high-speed TCP versions: Adding RED to the mix,'' in *Proc. IEEE 34th Conf. Local Comput. Netw.*, Zurich, Switzerland, Oct. 2009, pp. 569–576, doi: 10.1109/LCN.2009.5355190.

[22] Y. Song, D. Choi, and H. Lee, ''Designing a high performance SRAM-DRAM hybrid memory architecture for packet buffers,'' *IEICE Trans. Electron.*, vol. 102, no. 12, pp. 849–852, Dec. 2019, doi: 10.1587/transele.2019ECS6003.

[23] D. Edelstein et al., ''A 14 nm embedded STT-MRAM CMOS technology,'' in *IEDM Tech. Dig.*, San Francisco, CA, USA, Dec. 2020, pp. 11.5.1–11.5.4, doi: 10.1109/IEDM13553.2020.9371922.

[24] Y. Kwon, Y. Lee, and M. Rhu, ''TensorDIMM: A practical near-memory processing architecture for embeddings and tensor operations in deep learning,'' in *Proc. 52nd Annu. IEEE/ACM Int. Symp. Microarchitecture*, Columbus, OH, USA, Oct. 2019, pp. 740–753, doi: 10.1145/3352460.3358284.

[25] P. Rosenfeld, E. Cooper-Balis, and B. Jacob, ''DRAMSim2: A cycle accurate memory system simulator,'' *IEEE Comput. Archit. Lett.*, vol. 10, no. 1, pp. 16–19, Jan. 2011, doi: 10.1109/L-CA.2011.4.

[26] Agilent Techonology. (2007). *The Journal of Internet Test Methodologies*. Accessed: Apr. 5, 2023. [Online]. Available: https://test4tot.com/wp-content/uploads/2016/10/The-Journal-of-Internet-test-methodologies.pdf

[27] N. S. Kim, C. Song, W. Y. Cho, J. Huang, and M. Jung, ''LL-PCM: Low-latency phase change memory architecture,'' in *Proc. 56th ACM/IEEE Design Autom. Conf. (DAC)*, Las Vegas, NV, USA, Jun. 2019, pp. 1–6.

[28] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi, ''NVSim: A circuit-level performance, energy, and area model for emerging nonvolatile memory,'' *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 7, pp. 994–1007, Jul. 2012, doi: 10.1109/TCAD.2012.2185930.

[29] HP Labs. (2009). *CACTI 6.5*. Accessed: Apr. 5, 2023. [Online]. Available: https://www.hpl.hp.com/research/cacti/

YONGWOON SONG received the B.S. and Ph.D. degrees in computer science and engineering from Sogang University, Seoul, Republic of Korea, in 2012 and 2023, respectively. His research interests include computer architecture, memory systems, non-volatile memory, and embedded systems.

MUNHYUNG LEE received the B.S. and M.S. degrees in computer science and engineering from Sogang University, Seoul, Republic of Korea, in 2021 and 2023, respectively. His research interests include memory systems, non-volatile memory, automotive embedded systems, and deep neural networks.

HYUKJUN LEE (Member, IEEE) received the B.S. degree in computer science and engineering from the University of Southern California, Los Angeles, CA, USA, in 1993, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA, in 1995 and 2001, respectively.

From 2001 to 2011, he was a Senior Engineer with Cisco Systems Inc. Since 2011, he has been a Professor with the Computer Science and Engineering Department, Sogang University, Seoul, South Korea. He published numerous articles in the areas of memory systems, embedded systems, operating systems, and deep neural network accelerators.