

# Performance Analysis on Structure of Racetrack Memory

Hongbin Zhang\*, Chao Zhang†, Qingda Hu\*, Chengmo Yang‡ and Jiwu Shu\*

\*Department of Computer Science and Technology, Tsinghua University, Beijing, China

†Center for Energy-efficient Computing and Applications, Peking University, Beijing, China

‡Department of Electrical and Computer Engineering, University of Delaware, Newark, US

{zhanghb10, huqdl13}@mails.tsinghua.edu.cn, zhang.chao@pku.edu.cn, chengmo@udel.edu, shujw@tsinghua.edu.cn

**Abstract**—Racetrack Memory(RM) has attracted abundant attention of memory researchers recently. RM can achieve ultra-high storage density, fast access velocity and non-volatility. Former research has demonstrated that RM has potential to serve as on-chip cache or main memory. However, RM has more flexibility and difficulty in design space of main memory because it has more device level design parameters. The layout of macro unit (MU) needs trade-off among area, access performance and energy consumption, and its shift operation introduces extra dimension of design space. In this paper, we explore these design parameters and analyze their relationship in memory design space in both device and system levels. Based on the results, we also propose a hybrid MU structure to further optimize read intensive applications. Experimental results demonstrated the existence of regularity between design parameters and performance features. The optimized layout of racetrack MU is suggested for application areas such as big-data and IoT which need cost-effective and energy-efficient memory respectively. Together with hybrid MU structures, RM can be designed with more flexibility so that specific structures are suitable for specific applications which make “All stack optimization” possible in memory structure level.

## I. INTRODUCTION

Accompanying the development of cloud computing and big data technology, more and more CPU cores are being deployed in large scale computer systems in data centers. The gap between computation throughput and bandwidth of memory hierarchies continues to increase, and high static refresh power consumption further hinders the development of DRAM. With this trend, computer architects are seeking to bridge the gap with alternatives to traditional SRAM/DRAM memory technologies. Emerging non-volatile memory (NVM) technologies, such as phase change memory (PCM), spin-transfer torque RAM (STT-RAM), and resistance RAM (RRAM) have received significant attention regarding cache and main memory design [1], [2], [3], [4], [5]. Approaches have been proposed to improve the density, performance, energy efficiency, and stability of on-chip caches or main memory. However, the limitations of such approaches challenge a complete replacement of DRAM with them. For example, PCM can provide a higher storage density with lower leakage power consumption, but its poor write latency and limited lifetime are its main obstacle. STT-RAM provides a higher access performance than DRAM, but its density is several times smaller than that of DRAM, which prevents it from completely replacing DRAM.

Recently, a newly emerging NVM called racetrack memory (RM), which is also known as domain wall memory (DWM), has attracted significant attention of researchers. Previous research has demonstrated that this approach can achieve ultra-high density by integrating multiple domains in a tape-like nanowire [7], [12], [13]. Owing to its comparable access latency of SRAM and high write endurance, RM is a promising candidate for on-chip memory or caching [23], [14], [17], [21]. There also exist some approaches to compose racetrack as the main memory. A shift-sense address mapping policy (SSAM) has been proposed for reducing shift operations in racetrack-based main memory systems [6]. An ultra-low power memory based big-data computing platform is proposed, in which RM is designed to undertake part of logical computing like XOR while it works as main memory [18]. At the same time, page table is also optimized by leveraging the positions of access ports in RM to differentiate the state of page table entries [15]. A RM based stack operation mechanism is proposed since the sequential access structure of RM makes it well suited for a stack whose accesses display high temporal locality [16]. Furthermore, data placement mechanisms for optimizing its access latency and energy cost have been researched intensively [19], [20], [16].

Former research have studied how to design RM as cache or main memory and focused on how to reduce shift intensity, either at the system level or compiler level, aiming to reduce the latency and energy cost. However, RM technology is still new and there are many uncertainties in evaluating their actual impact on future memory hierarchy design. There is rare research to study the relationship between RM basic structure and performance of system at high level. Prior work [7] has proposed an organization involving overlapping RM cells, called a macro unit (MU), as the basic building block of an RM array. In this work, we give a detailed analysis and evaluation on main parameters of MU to discover the relationship between basic structure and RM system. Meanwhile, we evaluate a series of RM with different MUs using SPEC2006 benchmark at the system level. In addition, we propose a hybrid MU structure which uses both read-only and read write ports to further optimize certain applications. Based on the analysis of results, suggestions are given regarding RM design in different area such as big-data and IoT which need cost-effective and energy-efficient memory, respectively.

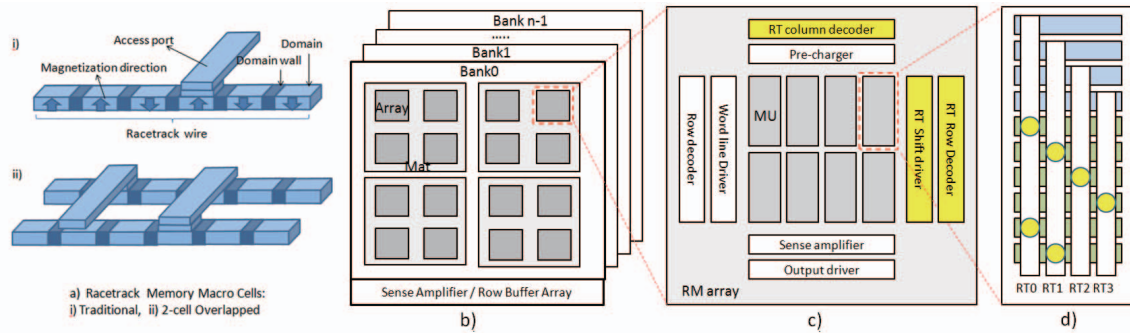


Fig. 1. The Racetrack Memory system. a) A RM cell; b) Overview of bank; c) Detail view of array; d) A macro unit;

The main contributions of this work are as follows:

1. We expand the NVsim simulator to evaluate racetrack memory. We analyze the design parameters of MU at structure level, and characterize the relationship between these arguments. Suggestions of RM based main memory design for typical applications such as big-data and IoT are given.

2. We expand the NVmain memory simulator to evaluate RM based main memory together with Gem5 system simulator. We simulate a set of RM-based main memory with typical MUs and evaluate them with various types of applications to exploit the relationship between performance and MU structure.

3. We propose a hybrid MU structure which use both read-only and read write ports to further optimize read intensive applications. The evaluation results show that hybrid MU is effective to improve the performance and decrease the energy cost for read intensive applications. Hybrid MU make “All stack optimization” possible in memory level.

The remainder of this paper is organized as follows. The background and motivation are introduced in Section 2. In Section 3, the design and simulation methodology are presented. Subsequently, the evaluation and analysis of macro unit design on structure level are shown in Section 4. In Section 5, the simulation and analysis of racetrack memory design on system level are introduced. In Section 6, the hybrid MU structure is proposed and the evaluation and analysis are given. Finally, the conclusions are presented in Section 7.

## II. BACKGROUND AND MOTIVATION

### A. Racetrack memory basics

A racetrack memory cell consists of a tape like stripe and several transistors, as illustrated in Figure 1a(i). The stripe is constructed from magnetic material, and is divided into several domains separated by domain walls. Each domain acts as an STT-RAM cell, and the magnetization direction of a domain is programmed to store either 0 or 1. Several transistors are connected to the stripe to perform read, write, and shift operations. These transistors are called access ports or shift ports. Each racetrack contains one or more access ports, and the data aligned with each port can be read/written by these accessing ports. In order to access other bits that are not aligned with a port, a shift operation must be performed to move these bits to specific access port.

The time required to access a data bit depends on its distance to the access port. If there is only one access port in each racetrack, then the average access latency may present a challenge for the design and integration of main memory. Thus, in general, architects design multiple access ports in each racetrack, evenly distributed along the track. The average shift latency can be reduced by adding access ports. However, the number of access ports is significantly limited by the sharply increasing energy consumption. As illustrated in Figure 1a(ii), in order to increase the density and area efficiency, racetracks are often overlapped together.

### B. Motivation

Racetrack memory has more design parameters than other NVMs, which makes it more complicated to compose the basic structure or data array because each parameter influences the features of the memory system. On the other hand, it also give more flexibility to compose the memory system with different specific structures to meet different requirements.

The MU has three important parameters, number of tracks, number of access ports, and domains one track has. The different combination of these parameters has different impact on the features of memory system such as read, write and shift latency and energy. Former work [7] has found the appropriate structure which aims to optimize one of the features. However, they did not give thorough investigation into the relationship between the design parameters and the features of memory system, and no further optimization is given.

At present, designing and implementing specific devices or software to optimize specific applications in certain area is the trend. Generally, different applications need certain specified feature of memory system. For example, memory-intensive applications in big data area need fast read velocity and huge capacity, while the embedded applications in IoT area need low energy cost and small area. This is the opportunity for RM to meet their requirements with customized structure design.

We design and simulate a series of basic units of racetrack memory with different structures and test them in different types of applications, aiming to find the relationship between design and feature. In addition, we propose a hybrid MU structure using both read-only and read write ports to further optimize read intensive applications. Based on this, customized

RM design will make “All stack optimization” possible in memory level.

### III. DESIGN AND SIMULATION METHODOLOGY

We perform our simulation and evaluation at both structure level and system level.

At structure level, We design a 32MB racetrack memory array under 45nm process in our extension of NVsim [8]. NVsim is a circuit-level model which facilitates the NVM array level exploration before real chip fabrication. It supports many types of NVMs, including PCM, RRAM, STT-RAM and Flash. Selected NVM types change the input cell parameters and keep the modeling of periphery circuitry unchanged. We reuse most simulation framework for peripheral circuitry in NVsim and add extra components described in pervious work [7], as the yellow part of Figure1(c) illustrate. The value and physical equations of device-level parameters are collected from previous works [25], [?], [27], [26]. The circuit level data of DRAM are collected from Micron’s website [10]. We evaluate several parameters, including area, latency and energy for each basic operation.

At system level, we simulate the RM memory by modifying the main memory simulator NVmain [9], which is a cycle-level main memory simulator designed to simulate emerging non-volatile memories at the architectural level. We simulate the RM memory by extending the STT-RAM object in NVM object structure tree of NVmain and adding the shift operation and logic into it. Before a RM array with given MU is to be evaluated, the parameters about timing and energy of basic operations are set up in its configuration file. The general memory system parameter as the number of channels, ranks, banks and columns are set up at the same time. We prepare config file for each MU structure individually, using the corresponding data obtained from NVsim.

We evaluate the performance and energy cost of RM using the cycle-accurate full system simulator gem5 [11] together with NVmain. Many new styles of memory can be simulated using gem5. We use its Syscall Emulation (SE) mode to simulate user space programs while system services are provided by the simulator. We compile gem5 and NVmain into gem5.opt to get a good balance between the speed of the simulation and the insight into the circumstances. We simulate 20 million instructions for each bench under ALPHA architecture.

We use basic array which has 32M bytes as one bank which is composed by three levels: mat, array and basic MUs, as illustrated in Figure1(b). We set up 1GB memory with one memory channel which has 2 ranks and each rank has 16 banks. The memory frequency is 800M Hz and the IO circuits work under DDR3 standard.

### IV. STRUCTURE LEVEL EVALUATION

We simulate 32MB racetrack memory array with 27 different MUs. The observed parameters including area, latency and energy of read, write and shift operations and leakage power. The aims of this evaluation is to find the relationship between design parameters of MU and features of memory array.

#### A. Basic macro unit candidates

The layout of an MU is determined by three parameters: the number of bits in each racetrack (Nb), the number of access ports in the MU (Np), and the number of racetracks in the MU (Nr). Figure1a(ii) illustrates the example of an MU with Nb=6, Np=2, and Nr= 2. In the rest of the paper, each MU design is labeled with MU-Nb-Np-Nr. For example, the MU in Figure1a(ii) is MU060202. These device level design parameters influence the area, performance, and energy of an RM array. For detailed discussion of MUs, we refer the reader to [7]. In this section, we mainly discuss the relationship between structure parameters and features of memory array.

According to the data given by designer, one 128F-long racetrack with 2F domain length generally has at most 64 domains. So we range Nb from 16 to 64. Secondly, since the size of the access port transistor is about 4 times the width of one racetrack, the area efficiency will be the best when four RM cells are overlapped [7], so we range Nr from 1 to 4. Thirdly, more access ports can reduce the shift distance but increase the energy cost, so we range Np from 1 to 32 in different MUs. What should be pointed out is the parameter of share degree (SD), which is defined as the number of domains that share one access port in one racetrack. It is an index of access port density and different SD indicates the difference of access ability. In order to depict the density of access ports, we range the SD from 4 to 32 in different MUs. We assemble these parameters and design 27 MU structure, and the detailed structure setup is displayed in Table 1.

TABLE I  
THE TYPICAL MACRO UNITS OF RACETRACK MEMORY.

No.	MacroUnit	SD	No.	MacroUnit	SD	No.	MacroUnit	SD
1	MU160101	16	10	MU323204	4	19	MU641604	16
2	MU160201	8	11	MU320401	8	20	MU640201	32
3	MU160402	8	12	MU320402	16	21	MU643202	4
4	MU160802	4	13	MU320404	32	22	MU643204	8
5	MU160804	8	14	MU320801	4	23	MU640401	16
6	MU160101	16	15	MU320802	8	24	MU640402	32
7	MU320101	32	16	MU320804	16	25	MU640801	8
8	MU321602	4	17	MU641601	4	26	MU640802	16
9	MU321604	8	18	MU641602	8	27	MU640804	32

#### B. Evaluation of area and latency

We evaluated the area and operation latency of RM composed by basic MU, and the result is displayed in Figure 2. The MUs in horizontal axis are ordered by Nb, Nr and Np successively in ascending order. It can be seen that there is clear regularity in the trend of total area. First, with the same Nb, the increasing of Nr helps to decrease the total area. For example, the total area of RM arrays with one track MU are all bigger than  $10mm^2$ , while the area of RM arrays with 2-track MU are all smaller than  $10mm^2$  and bigger than the area of RM arrays with 4-track MU. That is because overlapping structure improves the area efficiency. Second, with the same Nb and Nr, increasing the Np helps to decrease the total area as well. This is because, with the increasing of Np, the number of domains in charged by each port is decreasing, so the overhead

length at both ends of each track to accommodate overflowed domains is decreasing.

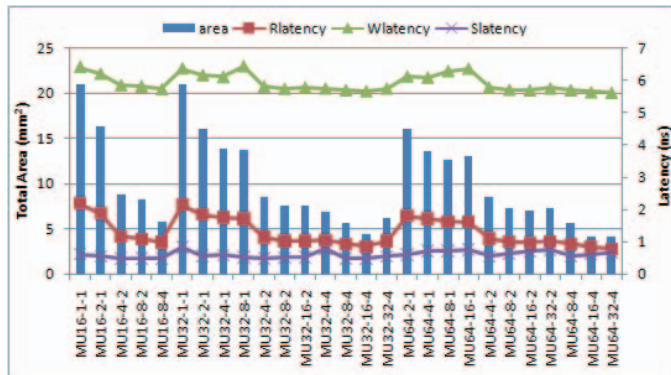


Fig. 2. Evaluation of area and latency;

At the same time, the read latency has approximately the same trend and regularity as the total area, though a smaller range instead. The tendency of write latency is not obvious and less similar with the trend of read latency. It can be observed that the MUs with lower overlap degree tend to have higher access latency ( $N_r=1$ ), while the ones with higher overlap degree have lower access latency ( $N_r=2$  or 4). This is because less peripheral circuit is needed to compose data array with MUs with higher overlap degree. The access latency is similar between MUs with  $N_r=2$  and MUs with  $N_r=4$ . On the other hand, the shift latency is relatively independent with design parameters of basic MU and has no significant change. The shift operation changes the magnetisation direction of domains by the current added at the end of the track. The current through the track will shift all domains opposite the current direction. Note that it will not cause mechanical movement of the material, but the change of domains magnetisation direction. Whatever the track length is and how many ports it has, the time spent in shifting one domain is relatively invariant.

### C. Evaluation of area and energy

We also evaluated the area and operation energy of RM composed by basic MU, and the result is displayed in Figure 3. The MUs in horizontal axis are in the same sequence as Figure 2. It is obvious that the read and write energy trend has the same trend as the total area, while the shift energy has no significant change. The reason is that less peripheral circuit is needed to compose data array with MUs with higher overlap degree, so it needs less energy to sense the data out or program data into the track. The shift energy of 64-bit long racetrack is slightly higher than that of 32-bit or 16-bit long racetrack. That is because the shift operation should change all the domains in one racetrack in order to shift one step forward, then the longer the track is, the more the energy needed.

### D. Evaluation of leakage power and share degree

The leakage power is displayed in Figure 4. Different MUs have different leakage power and they have no similar

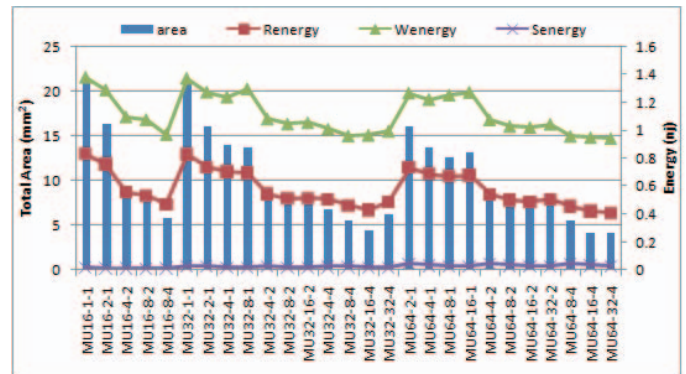


Fig. 3. Evaluation of area and energy;

regulation with latency or energy, but they have dependence with the share degree. Several MUs with higher leakage power such as MU320801, MU323204 and MU641601, have the same feature that their share degree is 4, while the MUs such as MU320101, MU640201 and MU640402 have a share degree of 32 and hence lower leakage power. It is not hard to understand that smaller share degree means more access ports and more transistors and then more leakage power.

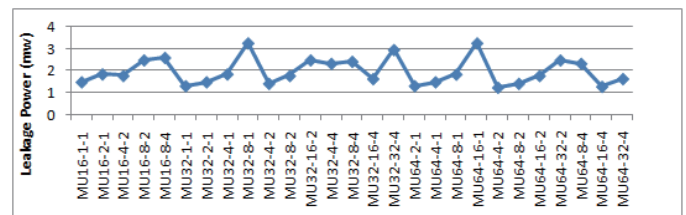


Fig. 4. Evaluation of leakage power;

In addition, we group the racetrack MUs by their share degree in 32, 16, 8, 4 respectively and compare their latency and energy. The results of the four series data have the similar trend and regularity as described before. Given the same share degree, the few tracks the MU has, the larger the array access latency is. The latency and energy has no big difference among MUs with different SDs. That means SD is the indicate of port density, it has no obvious effect on the latency and energy of data array.

In summary, smaller SD may decrease the shift operations to get required domain but increase the energy cost caused by more access ports. Larger SD may decrease the energy cost due to less access ports but spend more shift operations to get a required domain instead.

### E. Suggestions on MU selection

According to the data analysis and comparison above, we can conclude that RM has more design flexibility for different application area. For the applications such as big data which need fast access velocity and huge capacity, the RM memory designer can choose MUs with higher density and lower access latency, such as MU321604, MU641604 and MU643204. In order to further improve the performance, methods to decrease the shift operations are needed. For the

TABLE II  
EVALUATION SYSTEM SETUP.

Unit	Configurations
CPU	4 single Alpha cores, 3GHz, 1-way issue
L1	split I/D, 32KB/32KB, 4-way, 64B,LRU, private, R/W: 2/2-cycle, 0.074/0.074nJ, 23.4mW
L2	4MB shared by 2 cores, 8-way, 64B, LRU, R/W lat.: 10/10-cycle, R/W E: 0.407/0.386-nJ, 681.5mW
Memory.	Dual Channel DDR3, 1600MHz, 1GB, 100-cycle.
DRAM.	Memory latency and Energy Parameters from[10]
RM.	RM Latency and Energy Parameters from NVsim

applications which need low power cost and smaller area like IoT and embedded systems, MUs with lower access energy such as MU160804, MU320804, MU321604, MU641604 and MU643204 are suitable. Since MUs with fewer bits and fewer access ports per racetrack have less manufacture difficulty and better stability [21], [22], then MU160804, MU320804, MU321604 are more suitable for IoT application.

## V. SYSTEM LEVEL EVALUATION

### A. Experimental setup

In this section, we describe the system evaluation framework and the experimental setup. The detailed configurations are described in Table 2. We use full system simulator Gem5 to manipulate the server configuration, where the latency and energy data of DRAM comes from [10] and the configuration of RM bank is simulated by expanded NVmain described in section 3. The basic parameters of NVmain are got from NVsim with different MU setup discussed in previous section.

Regarding the workload, we select 21 workloads from the SPEC2006 benchmarks [28], including typical application areas, such as artificial intelligence, fluid simulation, video compression, programming language and speech recognition. The selected benchmarks represent the main aspects of computer applications at present and the application type includes both memory intensive and memory non-intensive ones.

We classify a benchmark as memory-intensive if its L2 Cache Misses per 1K Instructions (MPKI) is greater than 5 and otherwise we refer to it as memory non-intensive.

### B. The overall improvement of performance

We evaluate the performance using Instructions per Cycle (IPC). In order to obtain the overall performance improvement of RM over DRAM, we evaluated IPC of 27 set of RM with different MUs and get their geometric mean. We also evaluated the IPC of 2-dimension DRAM memory. The results is depicted in Figure 5.

The benchmark has been ordered by MPKI in ascending order. The MPKI is as low as 0.08 in the leftmost bench and as high as 96 in the rightmost bench. Figure5 shows that the benches with lower MPKI have higher IPC and the ones with

TABLE III  
THE BENCHMARKS FROM SPEC2006.

No.	Bench	type	IPC	MPKI	Application Area
1	bwaves	FP06	0.078	37.02	Fluid Dynamics
2	bzip2	INT06	1.387	1.03	Compression.
3	calculus	FP06	2.042	0.23	Structural Mechanics
4	gamess	FP06	0.227	16.16	Quantum Chemistry.
5	gcc	INT06	1.179	1.26	C Compiler
6	GemsFDTD	FP06	0.078	48.64	Computational Electromagnetics
7	gromacs	FP06	1.368	0.50	Molecular Dynamics
8	h264	INT06	0.615	3.72	Video Compression
9	hmmr	INT06	1.581	0.35	Search Gene Sequence
10	lbm	FP06	0.086	40.97	Fluid Dynamics.
11	leslie3d	FP06	0.198	20.22	Fluid Dynamics.
12	libquantum	INT06	1.965	0.19	Quantum Computing
13	milc	FP06	0.298	5.44	Quantum Chromodynamics
14	namd	FP06	1.745	0.26	Molecular Dynamics
15	omnetpp	INT06	0.775	2.25	Discrete Event Simulation
16	perlbench	INT06	0.254	11.47	Programming Language
17	povray	FP06	0.928	1.65	Image Ray-tracing
18	sjeng	INT06	0.065	53.14	Artificial Intelligence: chess
19	soplex	FP06	1.170	1.47	Linear Programming
20	sphinx3	FP06	0.033	96.17	Speech recognition
21	wrf	FP06	0.372	8.68	Weather

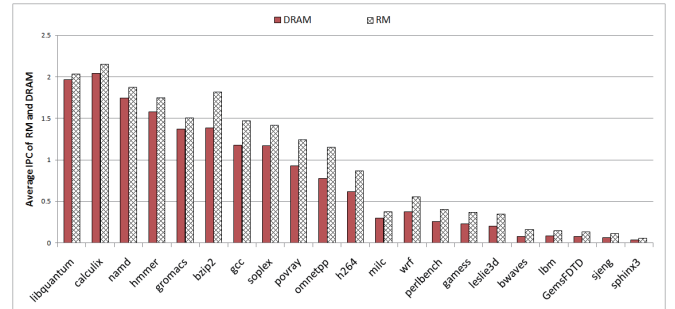


Fig. 5. The IPC of DRAM and RM in average;

higher MPKI have lower IPC, which means memory intensive bench has lower performance because of more memory accesses. It is obvious that each bench using RM has higher IPC than using DRAM and their range of improvement is different from each other. The results is depicted in Figure 6.

Figure 6 implies that for those benchmarks with higher MPKI, the improvement of IPC is more evident, which implies that the RM has more performance improvement for those memory intensive application. The range of improvement is 41% on average. The bench *bwaves* has the highest IPC

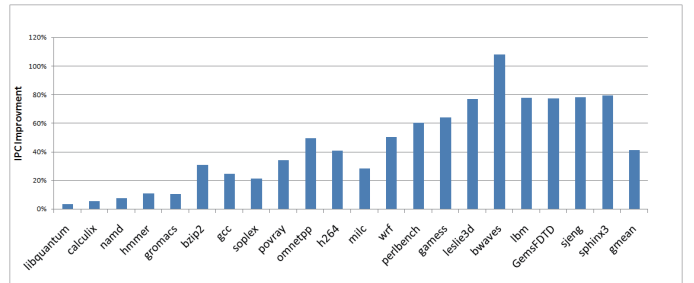


Fig. 6. The IPC improvement using RM against DRAM;

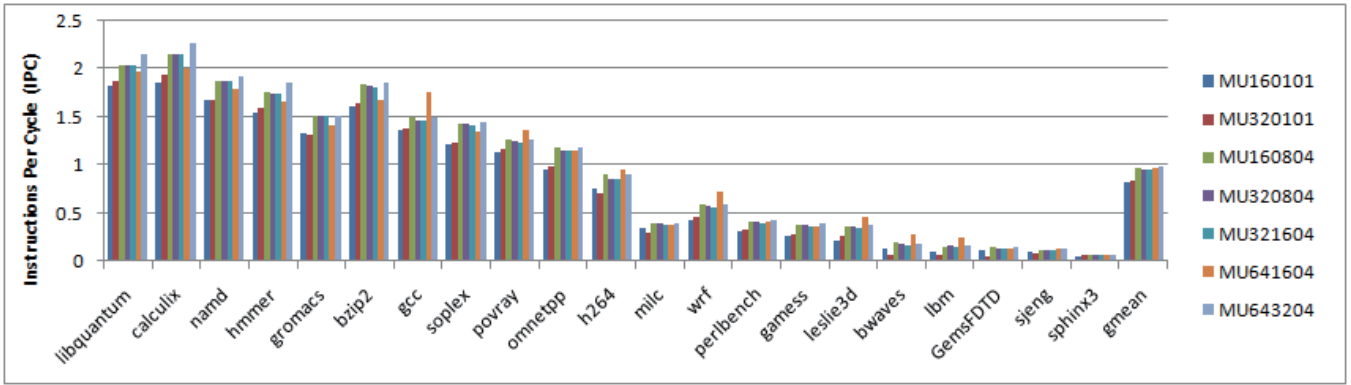


Fig. 7. The IPC of RM composed by different MUs;

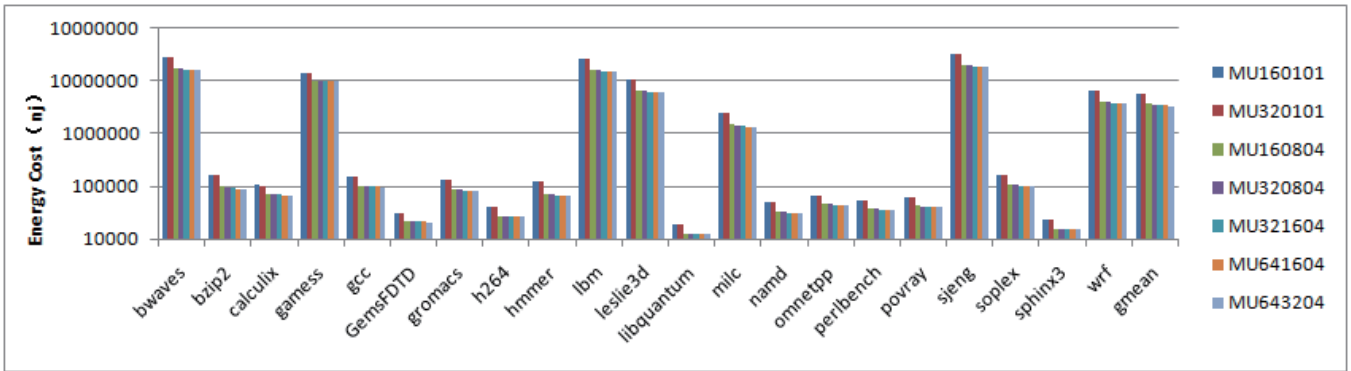


Fig. 8. The Energy Cost of RM composed by different MUs;

improvement more than 100%. This is because it has lower row buffer locality than others, which allows more memory access to get data from memory array instead of row buffer.

### C. The performance evaluation of typical MUs

We simulate and evaluate seven RM based memory with 21 benches ordered by MPKI in ascending order respectively. Five of RMs are composed by MUs described in last section which has lower latency and energy: MU160804, MU320804, MU321604, MU641604 and MU643204. In order to make comparison, we evaluate another two RMs composed by MU160101 and MU320101 which are not fabricated in overlapped structure. The evaluation results in Figure7 illustrate that the performance of the five MUs are better than the two obviously. Interestingly, MU641604 has better performance in benches which has higher MPKI, and MU643204 has better performance for benches with lower MPKI.

### D. The energy evaluation of typical MUs

In order to evaluate the energy in system level, we record the number of read, write and shift operation of each bench from the output of Gem5 and NVmain and calculate the total energy cost for each bench. The result in Figure8 shows that the five RMs using overlapped structure have smaller energy cost than the other two. The five overlapped MUs have no obvious difference in energy cost in each bench.

Based on the evaluation results of performance and energy on system level, together with the area analysis in former

section, we can conclude that MU641604 and MU643204 has better adaptability than other MU structures. They have better performance, lower energy cost and small area than others. At the same time, MU641604 is more suitable for memory intensive application and MU643204 is more suitable for non memory intensive applications. In order to get further optimization of area and energy for low energy cost application, we try to optimize the MU structure with hybrid ports.

## VI. HYBRID PORTS OPTIMIZATION

In order to make further optimization, we propose a hybrid MU structure. Based on the observation of system level evaluation, we found that about half of the benches have far more read operation than write operation. As shown in table 4, benches in left size of the table have 71% read operation in average. We propose a hybrid RM structure which has both read-only ports and read-write ports to make optimization aiming at this kind of benches or applications. First, given a specified MU, adding additional read-only ports and display them evenly in racetracks will shorten the average shift distance of read operation and thus improve the performance of RM. Second, since read-only port is much smaller than read-write port for its fewer transistor, the hybrid MU will have less energy cost than read write ports.

In order to verify this method, we design 5 MUs to evaluate their area, performance and energy. We group them into two groups to make compare: traditional MU and hybrid MU. The

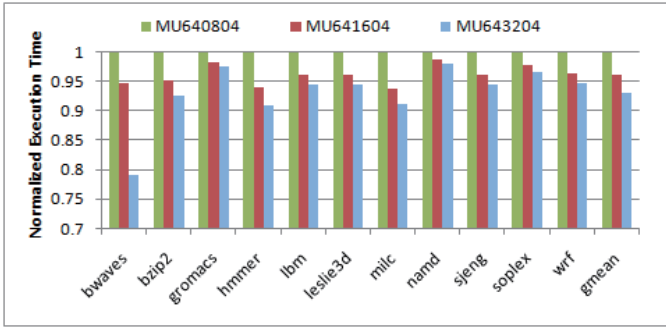


Fig. 9. The relative overall execution time of traditional MU;

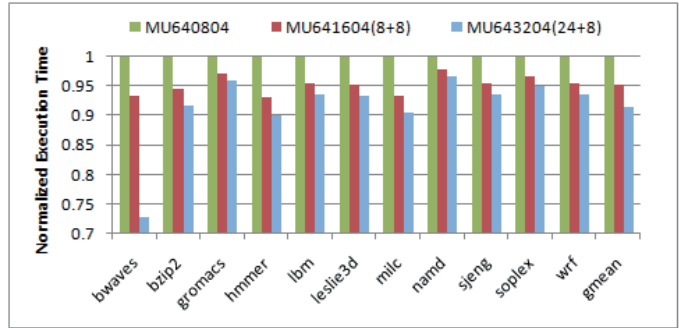


Fig. 11. The relative overall execution time of hybrid MU;

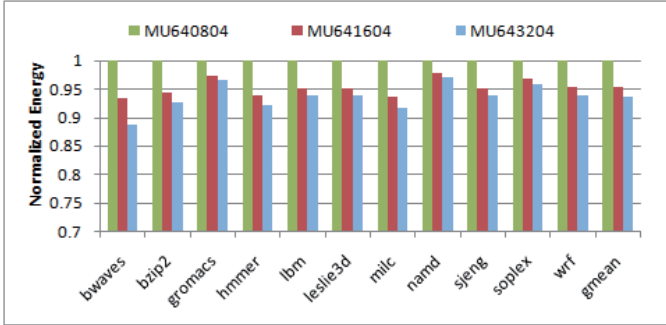


Fig. 10. The relative overall energy consumption of traditional MU;

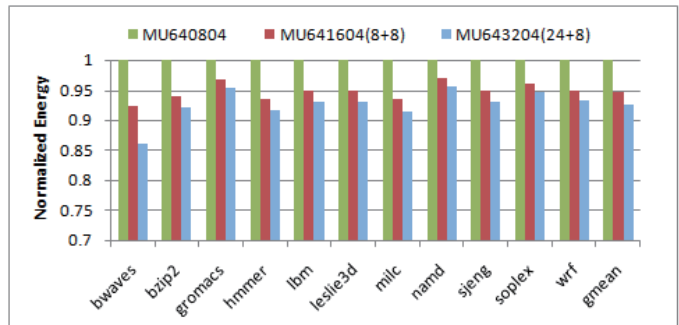


Fig. 12. The relative overall energy consumption of hybrid MU;

first group includes MU640804, MU641604 and MU643204, which use read write ports respectively. The second group includes MU640804, MU641604(8 read-only ports, 8 read-write ports) and MU643204(24 read-only ports, 8 read-write ports), in which the first one use read write ports and the last two use both read write ports and read-only ports. All the ports are distribute evenly in macro unit like Figure1d shows.

The first group represents the action to make optimization through adding traditional access ports and the second group represents another way through adding read-only ports. The aim of arrangement of 8 and 24 read only ports MU is to find out the effect and trend of different proportion of read-only ports will make.

TABLE IV  
THE READ AND WRITE OPERATION PROPORTION OF SPEC2006.

Bench	Read	write	Bench	Read	Write
bwaves	75%	25%	calculix	30%	70%
bzip2	83%	17%	games	1%	99%
gromacs	49%	51%	gcc	26%	74%
hmmer	85%	15%	GemsFDTD	13%	87%
lbm	75%	25%	h264	13%	87%
leslie3d	75%	25%	libquantum	26%	74%
milc	88%	12%	omnetpp	29%	71%
namd	44%	56%	perlbench	23%	77%
sjeng	75%	25%	povray	35%	65%
soplex	57%	43%	sphinx3	9%	91%
wrf	74%	26%			

We first simulate the MU641604(8+8) and MU643204(24+8) in NVsim and get their latency and energy of three basic operation. Then we evaluate execution time and energy of the RMs composed by five candidate

MUs respectively, using Gem5 and NVmain with the 11 read intensive benches of SPEC2006 within 200 million instructions. The methodology and process is the same as described in section 3. The results are depicted in Figure9 to Figure12.

For the traditional MU group, given MU640804 as base, MU641604 has 8 more read write ports which bring area decrease by 11%, performance improvement by 4% and energy decrease by 4.8% in average, and MU643204 has 24 more read write ports which bring area decrease by 15%, optimization by 7% and 6.4% in average. For the hybrid MU group, given MU640804 as base, MU641604 has 8 more read-only ports which bring area decrease by 14%, performance improvement by 4.8% and energy decrease by 5.3% in average, and MU643204 has 24 more read-only ports which bring area decrease by 17%, optimization by 8.5% and 7.4% in average. The relationship of these results is depicted in Figur13.

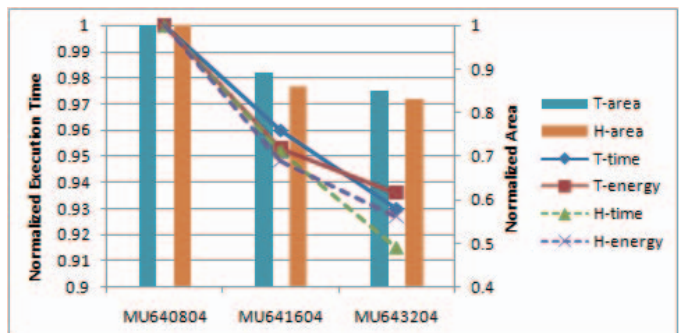


Fig. 13. The normalized optimization of hybrid MU;

The results imply that, for the read intensive benches or application, hybrid MU can bring more optimization than traditional MU in area, performance and energy respectively. At the same time, with the rising of proportion of read-only ports, the optimization extent is decreasing.

The optimization effect of benches which has larger read operation proportion is larger, and vice versa. So, with the scalability of RM structure and hybrid MU methodology, the RM based memory can be designed toward different applications in order to get optimal features. Customizable racetrack memory offer the opportunity to customize their memory design in different application for architecture designer. That make the trend "All stack optimization: from transistor to application" become into reality at memory level in many area, such as IoT and artificial intelligence etc.

## VII. CONCLUSION

Racetrack memory has potential to replace DRAM as main memory in modern computer system. The variety of RM design parameters introduces complexity to the designment of memory system, but brings flexibility at the same time. For memory intensive applications in big-data area, it can be designed to have higher density and higher access velocity with highly overlapped structure. For those applications in IoT area, structures designed with small area and lower energy are better. Using customized hybrid MU structure, RM have further optimization space for better results for read intensive applications. With the flexibility of RM structure, different design targets can be achieved through proper arrangement of design parameters. That make the trend "All stack optimization" possible in memory structure level.

## VIII. ACKNOWLEDGEMENTS

We thank anonymous reviewers for their feedbacks and suggestions. This work is supported by the National Natural Science Foundation of China (Grant No. 61232003), Samsung Electronics Co., Ltd., and Huawei Technologies Co., Ltd.. The paper corresponding author is Jiwu Shu.

## REFERENCES

- [1] Burr G W, Kurdi B N, Scott J C, et al. Overview of candidate device technologies for storage- class memory[J]. IBM Journal of Research and Development, 2008, 52(4): 449-464
- [2] B.C.Lee and others. Phase change memory architecture and the quest for scalability. Communications of the ACM. 2010.
- [3] M.K.Qureshi and others. Scalable high performance main memory system using phase-change memory technology. ACM SIGARCH. 2009.
- [4] B.C.Lee and others. Phase-change technology and the future of main memory. IEEE MICRO. 2010
- [5] R.A.Bheda and others. Energy efficient Phase Change Memory based main memory for future high performance systems. Proc of IEEE on Int Green Computing Conf and Workshops. 2011.
- [6] Qingda Hu, Guangyu Sun, Jiwu Shu, and Chao Zhang, "Exploring Main Memory Design Based on Racetrack Memory Technology," in Proceedings of the 26th ACM Great Lakes Symposium on VLSI (GLSVLSI 2016), May 18-20, 2016, Boston, MA, USA, pp. 397-402.
- [7] Chao Zhang, Guangyu Sun, Weiqi Zhang, Fan Mi, Hai Li, and Weisheng Zhao, "Quantitative Modeling of Racetrack Memory, A Tradeoff among Area, Performance, and Power," in Proceedings of the 20th Asia and South Pacific Design Automation Conference (ASP-DAC 2015), January 19-22, 2015, Chiba, Japan, pp. 100-105.
- [8] X. Dong et al. Nvsim: A circuit-level performance, energy, and area model for emerging nonvolatile memory. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 31(7), 2012.
- [9] Poremba M, Xie Y. NVMain: An Architectural-Level Main Memory Simulator for Emerging Non-volatile Memories[J]. IEEE Computer Society Annual Symposium on Vlsi, 2012, 59(5):392-397.
- [10] Micron. 8Gb: x4, x8, x16 DDR3L SDRAM Description. www.micron.com, 2015.
- [11] Binkert, N and Beckmann, B and Black, G and others. The gem5 simulator. SIGARCH Comput. Archit. 2011:1-7.
- [12] Yue Zhang, Chao Zhang, Jiang Nan, Zhizhong Zhang, Xueying Zhang, Jacques-Olivier Klein, Dafine Ravelosona, Guangyu Sun, and Weisheng Zhao, "Perspectives of Racetrack Memory for Large-Capacity On-Chip Memory: From Device to System." IEEE Transactions on Circuits and Systems, Vol. 63, No. 5, pp. 629-638, May 2016.
- [13] Guangyu Sun, Chao Zhang, Hehe Li, Yue Zhang, Weiqi Zhang, Yizi Gu, Yinan Sun, Jacques-Olivier Klein, Dafie Ravelosona, Yongpan Liu, Weisheng Zhao and Huazhong Yang, "From Device To System: Cross-Layer Design Exploration of Racetrack Memory," in Proceedings of the 18th Design, Automation and Test in Europe (DATE 2015), March 9-13, 2015, Grenoble, France, pp. 1018-1023.
- [14] Sun Zhenyu, Wu Wenqing, Li Hai, "Cross-layer racetrack memory design for ultra high density and low power consumption." in Proceedings of the Design Automation Conference IEEE, 2013:1-6.
- [15] Hoda Aghaei Khouzani, Pouya Fotouhi, Chengmo Yang, Guang R Gao. "Leveraging access port positions to accelerate page table walk in DWM-based main memory." Conference on Design, Automation and Test in Europe European Design and Automation Association, 2017:1454-1459.
- [16] Hoda Aghaei Khouzani, Chengmo Yang. "A DWM-Based Stack Architecture Implementation for Energy Harvesting Systems." Acm Transactions on Embedded Computing Systems 16.5s(2017):1-18.
- [17] Venkatesan, Rangharajan, et al. "DWM-TAPESTRI - An energy efficient all-spin cache using domain wall shift based writes." Conference on Design, Automation and Test in Europe EDA Consortium, 2013:1825-1830.
- [18] Wang, Yuhao, and H. Yu. "An ultralow-power memory-based big-data computing platform by nonvolatile domain-wall nanowire devices." in Proceedings of the 2013 International Symposium on Low Power Electronics and Design (ISLPED 2013):329-334.
- [19] Haiyu Mao, Chao Zhang, Guangyu Sun, and Jiwu Shu, "Exploring Data Placement in Racetrack Memory based Scratchpad Memory," in Proceedings of the 4th IEEE Non-Volatile Memory System and Applications Symposium (NVMSA 2015), August 19-21, 2015, Hong Kong, China, pp. 1-5.
- [20] Chen X, Sha H M, Zhuge Q, et al. Optimizing data placement for reducing shift operations on Domain Wall Memories[C]. Design Automation Conference. ACM, 2015:1-6.
- [21] Parkin S S, Hayashi M, Thomas L. Magnetic domain-wall racetrack memory[J]. Science, 2008, 320(5873):190-4.
- [22] Thomas L, et al. "Racetrack Memory: A high-performance, low-cost, non-volatile memory based on magnetic domain walls." Electron Devices Meeting IEEE, 2011:24.2.1-24.2.4.
- [23] Venkatesan R, Kozhikkottu V, Augustine C, et al. TapeCache: a high density, energy efficient cache based on domain wall memory[C]// Proceedings of the 2012 ACM/IEEE international symposium on Low power electronics and design. ACM, 2012:185-190.
- [24] Wang Y, Yu H. An ultralow-power memory-based big-data computing platform by nonvolatile domain-wall nanowire devices[C]// ACM/IEEE International Symposium on Low Power Electronics and Design. 2013:329-334.
- [25] Fukami, S., et al. "Current-induced domain wall motion in perpendicularly magnetized CoFeB nanowire." Applied Physics Letters 98.8(2011):1954.
- [26] Sun, Zhenyu, et al. "Multi retention level STT-RAM cache designs with a dynamic refresh scheme." Ieee/acm International Symposium on Microarchitecture IEEE, 2011:329-338.
- [27] Zhang, Y., et al. "Perpendicular-magnetic-anisotropy CoFeB racetrack memory." Journal of Applied Physics 111.9(2012):813-L7.
- [28] Henning, John L. "SPEC CPU2006 benchmark descriptions." Acm Sigarch Computer Architecture News 34.4(2006):1-17.