A Power and Temperature Aware DRAM Architecture

Song Liu, Seda Ogrenci Memik, Yu Zhang, and Gokhan Memik

Department of Electrical Engineering and Computer Science

Northwestern University, Evanston, IL 60208, USA

{sli646, seda, yzh702, memik}@eecs.northwestern.edu

ABSTRACT

Technological advances enable modern processors to utilize increasingly larger DRAMs with rising access frequencies. This is leading to high power consumption and operating temperature in DRAM chips. As a result, temperature management has become a real and pressing issue in high performance DRAM systems. Traditional low power techniques are not suitable for high performance DRAM systems with high bandwidth. In this paper, we propose and evaluate a customized DRAM low power technique based on Page Hit Aware Write Buffer (PHA-WB). Our proposed approach reduces DRAM system power consumption and temperature without any performance penalty. Our experiments show that a system with a 64-entry PHA-WB could reduce the total DRAM power consumption by up to 22.0% (9.6% on average). The peak and average temperature reductions are 6.1°C and 2.1°C, respectively.

Categories and Subject Descriptors

B.3.2 [**Memory Structure**]: Design Styles – Primary Memory

General Terms

Management, Design, Performance

Keywords

DRAM, Power, Temperature, Page Hit Aware Write Buffer

1. INTRODUCTION

Emergence of Chip Multi Processors (CMPs) pushes the computation capacity of computing systems to unprecedented levels. This capacity can only be maintained with an equally aggressive memory subsystem performance. Fully Buffered Dual Inline Memory Module (FB-DIMM) [5] has been proposed to feed the starving processors with larger main memory capacitance with increasing bandwidths. However, DRAM power consumption takes an increasingly larger share of the system power budget. Memory power issue is becoming further critical in battery-powered systems such as laptops.

Existing DRAM power management approaches focus on fully utilizing low power modes of the DRAM. However, DRAMs in high performance systems have fewer opportunities to use low power modes. Simply switching between different modes may incur significant performance penalty on the DRAM bandwidth. A technique that relates to DRAM's inherent nature is needed. Our goal is to develop a technique to explore DRAM power savings in the active mode. Specifically, our proposed technique reduces DRAM power consumption by improving DRAM page hit rate. Our technique does not impose any limitation on switching to low power mode for further power reduction.

In modern DRAM systems, I/O modules cannot access the DRAM array directly. The target row (or page) must be loaded to the I/O buffer before read or written, namely activated. Page hit rate is defined as the percentage of read/write operations that target an already activated page. When there is a page miss, the system has to spend time and power to activate the new target row. Therefore, page hit rate is a strong indication of the power efficiency of the DRAM.

The DRAM page hit rate could be improved by reordering DRAM accesses. However, reordering sometimes may hurt the system performance. From an architectural point of view, DRAM read operations are the bottleneck of system performance, while write operations are not the primary concern. Therefore an efficient reordering should not delay any read operation. We exploit this fact through an enhancement to the DRAM architecture. We introduce a buffering mechanism to hold write operations that may cause a page miss. These buffered writes are placed into the DRAM later when their target row is activated by read operations.

Since our technique reduces the power consumption of DRAM systems, the DRAM temperature is also impacted positively. Our experiments show that our proposed technique could reduce the total DRAM system power consumption by as much as 22.0% (9.6% on average) and

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2008, June 8–13, 2008, Anaheim, California, USA

Copyright 2008 ACM 978-1-60558-115-6/08/0006…$5.00

DRAM temperature by as much as 6.1 °C (2.1 °C on average) for over 20 SPEC CPU 2000 benchmarks[1].

The remainder of the paper is organized as follows. Section 2 discusses related work. Section 3 provides an overview of modern DRAM systems. Our proposed technique is described in Section 4. Section 5 presents experimental methodology and results. Finally, we conclude the paper in Section 6.

2. RELATED WORK

DRAM power consumption can comprise a large portion of the total system power. Modern DRAMs provide various low power modes to same DRAM power consumption. One example is Rambus DRAM (RDRAM), which has 4 power modes: active, standby, nap, and power down [13]. Traditional low power DRAM techniques focus on utilizing these low power modes efficiently to achieve best energy delay product [3, 4, 7]. We refer to this kind of approaches as power mode based low power techniques. These approaches are designed for applications with fewer DRAM accesses. Our goal however, is to improve the DRAM power efficiency in active mode by avoiding unnecessary page misses. This is achieved by accessing the DRAM in an intelligent way. Since our technique does not use low power mode or throttle the DRAM bandwidth, it does not incur performance penalties (in fact increasing DRAM page hit rate may improve the system performance). Moreover, our technique could easily cooperate with power mode based techniques to achieve more power saving for different kinds of applications

The DRAM-level prefetching technique by Lin et al. [9] could be considered as the closest related work. They introduce a small prefetching cache in the DRAM to improve system performance and reduce power consumption. However, their technique is designed for close page mode, where the page hit rate is forced to be zero. However, in a high performance DRAM system, open page mode is a more likely choice, which draws on the principle of data locality. In open page mode the benefit of prefetching on power reduction would be minimal if not non-existent. Our technique on the other hand, is devised upon and evaluated with a baseline system operating in the high performance open page mode.

3. BACKGROUND

In modern computers, synchronous dynamic random access memory (SDRAM) has replaced other types of DRAM due to its greater speed. Fully Buffered Dual In-line Memory Module (FB-DIMM) and Double Data Rate two (DDR2) SDRAM increase the DRAM bandwidth to keep pace with the increasing memory bandwidth demands by CMPs. In order to explore potential power and temperature optimizations in DRAM system, it is necessary to understand the DRAM operation mechanism. In this section, we briefly review modern DRAM technology.

3.1 DDR2 SDRAM

In modern DRAM systems, the memory array is hierarchically organized as banks, rows, and columns. Column is the minimum access unit of the SDRAM.

The DDR2 SDRAM, by Micron Technology Inc. [11], a representative architecture, has five kinds of commands: READ, WRITE, ACTIVATE, PRECHARGE, and REFRESH. The memory array in the SDRAM cannot be accessed directly by the I/O modules. Instead, before any READ or WRITE can be issued, a row in that bank must be loaded to the I/O buffer, namely opened or activated, by an ACTIVATE command. Only one row of a bank can be opened at the same time, therefore an opened row must be written back to the memory array (closed) by a PRECHARGE command. DRAM rows have to be refreshed periodically by REFRESH command and when other parts of the system are power down, the DRAM could maintain data by an automatic SELF REFRESH command.

The I/O buffer can be managed in two modes: the open page mode and the close page mode. When the SDRAM is in open page mode, an opened (activated) row is kept open for as long as possible, until this row is written back to the memory array by a REFRESH command or another row in the same bank is accessed. On the other hand, when the SDRAM is working in close page mode, any opened row is precharged immediately after an access. Due to the locality in memory accesses open page mode could provide better performance for most applications.

3.2 FB-DIMM (Fully Buffered DIMM)

FB-DIMM introduces an advanced memory buffer (AMB) between the memory controller and DDR2 SDRAM chips. In FB-DIMM system, the memory controller reads or writes through a serial interface to the AMB instead of the parallel-organized SDRAM chips. FB-DIMM provides necessary DRAM system parameters in serial presence detect (SPD), while keeping the organization of multiple DRAM chips transparent to the memory controller. Therefore AMB is the key point of increasing performance, portability, and reliability of FB-DIMM. AMB also offers error detection and correction, without posing overhead on the processor or the memory controller. However, the AMB unit has high power consumption, which makes the FB-DIMM more susceptible to thermal emergencies.

4. POWER AND TEMPERATURE AWARE DRAM ARCHITECTURE

This section describes our proposed technique for DRAM power reduction. We first analyze the power component of the DRAM to explore potential power savings. Then, we present our technique and analyze the overhead.

4.1 DRAM Power Modeling

DDR2 DRAM power consumption has three main components: background power, activate power, and read/write power [10]. Background power is power consumed to maintain DRAM accessibility in a certain mode. Different power modes have different background power consumptions. Power mode based techniques focus on reducing average background power consumption by using low power mode more often. Activate power is the power consumption of loading one row to the I/O buffer. Read/write power is the power consumption of reading or writing data stored in the I/O buffer. Read/write power is proportional to DRAM I/O bandwidth. Reduction in read/write power occurs at the expense of performance. Therefore, in order to reduce power consumption with no performance penalty, activate power should be optimized first.

In close page mode, activate power is proportional to DRAM access frequency. In open page mode, activate power is also a function of page hit rate. In other words, higher page hit rate means same number of read and write operations could be completed with fewer activate operations. Therefore, improving page hit rate could achieve power benefit without hurting performance. Therefore, we turn our attention towards opportunities to improve the page hit rate.

4.2 PHA-WB: Page Hit Aware Write Buffer

DRAM read operations pose the major bottleneck of system performance. While reading, the processor has to wait for the data before executing other instructions. Although instruction parallelism techniques can be used to hide a portion of this latency, for many application domains, the read latency is still one of the most important determinants of performance. On the other hand, DRAM write operations could possibly be delayed without hurting performance. Therefore, modern processors often write to the main memory through a write buffer. Similarly, we introduce a Page Hit Aware Write Buffer (PHA-WB) for DRAM page hit rate optimization.

PHA-WB is a buffer between the memory controller and DRAM systems. PHA-WB is transparent to the processor and memory controller. That is the memory controller could access the DRAM as if there is no PHA-WB. None of the read operations are delayed by PHA-WB, actually some read operations are accelerated since the page hit rate is higher.

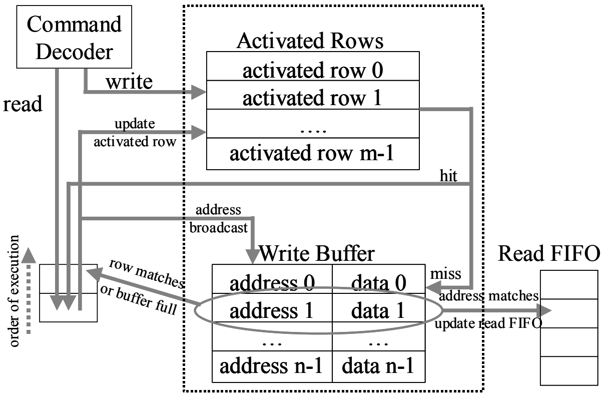

Figure 1. Block diagram of the PHA-WB scheme

Table 1. Power Consumption of PHA-WB with Different Size under 90 nm Technology

PHA-WB Size	PHA-WB Power Consumption (W)
8-entry	0.093969
16-entry	0.098326
32-entry	0.107038
64-entry	0.127537
128-entry	0.222912

Read operations pass through the buffer without delay. PHA-WB has to impact the read operations when the data read by the operation is residing in the buffer. In this scenario, the read operation will get the wrong data from the DRAM. However, while reading from the DRAM, PHA-WB compares the target address with the buffered data. If the target data is residing in PHA-WB, the buffered data replaces data from DRAM in the read FIFO. Since the delay of reading data from DRAM is much longer than searching the buffer (which is similar to reading data from a fully-associated cache), the correct data could be read without extra delay.

On the other hand, write operations have to go through PHA-WB before accessing the DRAM. PHA-WB checks whether the target row of the write operation is activated. If the target row is activated, the data is written to the DRAM; otherwise, the data is buffered in the PHA-WB. Data residing in PHA-WB could access the DRAM when the targeting row is activated by another read operation and thus the number of page misses is reduced. Another case of sending buffered data to the DRAM is when the buffer is full and another write operation needs buffering. In this scenario, the oldest data in PHA-WB will be written to the DRAM. However, when the PHA-WB has a large number of entries, choosing the oldest data may be expensive. An alternative solution is choosing a random entry to be vacated. Our experiments revealed that this will not degrade the optimization quality significantly. The page hit rate varies only by 1.37% on average between choosing the oldest versus choosing a random entry from a PHA-WB with 64 entries.

Another important aspect of the PHA-WB is that it can work with DMA engines. Specifically, a write operation to the memory occurs when a) the processor writes a value or b) the Direct Memory Access (DMA) engine writes data read from an input/output device. In either case, the value

Table 2. Experimental configuration of processor, memory system, and PHA-WB.

Parameters	Values
Processor	4-core, 2.5GHz
Instruction Cache (per core)	64kB, 2-way, 64B line, 1 cycle hit latency
Data Cache (per core)	64kB, 2-way, 64B line, 2 cycle hit latency, write-back
L2 Cache (shared)	16MB, 8-way, 128B line, 13 cycle hit latency, write-through
FB-DIMM	1GB, 512Mb per chip, 8 DQs per chip, 667MHz
DDR2 DRAM Chip	4 banks per chip, 16384 rows per bank, 256 columns per row, 4 Bytes per column
Burst Mode	Burst length of 8
Major DRAM Timing Parameter	Active to Read tRCD=15ns, Read to Read Data tCL=15ns, Precharge to Active tRP=15ns
No. of 64 Byte Entries in PHA-WB	16/32/64

that is written is not needed until a read operation to it occurs. Our PHA-WB will capture such cases and guarantee correct operation without any performance penalty.

4.3 Implementation of PHA-WB

For a FB-DIMM, the best choice is to implement the PHA-WB is in the AMB [6]. DRAM commands are decoded in the AMB, so that the target address could be analyzed in the AMB. There is a deep write buffer in the AMB between the data from the memory controller and the DDR I/O port. Therefore, we could replace this buffer with the PHA-WB, while maintaining the structure of the AMB. Another benefit of this design choice is that the PHA-WB is fully associated with all DRAM banks. Therefore the buffer entries could be used efficiently.

The number of entries in the PHA-WB is a tradeoff between system complexity and power/temperature savings. More entries will yield more DRAM power saving and temperature reduction at the expense of more complex circuitry for buffering the data and comparing target addresses. Considering each write operation will write 64 bytes to the DRAM, we compare different configurations of PHA-WB with 16, 32, and 64 entries, which correspond to 1kB, 2kB, and 4kB memory added to the AMB. Detailed results on the comparison of these configurations are presented in Section 5.

Figure 1 illustrates the structure of PHA-WB implemented in the AMB for a DRAM system with *m* independent DRAM banks, and *n* PHA-WB entries. The PHA-WB is enclosed in the dashed rectangle. PHA-WB has two main parts: the Activated Rows Table and the Write Buffer. Activated Rows Table is a table with *m* entries. Each entry traces the activated row of an independent bank. The Write Buffer is composed of a content addressable memory (CAM), which keeps write addresses, and a buffer array, which keeps write data. The PHA-WB works as follows. DRAM operations are decoded by the command decoder. Read operations are not subject to buffering, therefore, read operations are sent to the operation queue directly. On the other hand, for all write operations the Activated Rows Table must be checked for a possible match between their target row and the row that is already activated. If the target row is currently activated, the write operation is sent to the operation queue, if not, the operation is placed in the Write Buffer. For each operation that has just entered the operation queue, its target address is sent to the Activated Rows Table for an update. In the meanwhile, the address is broadcast in the Write Buffer. Each entry compares the broadcast address with its own target address. If these two addresses are identical, we refer to this case as an *address match*. If the two addresses are targeting the same row, we call it a *row match*. Address match is a special case of row match. Write operations with row matches are sent to the DRAM after the current operation, because this operation will hit an activated row. Write operations with address matches are also sent to the Read FIFO and replace the data read by the current read operation, because the data read from the DRAM is outdated. There is a third case when a buffered write operation would be issued. When the Write Buffer is full and a new write operation does not hit an activated row, a randomly selected operation in the PHA-WB is sent to the operation queue and the new write operation enters the PHA-WB.

We have estimated the power consumption of PHA-WB on a 1GB FB-DIMM system by modeling the broadcast structure and the buffer array using CACTI 3.2 [14]. The power overhead with different sizes is given in Table 1. The power consumption of PHA-WB is small compared to the DRAM power consumption, which is over 3W on average among our experimental workloads. We observe from the table that, the power overhead does not increase fast from 8-entry design to 64-entry design. However, 128-entry PHA-WB is much more power consuming than smaller designs. Therefore, 64-entry PHA-WB is a good trade off between DRAM power savings and buffer power overhead.

5. EXPERIMENTAL RESULTS

In this section, we first introduce the power and thermal models used in our evaluation. Then, we outline our experimental methodology and describe our benchmarks. Finally, we present the results demonstrating the effectiveness of our proposed technique.

5.1 Power and Thermal Modeling

We use a power model based on the data published by Micron Technology Inc. for DDR, DDR2, and DDR3 memory chips [12]. We utilize a RC-analogy based thermal model widely used in literature for the DRAM system, which is also similar to the model presented by Lin et al. [8]. We present static temperature values for each workload. We also adopted the AMB power model used by Lin et al. [8]. We have estimated the power consumption of PHA-WB by modeling the broadcast structure and the buffer array using CACTI 3.2 [14]. This power overhead of the PHA-WB has been considered in our evaluation.

5.2 Experimental System Configuration

We used M5 [2] as our architectural simulator to extract memory read and write traces for SPEC CPU 2000 applications. We assume a CMP with 4 processors and 1 GB DDR2 FB-DIMM. The major system parameters are listed in Table 2. Then, we analyze the DRAM page hit rate under different configurations with and without PHA-WB. System Power Calculator [12] is used to calculate DRAM power consumption. Finally, we calculate AMB power and static DRAM system temperature for each trace.

In order to evaluate the page hit rate, we define the memory mapping as follows. Read and write accesses are burst-oriented, in DDR2 SDRAM, with the burst length being programmable to either 4 or 8. The burst length determines the maximum number of column locations that can be accessed for a single read or write command. We chose burst length of 8 in our simulations. For 1GB DRAM module, 30 bits (bit 0 to 29) are used to describe the memory space. Bit 1 and 0 are mapped to each byte within a column. Bits 4 through 2 are mapped to adjacent columns that could be burst accessed from the DRAM. Therefore, burst read and write could access 32 bytes from one bank. Note that, the size of the L2 cache line is 128B, so each read operation reads 128 bytes from the DRAM. The write mechanism of L2 cache is write- through, so each write operation writes 64 bytes to the DRAM. Therefore, each time at least 64 bytes are accessed at the same time. We map bit 5 to different banks, so that 64 bytes could be accessed with one burst operation. Bit 10 through 6 are mapped to different sections of a row. There are 64 banks in the DRAM module and we have grouped them into groups of two, so there are 32 groups. Bits 15 through 11 are mapped to these groups. Finally, bits 29 through 12 are mapped to different rows within a bank.

5.3 Workloads

We used 20 applications from SPEC CPU 2000 benchmarks. In order to represent different DRAM usage, we organized 8 workloads with these applications. Each of these workloads has 4 applications. W1 has four instants of swim, which represents the application with highest DRAM accesses. W2 through W8 are various combinations of SPEC CPU 2000 benchmarks. W2, W3, and W7 represent applications with a high number of memory accesses. W5 and W6 represent applications with infrequent memory accesses. W4 and W8 represent cases between these two extremes. The workload contents are described in Table 3.

Table 3. Workload Mixes

Workload	Benchmarks
W1	swim, swim, swim, swim
W2	swim, lucas, applu, mgrid
W3	wupwise, apsi, fma3d, facerec
W4	equake, gap, gcc, wupwise
W5	ammp, apsi, vpr, parser
W6	mcf, vortex, mesa, gzip
W7	swim, lucas, wupwise, apsi
W8	swim, lucas, sixtrack, galgel

5.4 Results

We evaluate the impact of the PHA-WB on page hit rate, power consumption, and the static temperature of the DRAM. We performed a comparison of PHA-WB structures with different number of entries. Figure 2 shows the page hit rate of different workloads on systems without PHA-WB, 16-entry PHA-WB, 32-entry PHA-WB, and 64-entry PHA-WB. We observe that the page hit rate increases by up to 18.7% (9.7% on average) comparing an architecture without PHA-WB and one with a 16-entry PHA-WB. The maximum and average page hit rate improvement for the 64-entry PHA-WB is 25.4% and 14.4%. The introduction of even a small sized PHA-WB first makes a significant impact compared to the base case. As we increase the PHA-WB size further, we observe a gain; however, the additional benefits are lower because of the additional power consumption of the PHA-WB.

Figure 3 illustrates the power savings under different PHA-WB sizes, relative to a system without PHA-WB and shown in percentage. The power saving is a function of page hit rate and memory throughput. Some applications with high improvements in page hit rate fail to yield significant power savings because the DRAM throughput is very low. The power savings of the 16-entry PHA-WB design could be as high as 16.2% (6.9% on average). The average and peak page power savings for the 64-entry PHA-WB are 22.0% and 9.6%, respectively.

Figure 4 shows the reduction in DRAM temperature using different PHA-WB sizes compared to the base case without the PHA-WB. The temperature is reduced by up to 6.1 °C (2.1 °C on average) for the 64-entry PHA-WB and up to 4.5 °C (1.5 °C on average) for the 16-entry PHA-WB.

We observe that applications with high rate of DRAM accesses are likely to benefit from PHA-WB. On the other hand, applications with low rate of DRAM accesses are not likely to benefit from PHA-WB, because when there are few DRAM accesses, the background power dominates the total power consumption. In the most general case, a combination of a power mode based low power technique and our approach will succeed in reducing power consumption in both cases.

6. CONCLUSIONS

We have proposed and evaluated a Page Hit Aware Write Buffer (PHA-WB) in DRAM systems. Our proposed technique achieves power and temperature reduction by improving DRAM page hit rate with a write buffer. Since we do not delay read operations, our approach does not hurt the system performance. In fact, higher page hit rate improves the system performance. For a PHA-WB of 64 entries our proposed technique could achieve power savings by as much as 22.0% (9.6% on average) and reduce temperature by up to 6.4 °C (2.1 °C on average).

Our proposed technique does not pose any limitations on the use of any other types of idle mode power optimization schemes or thermal-emergency intervention methods. Such techniques can co-exist along with our proposed DRAM architecture. Our architecture would provide the first level of optimization at no performance cost and further aggressive optimizations or interventions could be applied by trading-off performance.

7. ACKNOWLEDGEMENTS

This work has been supported by NSF Grant # CNS-0546305.

8. REFERENCES

1. Standard Performance Evaluation Corporation. SPEC CPU2000. http://www.spec.org.
2. Binkert, N.L., R.G. Dreslinski, L.R. Hsu, K.T. Lim, A.G. Saidi, and S.K. Reinhardt, The M5 simulator: modeling networked systems. IEEE Micro, 2006. 26(4): p. 52-60.
3. Delaluz, V., A. Sivasubramaniam, M. Kandemir, N. Vijaykrishnan, and M.J. Irwin. Scheduler-Based DRAM Energy Management. in DAC'02. 2002.
4. Fan, X., C.S. Ellis, and A.R. Lebeck, Memory Controller Policies for DRAM Power Management, in ISLPED'01. 2001.
5. JEDEC, FBDIMM Specification: DDR2 SDRAM Fully Buffered DIMM (FBDIMM) Design Specification http://www.jedec.org/download/search/JESD2051.pdf.
6. JEDEC, FBDIMM: Advanced Memory Buffer (AMB) http://www.jedec.org/download/search/JESD82-20.pdf.
7. Lebeck, A.R., X. Fan, H. Zeng, and C. Ellis, Power Aware Page Allocation, in ASPLOS-IX. 2000.
8. Lin, J., H. Zheng, Z. Zhu, H. David, and Z. Zhang, Thermal Modeling and Management of DRAM Memory Systems, in ISCA'07. 2007.
9. Lin, J., H. Zheng, Z. Zhu, Z. Zhang, and D. H., DRAM-level prefetching for fully-buffered DIMM: design, performance and power saving, in ISPASS'07. 2007.

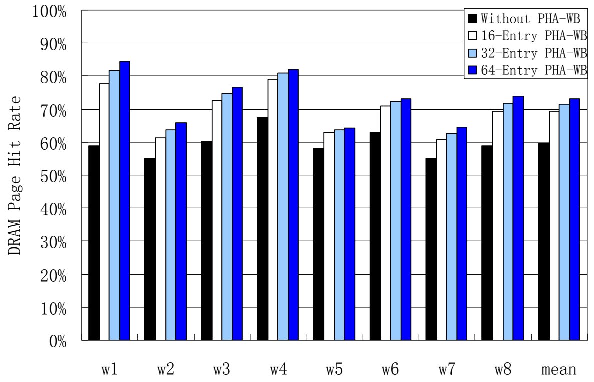

Figure 2. DRAM page hit rate for different workloads and different PHA-WB sizes

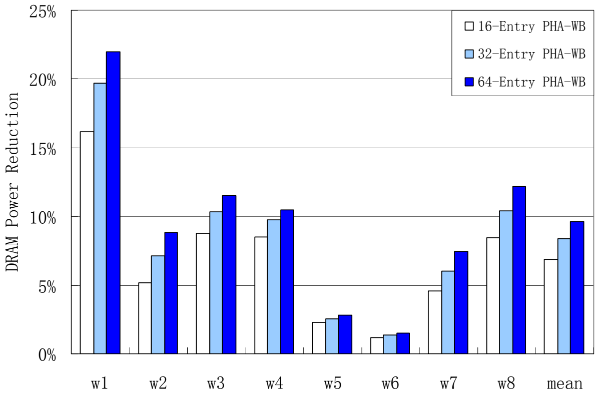

Figure 3. DRAM power reduction for different workloads with different PHA-WB sizes

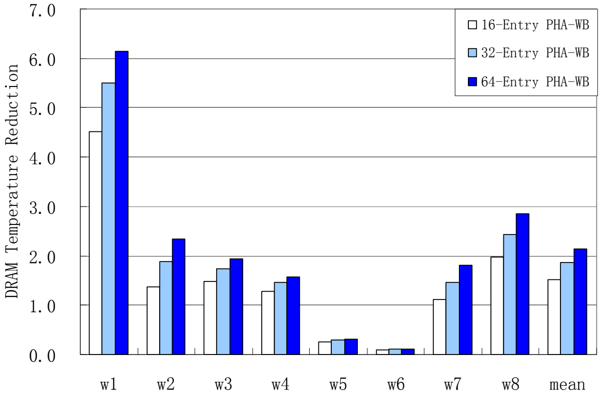

Figure 4. DRAM temperature reduction for different workloads with different PHA-WB sizes

10. Micron, Calculating Memory System Power for DDR2.
11. Micron, DDR2 SDRAM http://download.micron.com/pdf/datasheets/dram/ddr2/512MbDDR2.pdf.
12. Micron, System Power Calculator, http://www.micron.com/support/designsupport/tools/powercalc/powercalc.aspx.
13. Rambus, RDRAM, in www.rambus.com.
14. Shivakumar, P. and N.P. Jouppi, CACTI 3.0: An Integrated Cache Timing, Power, and Area Model WRL Research Report.