Modeling and Design Exploration of FBDRAM as On-chip Memory

Guangyu Sun
Center for Energy-efficient Computing and Applications
School of EECS, Peking University
Email: gsun@pku.edu.cn

Cong Xu and Yuan Xie
Computer Science and Engineering
Pennsylvania State University
Email: {czx102, yuanxie}@cse.psu.edu

***Abstract*—Compared to the traditional DRAM technology, floating body DRAM (FBDRAM) has many advantages, such as high density, fast access speed, long retention time, etc. More important, FBDRAM is compatible with the traditional CMOS technology. It makes FBDRAM more competitive than other emerging memory technologies to be employed as on-chip memory. The characteristic variance of memory cells caused by process variations, however, has become an obstacle to adopt FBDRAM. In this work, we build a circuit level model of FBDRAM caches with the consideration of process variations. In order to mitigate the impact of process variations, we apply different error correction mechanisms and corresponding architecture-level modifications to FBDRAM caches and study the trade-off among reliability, power consumption, and performance. With this model, we explore the L2 cache design using FBDRAM and compare it with traditional SRAM/eDRAM caches in both circuit and architectural levels[1].**

I. Introduction

The increasing number of processor cores integrated on a single chip and the growing bandwidth gap between on-chip memory and off-chip I/O argue for more and more on-chip memory in future memory systems. The traditional SRAM technology cannot satisfy this requirement due to its low density and scalability issues. Recently, many emerging memory technologies, such as floating body DRAM (FBDRAM), spin-torque-transfer random access memory (STTRAM), and phase-change random access memory (PRAM), are extensively researched to attack the so called *memory wall*. Compared to the traditional SRAM technology, these emerging memories have common advantages, such as higher density, low standby power, better scalability.

Among these emerging memory technologies, the FBDRAM is more competitive than others to be used as on-chip memory because of several reasons [1], [2], [3], [4], [5], [6]. First, the access latency of FBDRAM is comparable to that of SRAM. The detailed modeling will be introduced in Section III. Second, the FBDRAM has almost the highest density among these memory technologies. As shown in Table I, the cell sizes of different memory technologies are compared. We can find that the FBDRAM has the smallest cell size among these technologies. Thus, with the same area, much more memory can be integrated on-chip, when we replace SRAM with FBDRAM. Third, the write cycle of FBDRAM is in the level of 10^{15} [1], which is large enough to avoid endurance problems. Fourth, the process technology of FBDRAM is compatible with the CMOS technology. This is the unique advantage of FBDRAM over other emerging memory technologies such as STTRAM and PRAM, which need extra process steps in fabrication.

TABLE I
Comparison of cell sizes and write cycles.

Tech.	SRAM	STTRAM	PRAM	eDRAM	FBDRAM
Size (F^2)	80-140	8-20	4-20	8-20	4-8
Write Cycle	10^{15}	10^{15}	10^{8}	10^{15}	10^{15}

[1]This work is supported in part by SRC grants and NSF 0643902, 0702617, 0903432, and 1017277.
978-3-9810801-8-6/DATE12/©2012 EDAA

Although FBDRAM has attracted more and more attention, prior research on FBDRAM mainly focuses on fabrication, device modeling, and circuit design. There lacks a model to efficiently evaluate the properties of FBDRAMs with different configurations, especially when the impact of process variations are considered. Due to process variations, the characteristic of FBDRAM cells on a single chip are not identical but follow a statistical distribution [1]. To the best of our knowledge, however, there is no research on exploration of trade-off among performance, power consumption, and reliability for FBDRAM designs in the circuit and architecture levels. More important, the system level benefits, which are not studied in prior work, need to be shown before replacing traditional on-chip SRAM/eDRAM with FBDRAM.

These issues are addressed in this work. Our contributions are as follows: (1) We present a circuit level modeling of FBDRAM with consideration of process variations. (2) Based on the model, a tool is built by extending the CACTI to evaluate performance, power consumption, and reliability of FBDRAM designs. (3) By adjusting the strength of error correction mechanisms, the trade-off among these properties is explored to mitigate the impact of process variations. (4) The architectural evaluation is performed and compared to the counterparts using SRAM/eDRAM. The benefits of using FBDRAMs as on-chip memory are studied and the optimized designs are discussed.

The rest of paper is organized as follows. Section II introduces the background of the FBDRAM technology. The modeling of performance and power consumption for FBDRAM is presented in Section III. The impact of process variations is also studied at the same time. In order to mitigate the impact of process variations, different error correction strategies and the corresponding architectural level modification are introduced in Section IV to explore the trade-off among performance, power consumption, and reliability. The circuit and architectural level evaluations are presented in Section V.

II. Background

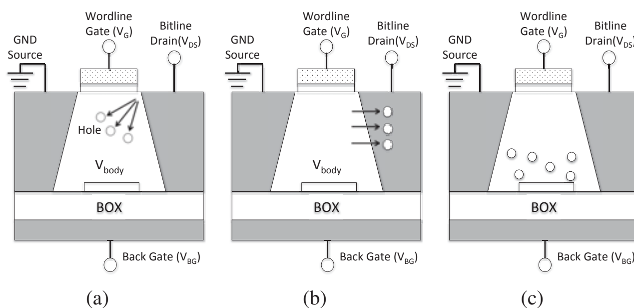

Fig. 1. Illustration of read/write operations to an FBC: (a) SET operation, (b) RESET operation, (c) read operation [7].

The floating body effect is the effect of the body potential in a transistor, which is first realized by the silicon on insulator (SOI) technology. The transistor's body forms a capacitor against the

insulated substrate. The charge accumulates on this capacitor and may cause adverse effects. At first, the effect is supposed to have a negative impact on the SOI devices. For example. it causes off state leakages, resulting in higher current consumption. For the traditional 1T1C DRAM implemented with SOI, the effect aggravates the loss of information from the memory cells.

While floating body effect presents a problem in SOI DRAM chips, it is exploited later as the underlying principle of FBDRAM. The capacitor appears between the bottom of the isolated tub and the underlying substrate allows a DRAM-like cell to be built without adding a separate capacitor. The floating body effect takes the place of the conventional capacitor. Consequently, the FBDRAM cell (FBC) is also called as 1T DRAM cell.

As shown in Figure 1, the FBC consists of a MOSFET whose body is electrically floating and is used as a storage node of electric charge. An nMOSFET formed on PD-SOI or FB-SOI can be chosen as a candidate to realize this concept. The source of the FBC is biased to $0V$ (GND) in both read and write operations. The drain is connected to a bit line (BL), and the gate is connected to a word line (WL). To write data '1' (also known as SET operation), the voltage applied on WL (V_G) and BL (V_D) are well controlled so that the nMOSFET is operated in saturation status. It leads to impact ionization that injects holes into the body. A time constraint is required to promise the successful SET operation. The drain current (I_{DS} are decided by V_G and V_D. To write data '0' (also known as RESET operation) the voltage applied on body (V_B)) and V_G are controlled to make the *pn* junction between the body and the drain forward-biased, ejecting the stored holes from the body. Similarly, a write period is also required to achieve a current state in RESET operation. To read the data, the voltages are applied to WL and BL and the nMOSFET is operated in a linear ohmic region. With a proper V_G, the I_{DS} caused by the body effect depends on the number of holes stored in the body. Thus, the difference between state '0' and '1' can be sensed. In order to differentiate the two states, the V_G needs to be controlled in a proper region. This is called the "program window" [7]. More details will be introduced in the next section.

Data stored in an FBC is volatile and needs to be refreshed, because holes are generated in the body of the data state '0' through the PN junction reverse-bias leakage current between the body and the source/drain and they need to be bailed out in order to maintain the difference in the number of holes between the data '1' and the data '0'. Fortunately, the data retention time of an FBC is much longer than that of an eDRAM cell [1].

III. FBDRAM Modeling

In this section, we analyze the impact of process variations on characteristic of FBDRAM cells and provide the circuit level modeling. In addition, different error correction strength are introduced to explore the design trade-off.

A. Timing and Power Estimation

The generic timing and power modeling is derived from the previous versions of CACTI [8] by several enhancements. For example, we first estimated the turn-on resistance, gate/drain capacitance and other parasitic resistance/capacitance from the device characteristic of FBC. As one of its most attractive features, FBC has very small capacitance of the source/drain (S/D) region because SOI device eliminates the souce/drain-to-body diffusion capacitance. Furthermore, threshold-voltage-feedback loop is used to write "1" in FBC to reduce the operation voltage. To enhance the efficiency of the threshold-voltage-feedback loop, relative thick gate oxide is used. As a result, the gate capacitance of FBC is even smaller than the conventional SOI MOSFET. Thus, the RC delay related to wordline/bitline charge/precharge of FBDRAM is significantly faster than SRAM given the same array size. We also calculate the interconnect wire resistance and wire capacitance with taking thermal impact into considerations by latest ITRS report [9]. Consistent with CACTI, as a system-level model, we keep the modeling of the device at a reasonable granularity. Then a simplified version of Horowitz's timing model [10] was involved in calculating the delay of each logic component as follows,

$$\text{Delay} = \tau\sqrt{\left(ln\frac{1}{2}\right)^2 + \alpha\beta} \tag{1}$$

where α is the slope of the input, $\beta = g_m R$ is the normalized input transconductance by the output resistance, and $\tau = RC$ is the RC time constant. The overall memory access latency is estimated by combining all the timing values of circuit components together.

Similar to the delay estimation, our energy model followed the CACTI tool to first compute the capacitances for each unit. The dynamic energy consumption can be modeled as

$$E_{dynamic} = C \times V_{DD}^2 \times P \tag{2}$$

where C is the equivalent capacitance of a node and "P" is the switching probability of that node. Total dynamic energy of the FBDRAM array consists of the dynamic energy consumed in the memory cells, the wordline circuitry, the bit line circuitry including column multiplexer, the drivers, and the precharge circuitry.

B. Data Sensing Model

Like other emerging memory technologies such as STT-RAM, PCRAM and ReRAM, the bit information of FBDRAM can be treated as the equivalent resistance state of the memory cell. To read the state of the FBC, a specific voltage is applied to the FBC to produce a current. A current-voltage converter is introduced in such current-mode sensing scheme. The converter behaves as the first-level sense amplifier, and the CACTI-modeled voltage sense amplifier is still kept as the final stage of the sensing scheme. The current-voltage converter senses the current difference ΔI and then it is converted into a voltage difference ΔV. As to the implementation, we refer to a previous current-voltage converter design [11]. This sensing scheme is similar to the hybrid-I/O approach [12], which can achieve high-speed, robust sensing, and low power operation. To avoid unnecessary calculation, the current-voltage converter is modeled by directly using the HSPICE-simulated values and building a look-up table of delay, dynamic energy, and leakage power.

C. Modeling of Process Variations

Due to process variations (PV), the characteristic of FBCs on a single chip may vary a lot from each other. For a FBDRAM design without consideration of PV effects, the errors in read/write operations cannot be detected and recovered by traditional error correction techniques. In this subsection, we apply process variations to our model and study the PV effect on FBDRAM.

Note that process variations have effects on both FBCs and the peripheral circuitry of FBDRAM. The experimental results show that effect on FBCs dominates. Thus, we mainly focus on the modeling of PV in FBCs. Several important properties of FBCs, which are related to access error rates, are discussed.

The characteristic of an FBC is mainly decided by several key parameters in fabrication, which include effective channel length, threshold voltage, gate thickness, etc. In order to model the spatial

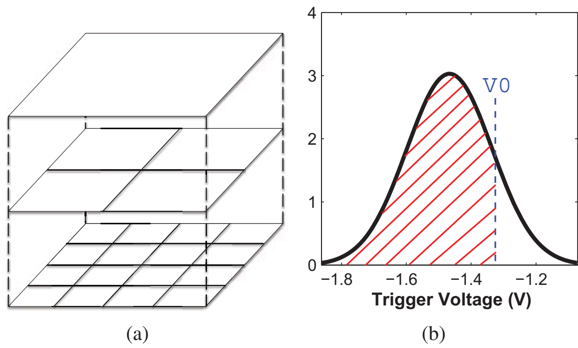

Fig. 2. (a) Modeling of process variations; (b) Distribution of trigger voltages for SET operations (45nm).

correlations of process variations among FBCs, we employ an extensively used modeling method [13]. We first divide the area of the die into regions using a multi-level quad-tree partitioning, as shown in Figure 2 (a). For each level i, the die area is partitioned into 4^i squares. In the example shown in Figure 2 (a), the first or top level 0 has a single region for the entire die and the last or bottom level has 16 regions. We then apply an independent random variable with each region to represent a component of the total intra-die parameter variation. The total variation for a parameter of a device is then composed as the sum of intra-die components from all levels, where level ranges from 0 to 3 in the example of Figure 2 (a).

With this method, we can calculate the parameters of all FBCs using the Monte Carlo simulations. Similar to prior research [13], the total variation follows the normal distributions. In order to achieve an accurate estimation, the variance (σ) of each parameter is calibrated so that the simulated results match the distribution measured from the real fabricated FBCs. After calculating the characteristic of FBCs, we can model the properties of a whole FBDRAM (as in previous two subsections) and analyze the error rates of read/operations with different design configurations.

The error rate of programming (SET/RESET) a FBDRAM is decided by two factors: (1) programming timing and (2) programming current. In other words, the access can only succeed when the timing is long enough and the current is high enough. In current design of FBDRAMs, the programming timing is well designed to make sure it is long enough to performance SET/RESET operations. Consequently, the error rate of programming a FBDRAM is decided by the trigger voltages in SET/RESET operations. The simulation results show that the characteristic of FBCs also follow the normal distributions.

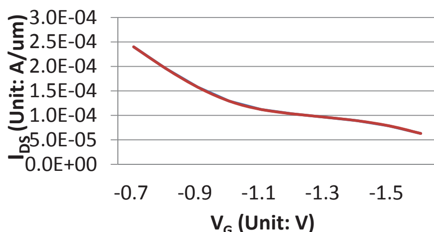

Fig. 3. Relationship between drain current and the gate voltage for SET.

Figure 2 (b) shows an example distribution of trigger voltage for SET operations. The trigger voltage is defined as the minimum program voltage (V_G) required to change the storage state of an FBC. As shown in Figure 2 (b), if the programming voltage (V_G) of FBDRAM is set as V_0, the shadow area(cumulative distribution) represents the probability that an FBC can be successfully programmed to bit '1' in the SET operation. Apparently, the error probability in SET operations can be reduced by increasing V_G. However, drain current (V_{DS}) in write operation is also increased with V_G at the same time, as shown in Figure 3. Thus, increasing V_G will induce more energy consumption. In order to achieve a feasible error rate (e.g. 10^{-9}), V_G has to be set much higher than the mean value of its distribution. It means that many FBCs are over-programmed and consume more energy consumption. Note that the distributions of trigger voltages are similar for RESET and SET operations but with different variances due to different program mechanisms. Due to the page limitation, we only show the distribution for SET operations.

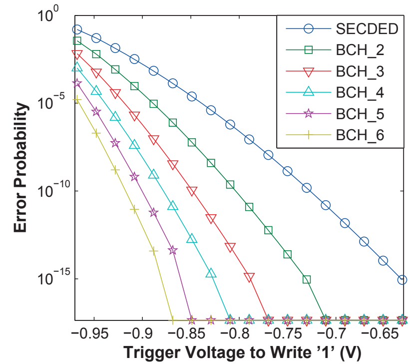

Fig. 4. Relationship between error probabilities and trigger voltages with different ECCs for SET operations (45nm, 32M FBDRAM.)

For the read operations, the access error rate is decided by the FBC's characteristic named as program window, which also follows the normal distribution. The program window is defined as the range of V_G, in which the difference of state '0' and '1' can be sensed in the read operation. If V_G in read operation is set too large, all stored data will be read out as bit '1'. On the contrary, all bits will be read as '0' when V_G is too small. For a single FBC, the largest read margin is obtained by setting V_G at the middle of the program window. With PV effects, the program window of FBCs also follows a statistical distribution. In order to achieve the highest probability of success in read operations, V_G is set based on the mean value of program window. Thus, increase the size of program window can reduces the error rates in read operations. The program window size can be improved by increasing I_{DS}. For the similar reason in write operations, increasing I_{DS} also causes extra power consumption.

Another important characteristic of the FBC is the retention time. The retention time is close related to the refreshing power of FBDRAM. Although the retention time is also statistically distributed, the prior research has shown that the retention time of FBDRAM is in the level of several seconds, which is much larger than that of eDRAM [1]. It means that the refreshing power consumption of FBDRAM contributes a little (less than 5% in this work) to total power consumption. Thus, the retention time is not the concern of this work.

Due to process variations, proper design margin need to be reserved for FBDRAM, and error correction mechanisms are necessary to achieve a required error rate. In next section, we explore error control in FBDRAM design with different error correction codes (ECC) and design margin. The trade-off among reliability, power consumption, and performance are studied.

IV. Error Control in FBDRAM

In this section, we discuss how to choose the ECC strength and corresponding design margin under a fixed error rate constraint. The

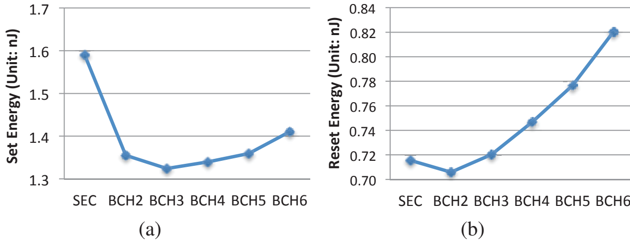

Fig. 5. Energy consumption of SET/RESET operation for a 32MB FBDRAM with different ECCs (45nm, error probability:10^{-9}).

optimized design point can be found for power reduction. Based on the characteristic of the FBDRAM, we propose to separate SET from RESET operations to enable the optimized design.

A. Error Correction Codes

Assume that the number of bits accessed in each read/write operation is N. The probability that errors happen in each read/write operation (P_{err}) is calculated as follows:

$$P_{err} = 1 - \Phi(V)^N \tag{3}$$

, where $\Phi(V)$ is the CDF calculated from the distribution of the trigger voltage for read/write operation discussed in the last section. In modern memory design, the ECC, such as parity check and SECDED, are used to ensure the access correction [14]. Assume K bits ECC can correct M bits of errors. After adding the ECC in the FBDRAM, the error probability is changed to the follows:

$$P_{err} = 1 - \sum_{i=0}^{M} \binom{N+K}{i} \cdot \Phi(V)^i \cdot (1-\Phi(V))^{N+K-i} \tag{4}$$

Figure 4 shows the error probabilities of SET operations when different programming voltages and ECCs are used in an FBDRAM design. The *BCH_i* represents the BCH code that can correct i-bit errors in each operation. Since the FBDRAM is supposed to be used as on-chip memory, the number of bits accessed in each operation is equal to the size of a cache line. In this work, the cache line size is $64Byte$ or $512Bit$, which is widely adopted in modern microprocessors.

The results in Figure 4 show that, in order to achieve an acceptable error probability for SET operations, the program write voltages have to be set much higher than the mean value in Figure 2 (a) (about -1.5V). The energy consumption is increased by more than 20% at the same time, due to the higher current I_{DS}. The results also show that increasing the strength of ECC can significantly reduce the program voltage to achieve the same error probability. Thus, using stronger ECC can reduce the energy consumption of a SET operation. With a stronger ECC, however, more bits are accessed in each write operation because the length of ECC is increased. In addition, more energy is consumed in the ECC encoding/decoding process.

With the same error rate constraint, we can trade ECC strength with the programming voltage to find the optimized write energy consumption. Figure 5 (a) shows the energy consumption of SET and RESET operations for a 32MB FBDRAM with different ECCs. For the SET operation, the energy consumption is significantly reduced after changing the ECC from *SECDED* to *BCH_2*. It is because the programming current can be greatly reduced after increasing the ECC strength. When we further increase the ECC strength, the write energy, however, is not decreased much and may be increased. The reason comes in two folds: (1) the reduction of programming current is not significant, as shown in Figure 4, (2) the energy overhead caused by ECCs is increased. The minimum energy consumption exists when *BCH_3* is used as the ECC. Using stronger ECCs not only causes more energy consumption, but also induces more overhead in access latency and area. Table II shows the overhead of access latency and extra bits for different ECCs strength, respectively. Although using *BCH_3* achieves the minimum energy consumption for the SET operation, the overhead in access latency and area offsets the benefits. Consequently, *BCH_2* is the best choice of ECC for SET operations in this work.

The case for a RESET operation is different from that for a SET operation. As shown in Figure 5 (b), the access energy consumption is only reduced a little when we change the ECC from *SECDED* to *BCH_2*. The reason is also in two folds. First, the RESET operation of FBDRAM is more tolerant to process variations so that the programming voltage of the RESET operation is not reduced much, when we increase the ECC strength. Second, the energy consumption of changing the FBC's status in the RESET operation is much lower than that in the SET operation. The energy consumption consumed on the peripheral circuitry becomes more dominating in RESET operation. Thus, we cannot gain much benefit from using stronger ECCs. The results show that the minimum RESET energy consumption happens when the *BCH_2* is employed as the ECC.

TABLE II
OVERHEAD OF USING DIFFERENT ECCS.

	SEC	BCH2	BCH3	BCH4	BCH5	BCH6
Extra Bits	11	20	30	40	50	60
Enc. Lat. (ns)	0.04	0.07	0.1	0.15	0.18	0.22
Dec. Lat. (ns)	0.7	1.3	2.1	2.9	3.7	4.5

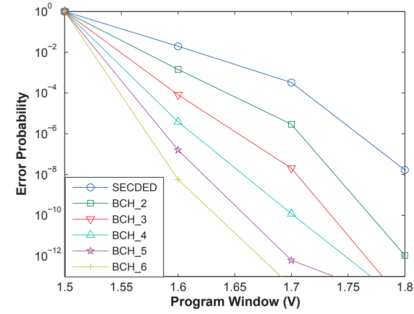

Fig. 6. Relationship between error probabilities and program window with different ECCs for read operations (45nm, 32MB FBDRAM).

For the read operation, the reliability is decided by the program window, which can be controlled through I_{DS}. Similarly to write operations, increasing I_{DS} causes more energy consumption and using stronger ECCs can help to reduce I_{DS}. Figure 6 shows the error probability in read operations when different I_{DS} and ECCs are used in a 32MB FBDRAM. The minimum read energy consumption also happens when the *BCH_2* is employed as the ECC.

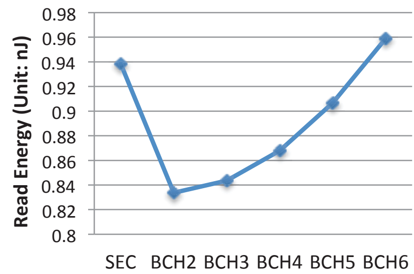

Fig. 7. Energy consumption of the read operation to a 32MB FBDRAM with different ECCs (45nm, error probability:10^{-9}).

B. Separating SET from RESET

In the write operations, the V_G required to achieve successful SET and RESET operations are different. For example, the mean value of V_G required for a successful SET operation is about $-1.5V$, but the mean V_G required for a RESET operation is about $-0.5V$. In the traditional write operation of RAMs, the bit '1' and bit '0' are programmed at the same time. Since the FBCs of a cache line share the wordline, which is used to control V_G. The V_G of all FBCs have to be set as $-0.5V$ in order to operate SET and RESET operations at the same time. According to results in Figure 4, the V_G of $-0.8V$ is enough to achieve a low error rate in SET operations with BCH_2. Thus, the bits for SET operations are over-programmed, and the BCH_2 ECC cannot help reduce the energy consumption in SET operations.

In order to leverage the benefits of strong ECC in SET operations, we propose to divide the write operations in two steps. The SET and RESET operations are separated from each other and finished in serial. There are two drawbacks in such a scheme: (1) the write latency is increased to the sum of timing for a SET and a RESET operation, (2) extra control logic is required, which results in extra area and energy overhead. For the write latency, the overhead is trivial because the RESET latency is much smaller than that of SET latency. The increase in write latency for a 32MB FBDRAM is about 17%. Since the write operation is normally not on the critical path of the memory access, the increase of write latency cause little degradation in performance. The control logic is not complicated, we need one extra multiplexer on the driver of each bit line. Due to the page limitation, we cannot provide all details. The simulation results show that the increase in area and energy consumption is negligible.

V. Evaluation Results

In this section, we first compare FBDRAM caches and traditional SRAM/eDRAM caches, in respect of timing, energy consumption, and the area. Then, we choose a specific configuration and show the experimental results in the system level for different workloads.

A. Circuit Level Evaluation

Figure 8 (a) compares the read latency of SRAM, eDRAM and FBDRAM caches with different capacity. Note that SECDED is used in both SRAM and eDRAM caches, and BCH_2 is used in FBDRAM caches. The results for FBDRAM using SECDED are not shown since they are very close to those using BCH_2. Figure 8 (a) shows that FBDRAM cache has longer read latency than that of SRAM, when the cache capacity is small ($< 2MB$). As the cache capacity increases, the read latency of FBDRAM caches becomes smaller than that of SRAM caches because of the smaller cell size of FBCs. Consequently, we can expect improvement of performance when we replace SRAM with FBDRAM for caches of large capacity. The results for eDRAM caches are also shown in the figure. We can find that the read latency of FBDRAM caches is always smaller than that of eDRAM caches with the same capacity. Note that the latency result also includes the timing consumed on the bus and memory controller.

The comparison of write latency is shown in Figure 8 (b), which is different from that of read latency. We can find that the write latency of FBDRAM caches are not comparable to those of SRAM and eDRAM ones until the cache capacity is larger than 64MB. It is because the latency of the SET operation is set to about $10ns$ in order to promise enough programming timing. Fortunately, we have argued that the long write latency can be hidden in the system level.

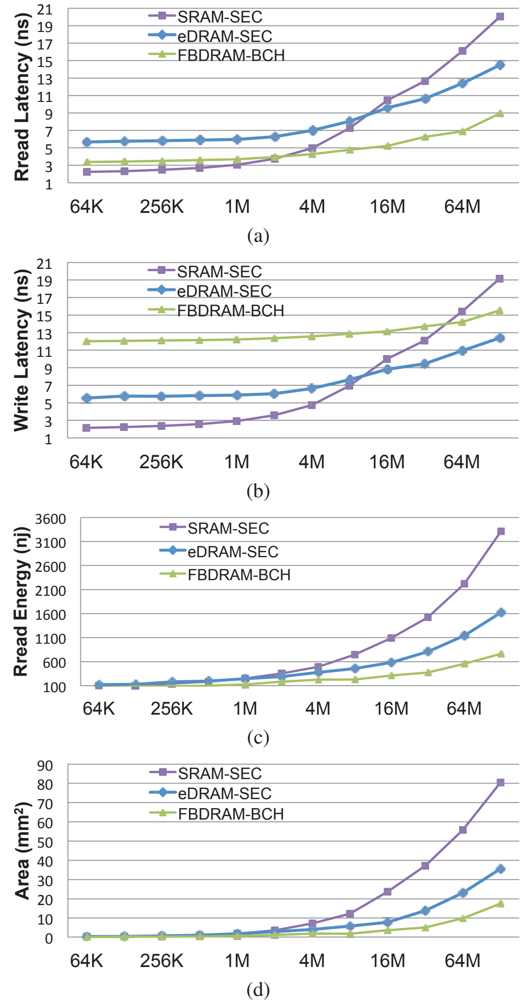

Fig. 8. Comparison of (a) read latency, (b) write latency, (c) read energy, and (d) the area.

Figure 8 (c) shows the comparison of read energy among different memory technologies. We can find that the difference is not significant for FBDRAM/SRAM/eDRAM caches of small capacity. Actually, the read energy of FBDRAM caches is always smaller than that of SRAM and eDRAM caches. As the capacity increases, the energy on peripheral circuitry become more dominating. Thus, we can save more energy consumption with FBDRAM because of its smaller cell size. Note that the RESET energy of FBDRAM has the similar trend as the read energy. The comparison between SET energy of FBDRAM and write energy of SRAM/eDRAM has the similar trend as the comparison of write latency. Due to page limitation, these results are not shown.

Figure 8 (d) presents the area comparison for different caches. We can find that the results are quite similar to those of read energy in Figure 8 (c). It is reasonable because the latency of peripheral circuitry is close related to the cache area. Consequently, as the peripheral circuitry latency become dominating in caches of large capacity, we can also find significant reduction in the area if we replace SRAM/eDRAM caches with FBDRAM caches.

B. System Level Evaluation

For the system level simulation, we use the ZESTO [15] simulator to measure performance. It is configured to model an eight-core

processor. Each core is similar to Intel core i7 with a 3GHz frequency. There are two levels of caches. The private IL1/DL1 caches are SRAM based and the capacity is fixed to 64KB. The L2 caches are shared among cores and can be implemented with SRAM, eDRAM, or FBDRAM. In the baseline configuration, the L2 cache is 4MB SRAM based cache. The baseline cache can be replaced with either eDRAM (16MB) or FBDRAM caches (32MB) of the similar area. The simulator captures data addresses from all loads, stores, and prefetch operations. We use the information to calculate the memory access intensity, and use it to compute the energy consumption of the cache hierarchy. Our workloads are sets of multiprogrammed benchmarks from SPEC2006 and PARSEC [16]. We randomly choose benchmarks from the full set and mix them together to help us create a diverse set of cache access patterns.

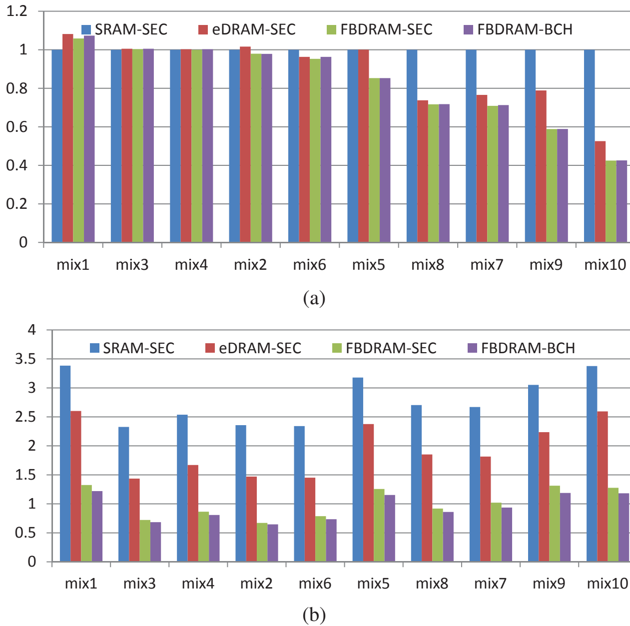

Fig. 9. (a) Normalized execution time with different caches, (b) Power consumption results for different caches (unit: Watt).

Figure 9 (a) compares the system performance of using different caches for 10 different workloads. For the first four workloads, the working sets are not very large so that we cannot gain much benefits from increasing the capacity of the L2 cache. Thus, the performance is not improved when we replace the 4M SRAM cache with 16M eDRAM or 32M FBDRAM. Due to the long read latency of eDRAM and long write latency of FBDRAM, the performance may even decreases for some workloads (e.g. mix1). For the last six workloads, the working sets become larger so that the performance is significantly improved when we use FBDRAM L2 cache. Note that label "FBDRAM-SEC" represents the results of using the FBDRAM cache, which employs SECDED instead of *BCH_2* as the ECC. The results show that the system performance is only degraded less than 1% when we apply stronger ECC (*BCH_2*) to the FBDRAM cache. Compared to the cases of using SRAM L2 cache and eDRAM L2 cache, the performance is increased by 12% and 6% on average, respectively.

Figure 9 (b) shows the comparison of power consumption results. Compared to the SRAM cache and the eDRAM L2 cache, the FBDRAM L2 cache consumes less power because of low leakage power consumption and low data refreshing rate. When the FBDRAM cache with SECDED is used, the power consumption is reduced by 64% and 48% on average, compared to the SRAM cache and eDRAM cache respectively. After we apply *BCH_2* to the FBDRAM cache as ECC, the power consumption can be further reduced by about 6%.

VI. Conclusions

The advantages of FBDRAM make it competitive to replace SRAM/eDRAM for the future multi-/many-core systems. Based on our FBDRAM model with the consideration of process variations, we explore FBDRAM cache design using different ECCs and corresponding design margin. The trade-off among performance, power consumption, and reliability is studied. Under a fixed error rate constraint, the FBDRAM with *BCH_2* is considered to be a good choice, in respect of power and performance. We compare FBDRAM caches with SRAM/eDRAM caches in both circuit and system levels. The results show that performance is improved and power is reduced, when we replace SRAM/eDRAM caches with FBDRAM caches.

Acknowledgment

The authors would like to thank Dr. Zhichao Lu and Dr. Jin Ouyang for making available some data and simulations.

References

[1] Z. Lu, N. Collaert, M. Aoulaiche, and *et al*, "A novel low-voltage biasing scheme for double gate FBC achieving 5s retention and 10^{16} endurance at 85C," *IEDM*, pp. 12.3.1–12.3.4, 2010.

[2] J. Kim, S. Chung, T. Jang, and *et al*, "Vertical double gate Z-RAM technology with remarkable low voltage operation for DRAM application," *VLSIT*, pp. 163–164, 2010.

[3] I. Ban, U. Avcl, D. Kencke, and P. Chang, "A scaled floating body cell (FBC) memory with high-k+metal gate on thin-silicon and thin-BOX for 16-nm technology node and beyond," *VLSIT*, pp. 92–93, 2008.

[4] J. G. Fossum, Z. Lu, and V. P. Trivedi, "New Insights on Capacitorless Floating-Body DRAM Cells," *IEEE Electron Device Letters*, pp. 513–516, 2007.

[5] Z. Lu, J. G. Fossum, and *et al*, "A Novel Two-Transistor Floating-Body/Gate Cell for Low-Power Nanoscale Embedded DRAM," *IEEE TED*, pp. 1511–1518, 2008.

[6] Z. Lu, J. G. Fossum, and Z. Zhou, "A Floating-Body/Gate DRAM Cell Upgraded for Long Retention Time," *IEEE Electron Device Letters*, pp. 731–733, 2011.

[7] J.-T. Lin and M. Chang, "A new 1t dram cell with enhanced floating body ef," in *MTDT06*, 2006, pp. 23–27.

[8] S. Thoziyoor, N. Muralimanohar, J.-H. Ahn, and N. P. Jouppi, "CACTI 5.1 technical report," HP Labs, Tech. Rep. HPL-2008-20, 2008.

[9] International Technology Roadmap for Semiconductors, "Process Integration, Devices, and Structures 2010 Update," http://www.itrs.net/.

[10] M. A. Horowitz, "Timing models for MOS circuits," Stanford University, Tech. Rep., 1983.

[11] E. Seevinck, P. J. van Beers, and H. Ontrop, "Current-mode techniques for high-speed VLSI circuits with application to current sense amplifier for CMOS SRAM's," *IEEE Journal of Solid-State Circuits*, vol. 26, no. 4, pp. 525–536, 1991.

[12] Y. Moon, Y.-H. Cho, H.-B. Lee *et al.*, "1.2V 1.6Gb/s 56nm $6F^2$ 4Gb DDR3 SDRAM with hybrid-I/O sense amplifier and segmented sub-array architecture," in *ISSCC09*, 2009, pp. 128–129.

[13] A. Agarwal, D. Blaauw, and V. Zolotov, "Statistical timing analysis for intra-die process variations with spatial correlations," in *ICCAD03.*, nov. 2003, pp. 900 – 907.

[14] J. Wang, S. Yaldiz, X. Li, and L. T. Pileggi, "Sram parametric failure analysis," *DAC*, 2008.

[15] G. Loh, S. Subramaniam, and Y. Xie, "Zesto: a cycle-level simulator for highly detailed microarchitecture exploration," in *Proceedings of the International Symposium on Performance Analysis of Systems and Software*, 2009, pp. 53–64.

[16] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The parsec benchmark suite: characterization and architectural implications," in *Proceedings of the International Conference on Parallel Architectures and Compilation Techniques*, 2008, pp. 72–81.