Energy Exploration and Reduction of SDRAM Memory Systems

Yongsoo Joo, Yongseok Choi, Hojun Shim,

Hyung Gyu Lee, Kwanho Kim, Naehyuck Chang[*]
School of Computer Science & Engineering, Seoul National University, Korea
naehyuck@snu.ac.kr

ABSTRACT

In this paper, we introduce a precise energy characterization of SD-RAM main memory systems and explore the amount of energy associated with design parameters, leading to energy reduction techniques that we are able to recommend for practical use.

We build an in-house energy simulator for SDRAM main memory systems based on cycle-accurate energy measurement and state-machine-based characterizations which independently characterize dynamic and static energy. We explore energy behavior of the memory systems by changing design parameters such as processor clock, memory clock and cache configuration. Finally we propose new energy reduction techniques for the address bus and practical mode control schemes for the SDRAM devices. We save 10.8mJ and 12mJ, 40.2% and 14.5% of the total energy, for 24M instructions of an MP3 decoder and a JPEG compressor, using a typical 32-bit, 64MB SDRAM memory system.

Categories and Subject Descriptors

B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aids; C.4 [**Performance of Systems**]

General Terms

Design, Experimentation, Measurement, Performance

Keywords

low power, memory system, SDRAM

1. INTRODUCTION

Recently, even battery-operated portable systems have become equipped with high-performance memory systems to accommodate high-end applications that have previously been running on desktop computers. Thus, memory systems become dominant power consumers, which has inspired the development of low-power memory

---

[*]Corresponding author

buses and memory devices. Off-chip main memory systems are major energy consumers even in hand-held embedded systems.

Previous power reduction work tends to focus on optimization techniques rather than the cost function, *i.e.*, power consumption models. Although these power reduction techniques are innovative, they are not ready for real implementation because they are based on simple energy models and do not consider cooperation with memory buses, bus drivers and memory devices. Simple capacitance models have been regarded as sufficient, and even transition counts are often treated as a power metric. System buses are mostly modeled as if they were composed of large capacitors, and even peripheral devices are often considered as capacitive loads [6]. SRAM-based memory is often used and the models of these systems have constant energy per access [4], while main memory systems composed of DRAMs or SDRAMs are more dominant power consumers than cache memories.

However, simple power consumption models may lead to seriously misguided power reduction practices [1], because it is important to take into account device-level access protocols in power reduction for memory systems. In this paper, we build a cycle-accurate energy simulator for off-chip SDRAM main memory systems and introduce energy reduction methods that are generally applicable to real implementations. In order to build a precise simulator, we perform intensive energy measurement and characterization. This paper presents state-machine-based energy characterization of memory systems, taking simultaneous account of memory buses, bus buffers, memory devices, and interaction between the processor, cache and main memory. We modify a switched-capacitor energy measurement technique [2] to account for leakage current as well. We borrow the energy consumption models of modern bus specifications introduced by chang *et al.*[1].

In this paper, we explore the energy behavior of SDRAM main memory systems considering these factors. Real application traces, which are acquired by special hardware support, drive the simulator. The simulator precisely replays the memory transactions together with the configurable cache memories and clock frequencies of the processor and memory system. Finally, we introduce practical energy reduction schemes.

2. ENERGY CHARACTERIZATION

2.1 Energy state machine

Finite state machines are widely used to describe the operation of digital systems. In this paper, we introduce an energy state machine to describe the energy consumption behavior of digital systems. A finite state machine M is four-tuple (S, Σ, δ, s_0) where $S = (s_0, ..., s_n)$ is a set of finite states, Σ is a finite input alphabet, $\delta : \Sigma \times S \to S$ is a state transition function, and s_0 is the initial state. We label each arc that denotes state transition δ with a finite set of state transitions $T = (t_0, ..., t_m)$. Fig.1 illustrates the variation in power supply current i_{dd} of asynchronous and synchronous devices, and justifies the static and dynamic energy association of the energy state machine.
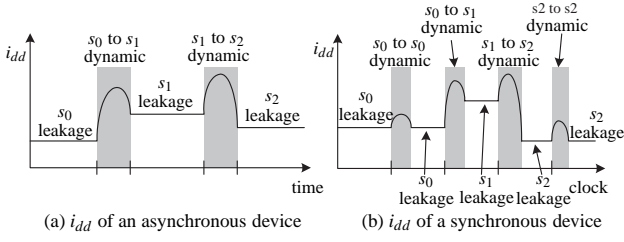
Figure 1: i_{dd} variation due to state change.

(a) asynchronous energy state machine
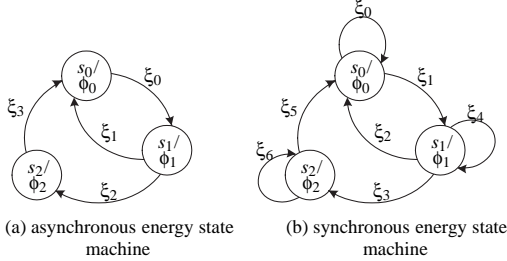
(b) synchronous energy state machine

Figure 2: The energy state machine.

Asynchronous devices consume dynamic energy when strobe signals are issued. The strobe signal changes the device state, thus leading to variation of the static energy consumption. Synchronous devices consume dynamic energy at each clock edge rather than at the logical state change.

Definition 1. *The energy state machine Π is a three-tuple (M, Φ, Ξ) where M is a finite state machine, $\Phi = (\phi_0, ..., \phi_n)$ is the leakage energy associated with the state $S = (s_0, ..., s_n)$, and $\Xi = (\xi_0, ..., \xi_m)$ is the dynamic energy associated with the transition $T = (t_0, ..., t_m)$.*

Fig. 2 shows examples of an asynchronous and a synchronous energy state machine. Each state and transition is associated with a static energy consumption ϕ_i and a dynamic energy consumption ξ_j, respectively.

Remark 1. *Let us consider an asynchronous state machine M_a. If the tenure time of $s_i = t$ such that $s_i \in M_a$, the static energy consumption is $t\phi_i$.*

Remark 2. *Let the clock period of a synchronous state machine M_s be τ. If M_s remains in s_i such that $s_i \in M_s$ for n consecutive clock cycles, the tenure time of state s_i is $n\tau$ and thus the static energy consumption is $n\tau\phi_i$.*

Remark 3. *The dynamic energy consumption Ξ does not vary with the tenure time of states and the clock frequency.*

Remark 4. *Previous current models associate a value of i_{dd} with each state. Let τ be a memory clock period. When a mode change occurs such that $s_i \xrightarrow{t_j} s_k$, the i_{dd} of s_k is determined by $\frac{\xi_j + n\phi_k}{n}$ where n is the tenure time of s_k in number of clock cycles. Because previous current models do not take n into account, the dynamic energy is under-estimated or the static energy is over-estimated.*

2.2 Annotation of the energy state machine

To complete the energy state machine, we have to annotate the energy values Φ and Ξ. To start with, we need to distinguish the energy consumption behavior of target devices before deciding a characterization method. So we further classify dynamic energy

consumption into *HDD* (Hamming-distance-Dependent Dynamic) energy, *WDD* (Weight-Dependent Dynamic) energy and *CD* (Common-mode Dynamic) energy. We also classify static energy, Φ, into *WDS* (Weight-Dependent Static) energy and *CL* (Constant Leakage) energy. Switching activity causes dynamic energy consumption by CMOS circuits. This switching activity is largely dependent on the Hamming distance of the data between the current and previous clock cycles in the static CMOS circuits; this is HDD energy. The dynamic CMOS circuits precharge before every evaluation. They draw a large current if the circuit has been discharged in the previous evaluation (usually the previous clock cycle); but they draw only a small current if the circuit has not been discharged. The energy consumption is mainly proportional to the weight[1] of the current data: the WDD energy. We do not need to annotate the energy values of each internal component of a device. So, a significant portion of the dynamic energy is represented in the form of common-mode values: the CD energy. Passive pull-ups or BiC-MOS drivers consume more static energy when the signal level is 0. This energy is inversely proportional to the duty ratio of the signal which is the WDS energy. Devices may also consume a constant leakage power supply current which is variable to a state of a device: the CL energy.

While some previous characterizations have ignored WDD or CL energy, most modern memory devices consume significant WDD and CL energy. In addition, previous characterizations have often averaged out instantaneous energy variations, but these variations are essential information in high-level energy reduction. For various reasons, devices are often operated slower than their maximum speed, (*i.e.* we issue strobe signals that are longer than necessary for asynchronous devices, or use slower clocks for synchronous devices). When we do not operate a device at full speed, the static energy consumption increases but dynamic energy does not change. This forces us to use intensive energy characterization to distinguish static energy from dynamic energy, which is difficult to come up with using conventional average power measurement.

Taking these problems into account, we decided to use a real-time cycle-accurate energy measurement technique [2] to characterize the state-machine-based energy characterization for memory devices. We sample the the supply voltage, v_s, at many points to determine the static energy consumption by measuring the slew rate. Because there is static energy consumption in a given state s_j, v_s continuously decreases with a constant slope. The leakage energy consumption is denoted by

$$\phi_j = \frac{C_s}{2} \frac{v_s^2(t + \Delta t) - v_s^2(t)}{\Delta t}, \qquad (1)$$

where C_s is the switched capacitor.

3. ENERGY CONSUMPTION OF MEMORY SYSTEM COMPONENTS

3.1 Energy consumption of memory buses

The energy consumption of memory buses is denoted by a energy state machine $\Pi_B = (M_B, \Phi_B, \Xi_B)$. The set of states M_B is (s_0, s_1), which represents driven low and driven high states respectively. The power consumption of LVT and GTL+ bus and bus drivers have already been studied [1]. We compose an energy state machine for an LVT bus (these are commonly used high-performance memory buses for embedded systems) by converting the power values to cycle-accurate energy values. We assume that the LVT drivers have bus-hold functions, which have been devised to avoid excessive WDS energy by eliminating external pull-up resistors. State s_2 and s_3 in Fig. 3 (c) are bus hold states. We control the output enable no later than the input change in order to keep rise and fall time. Thus no state change such as $s_0 \rightarrow s_3 \rightarrow s_1$ is allowed. This guarantees that $\xi_4 = \xi_5 = \xi_6 = \xi_7 = 0$. In addition, $\phi_2 = \phi_3 = 0$.

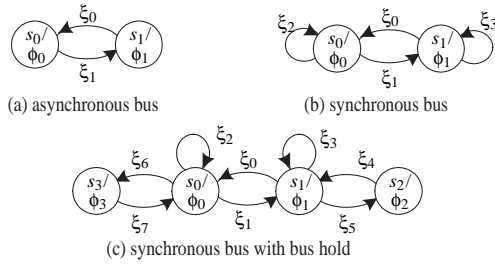[1]the number of 1's or 0's according to the logic structures.

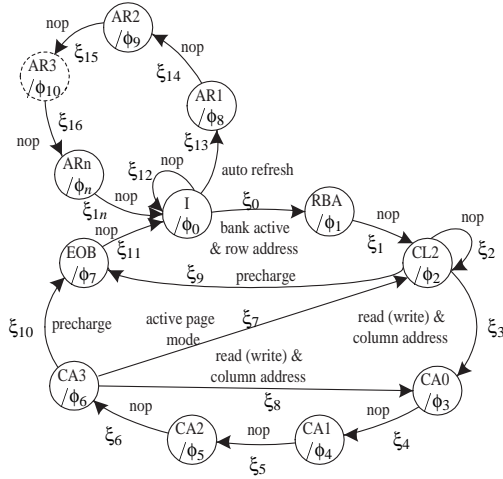**Figure 3: Energy state machine of off-chip memory buses.**

Figure 4: Energy state machine of an SDRAM.

We have constructed a 2-inch bi-directional bus, with a transmission line capacitance of 2.7pF, using a Fairchild 74LVT245, which has values of $\Xi_{aB} = (0.55, 0.55)$ and $\Phi_{aB} = (0.0053t, 0)$ for asynchronous buses, and $\Xi_{sB} = (0.55, 0.55, 0, 0)$ and $\Phi_{sB} = (0.0053\tau, 0)$ for synchronous buses. The units are nJ/bit, t is the tenure time of the asynchronous bus in nS, and τ is the clock period of the synchronous bus in nS.

3.2 Energy consumption of memory devices

In this paper, we focus on SDRAM devices, which are popular in equipment ranging from hand-held devices to desk-top computers. SDRAM has various modes of operation, but we consider two of the most important operations: burst-mode access and $\overline{\text{CAS}}$ before $\overline{\text{RAS}}$ refresh (Fig. 4).

Table 1 shows value of coefficient Ξ for our SDRAM. Let $f_1(\cdot)$

Table 1: Dynamic energy consumption of an SDRAM, Ξ (nJ/bit).

Ξ	Energy Cost
ξ_0	$\Theta_{ra} + c_{ra}f_1(A_r)$
$\xi_1 = \xi_2 = \xi_7 = \xi_{11} = \xi_{12}$	0
$\xi_3 = \xi_8$	read: $\Theta_{car} + c_{do}f_1(D_0) + c_{car}f_1(A_c)$ write: $\Theta_{caw} + c_{caw}f_1(D_0) + c_{caw}f_1(A_c)$
$\xi_4 = \xi_5 = \xi_6$	read: $\Theta_{car} + c_{do}f_1(D_i)$ write: $\Theta_{car} + c_{di}f_1(D_i)$
$\xi_9 = \xi_{10}$	Θ_{pr}
$\xi_{13} + \ldots + \xi_{1n}$	Θ_{rf}

Table 2: Common Mode Dynamic Energy, Θ (nJ/bit).

Symbol	Description	Samsung	Micron
Θ_{ra}	row active	1.625	1.158
Θ_{car}	column active (read)	1.80	0.929
Θ_{caw}	column active (write)	0.681	0.783
Θ_{pr}	precharge	0.149	0.261
Θ_{rf}	refresh	4.51	5.30

Table 3: Coefficient of Ξ (nJ/bit).

Coefficient	Description	Samsung	Micron
c_{ra}	row address input	0.16	0.05
c_{do}	data output (read)	0.11	0.12
c_{di}	data input (write)	0.07	0.09
c_{car}	column address input to read	0.15	0.19
c_{caw}	column address input to write	0.14	0.16

be the number of 1's in $\cdot$. Among Ξ, CD is the dominant component. Table 2 shows the CD energy of K4S280832B-TC1L and MT48LC16M8A2-7E SDRAM devices from two major vendors, Samsung and Micron respectively. They both have a 128Mbit capacity with a 4M address space $(12 \times 10) \times$ 8bit data $\times$ 4 banks.

4. HYBRID ENERGY ESTIMATOR

The energy consumption of main memory systems is widely variable depending on the processor, cache memory, system bus, bus clock frequency and so on. This section introduces a cycle-accurate energy simulator for off-chip main memory systems. Fig. 5 illustrates the organization of the simulator.

4.1 Processor

The processor generates a memory trace for a given application program and data. This is the most complex and time consuming part of system-level energy simulation. We analyze a real target processor connected over a PCI bus interface, and obtain exact memory traces in virtually real time. Our target processor is ARM7TDMI. Once a trace has been captured by the hardware, it is transferred to the cache simulator.

4.2 Cache

The cache module has separated instruction cache and data cache. We can configure index size, associativity and cache block size. The LRU replacement policy and critical-word-first scheme remain at their default setting.

4.3 Memory system

Memory system is composed of SDRAM controller, memory bus and memory device. We built a SDRAM controller with configurable bus clock frequency, burst length, $\overline{\text{CAS}}$ latency, burst refresh, precharge mode, and so on. We can also configure bus drivers, driver I/O capacitance, transmission line capacitance and load capacitance. We can configure capacity and organization of the memory devices.

5. ENERGY REDUCTION PRACTICES

Table 4: Static energy consumption, Φ (nJ/bit).

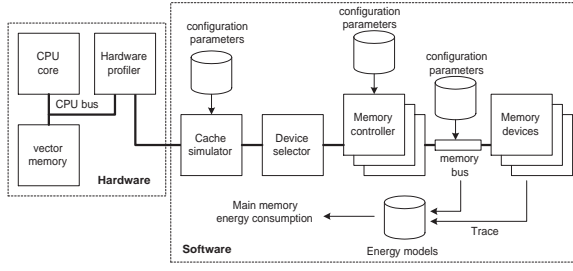| Φ	samsung	micron
$\phi_1 = \phi_2 = \phi_3 = \phi_4 = \phi_5 = \phi_6$	0.0058τ	0.0044τ
$\phi_0 = \phi_7 = \phi_8 = \phi_9 = \phi_{10} = \phi_n$	0.0019τ	0.0044τ

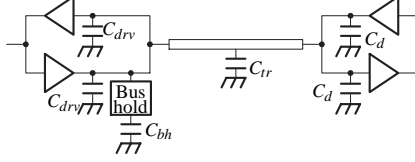**Figure 5: Estimation of main memory energy consumption.**

Figure 6: Data bus model.

5.1 Experimental setup

We have set up a typical SDRAM main memory systems for embedded applications. It has 32bit data width and 64MB capacity with four K4S280832B-TC1L devices. The length of the bus is 2 inches.

Fig. 6 shows the data bus model. C_{drv}, C_{bh}, C_{tr} and C_d are the I/O capacitance of the LVT driver, the effective bus-hold logic capacitance, the 2-inch transmission line capacitance and the I/O capacitance of the SDRAM. The energy state machine follows the arrangement shown in Fig. 3 (b). The energy consumption is given by

$$\xi_0 = \xi_1 = \frac{1}{2}(2C_{drv} + C_{bh} + C_{tr} + 2C_d)V_{DD}(V_{OH} - V_{OL}) + E_{drv},$$
$$\xi_2 = \xi_3 = 0, \phi_0 = 0.0053\tau, \phi_1 = 0. \qquad (2)$$

E_{drv} is the dynamic energy of the LVT driver, and includes the bias-overlapping current. Typically, $E_{drv} = 0.44$nJ, $C_{drv} = 4$pF, $C_{bh} = 0.5$pF, $C_{tr} = 2.6$pF, $2C_d = 5.25$pF and $V_{DD} = 3.3$V. V_{OH} and V_{OL} are output high and output low voltage of the bus driver, respectively. In order to accommodate many memory chips for large capacity, CBT bus switches are commonly used to reduce the load capacitance, and thus these values are still justified.

Fig. 7 shows the common configuration of SDRAM address buses. In this case too, the state machine also obeys that Fig. 3 (b). Energy consumption is the same to that of data bus except

$$\xi_0 = \xi_1 = \frac{1}{2}(C_{drv} + C_{tr} + 4C_a)V_{DD}(V_{OH} - V_{OL}) + E_{drv} \qquad (3)$$

where C_a is the input capacitance of the SDRAM, with a typical value of 3.75pF.

We have chosen two embedded application programs for the reduction practices: an MP3 decoder and a JPEG compressor with a 512×512 gray-scale Lenna image.
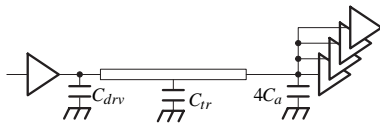

Figure 7: Address bus model.

Table 5: SDRAM memory bus energy consumption by design parameters (CJPEG, mJ/mW, 24M instructions).

Parameter		Address bus		Data bus	
		HDD	WDS	HDD	WDS
Processor clock (f_P) MHz	100	1.27/4.46	11.79/41.36	4.13/14.48	1.05/3.67
	133	1.27/5.71	9.20/41.32	4.13/18.54	1.05/4.70
	200	1.27/7.97	6.58/41.24	4.13/25.86	1.05/6.55
	266	1.27/9.90	5.29/41.17	4.13/32.13	1.05/8.14
	400	1.27/13.06	4.00/41.04	4.13/42.37	1.05/10.74
f_M =66MHz, 8KB/4word/2way-set-associative cache					
Memory clock (f_M) MHz	33	1.27/7.75	6.72/40.94	4.13/25.16	2.09/12.75
	66	1.27/9.90	5.29/41.17	4.13/32.13	1.05/8.14
	83	1.27/10.50	5.00/41.23	4.13/34.05	0.84/6.90
	100	1.27/10.68	4.92/41.31	4.13/34.63	0.70/5.85
	133	1.27/11.22	4.69/41.35	4.13/36.41	0.52/4.61
f_P =266MHz, 8KB/4word/2way-set-associative cache					
Cache size KB	1	5.78/22.44	10.09/39.18	16.44/63.83	4.48/17.40
	2	4.63/20.55	8.82/39.20	13.70/60.84	3.41/15.13
	4	3.91/19.01	8.05/39.15	12.14/59.05	2.73/13.29
	8	1.27/9.90	5.29/41.17	4.13/32.13	1.05/8.14
	16	0.62/5.59	4.65/42.03	2.17/19.61	0.53/4.79
f_M =66MHz, f_P =266MHz, 4word/2way-set-associative cache					
Cache set	1	1.30/10.04	5.31/41.15	4.11/31.86	1.14/8.86
	2	1.27/9.90	5.29/41.17	4.13/32.13	1.05/8.14
	4	1.36/10.36	5.40/41.13	4.45/33.90	1.09/8.27
	8	1.47/10.87	5.60/41.46	4.81/35.63	1.15/8.52
f_M =66MHz, f_P =266MHz, 8KB/4word cache					
Block size	4	1.27/9.90	5.29/41.17	4.13/32.13	1.05/8.14
	8	0.80/6.68	5.13/42.58	5.15/42.79	1.48/12.28
f_M =66MHz, f_P =266MHz, 8KB/2way-set-associative cache					

5.2 Energy reduction of memory buses

5.2.1 Exploration of memory bus energy

We perform a SDRAM memory system simulation with design parameters and have the energy information of bus and device. Table 5 shows the energy consumption behavior of memory bus according to the system design parameters. The memory bus energy consumption is composed of WDS and HDD energy.

WDS energy in the address bus is directly proportional to the execution time, which is in turn determined by the processor clock frequency and cache miss ratio. A higher memory clock frequency also reduces the WDS energy but not so significantly. The HDD energy of address bus is directly proportional to the number of addresses issued. Therefore, the more memory transaction that are performed, the more HDD energy is consumed. The memory transaction count is closely related to the cache miss ratio, but it is independent of the clock speed of the CPU and the memory system. Generally, the burst length of memory transaction is equal to the cache block size. Thus, a large block size in the cache reduces the number of memory transactions.

A conventional data bus configuration does not waste WDS energy in the idle period because it utilizes the bus-hold logic of the bi-directional bus driver. The WDS energy consumption of the data bus is exactly proportional to the cache miss ratio times the clock period of the memory bus. The HDD energy of the data bus grows as the count of memory transaction increases.

To reduce the energy consumption of the memory bus, we need to provide enough cache memory, or associativity exceeding the knee-point. Higher CPU and memory clock frequency is also helpful in reducing energy consumption of memory bus.

5.2.2 Energy reduction for the address bus

The major portion of energy consumption in the address bus is the WDS energy. Due to the conventional address bus architecture of the SDRAM, the address bus remains driven in most idle time and wastes WDS energy. The shaded areas in Fig. 8 represent the valid range of the address bus.

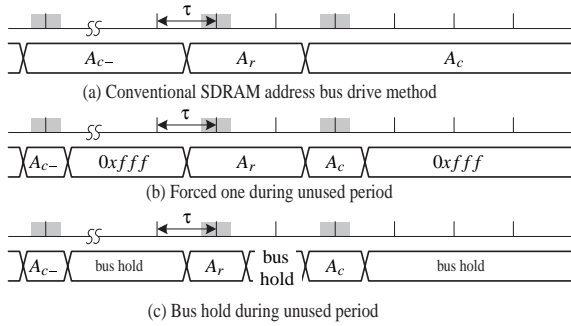We propose to minimize static energy consumed during idle time

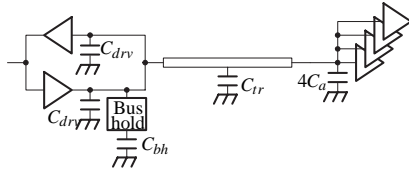**Figure 8: Address bus energy reduction schemes.**

Figure 9: Address bus energy reduction using bus-hold function.

when the address bus is not valid. We suggest using two static energy reduction schemes: *forced 1* and *bus hold* (Fig. 9). *Forced 1* scheme makes the idle period (white areas) high and thus saves WDS energy. The *forced 1* is economic but slightly increases dynamic energy. The *bus hold* scheme is a costly solution because we have to use bi-directional buffers for uni-directional drive in order to utilize the bus-hold function. The *bus hold* scheme induces a minor increase in switching capacitance (0.5pF). Table 6 shows that the address bus energy of MP3 decoder and JPEG compressor is reduced to 8.5% and 24.6% respectively.

5.3 Energy reduction for memory devices

5.3.1 *Exploration of memory device energy*

In practice, it is important to analyze energy behavior by changing conventional design parameters such as processor clock speed, memory clock speed and cache configuration depending on the application program. Table 7 shows how the energy behavior of the SDRAM devices is affected by changing these parameters.

When we increase the processor and memory clock frequencies, the total execution time decreases and thus both CL energy and CD energy decrease. A higher processor clock frequency results in smaller CL energy in idle mode while higher memory clock frequency reduces the CL energy in active mode. The CL energy rapidly decreases as the clock frequencies increase, because the total execution time directly affects CL energy. On the other hand, the CD energy slightly decreases because the number of refresh cycles is reduced. These clock frequencies do not affect the WDD

Table 6: Energy reduction for the SDRAM memory buses (mJ, 24M instructions, $f_M =$66MHz, $f_P =$266MHz, 8KB/4word/2way-set-associative cache).

App.	Reduction technique	HDD	WDS	Total	%
MP3	conventional	0.32	4.27	4.59	100.0
	forced 1	0.37	0.09	0.46	10.1
	bus hold	0.33	0.06	0.39	8.5
CJPEG	conventional	1.27	5.29	6.56	100.0
	forced 1	1.73	0.44	2.17	33.2
	bus hold	1.34	0.27	1.61	24.6

Table 7: Design parameters for SDRAM memory energy consumption (CJPEG, mJ/mW, 24M instructions).

Parameter		CL	CD	WDD	Total
Processor clock (f_P) MHz	100	19.0/66.6	65.8/230.8	2.3/8.1	87.1/305.5
	133	15.3/68.9	63.5/285.2	2.3/10.4	81.2/364.4
	200	11.7/73.0	61.2/383.1	2.3/14.5	75.1/470.6
	266	9.8/76.5	60.0/467.1	2.3/18.0	72.2/561.6
	400	8.0/82.3	58.9/604.3	2.3/23.7	69.2/710.3
$f_M =$66MHz, 8KB/4word/2way-set-associative cache					
Memory clock (f_M) MHz	33	14.2/86.8	61.1/372.3	2.3/14.1	77.7/473.2
	66	9.8/76.5	60.0/467.1	2.3/18.0	72.2/561.6
	83	8.9/73.8	59.8/493.3	2.3/19.0	71.1/586.1
	100	8.8/73.7	59.7/501.1	2.3/19.4	70.8/594.2
	133	8.0/70.5	59.6/525.4	2.3/20.4	69.9/616.2
$f_P =$266MHz, 8KB/4word/2way-set-associative cache					
Cache size KB	1	24.7/96.1	275.8/1070.9	10.8/41.8	311.3/1208.7
	2	20.9/93.0	222.9/990.0	9.0/40.1	252.9/1123.1
	4	18.6/90.6	191.4/930.7	8.0/39.0	218.1/1060.4
	8	9.8/76.5	60.0/467.1	2.3/18.0	72.2/561.6
	16	7.6/69.1	30.0/270.9	1.1/9.8	38.7/349.8
$f_M =$66MHz, $f_P =$266MHz, 4word/2way-set-associative cache					
Cache set	1	9.9/77.0	60.6/469.7	2.2/17.2	72.8/563.9
	2	9.8/76.5	60.0/467.1	2.3/18.0	72.2/561.6
	4	10.1/77.2	64.9/494.3	2.5/19.3	77.6/590.8
	8	10.5/78.1	70.8/524.5	2.8/20.8	84.2/623.4
$f_M =$66MHz, $f_P =$266MHz, 8KB/4word cache					
Block size	4	9.8/76.5	60.0/467.1	2.3/18.0	72.2/561.6
	8	9.5/78.9	66.2/550.0	2.2/17.8	77.9/646.8
f_M @66MHz, f_P @266MHz, 8KB/2way-set-associative cache					

energy. The cache configuration also significantly affects the CL, CD and WDD energy. The energy consumption shows a distinct knee-point as we increase memory size and the associativity. The knee-points are at 4KB/2way and 8KB/direct-mapped for the MP3 and JPEG compressor respectively. A larger block size results in energy loss.

5.3.2 *SDRAM mode control*

Table 1 shows that HDD energy is not consumed by SDRAM devices. Although WDD energy is consumed, its possible variation is only 10 to 15% of the CD energy. Actual variation are much less. This implies that the switching activity reduction of the SDRAM address bus and data bus would not be effective in energy reduction for SDRAM devices although it is a powerful in energy reduction for memory buses. Although we aggressively eliminated one third of WDD energy with our elaborated encoding scheme, we only reduced the total energy by around 1%. Thus, SDRAM mode control schemes must be introduced to reduce energy consumption significantly. SDRAM mode control schemes may be divided into two categories. The first is to shut down devices that are not being used, as to send them to a low-energy state [3]. The second is to force SDRAM devices to active or idle mode in order to minimize the cycle time [5]. This policy is not driven by the energy model, but reducing the cycle time reduces the total energy consumption as well as run time. The scheme requires correct break-even time for shutting down the devices, and the second scheme requires a correct estimation of the row hit behavior of SDRAM devices.

We have investigated the idle time distributions needed for effective mode control as shown in Fig. 10. At first, we investigate idle time distribution between consecutive memory operations (Fig. 10 (a)). Shutting down the SDRAM is conducted according to the time distribution. Second, we also investigate idle time distribution between consecutive row hits (Fig. 10 (c)). If consecutive burst-mode access refers the same row in the SDRAM device, we need not precharge and reactivate the row. The idle time between consecutive row hits determines the active and idle mode change scheme.

Modern SDRAMs have power down modes. SDRAMs may enter power down mode from active mode or idle mode. We call these transitions active power down *APD* and idle power down
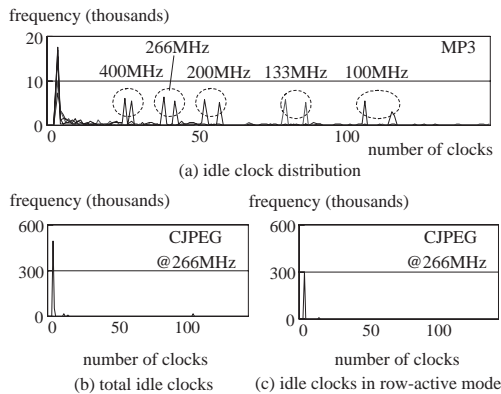
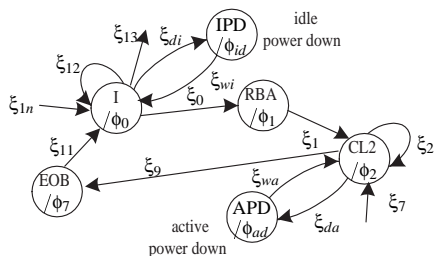Figure 10: Idle clock distributions (8KB/2way/4word cache).

Figure 11: Power down mode in SDRAMs.

App.	Reduction	CL	CD	WDD	Total	%
MP3	active page	12.0	14.3	0.3	26.6	123.3
	auto prechg	6.5	14.7	0.4	21.6	100.0
	delayed prechg	7.1	12.9	0.2	20.2	93.7
	auto prechg, power down	1.6	14.7	0.4	16.6	77.0
	delayed prechg, power down	2.3	12.9	0.2	15.4	71.5
CJPEG	active page	21.0	56.5	2.2	79.7	111.8
	auto prechg	9.8	59.2	2.3	71.3	100.0
	delayed prechg	12.6	54.6	2.0	69.2	97.0
	auto prechg, power down	4.9	59.2	2.3	66.4	93.1
	delayed prechg, power down	8.4	54.6	2.0	65.0	91.1

IPD, separately. For the Samsung K4S280832B-TC1L, $\phi_{id} = 1.6 \cdot 10^{-4}\tau$ and $\phi_{ad} = 7.5 \cdot 10^{-4}\tau$, while $\xi_{di} = \xi_{da} = \xi_{wi} = 0$ and $\xi_{wa} = 0.54$. It is very important to include dynamic cost Ξ in calculating mode control energy overhead. Otherwise, the mode control energy overhead will be greatly under-estimated or the CL energy will be greatly over-estimated as in [5]. For $I \to IPD \to I$, the energy cost is $(0.0038 + 1.6 \cdot 10^{-4}n)\tau$, where n is the length of time spent in *IPD* state, in clock cycles. Since $\xi_{di} = \xi_{wi} = 0$, there is no additional energy overhead in mode control although there is a loss of performance.

The precharge command changes the SDRAM's state from active to idle. The default mode of commercial SDRAM controller is auto precharge that immediately sends the SDRAM devices into the idle mode after every burst-mode transaction. We can optionally configure the SDRAM devices to stay in the row-active mode after burst mode transactions. The time spent in the row-active mode is determined by the refresh period, row miss and the forced precharge command. Delayed precharge sends the SDRAM device from the row-active mode to the idle mode after a predefined time-out value. Commercial SDRAM controllers commonly set up the time-out value to be 256 clock steps. However, Fig. 10 (c) shows that the dominant spectrum is located at 3 clock which is desirable time-out value. Table 8 shows the aggregate energy reduction of a SDRAM memory devices when we run the MP3 decoder and the JPEG compressor.

5.4 Low-power SDRAM main memory system

The energy reduction schemes for memory buses and devices are orthogonal (independent). The aggregate energy reduction amount will be the summation of all the schemes. We applied these reduction schemes to 24M instructions of MP3 decoder and 24M instructions of JPEG compressor with a 512×512 gray-scale Lenna image, using a 266MHz processor clock, a 66MHz memory clock, a 8KB/4word/2way-set-associative cache and a 32-bit 64MB Samsung SDRAM main memory system. We use bus-invert coding for

the data bus, *bus hold* for the address bus, delayed precharge and idle-mode power down for the SDRAM devices. The result for MP3 is 40.2% reduction: 26.8mJ to 16.0mJ. For JPEG compressor, we have 14.5% reduction: 83.0mJ to 71.0mJ.

6. CONCLUSION

Memory systems are dominant energy consumers, and thus many energy reduction techniques for memory buses and devices have been proposed. For practical energy reduction, we have to take into account interactions between a processor and cache memories, as well as application programs; and the energy characterization of memory systems must be sufficiently accurate. In this paper, we introduced an energy simulator for memory systems which is accelerated by special hardware support, while maintaining accuracy. We explored the energy behavior of memory systems and achieved low-power consumption for given applications by changing the processor and memory clock frequencies and cache configuration.

Our simulator is based on the precise energy characterization of memory systems including buses, bus drivers and memory devices by a cycle-accurate energy measurement technique. We characterize the energy consumption of each component by an energy state machine whose states and transitions are associated with dynamic and static energy costs. Our approach easily characterizes the energy consumption of complex SDRAMs. The energy simulator enables us to devise practical energy reduction schemes that are justified by actual data and achieve real reductions of the total energy consumption of main memory systems.

7. REFERENCES

[1] N. Chang, K.-H. Kim, J. Cho, and H. Shin. Bus encoding for low-power high-performance memory systems. In *Proceedings of ACM/IEEE Design Automation Conference*, pages 800–805, 2000.

[2] N. Chang, K.-H. Kim, and H. G. Lee. Cycle-accurate energy consumption measurement and analysis: case study of ARM7TDMI. In *Proceedings of International Symposium on Low Power Electronics and Design*, pages 185–190, July 2000.

[3] X. Fan, C. S. Ellis, and A. R. Lebeck. Memory controller policies for dram power management. In *Proceedings of International Symposium on Low Power Electronics and Design*, pages 129–134, April 2001.

[4] P. Hicks, M. Walnock, and R. M. Owens. Analysis of power consumption in memory hierarchies. In *Proceedings of International Symposium on Low Power Electronics and Design*, pages 239–242, 1997.

[5] S. Miura, K. Ayukawa, and T. Watanabe. A dynamic-SDRAM-mode-control scheme for low-power systems with a 32-bit risc cpu. In *Proceeding of International Symposium on Low Power Electronics and Design*, pages 358–363, April 2001.

[6] M. R. Stan and W. P. Burleson. Bus-invert coding for low power I/O. *IEEE Transactions on VLSI*, 3(1):49–58, March 1995.