Static Energy Reduction Techniques in Microprocessor Caches

Heather Hanson, Stephen W. Keckler, Doug Burger
Computer Architecture and Technology Laboratory
Department of Computer Sciences
Tech Report TR2001-18
The University of Texas at Austin

www.cs.utexas.edu/users/cart

cart@cs.utexas.edu

ABSTRACT

Managing power and energy consumption has become a primary consideration for microprocessor design. This report examines the effect of technology scaling on static power and energy dissipation and evaluates three techniques to reduce static energy in primary and secondary microprocessor caches. We examine the energy and performance tradeoffs associated with each technique and present the leakage-reduction configurations that minimize the energy-delay product. Our experimental results show that in the best case, the energy-delay product is reduced by 2% in the level-1 instruction cache, 7% in the level-1 data cache, and a factor of 50 in the level-2 unified cache. This technical report is an updated edition of a Masters Report submitted in May, 2001 by Heather Hanson to the Department of Electrical and Computer Engineering at The University of Texas at Austin.

Table of Contents

List of Tables

List of Equations

List of Figures

Chapter 1 Introduction

Managing power and energy consumption have become important design consideration for microprocessors. With each generation of semiconductor fabrication technology, transistors are engineered to be smaller and faster. Microprocessor performance improvements from using higher clock frequencies and more devices per chip have been accompanied by an increase in dynamic power dissipation from transistor switching activity. As fabrication technology scales to sub-180nm device sizes, static power due to increased subthreshold leakage current is emerging as a significant contributor to microprocessor power. While most existing low-power design techniques target dynamic power, future technology generations will require additional circuit and architectural mechanisms to reduce static power. This report investigates the source of static power, and then evaluates three techniques to reduce static power and energy in microprocessors.

1.1 MICROPROCESSOR POWER TRENDS

Microprocessor power consumption has been increasing with each product generation for high-performance systems. Increased power translates to more heat generated by integrated circuits, which in turn leads to slower switching speeds and degraded reliability. Power and energy constraints limit product design throughout the spectrum of embedded microcontrollers through high-end servers. In embedded systems, such as cell phones and other consumer products, chip temperature must be regulated without sophisticated cooling systems, and energy must be low enough for a reasonable battery life. Supplying current to many desktop personal computers or to enterprise server systems requires a substantial amount of electricity. The trend of increased power consumption through product generations has caused power budgets to displace manufacturability as a leading constraint for microprocessor performance [1].

1.2 POWER

The total power in complementary metal-oxide semiconductor (CMOS) integrated circuits is described by Equation 1.

Equation 1: CMOS power

$$P = \frac{1}{2} C \cdot V_{DD} \cdot V_{swing} \cdot a \cdot f + I_{leakage} \cdot V_{DD} + I_{sc} \cdot V_{DD} \quad [2]$$

In this equation, **C** is the device and interconnect capacitance, $\mathbf{V_{DD}}$ is the supply voltage, and $\mathbf{V_{swing}}$ is the voltage range through which a signal switches, typically equal to V_{DD} for static CMOS circuits. The coefficient **a** is an activity factor that represents the fraction of transistors switching, and **f** is the operating frequency. $\mathbf{I_{leakage}}$ is the leakage current that flows through

transistors while they are nominally off. $\mathbf{I_{sc}}$ is the short-circuit current due to an electrical path between power supply and ground, such as the momentary connection made while gate outputs switch. The equation's first term describes the dynamic power and the second term is static power; the third component is the short-circuit power, which is usually negligible for static CMOS circuits.

1.2.1 Dynamic Power

As Equation 1 indicates, dynamic power is proportional to frequency, capacitance, and the square of the supply voltage. Supply voltage and capacitance per transistor decrease as technology scales to smaller device sizes. However, the clock frequency and number of transistors integrated per chip have approximately doubled every generation [1]. The combined effect is a net increase in dynamic power as technology scales to future fabrication processes. Dynamic power has been the dominant source of power consumption, but the ratio of dynamic to static power is shifting as leakage current increases with each technology generation.

1.2.2 Static Power

Static power is dissipated by leakage current flowing through transistors while they are nominally off. Leakage current was negligible in technology generations prior to the 180nm node, but is increasing as fabrication technology scales to smaller devices as a side effect of reduced supply voltages. Supply voltages are lowered to maintain reasonable electric fields and reduce dynamic power. Transistor threshold voltages must then be reduced to maintain fast switching capability and provide sufficient noise margin with low supply voltages.

Unfortunately, reducing the threshold voltage causes leakage current to increase exponentially. Leakage current also increases exponentially with increasing temperature. As transistors become leakier and more transistors are integrated on each chip in successive process generations, static power increases. One estimate is that static power will account for 26% of total power dissipated per chip for a 130nm technology, and 56% for a 100nm process at a chip junction temperature of 110° C [3]. The contributions of dynamic and static power for this projection are illustrated in Figure 1 and Figure 2.

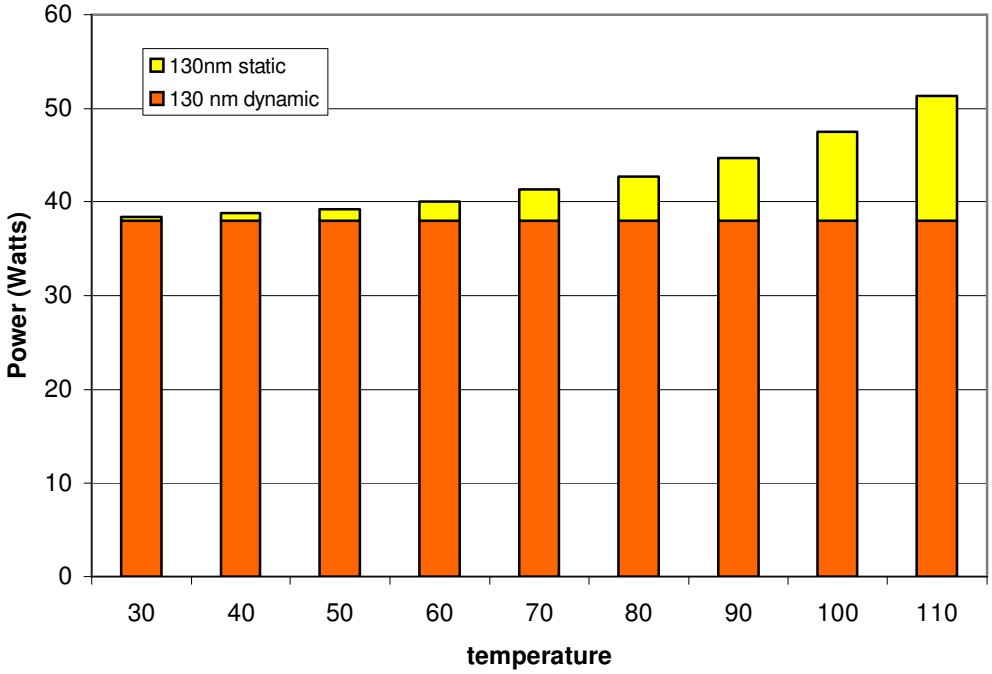


Figure 1 Static and Dynamic Power for 130nm Technology Generation

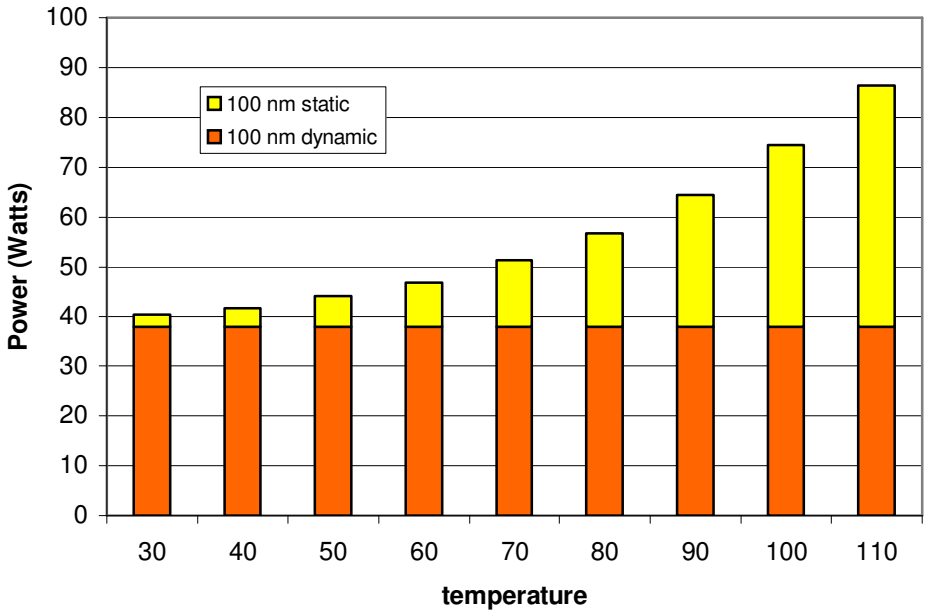


Figure 2 Static and Dynamic Power for 100nm Technology Generation

Static power reduction is an emerging research area, as traditional low-power techniques for reducing dynamic power are no longer sufficient to curb the steady increase in microprocessor power.

1.2.3 Power and Energy Reduction Research

Several solutions for reducing static power and energy in microprocessors target on-chip memory structures. Caches, in particular, present an opportunity to reduce a significant amount of leakage current since they contain a large fraction of a microprocessor's transistors. We examine three techniques that effectively raise the threshold voltage, V_T, of SRAM (static random-access memory) cells in on-chip caches in this study.

The first technique uses a combination of low-leakage and high-performance transistors in the cache structure, and is named dual-V_T for the two levels of transistor threshold voltage. In this technique, the SRAM memory cells contain high-V_T devices to have minimum leakage throughout the array, and the peripheral circuits employ low-V_T transistors for fast access times. The mix of transistor types achieves a constant rate of leakage reduction, determined during the circuit design. This technique does not adapt to program behavior, but is a simple implementation that requires no additional circuitry or control hardware.

A second technique also uses two threshold voltage levels, though with a different effect. Powell, et al. demonstrates a technique named gated-V_{DD} that disconnects memory cells from power or ground supplies through an extra, high-threshold voltage transistor [4]. The technique reduces leakage current when the gating transistor is deactivated, causing memory cells to lose stored data.

A third technique dynamically changes transistors' substrate bias to increase the threshold voltage, reducing leakage current while the circuit is in an idle mode. This technique, named MTCMOS, preserves the memory cells' contents as it selectively transitions cache lines into and out of a low-leakage idle mode [5].

We examine the effect of current state-of-the-art leakage reduction mechanisms by incorporating these three power-reducing techniques into an architectural simulator, and measuring microprocessor performance and energy expenditure. Our experiments show that each technique is effective in reducing static energy—static power dissipated through time—in on-chip caches at the expense of some degradation in microprocessor performance.

1.3 SUMMARY

Power has become a primary design constraint for microprocessors [6]. Static power is becoming a larger component of microprocessor power, and will require innovative power management techniques to ensure reliable operation within current supply, battery, and thermal requirements. This report examines the effect of technology scaling on leakage current, the source of static power dissipation, and evaluates techniques to reduce static energy in microprocessor caches. The report is organized as follows. Chapter 2 explores the relationship of fabrication process scaling, leakage current and static power and energy. It explains how the physical properties and operating conditions of CMOS transistors determine leakage current, and the trend of increasing static power due to leakage current. Chapter 3 contains details of our investigation of three leakage-reduction techniques, including memory cell circuits, our

architectural simulator, and the experimental methodology of simulating the effect of leakage-reducing circuits on microprocessors' energy and performance. Chapter 4 presents our simulation results with leakage-reduction techniques applied to on-chip caches. We examine the energy and performance tradeoffs associated with each technique. Chapter 5 provides information on related work in the area of static energy modeling. This report concludes with summary comments in Chapter 6.

Chapter 2 Leakage Current Models

Static power in CMOS integrated circuits is negligible for technology generations introduced prior to the 180nm node, but is increasing dramatically as technology scales to smaller transistors and lower supply voltages. The increase in static power is due to transistor fabrication parameters and operating conditions that increase leakage current, which flows through transistors while they are nominally off. This section begins with an overview of CMOS leakage current and indicates how leakage current behavior is changing in future fabrication processes.

2.1 CMOS CURRENT DEFINITION

Classical CMOS transistor current equations describe drain current, I_{DS}, in subthreshold, linear and saturation regions, which are defined by the relationship between the gate-to-source voltage $V_{GS,}$ drain-to-source voltage V_{DS}, and threshold voltage V_T:

Equation 2: drain current in subthreshold region, $V_{GS} < V_T$

$$I_{DS}(subthreshold) = 0$$

Equation 3: drain current in linear region, $V_{GS} > V_T$ and $V_{DS} < (V_{GS} - V_T)$

$$I_{DS}(linear) = \frac{\mu_n \cdot C_{ox}}{2} \cdot \frac{W}{L} \cdot \left[2 \cdot (V_{GS} - V_T)\, V_{DS} - V_{DS}^{\,2}\right]$$

Equation 4: drain current in saturation region, $V_{GS} \geq V_T$ and $V_{DS} \geq (V_{GS} - V_T)$

$$I_{DS}(saturation) = \frac{\mu_n \cdot C_{ox}}{2} \cdot \frac{W}{L} \cdot \left[(V_{GS} - V_T)^2 \cdot (1 + \gamma \cdot V_{DS})\right]$$

In these equations, **W** and **L** are the transistor channel width and length, respectively. $\mathbf{C_{ox}}$ is the oxide capacitance, μ is the mobility, and γ is the body effect parameter [7].

The linear and saturation equations describe a transistor's on-current. The on-current level determines the transistor's switching speed: higher currents are able to discharge capacitive loads more quickly. Note that the on-current increases as the potential difference between V_{gs} and V_T increases. In older CMOS technology generations, on-current is several orders of magnitude greater than the off-current, which is usually approximated as zero. However, in contemporary and future CMOS processes, the off-current is no longer negligible, due to increasing transistor leakage currents. Transistor leakage current is a combination of current through three paths within the device: through the gate oxide, from drain and source regions to the substrate (known as reverse-bias current), and also through the channel between drain and source regions. The subthreshold leakage current between source and drain is currently the largest of these components, and will increase in future fabrication technologies as threshold voltages are reduced. For the purposes of this study, we neglect the contribution of gate and

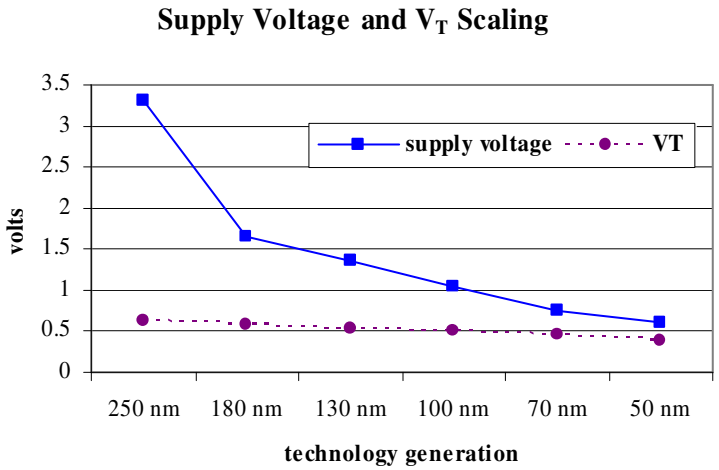


Figure 3 Supply Voltage and Threshold Voltage Scaling with Technology Generations

reverse-bias leakage currents and equate the terms *leakage current,* I_{off}, and *off-current* with subthreshold leakage current.

As supply voltages scale to smaller values in order to reduce dynamic power and maintain reasonable electric fields, threshold voltages must be reduced to provide sufficient noise margins for input signals. In older technology generations, the threshold voltage for silicon CMOS devices was typically about 0.7 volts; as supply voltage is lowered, VT is engineered to lower values. Figure 3 shows our projected voltage supply and threshold voltage scaling through a range of process technologies. As the envelope of supply voltage narrows, threshold voltage values diminish. Reduction of VT has a significant impact on-current. Lowering the threshold voltage allows the transistor to switch quickly and operate reliably with a low supply voltage, but results in more leakage current. The exponential increase in Ioff for an SRAM cell is shown in Figure 4; in this example of a 70nm device, the supply voltage is 0.75 volts, and VT varies throughout a range of 0.23 to 0.65 volts.

Transistors in technology generations introduced prior to the 180nm node have high threshold voltages and a large difference between the supply voltage and threshold voltage, resulting in a device with high drive current yet low leakage current. In contemporary and future technology generations, the separation between on-current and off-current will diminish because of the reduction of supply voltages and V_T. To achieve a high on-current with a reduced V_T, the off-current will be high; to maintain a low off-current, the saturation current will be low. Beginning with the 180nm node, fabrication processes provide two types of transistors, a high-V_T transistor for lower power and a low-V_T transistor for faster switching speed. Circuit designers can select low-V_T transistors for critical paths and high-V_T transistors in power-critical circuits and paths with timing slack [8][9]. Two of the static power reduction techniques we evaluate in this report use combinations of low-V_T and high-V_T transistors to balance speed and power requirements.

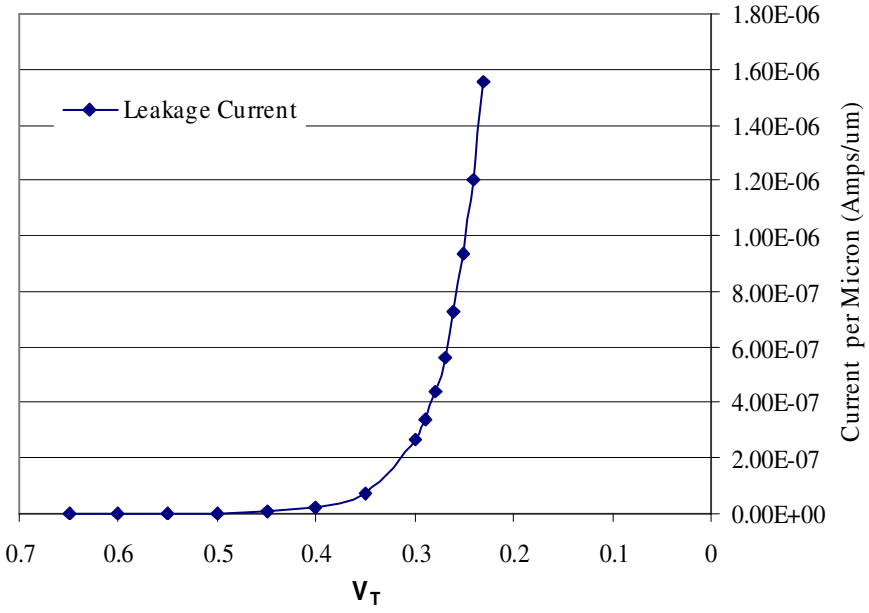


Figure 4 : Leakage Current as a Function of Threshold Voltage

2.2 TECHNOLOGY SCALING

Semiconductor physics dictates that leakage current will increase as V_T values decrease. The magnitude of I_{off} for a transistor in a future process technology is less predictable, since the physical parameters that determine the current value are not yet fully defined. Without direct measurements available, we present several projections and models, and discuss trends of current and power increases.

2.2.1 Projections

One prediction asserts that a 15% reduction in VT results in a five-fold increase in Ioff [10]. The International Technology Roadmap for Semiconductors (ITRS) produced by the Semiconductor Industry Association predicts that Ioff will double with each generation for both high-performance (low VT, high leakage) and low-power (high VT, low leakage) transistors, with Ioff for high-performance transistors in the nA/micron range and for low-power transistors in the pA/micron range [11].

2.2.2 Analytical Models

The HSPICE circuit simulator relies on analytical and semi-empirical transistor models to describe I_{off}. In our experiments, we use a transistor model governed by Equation 5:

Equation 5: simulated drain current

$$I_D = I_{on} \cdot e^{(V_{GS} - V_{on}) \cdot (q / nkT)}$$

where $V_{on} = V_T + \frac{nkT}{q}$

and $n = 1 + \frac{q \cdot NFS}{C_{ox}} + \frac{C_d}{C_{ox}}$.

In this equation, $\mathbf{I_D}$ is the leakage current (drain current under leakage conditions), $\mathbf{I_{on}}$ is the on-current, $\mathbf{V_{GS}}$ is the gate-to-source voltage difference, and $\mathbf{V_T}$ is the threshold voltage. The combined terms $\frac{nkT}{q}$ are the thermal voltage. The model shown here is the HSPICE level-3 model, in which the parameter ***n*** depends upon the drain capacitance, $\mathbf{C_d}$ and oxide capacitance, $\mathbf{C_{ox}}$, and an indication of the oxide interface quality, **NFS** [12].

2.2.3 Current Measurements

We simulated current flow through memory cells with the HSPICE circuit simulator and level-3 HSPICE transistor models derived from the CACTI 2.0 cache model for an 800nm technology [13]. NMOS and PMOS (N-type metal-oxide semiconductor and P-type metal-oxide semiconductor) parameters such as oxide thickness and junction capacitance are scaled to fit each process technology. Figure 5 shows projected leakage currents for 180nm through 50nm technology generations at room temperature, 25 – 30° C, normalized to transistor width in units of Amps/micron for three projections: linear scaling, an industry roadmap, and our experimental HSPICE models. The projections incorporate differing expectations of threshold voltage scaling and process parameters, with the result that leakage current projections vary by several orders of magnitude across a range of future technology generations.

2.4 TEMPERATURE DEPENDENCE

Leakage current is strongly dependent on transistor junction temperature. In this report, we refer to data at several temperatures, reflecting the range of reported temperatures from original sources. Typical room temperature is 25° to 30° C; and chips can reach higher temperatures of 80° to 110° C during operation. As a chip's temperature increases, leakage current increases, leading to a "self-heating" effect where increased heat generation from static power induces more static power dissipation. With the self-heating feedback effect, it is critical to reduce leakage current sources to control total power dissipation. Figure 6 shows the effect of temperature on leakage current per micron of transistor width for a range of technology generations as measured with HSPICE transistor models. For all technology generations, the leakage current at an operating temperature of 100° C is substantially higher than room temperature and cooler temperatures.

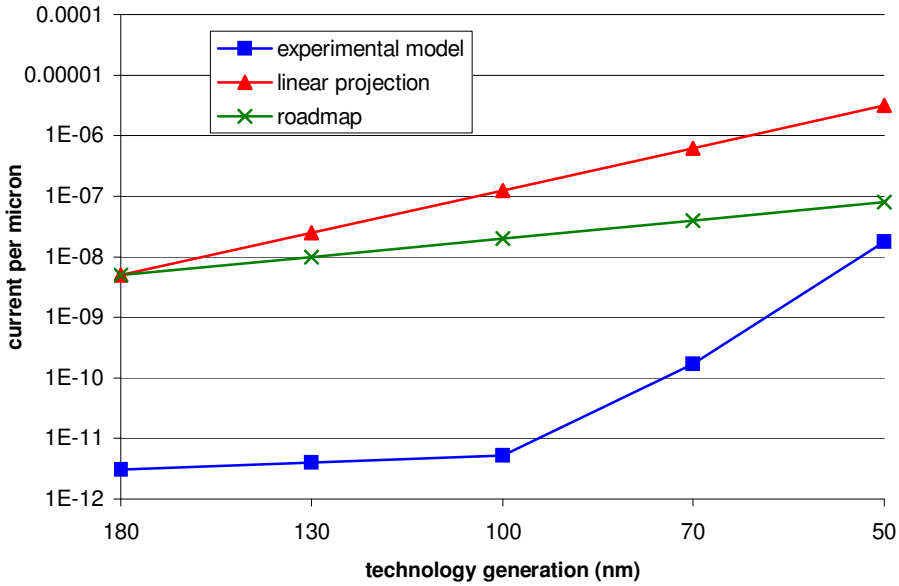


Figure 5 Leakage Current Projections

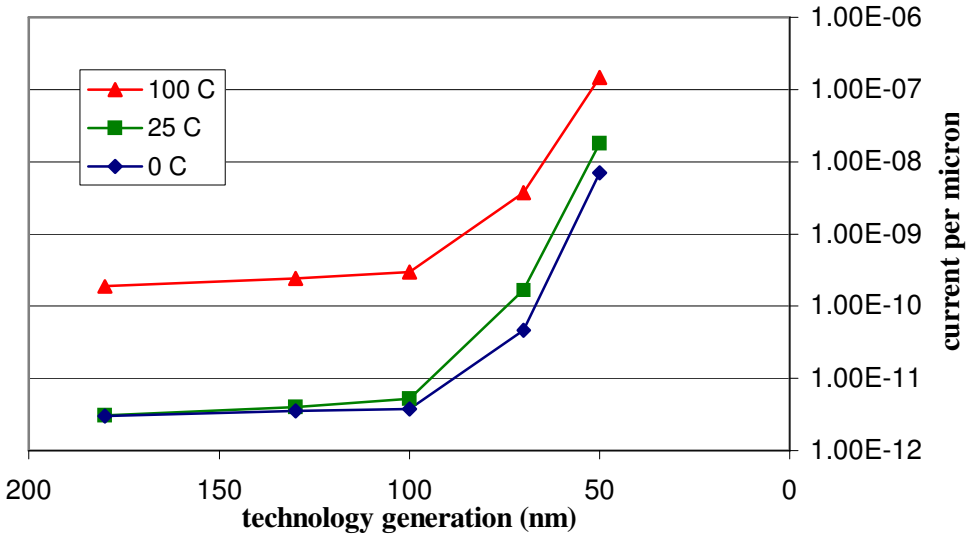


Figure 6 Leakage Current Temperature Dependence

2.5 CACHE MODEL

To quantify the effect of leakage current on microprocessor cache energy, we applied current measurements and projected supply voltage levels to estimates of cache capacity across a range of technology generations. First, we measured leakage current by simulating an SRAM memory cell circuit and measuring subthreshold leakage current through the circuit for each technology generation in the range of 180nm through 50nm. We adapted CACTI's memory cell,

which was designed for an 800nm technology, to our study by linearly scaling transistor widths for each generation [13]. The drain current is proportional to the ratio of transistor width to length; by scaling both width and length by the same value, drain current measured in different technology generations is not skewed by transistor sizing. Note that in this memory cell, the NMOS transistors are larger than the PMOS transistors to optimize the cell for read accesses.

Table 1: Technology Parameters for an SRAM Memory Cell

	180nm	*130nm*	*100nm*	*70nm*	*50nm*
Supply voltage (volts)	1.65	1.35	1.10	0.75	0.60
$\mid V_T \mid$ (volts)	0.60	0.54	0.50	0.46	0.38
PMOS width (microns)	0.54	0.38	0.30	0.21	0.15
NMOS width (microns)	1.08	0.77	0.60	0.42	0.30

We calculated the amount of static power dissipated by caches for a range of fabrication technologies by multiplying the leakage current per cell and projected supply voltage to find the static power dissipated per SRAM cell. Table 1 shows supply voltages and threshold voltages, respectively, for each technology generation in the range from 180nm to 50nm. Then, we projected cache capacities for each technology generation and calculated static power dissipated per cache using the approximation that all transistors in a cache are in the SRAM array (neglecting decoders, sense amps, etc.). Although smaller transistor widths are employed as technology scales to smaller minimum device sizes, the leakage current per transistor width increases each generation, and more transistors fit on a chip. The increase in leakage current outweighs the reduction in supply voltage at each generation, for a net effect of increased static power dissipation with each fabrication generation, illustrated in Figure 7. The graph plots power dissipation for projected cache sizes for future technology generations. The combined effect of large memory structures and large leakage current results in expected power dissipation approaching 100 watts for our experimental models of low-leakage transistors, and nearing a kilowatt for high-performance devices. The linear projection is shown as a reference for the extreme range of power dissipation if leakage current increases by a factor of 5 each generation.
In the ITRS documents, the Semiconductor Industry Association warns that leakage current will become a serious problem as technology scales to smaller devices. The roadmap charts static power reduction needed to maintain reasonable operation, suggesting that leakage currents in future generations will exceed heat sink capabilities with high-performance designs and battery limitations for low-power designs. According to the roadmap, static power reduction required from circuit and system techniques jumps from 0 percent for the 180nm node to 65 percent at 130nm, and continues increasing to 95 percent for 70nm technologies [11].

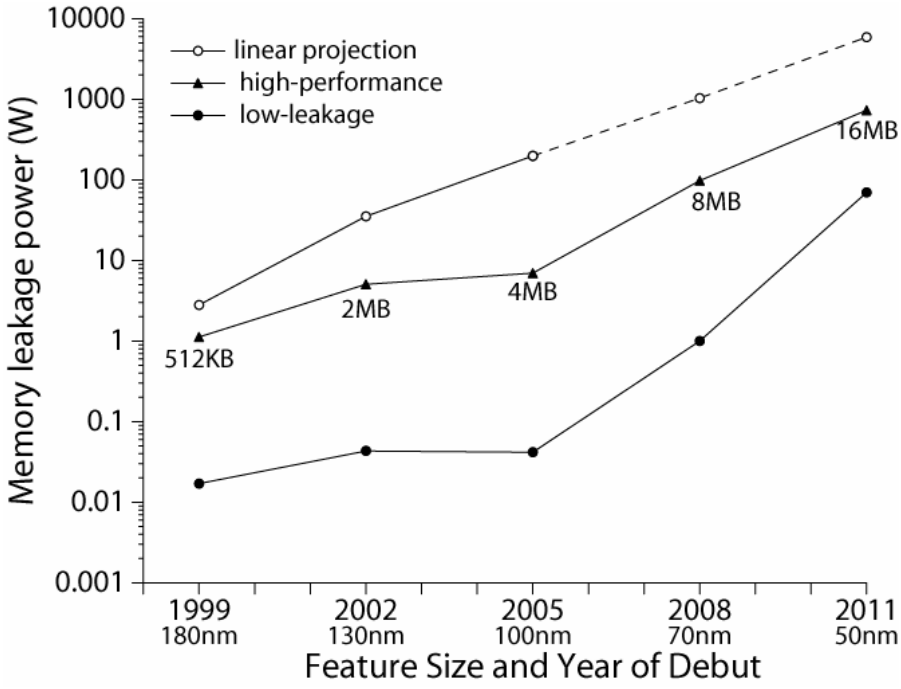


Figure 7: Power Expended in Memory Arrays with Projected Cache Capacities

2.6 SUMMARY

Static current is projected to become a substantial fraction of the total power dissipated by microprocessors due to an increase in leakage current through CMOS transistors. As supply voltages and threshold voltage scale to smaller values each successive generation of fabrication technology, leakage current increases exponentially. Leakage current is also exponentially dependent upon operating temperature, leading to a self-heating effect in which heat from power dissipation results in increased leakage current. We use analytical models and linear projections as estimates of leakage current values in future technologies to predict the static power demands of on-chip caches, which contribute a large part of a microprocessor's static energy consumption. The next chapter introduces circuit and architectural techniques to reduce static power and energy dissipation in of SRAM memory structures.

Chapter 3 Static Energy Reduction

Several research and industrial groups have introduced circuit and architectural techniques to curb static power and energy. This chapter presents three leakage-reduction techniques and our experimental methodology for applying these techniques to on-chip microprocessor caches.

3.1 DUAL-V_T

One solution for lowering leakage current, named dual-V_T, uses a mix of transistors tailored to the circuit's function—high-performance transistors on the critical path, and low-leakage transistors in areas that have more slack for delay [9][3]. With this method, leakage current is engineered at design time, rather than controlled dynamically during operation. The technique may be implemented in a cache by instantiating low-leakage transistors in the memory array and fast, leaky transistors in other areas of the circuit. The memory array contains the majority of transistors in a cache, providing a substantial reduction in leakage current when memory cells contain low-leakage devices.

3.2 GATED-V_{DD}

Another circuit technique to reduce leakage current adds a low-leakage transistor between a circuit and the power or ground connection (or both) [4]. This technique is named gated-V_{DD} to describe the additional transistor that acts as a gate opening and closing a connection to the power supplies. Figure 8 shows a schematic diagram of the gated-V_{DD} technique applied to an SRAM memory cell; in this example, an NMOS gating transistor is placed between the memory cell and ground. All memory cells in a cache line may share a gating transistor to reduce control complexity and amortize the extra transistor area required by this technique.

As shown in the diagram, the `active` signal controls the leakage mode of the circuit. While the `active` signal is asserted, the subcircuit is connected to power supplies and functions as a standard memory cell. To place the circuit in idle mode, the `active` signal is deasserted, turning the low-leakage transistor off and interrupting the current path through the circuit. In addition to creating a bottleneck for leakage current, the extra transistor increases the effective threshold voltage for the other NMOS transistors in the cell due to the "body effect" of transistors connected in series.

Leakage current is reduced when the gating transistor disconnects subcircuits from power supplies, reducing the static power dissipated by the circuit and reducing energy consumption throughout the duration the circuit is in the low-leakage idle mode. When the gated-V_{DD} technique is applied to memory structures, clamping the leakage current by disconnecting memory cells from the power supplies causes the memory cell to lose its stored contents. Read or write accesses to an idle memory cell result in cache misses, which leads to dynamic energy expenditure to refill data from another level of memory hierarchy.

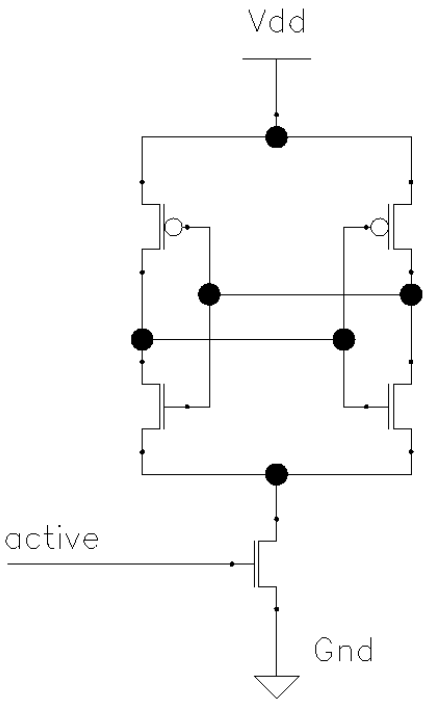


Figure 8 Gated-V_{DD} Schematic

To avoid performance degradation from extra misses and additional energy to re-fetch data, one solution developed concurrently by Kaxiras, et al. [14] and our research group is to disable cache lines only when there is a high probability that the contents will not be needed again. Kaxiras, et al. name the window of time in which data contained in a cache line is useful as the *decay interval*. The interval length is determined by cache access patterns; after a decay-interval length of time since the last access, data is not likely to be useful. In the gated-V_{DD} scheme, counters control the gating transistor, which disables each cache line after a decay interval has elapsed since its last access.

3.3 MTCMOS

Another technique is a dynamic multi-threshold CMOS circuit called MTCMOS. In this technique, the supply voltage and ground voltage levels are altered to bias transistors such that their effective threshold voltages are higher, reducing leakage current [5]. The technique may be applied to combination logic or memory structures; when used in an SRAM, the technique allows reduced leakage current while preserving memory state. Figure 9 shows a schematic of an MTCMOS memory cell. The transistors' source and substrate terminals are connected to separate electrical nets. When `sleep` is asserted, the power supply connected to the substrate in the PMOS transistors is forced higher than the standard levels by a pair of diodes. The larger substrate voltage levels lead to a voltage potential difference between source and substrate terminals of the transistor, which raises the effective threshold voltage and reduces leakage current. Similarly, a voltage difference between the source and substrate of the NMOS transistors is applied by separating the ground potentials. When an MTCMOS memory cell is in its normal

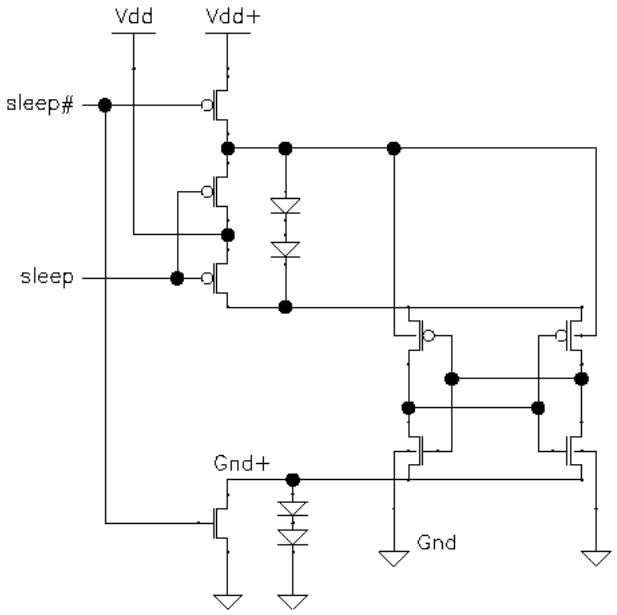


Figure 9 MTCMOS schematic

mode, the `sleep` signal is deasserted and there is no voltage difference between power supplies or between ground nodes [5]. The MTCMOS memory cell is an SRAM cell that behaves as a standard leaky memory cell while the cache line is active and a low-leakage cell with a longer access time when the cache line is asleep. Like gated-V_{DD}, MTCMOS circuits require control circuitry to determine when to disable cache lines, and reduce static energy consumption by lowering leakage current while the circuit is in an "idle" mode.

3.4 EXPERIMENTS

In this study, we use energy—power dissipated through time—to encapsulate the effects of leakage-reduction techniques throughout a program's execution. We calculated the total energy consumed throughout execution of each benchmark by tabulating the static and dynamic energy expenditure for accesses to each cache and summing the energy consumed by individual components of the system. Leakage currents and energy values were measured using the HSPICE circuit simulator. The clock rate was calculated using a 16 fanout-of-four inverter delay metric [15] for a 70nm technology to illustrate the effects of leakage current at a technology several generations beyond current production technology. The remainder of this chapter describes leakage currents and energy parameters for our experiments.

3.4.1 Experiment Parameters

Table 2 summarizes the experimental parameters used in this study for calculating static and dynamic energy. In this table, I_{max} is the projected leakage current when the SRAM cell is active and I_{min} is the leakage energy when the cell is disabled. In each experiment, $V_T = 0.4V$

Table 2: Experimental Parameters for Energy Calculations

70 nm Technology			**Per-Bit Leakage Current**		**Energy per transition**	**Dynamic Energy Per Cache Access**			
Technique	**Clock Rate (GHz)**	**V_{DD} (Volts)**	**Imax (nA)**	**Imin (nA)**	**Eswitch (fJ)**	**E_{IL1} (nJ)**	**E_{DL1} (nJ)**	**E_{L2} (nJ)**	**Epins (nJ)**
Baseline	2.5	0.75	1941	-	-	0.07	0.07	4.5	0.9
Dual-V_T	2.5	0.75	-	26	-				
Gated-V_{DD}	2.5	0.75	1939	9.7	0.35				
MTCMOS	2.5	0.75	1941	12	50				

for high-threshold voltage transistors and 0.2V for low-threshold voltage transistors. E_{switch} approximates the energy required to switch the cell between the active and inactive modes. To measure the dynamic energy expended in cache hits and misses, we modified the cache tool CACTI 2.0 [13] to use our projected parameters for a 70nm process technology. The E_{IL1}, E_{DL1}, and E_{L2} parameters are the read-access energies for the 32KB 2-way set-associative primary caches and a 2MB 4-way set associative secondary data cache. The energy to drive package pins for off-chip memory accesses to service L2 misses is represented by E_{pins}.

The total dynamic energy is calculated as the number of cache accesses multiplied by the appropriate energy per access parameter, plus the number of transitions into idle mode multiplied by the energy per transition (where applicable). To compute the dynamic energy expended in cache accesses, we make the following approximations:

1) level-1 cache miss energy is equal to two cache hit accesses (one for the initial miss plus another for loading data)
2) level-2 cache miss energy is equal to two cache hit accesses plus the energy to drive 32 address pins for off-chip memory accesses

3) any power consumed outside the CPU chip is not included in this study.

The approximation that one cache miss is equivalent to two cache hits presumes a cache circuit in which tags and data are accessed in parallel to provide a fast time. If tags were accessed first, followed by data access if the requested cache line were resident in the cache, the dynamic energy cost of a miss would be lower. However, each cache access would be slower, reducing system performance.

We estimate the energy to drive the I/O pins to fetch data from off-chip memory with a simple model based on the following equation:

Equation 6: pin energy

$$E_{pin} = 1.3\ C_{pin}\ V_{pin}^2 \quad [16].$$

We set $\mathbf{C_{pin}}$ = 10pF, according to the multi-chip module estimates in [16] and use an I/O pin supply voltage of $\mathbf{V_{pin}}$= 1.5V [17]. With 32 address pins switching, the energy cost is 0.9nJ per off-chip access. We account only for the pin energy that is expended in driving the address to the pins of the CPU, and not energy expended to receive data.

Static energy is computed as the static power per cycle multiplied by the number of cycles of program execution; static power is the leakage current per bit multiplied by the number bits, then multiplied by the supply voltage. In our calculations, we apply the approximation that all transistors are in the memory array; this approximation neglects the leakage current due to the small fraction of transistors in the peripheral circuitry.

3.4.2 Baseline

We compare the energy consumption and performance of the leakage-reduction techniques to a baseline case to evaluate the experimental techniques' effectiveness in lowering static energy and impact on performance. The baseline in this study is a high-performance cache without leakage current control. Each transistor in the SRAM cell has a low threshold voltage of 0.2V for faster switching time, and has a high leakage current, I_{max}, at all times. The baseline case has the maximum performance and maximum energy consumption throughout program execution.

3.4.3 Dual-V_T

A dual-V_T cache has low-leakage transistors in the memory array and high-leakage transistors in other components. In this study, we account for static energy only in the memory array, and thus only list the reduced-leakage current, I_{min}, in the table of parameters. The dual-V_T technique does not transition between idle and active states and does not incur extra misses.

3.4.4 Gated-V_{DD}

In the gated-V_{DD} technique, I_{max} is the leakage current when the memory cell is in the active state, and I_{min} is the leakage current when the memory cell is disconnected from the power supplies. The gating transistor has a high threshold voltage of 0.4V, and the other SRAM cell transistors' threshold voltages are the low-V_T value of 0.2V. The value of E_{switch} is based on the gate capacitance of the activation transistor and the wire capacitance to reach all cells in a cache line.

3.4.5 MTCMOS

Table 2 summarizes the parameters of an MTCMOS SRAM array that controls leakage current on the granularity of a cache line. The time and energy to enter and exit sleep mode depend directly on the effective capacitance of the well that contains the PMOS transistors in the

SRAM cell. By assuming that the time to switch a cache line into or out of sleep mode is a single cycle, we account for well capacitance up to 30 times that of the combined source and drain capacitances of the transistors in a well. The MTCMOS parameters depend upon the circuit and fabrication mask design; implementing this technique on silicon could require partitioning the wiring and the number of transistors per well to maintain a one-cycle wakeup time. E_{switch} is the energy required to charge the block's well plus the energy consumed to discharge the source terminals of the NMOS transistors.

3.5 SUMMARY

Several techniques have been proposed to curb static power in microprocessors by reducing leakage current in large on-chip caches. Two techniques, gated-V_{DD} and MTCMOS, use decay intervals to selectively disable cache lines after they are no longer hold useful information. The dual-V_T SRAM is designed to have low-leakage transistors in the memory array and fast, high-leakage transistors elsewhere in the circuit. Gated-V_{DD} adds an extra high-V_T transistor that throttles leakage current, and the MTCMOS technique dynamically raises the threshold voltage of all memory cell transistors. We examined the energy and performance characteristics of these three techniques by simulating benchmarks in an architectural simulator. After gathering data from simulations, we estimated the total energy required by the program as the sum of static and dynamic energy components.

Chapter 4 Experimental Simulations

In this section, we present our experimental methodology for implementing leakage reduction techniques in an architectural simulator and compare tradeoffs of performance and energy reduction for each of the three leakage-reduction techniques: dual-V_T, gated-V_{DD}, and MTCMOS. We calculate energy consumption and measure performance in terms of instructions per cycle by simulating execution of a benchmark suite for each technique. We use the energy-delay product metric to balance the benefits of lower leakage with the penalty of reduced performance.

4.1 EXPERIMENTAL METHODOLOGY

To evaluate the effectiveness of the dual-V_T, gated-V_{DD}, and MTCMOS leakage-reduction techniques, we modified a version of the SimpleScalar simulator [18]. We added the capability to discard cache lines or put them to sleep after a specified decay interval had passed since the last access to the cache line. We chose decay intervals of 1K, 8K, and 64K clock cycles to capture approximately 95%, 99%, and more than 99% of cache line accesses for our benchmark suite. The benchmark suite for this study consists of five SPEC2000 benchmarks: `mcf`, `vpr`, `eon`, `equake`, and `gcc`, compiled for the Alpha instruction set.

The execution core is configured as a 4-wide superscalar pipeline organization roughly comparable to the Compaq Alpha 21264 [19]. The memory hierarchy consists of a 32KB 2-way set associative level-1 instruction cache with a single cycle hit latency, a 32KB 2-way set associative level-1 data cache with a 3-cycle hit latency, and a unified 2MB level-2 cache with a 12-cycle hit latency. When, data bits transition into an idle mode in the gated-V_{DD} and MTCMOS techniques, cache tags are kept in the active state to provide fast lookup times. For gated-V_{DD}, only "clean" lines that do not require a write back to the memory hierarchy are disabled; "dirty" lines are kept in the active state.

In each experiment, we applied a leakage reduction technique to one cache and simulated benchmark execution with our modified SimpleScalar simulator. Simulations executed 1 billion instructions after fast-forwarding through the first 500 million instructions. During simulations, we measured several attributes of program execution: instructions per clock cycle (IPC), active and inactive durations for each cache line, the number of hits and misses in each level of the hierarchy, and the number of times any cache line is enabled or disabled.

4.2 CACHE ACCESS LATENCY

If a cache circuit design could compensate for slower SRAM cells in the memory array and achieve cache accesses in the same number of clock cycles as an array with high-performance transistors, the dual-V_T technique would have reduced static power and without performance penalty. However, with aggressive clock speeds in future technologies, a few picoseconds of additional delay due to slow SRAM transistors could mean a dual-V_T cache access requires more cycles than a cache with purely high-performance transistors. Techniques

such as dual-V_T that add latency to each cache access can affect the microprocessor's performance and lengthen program execution time, expending more leakage energy per program and reducing the techniques' effectiveness. To evaluate the effectiveness of the dual-V_T technique, we investigate the effects of additional latency on the processor's performance and static energy consumption.

4.2.1 Performance

Figure 10 through Figure 12 show the measured IPC for a range of cache-hit latencies across the benchmark suite. In the level-1 instruction cache, the IPC harmonic mean drops from 1.65 to 0.41, a 74% reduction in performance from zero to two additional cycles of hit latency. The processor is less sensitive to additional delays in the level-1 data cache. IPC values dip from a mean of 1.64 to 1.50, an average performance reduction of 4% when the DL1 cache latency increases by two cycles. Additional latency in the level-2 cache causes the least impact on performance, with an average of 2% decrease in IPC for two extra cycles of latency.

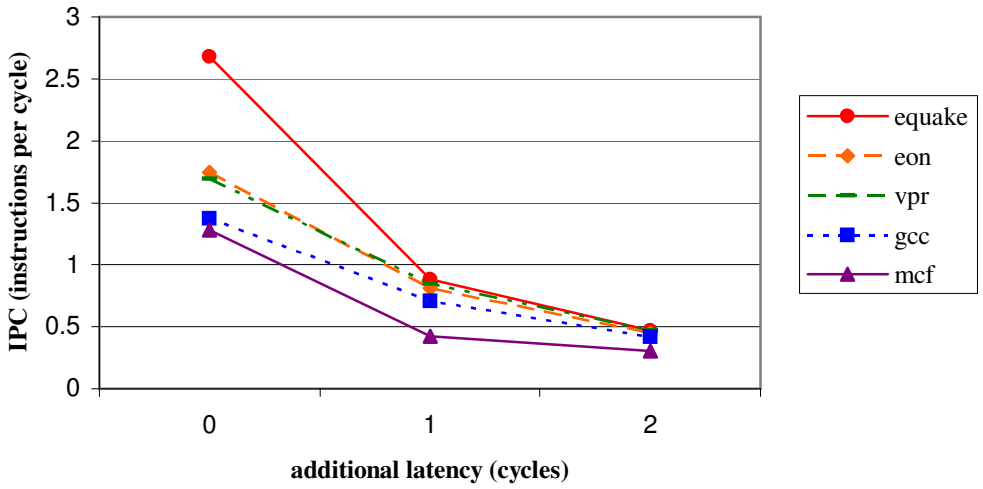


Figure 10 Processor Performance with Dual- V_T Level-1 Instruction Cache

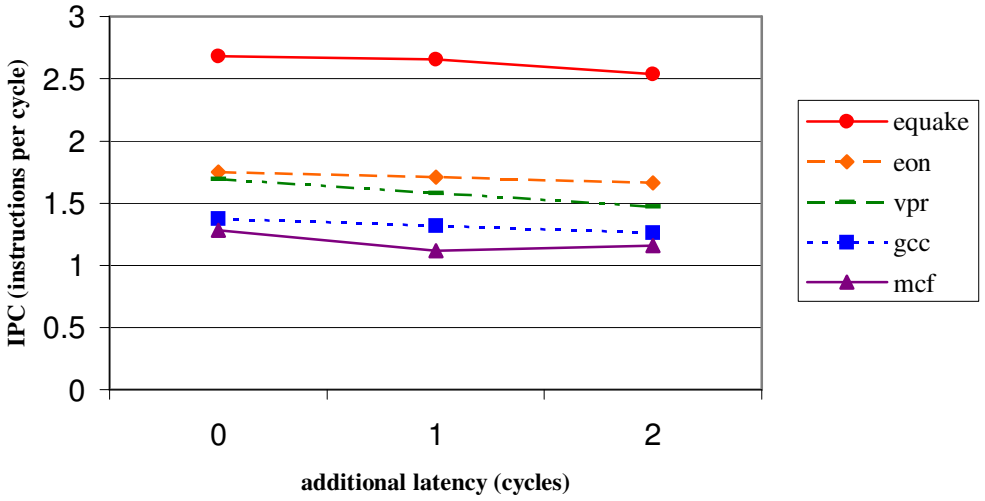


Figure 11 Processor Performance with Dual- V_T Level-1 Data Cache

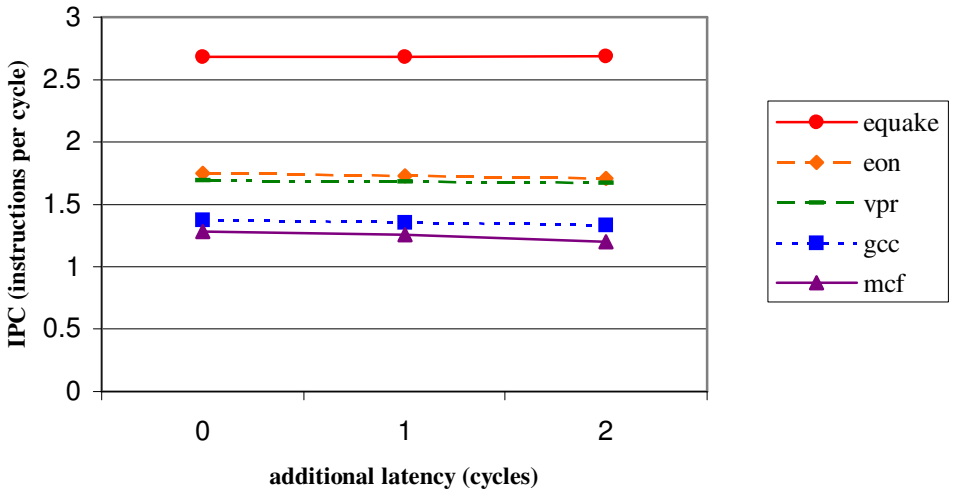


Figure 12: Processor Performance with Dual-V_T Level-2 Cache

4.2.2 Energy

Additional latency per cache access can extend program execution time, increasing the static energy expended. Figure 13 through Figure 15 relate longer cache access times to increased static energy. In the level-1 instruction cache, static energy increases by 157% for one additional cycle and 387% for two additional cycles of IL1 cache latency. In the level-1 data cache, the static energy reductions for one and two additional cycles of latency are 5% and 9%, respectively. The unified level-2 cache shows a 1% increase in static energy for each additional cycle of latency.

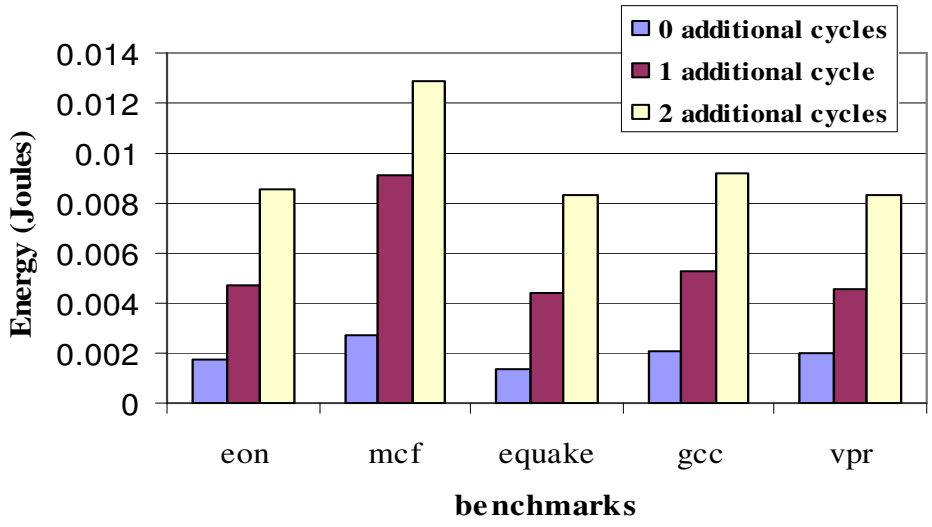


Figure 13: Static Energy with Dual-V_T Level-1 Instruction Cache

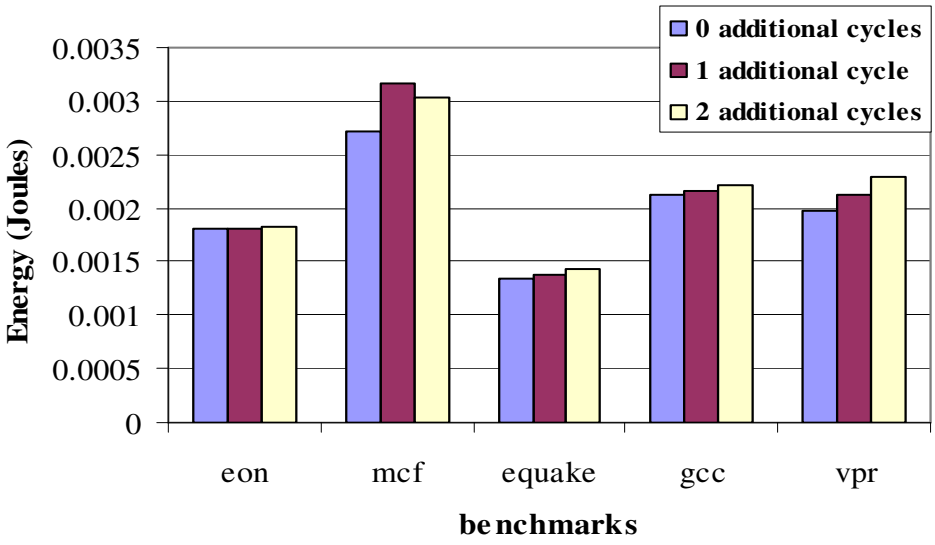


Figure 14: Static Energy with Dual-V_T Level-1 Data Cache

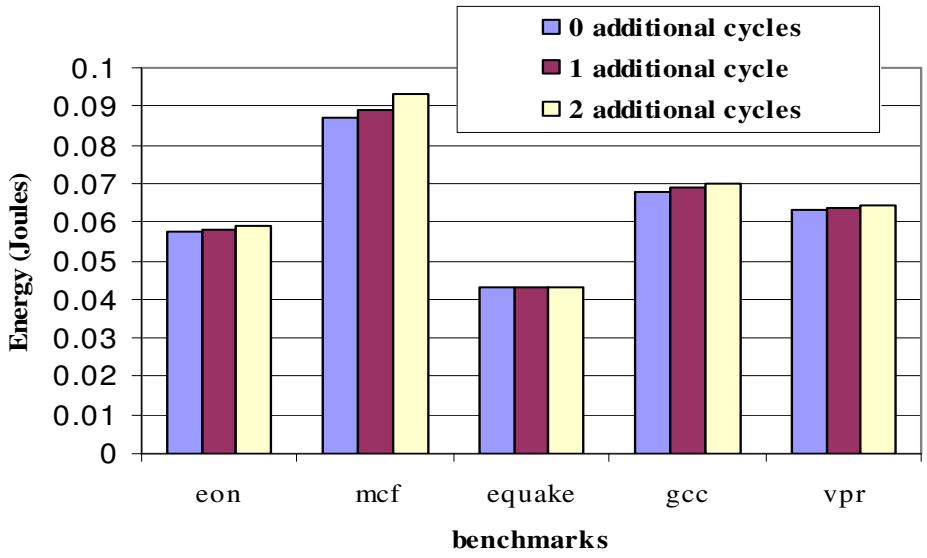
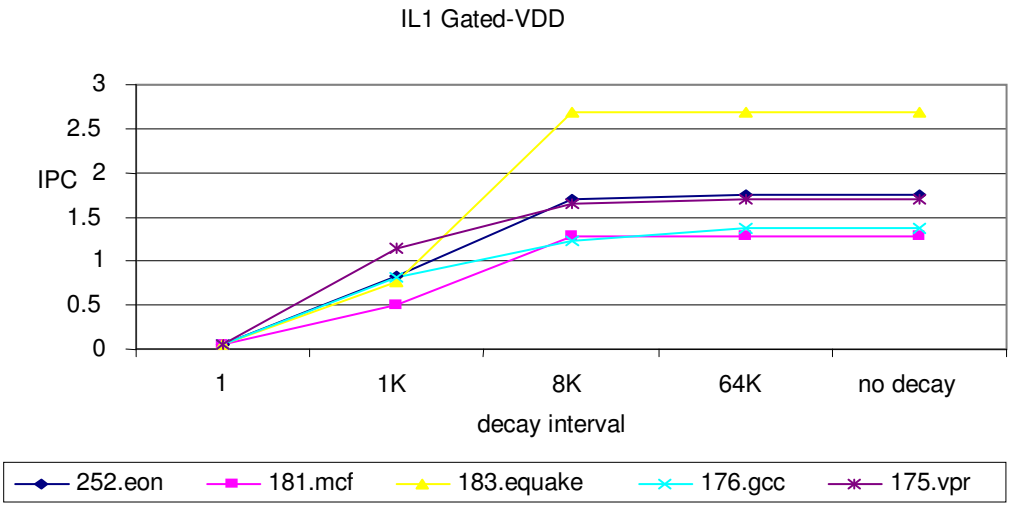


Figure 15: Static Energy with Dual-V_T Level-2 Cache

4.3 DECAY INTERVALS

The energy savings and impact on performance of gated-V_{DD} and MTCMOS techniques depend upon the decay interval used to disable cache lines. In this experiment, the decay interval is based on profile information, and does not change during program execution. Figures 16-18 and 20-22 show measurements for IPC, static energy, and dynamic energy over a range of decay intervals used with the gated-V_{DD} and MTCMOS techniques: 1K, 8K, and 64K. For comparison, we also test these techniques with an immediate-disable policy and an infinite decay interval (no disable).



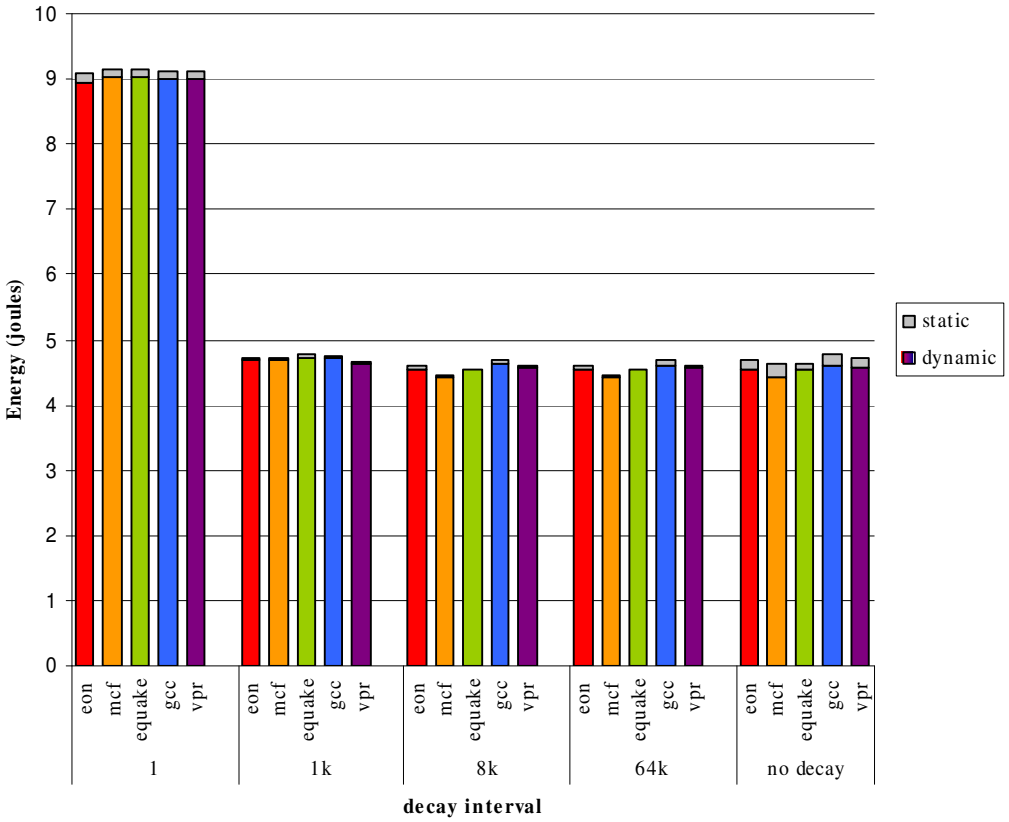
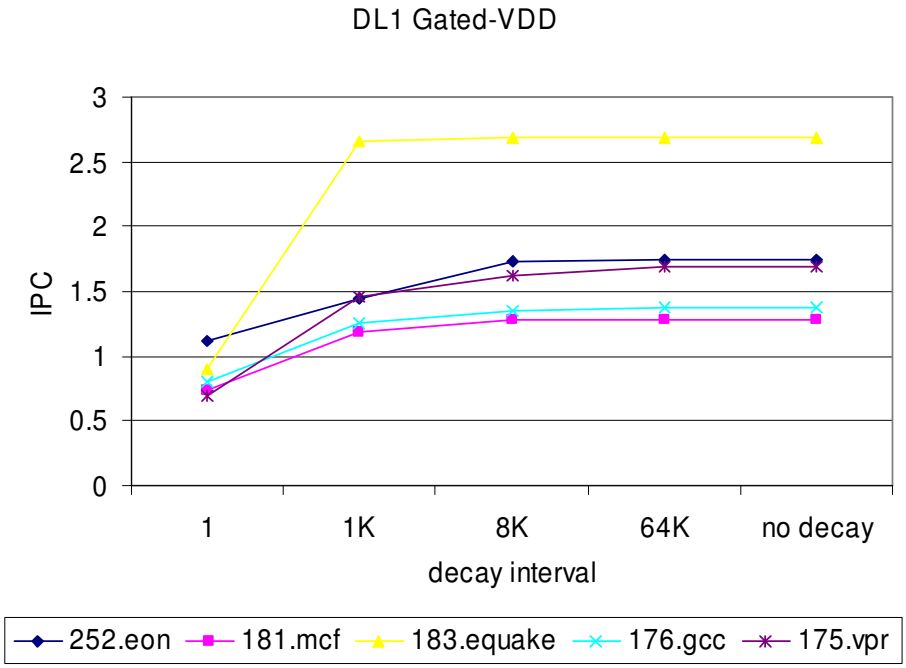


Figure 16: Gated-VDD Level-1 Instruction Cache IPC and Energy



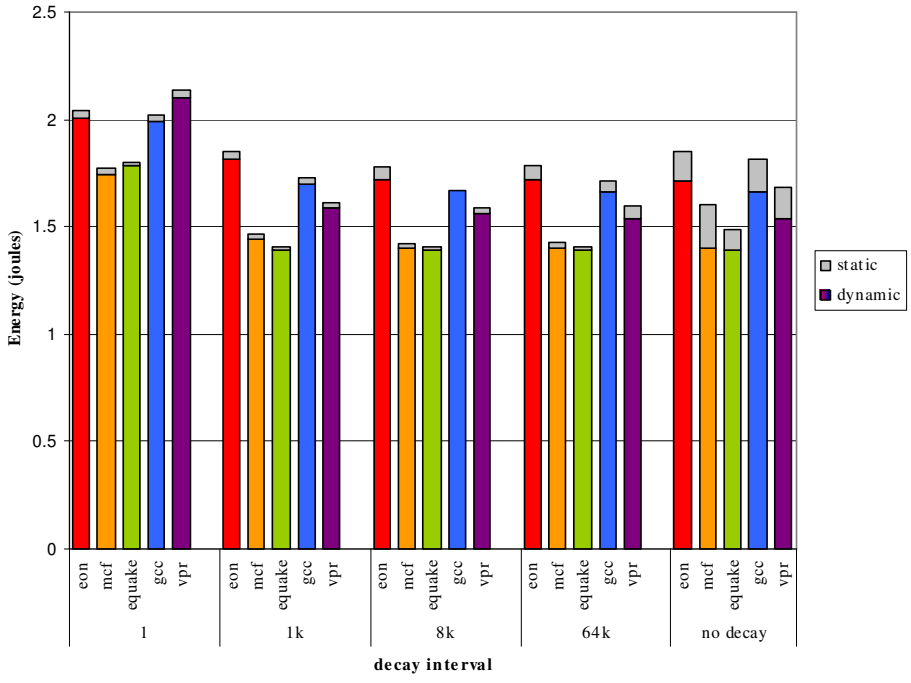
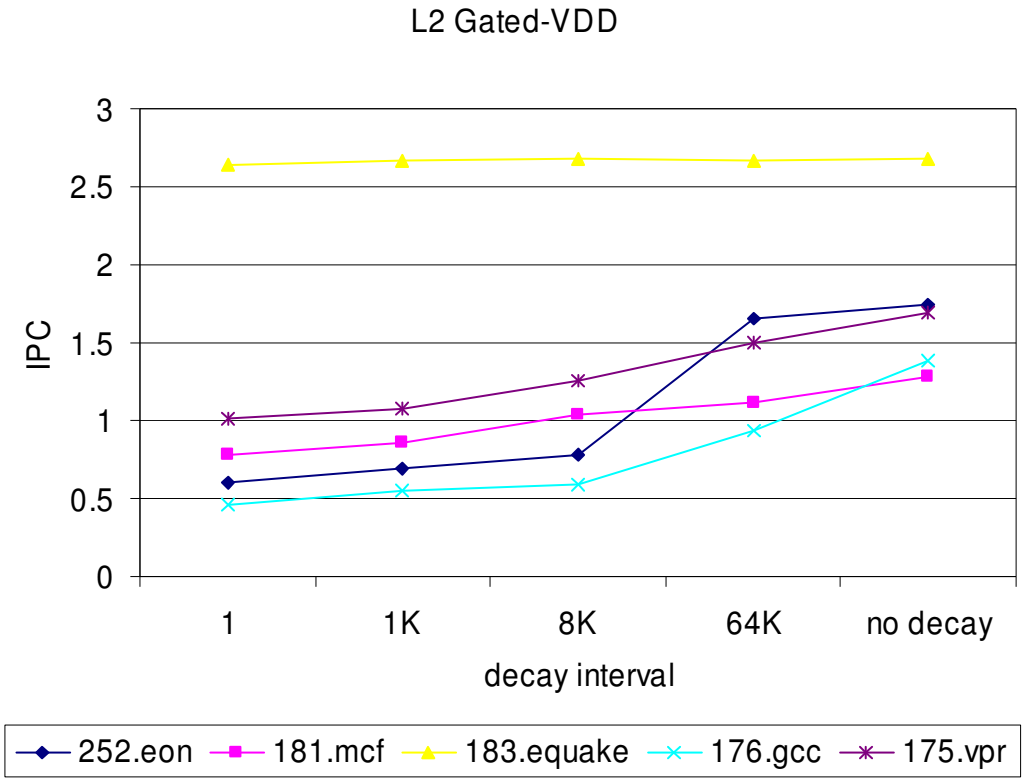


Figure 17: Gated-V_{DD} Level-1 Data Cache IPC and Energy



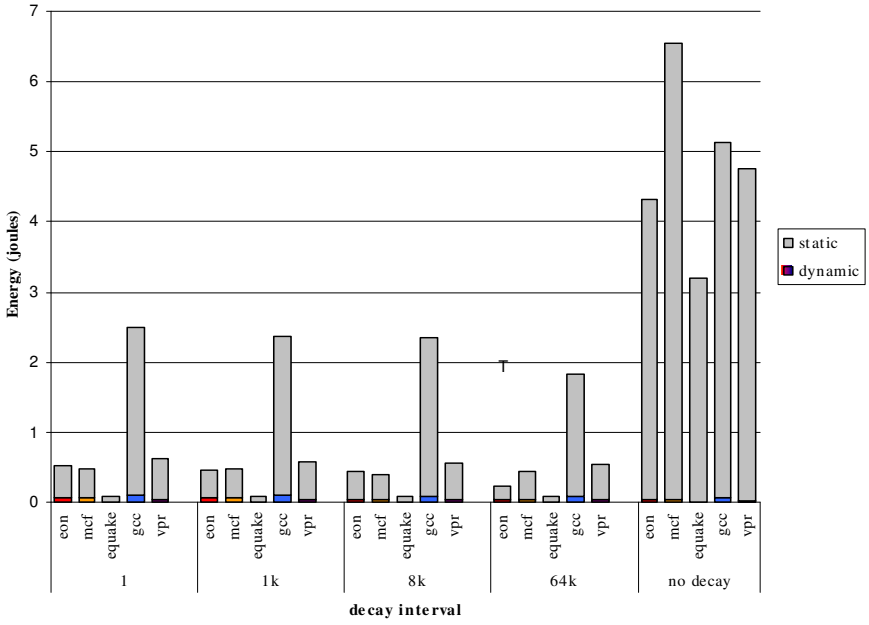


Figure 18: Gated-V_{DD} Level-2 Cache Measurements

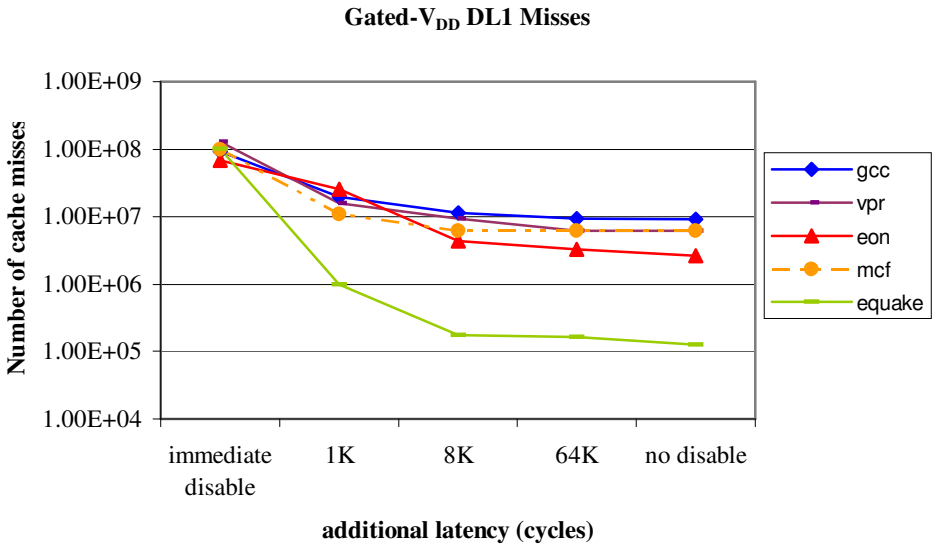


Figure 19: Gated-V_{DD} Level-1 Data Cache Misses

For the gated-V_{DD} technique, IPC increases with increasing decay intervals up to 64K cycles, for all cache structures. Figure 19 shows an example of how the decay interval size affects the number of misses in a gated-V_{DD} DL1 cache; when the decay interval is too small for the cache access pattern of a program, the number of misses is higher due to attempts to access cache lines which have been invalidated. As the decay interval increases to accommodate most useful accesses, the number of cache misses approaches the number of misses that would occur without the gated-V_{DD} technique. The gated-V_{DD} technique has a high dynamic energy cost associated with accessing inactive blocks due to re-fetching data, which is reflected by the increase in energy consumption at small decay intervals. The level-1 instruction cache for gated-V_{DD} with a 1 cycle decay interval, which disables cache lines immediately after use, uses an average of 93% more total energy, almost twice the amount energy required for the 1K cycle decay interval. For the gated-V_{DD} technique, total energy decreases with increased decay intervals. The optimal decay interval for each gated-V_{DD} cache is 64K cycles, which keeps data valid in the cache and reduces the number of re-fetches to other memory hierarchy levels.

The MTCMOS technique has a smaller energy penalty for turning cache lines off earlier. Instead of re-fetching data after an access to an idle cache line, the MTCMOS circuit transitions from sleep state to awake. The performance penalty is the wakeup time for the cache—shorter than re-fetching data from another level of memory hierarchy—and the energy penalty is increased static energy from longer execution time. In the MTCMOS experiments, IPC increases with increasing decay interval size up to the 8K-decay interval for level-1 caches, where it reaches a plateau. When MTCMOS techniques are applied to level-2 caches, the IPC is essentially constant, independent of the decay intervals. The level-2 cache is accessed infrequently, and has a baseline hit latency of 12 cycles; an additional one-cycle delay to wake up sleeping cache lines for the occasional level-2 access does not noticeably degrade performance. Figures 20 through 22 show MTCMOS performance and energy measurements.

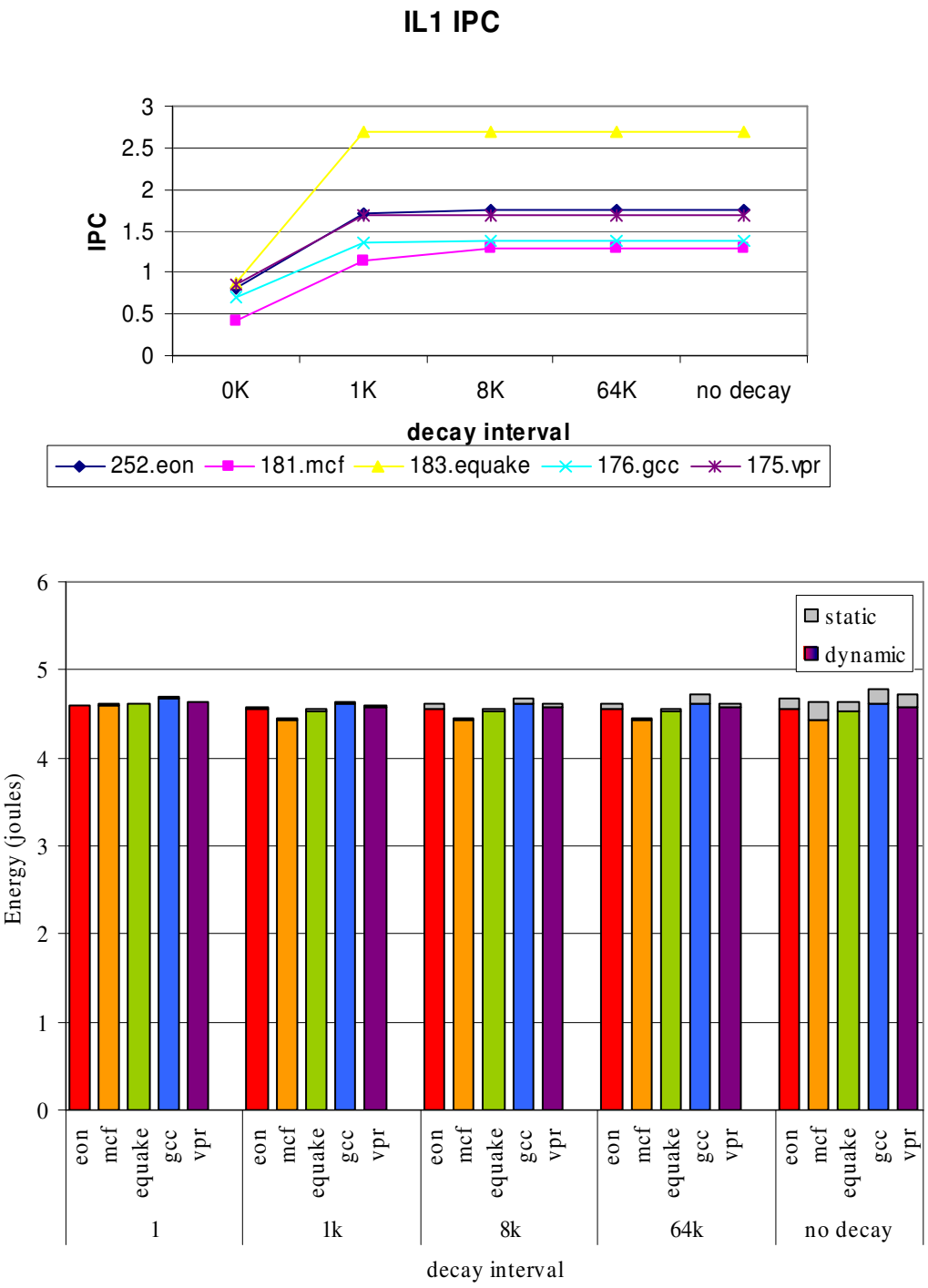
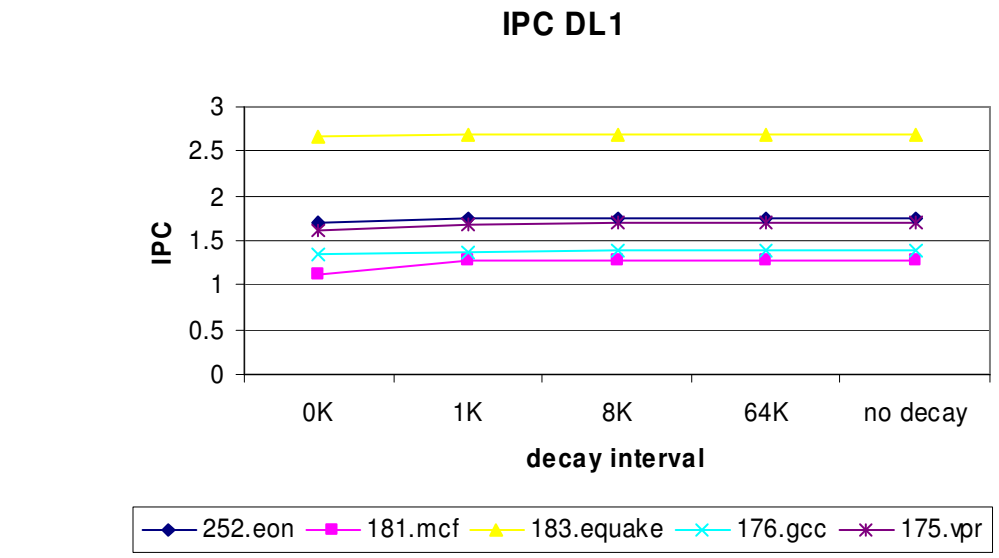


Figure 20: MTCMOS Level-1 Instruction Cache Measurements



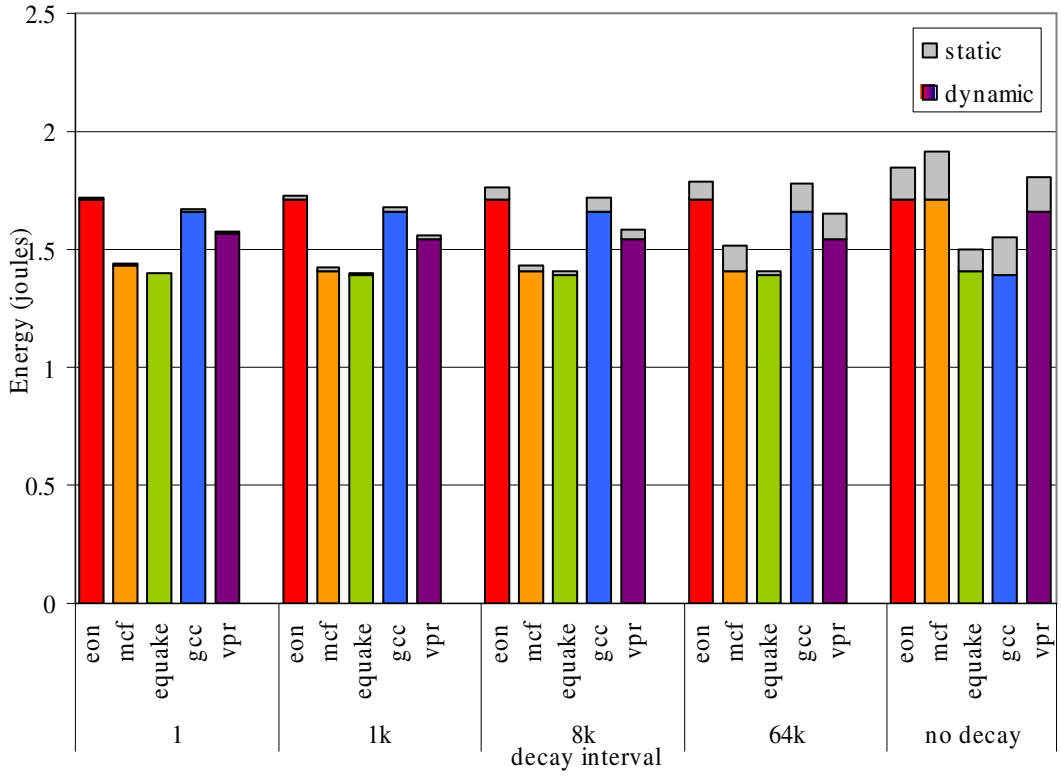
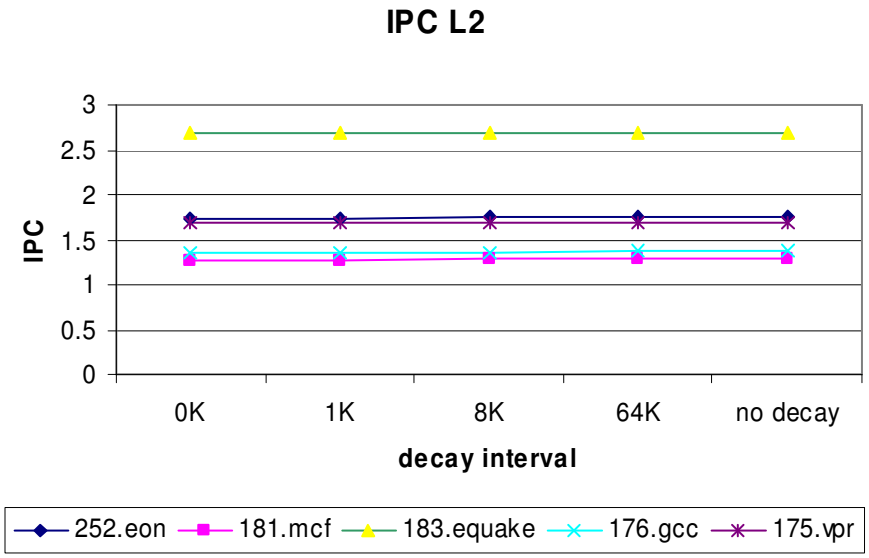


Figure 21: MTCMOS Level-1 Data Cache Measurements



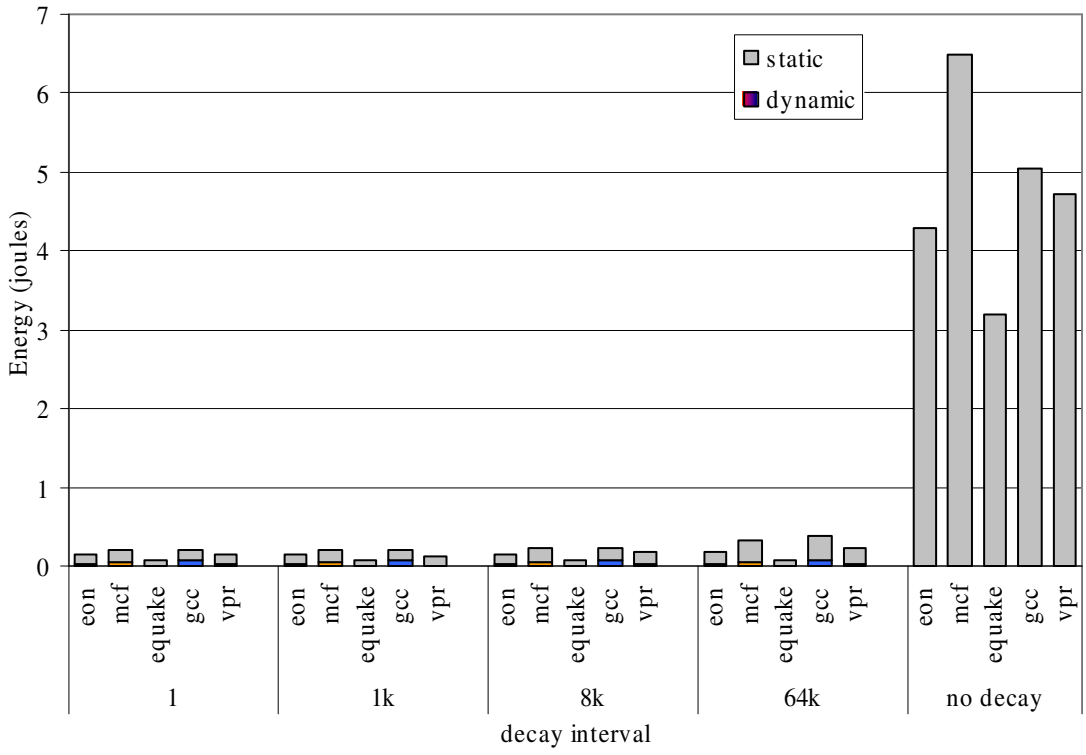


Figure 22: MTCMOS Level-2 Cache Measurements

Total energy tends to increase with longer decay intervals in the MTCMOS case because more cache lines are awake and leaking. For MTCMOS, the best decay interval is 1K cycles for level-1 caches; this decay interval minimizes both leakage energy and dynamic energy penalties for switching the cache blocks in and out of sleep mode. The level-2 cache has no significant difference between immediate sleep, 1K, and 8K cycles. In this case, the immediate sleep option is preferred since it does not require counters to implement the decay interval control mechanism.

4.3.1 Control Mechanisms

A drawback of any technique that relies on a decay interval, such as the MTCMOS and gated-V_{DD} strategies in this study, is the overhead of implementing the idle-mode control logic. One alternative for reducing the control complexity in the MTCMOS technique is to keep all cache blocks in idle mode by default, waking them up briefly when as they are accessed before putting them back to sleep. While minimizing leakage current, this has the effect of increasing the latency of every cache access by the wakeup penalty. This strategy is attractive for the level-2 cache but has a negative impact on level-1 caches. A second alternative is to place cache lines in a set-associative cache into standby mode based on the existing replacement control logic. For example, in a four-way set associative cache, cache lines in the two least-recently-used (LRU) sets could be placed in standby mode. The leakage energy is only reduced by 50%, but additional hardware complexity is minimal.

4.4 ENERGY/PERFORMANCE COMPARISON

Each technique degrades performance to some extent in return for lower energy consumption. In this section, we compare each technique against the baseline case to examine the energy/performance tradeoffs in detail. The behavior of the gated-VDD and MTCMOS techniques depends upon the decay interval. An optimal decay interval size is a factor of program cache access patterns and circuit parameters unique to each leakage-reduction technique. To evaluate the gated-VDD and MTCMOS techniques, we select a fixed decay interval with the minimum energy-delay product for the benchmark suite, for each cache individually. The experimental results are summarized in Table 3 through Table 5 for simulations of 1 billion cycles, reported as the harmonic mean of the results for simulated program execution across the benchmark suite. The energy-delay product is calculated by the total energy divided by IPC to maximize energy savings and minimize impact on performance.

Table 3: Experimental Results for Level-1 Instruction Cache

Technique	Optimal Decay Interval	IPC	Total Energy (Joules)	Dynamic Energy (Joules)	Leakage Energy (Joules)	Energy-Delay (E/IPC)
Baseline	-	1.645	4.688	4.539	0.141	2.663
Dual-V_T	-	0.680	4.525	4.520	0.005	6.181
Gated-V_{DD}	64K	1.641	4.584	4.539	0.039	2.613
MTCMOS	8K	1.644	4.580	4.539	0.035	2.607

Table 4: Experimental Results for Level-1 Data Cache

Technique	Optimal Decay Interval	IPC	Total Energy (Joules)	Dynamic Energy (Joules)	Leakage Energy (Joules)	Energy-Delay (E/IPC)
Baseline	-	1.645	1.679	1.530	0.141	0.942
Dual-V_T	-	1.540	1.520	1.518	0.002	0.898
Gated-V_{DD}	64K	1.643	1.571	1.531	0.030	0.885
MTCMOS	1K	1.639	1.547	1.530	0.017	0.874

Table 5: Experimental Results for Level-2 Cache

Technique	Optimal Decay Interval	IPC	Total Energy (Joules)	Dynamic Energy (Joules)	Static Energy (Joules)	Energy-Delay (E/IPC)
Baseline	-	1.6451	4.5403	0.004092	4.5133	2.4245
Dual-V_T	-	1.6249	0.0837	0.004094	0.0610	0.0423
Gated-V_{DD}	64K	1.3863	0.2392	0.0048	0.2254	0.1117
MTCMOS	0	1.6259	0.1397	0.0042	0.1149	0.0723

4.4.1 Energy

Figure 23 through Figure 25 show the total energy required for program execution for each leakage-reduction technique. Each of the three techniques in this study reduces leakage energy compared to the baseline case of a standard, high-performance SRAM cell. In most benchmarks, the dual-V_T technique with a single cycle of additional latency requires less energy than either gated-V_{DD} or MTCMOS. With the transistor parameters used in this study and a single additional cycle penalty for both dual-V_T and MTCMOS techniques, the static energy for dual-V_T and MTCMOS techniques is approximately the same for the array of data bits in the level-2 cache. However, the cache tags were kept awake to provide fast lookup times, causing more static energy due to the tag bits' higher leakage current.

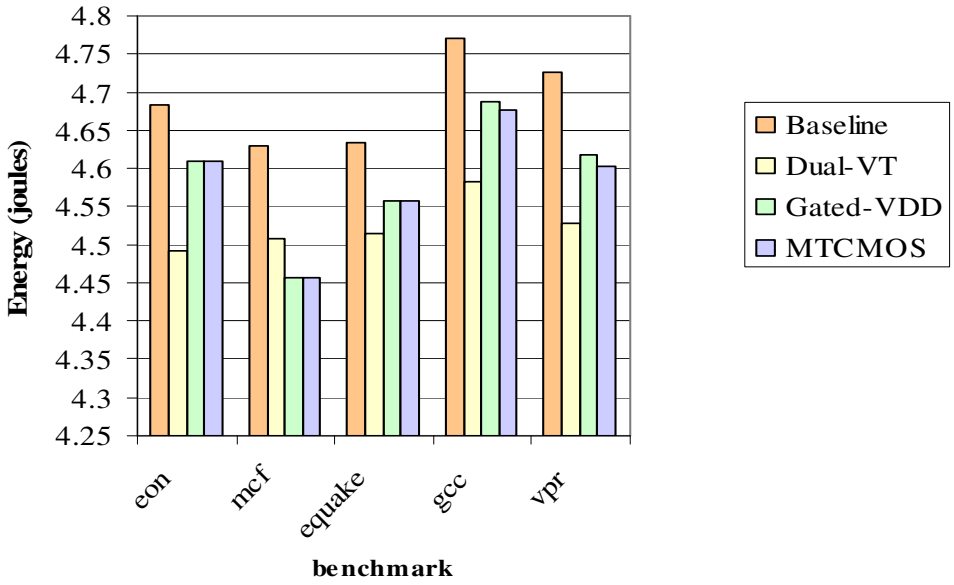


Figure 23: Total Energy with Leakage-Reduction Techniques Applied to IL1

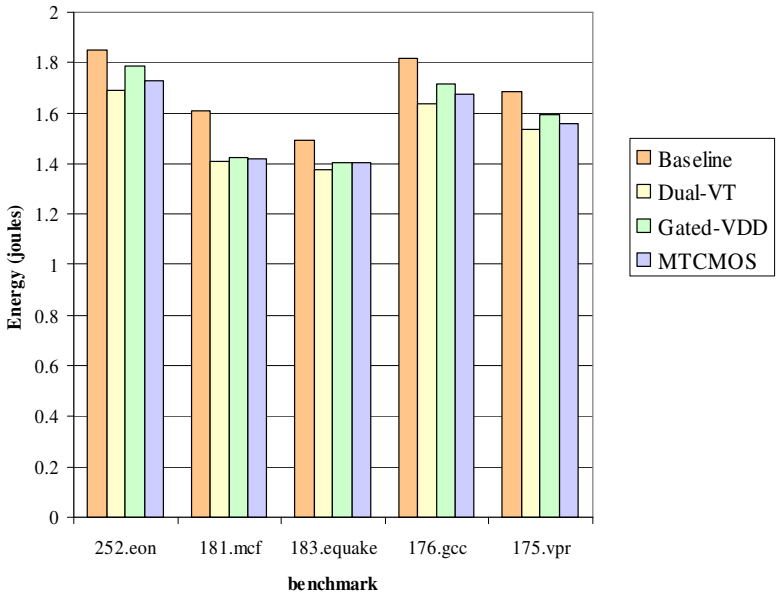


Figure 24: Total Energy with Leakage-Reduction Techniques applied to DL1

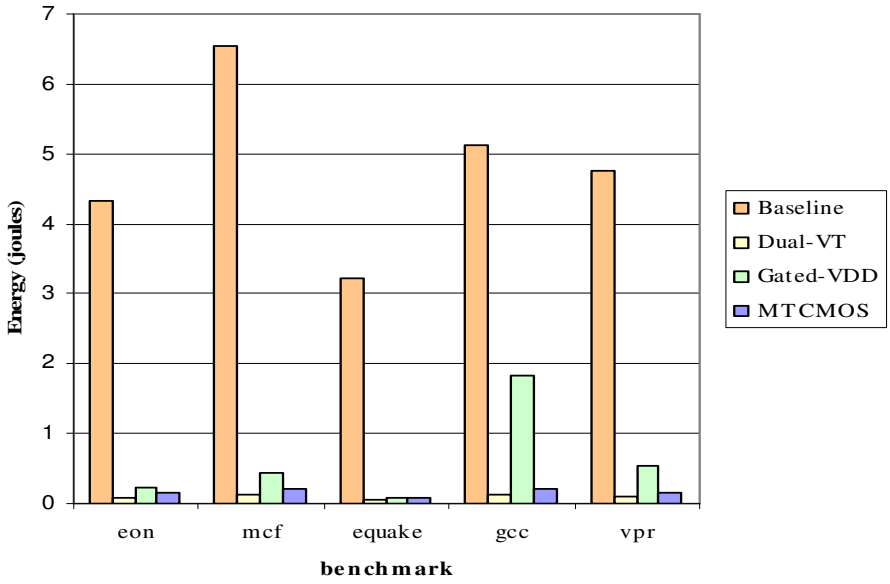


Figure 25: Total Energy with Leakage-Reduction Techniques applied to L2

The gated-V_{DD} memory cells have leakage currents per cell approximately equal to the MTCMOS circuit, but since idle cache lines lose their data, the gated-V_{DD} technique extends program execution time to handle additional cache misses and thus spends more cycles leaking. To achieve an energy reduction comparable to MTCMOS, the gated-V_{DD} technique needs a finely tuned decay interval and idle-mode control mechanism to avoid accesses to idle cache lines. Also, in our implementation of gated-V_{DD}, dirty cache lines are kept alive, reducing the opportunity to save energy.

4.4.2 Energy-Delay

The dual-V_T technique does not require additional dynamic energy to reduce static energy, and thus has an advantage over the other techniques in terms of total energy reduction. However, the price of lower energy is reduced performance. Considering the energy-delay product as $\frac{energy}{IPC}$, the MTCMOS technique provides a better tradeoff of energy and performance for the level-1 caches than the dual-V_T or gated-V_{DD} techniques. Figure 26 though Figure 28 show the energy-delay product for each technique applied to a cache level.

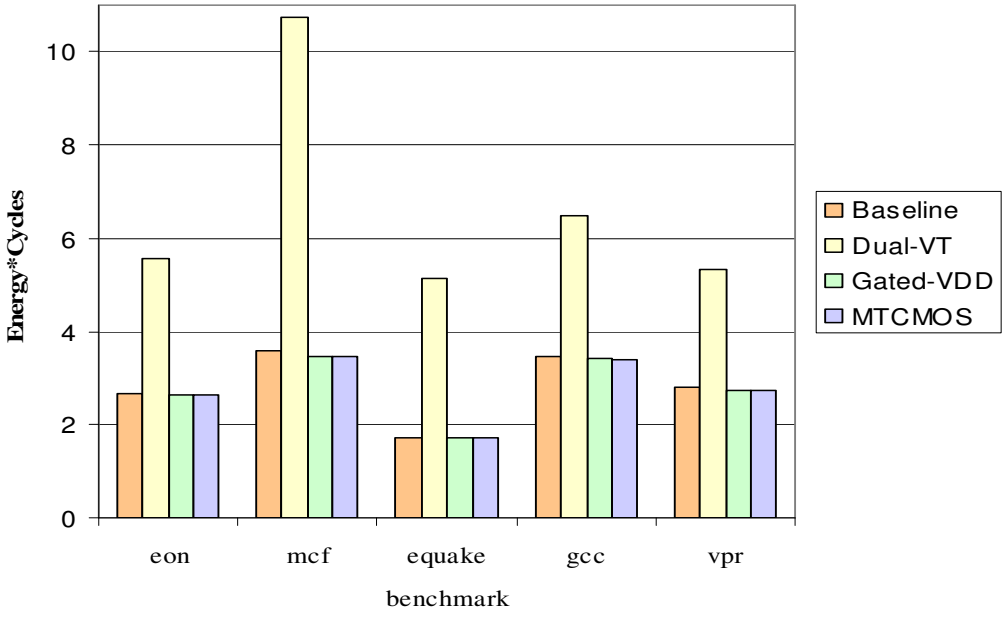


Figure 26 : Energy-Delay Product for Leakage Reduction in Level-1 Instruction Cache

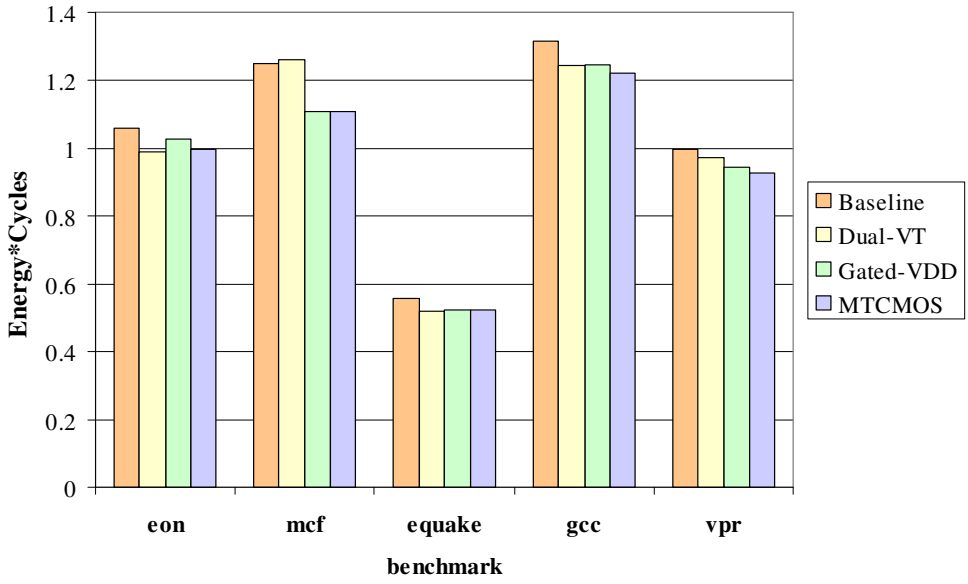


Figure 27: Energy-Delay Product for Leakage Reduction in Level-1 Data Cache

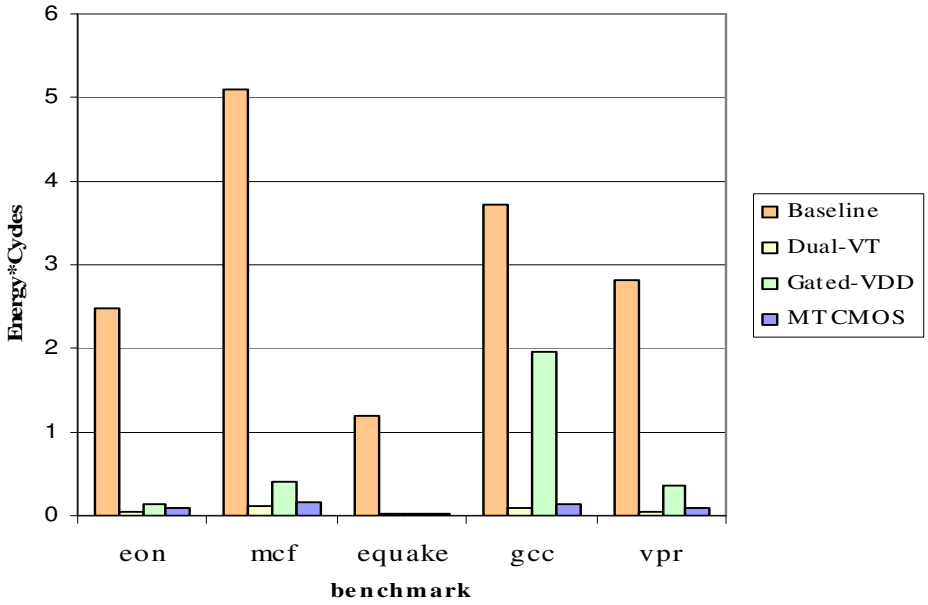


Figure 28: Energy-Delay Product for Leakage Reduction in Level-2 Cache

With a one-cycle wakeup time for sleeping cache lines in the MTCMOS case and a decay interval that captures most accesses, the additional program execution time is negligible in our simulations with an out-of-order processor core, and as a result, this technique provides a smaller energy-delay product than the other methods for primary caches. Additional latency per access in the secondary cache does not have a strong impact on the processors' IPC. The optimal MTCMOS configuration for L2 caches chosen in this study is equivalent in performance to the dual-V_T technique—in both cases lines are in a low-leakage mode, whether sleeping or fixed, with a 1-cycle additional latency. Thus, the two techniques have the same IPC. The dual-V_T technique has a smaller energy value than MTCMOS and gated-V_{DD} techniques, with the effect of the lowest energy-delay product for the secondary cache. The gated-V_{DD} results are similar to the MTCMOS and dual-V_T methods for most benchmarks, with the exception of greater expenditure for execution of `gcc` with the gated-V_{DD} cache.

4.5 WAKEUP LATENCY

Although leakage reduction techniques attempt to reduce static energy consumption, the performance penalties they can impose act in opposition to such savings and can reduce the techniques' effectiveness. In particular, if a program takes more time to complete with leakage reduction techniques enabled, then all remaining leaky components of the chip will leak for a longer period of time. In this section, we investigate the effects of additional latency on processor performance and static energy consumption. In dual-V_T and gated- V_{DD} , delays are manifested in cache access time overhead, while the most interesting variable for MTCMOS is the time to wake a sleeping line.

DUAL-VT : Cache access time for dual-V_T can increase if the speed reduction of the higher threshold devices in the cache is significant. Likewise, the high-V_T cut-off transistor implemented in a gated- V_{DD} strategy could also increase overall cache access time. The increase in access latency can extend the execution time of the program and degrade performance. Graphs in the left column of Figure 29 show the performance degradation for processors accessing dual- V_T caches as the access latency is increased by one and two cycles. The IPC values are calculated as the harmonic mean of measured IPC results from all five benchmarks. Figure 29a shows the IPC for the level-1 instruction cache drops from 1.65 to 0.41, a substantial 74% reduction in performance as the latency increases by 2 cycles. The processor is less sensitive to additional delays in the level-1 data cache, as illustrated in Figure 29b. The mean IPC values dip from 1.64 to 1.50, an average performance reduction of 4% when the DL1 cache latency increases by two cycles. Figure 29c shows that additional latency in the level-2 cache causes the least impact on performance, with an average of 2% decrease in IPC for two extra cycles of latency. The right column of Figure 29 indicates how longer access times translate into increased static energy for individual program execution. In addition, the harmonic mean over the full benchmark suite is reported in this discussion on sensitivity trends. In the level-1 instruction cache, the mean static energy increases by 157% for one additional cycle and 387% for two additional cycles of IL1 cache latency. Figure 29d shows how each extra cycle of latency adds to static energy consumption for each program in the benchmark suite. The short bars in Figure 29e indicate that static energy of the level-1 data cache is not as strongly affected by additional access latency. In the DL1, the static energy increases for one and two additional cycles of latency are 5% and 9%, respectively. The unified level-2 cache shows an overall 1% increase in static energy for each additional cycle of latency. Figure 29d illustrates that the static energy consumption depends upon program behavior; the increase is more pronounced in the benchmarks mcf and gcc than in equake. For reference, tables 3 through 5 report performance and energy data from simulated dual-V_T caches.

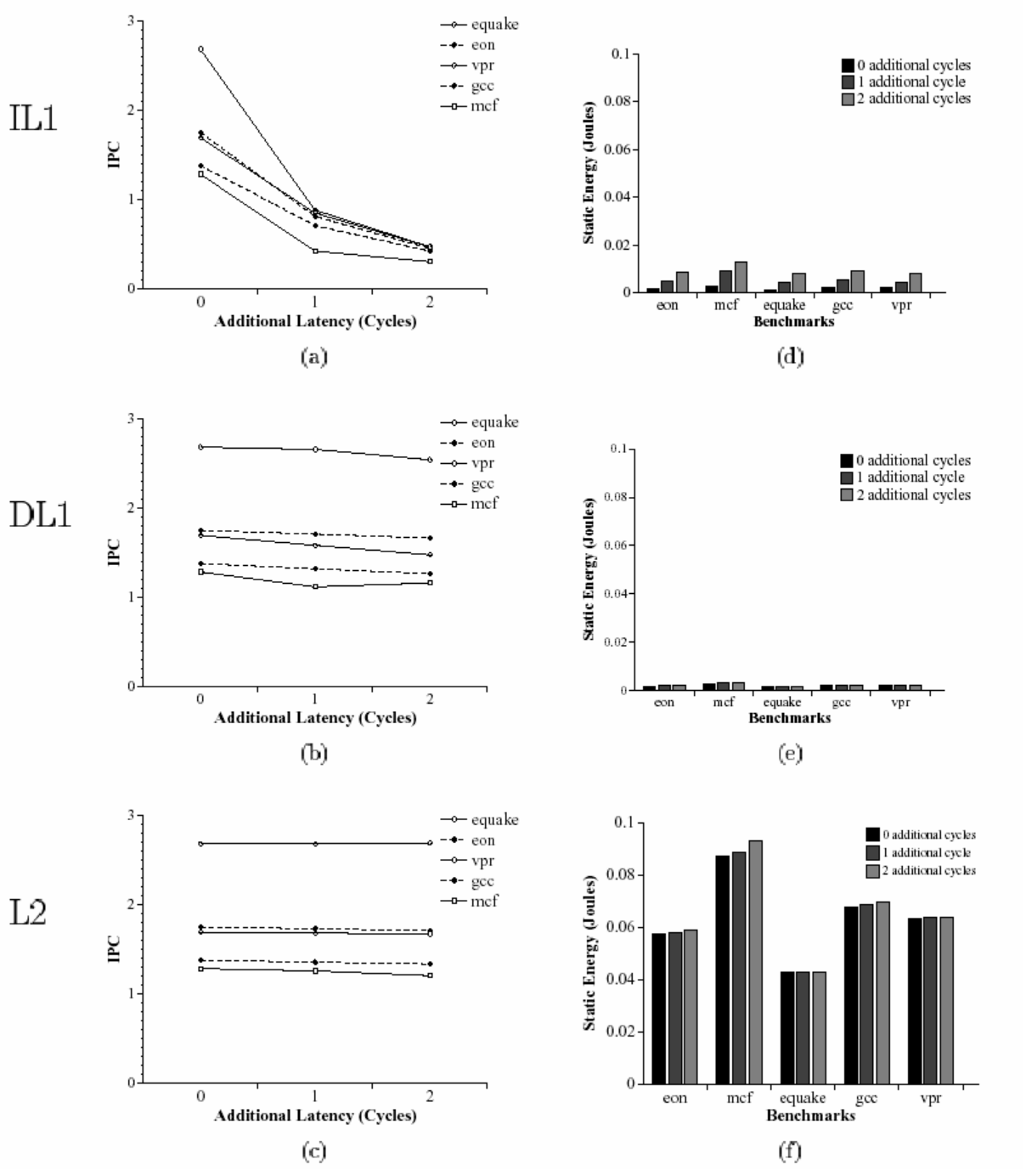


Figure 29: IPC and Energy Sensitivity to Access Delay for L1 and L2 Dual-V

Table 6 : Dual-V_T Sensitivity to Additional Access Delay in IL1 Cache

IPC	Cycles of additional delay in IL1		
benchmark	0	1	2
252.eon	1.75	0.81	0.45
181.mcf	1.28	0.42	0.30
183.equake	2.68	0.88	0.47
176.gcc	1.38	0.71	0.42
175.vpr	1.69	0.85	0.47

E/IPC	Cycles of additional delay in IL1		
benchmark	0	1	2
252.eon	2.60	5.56	10.05
181.mcf	3.45	10.72	14.94
183.equake	1.69	5.13	9.61
176.gcc	3.35	6.50	10.95
175.vpr	2.70	5.34	9.67

Table 7: Dual-V_T Sensitivity to Additional Access Delay in DL1 Cache

IPC	Cycles of additional delay in DL1		
	0	1	2
252.eon	1.75	1.71	1.66
181.mcf	1.28	1.12	1.16
183.equake	2.68	2.66	2.54
176.gcc	1.38	1.32	1.26
175.vpr	1.69	1.58	1.48

E/IPC	Cycles of additional delay in DL1		
	0	1	2
252.eon	0.98	0.99	1.00
181.mcf	1.10	1.26	1.18
183.equake	0.52	0.52	0.53
176.gcc	1.20	1.24	1.28
175.vpr	0.91	0.97	1.04

Table 8: Dual-V_T Sensitivity to Additional Access Delay in L2 Cache

IPC	Cycles of additional delay in L2		
	0	1	2
252.eon	1.75	1.73	1.71
181.mcf	1.28	1.26	1.20
183.equake	2.68	2.68	2.69
176.gcc	1.38	1.36	1.34
175.vpr	1.69	1.68	1.67

E/IPC	Cycles of additional delay in L2		
	0	1	2
252.eon	0.05	0.05	0.05
181.mcf	0.10	0.10	0.11
183.equake	0.02	0.02	0.02
176.gcc	0.10	0.10	0.10
175.vpr	0.05	0.06	0.06

MTCMOS: While MTCMOS does not suffer from additional latency to access cache lines in an awake state, its effectiveness does depend on the speed at which cache lines can be re-awakened. Additional clock cycles used to awaken sleeping cache lines can extend the program execution time, with the effect of reducing processor performance and increasing the static energy expended. The wakeup transition time is determined by the circuit configuration and physical parameters; this section explores the sensitivity of the MTCMOS technique applied to primary and secondary caches as the experimental wakeup penalty is varied from 1 to 10 cycles.

Graphs in the left column of Figure 30 show the combined effect of decay interval and wakeup latency on processor performance, charted as the harmonic mean of IPC and the harmonic mean of the static energy for program execution of all benchmarks in the suite. In Figure 30a, b, and c, the processor's performance is plotted as a function of the wakeup latency for four cache decay intervals: immediate sleep, 1K, 8K, and 64K cycles. Graphs in the right column of Figure 30 show the static energy consumption expended by the processor as a function of the wakeup latency for four cache decay intervals: immediate sleep, 1K, 8K, and 64K processor cycles. Unlike the dual-V_T scenario in which extra latency affects each cache access, MTCMOS caches incur extra latency only for accesses to sleeping cache lines.

An MTCMOS level-1 instruction cache causes the largest performance degradation in IPC when short decay intervals with long wakeup latencies are employed, as illustrated in Figure 30a. For an IL1 cache with an MTCMOS immediate sleep policy, the measured IPC drops by 93% when the wakeup penalty is ten cycles compared to a wakeup penalty of 1 cycle. For a larger decay interval of 64K cycles, when most useful cache lines are kept awake, the IPC is reduced by less than 1% when the wakeup penalty is increased from 1 to 10 cycles. With a decay interval of 8K, the best-case interval in this study for MTCMOS IL1 caches, the IPC is 1.35% lower for a ten-cycle wakeup time. Figure 30d shows that an MTCMOS IL1 cache with an immediate sleep mode uses 18 times more static energy with a wakeup penalty of 10 cycles than with a 1 cycle penalty. However, since dynamic energy dominates the total energy for the primary caches, the total IL1 cache energy consumption increases by only 3%. With a decay interval of 64K, the program execution time is not noticeably affected, and the static energy is essentially unchanged.

The MTCMOS DL1 cache also causes performance degradation with short decay intervals. As Figure 30b illustrates, an MTCMOS DL1 cache with an immediate sleep policy causes an IPC drop of 31% from 1-cycle to 10-cycle wakeup penalties. The extra execution time for this case leads to an additional 3mJ of static energy, an 86% increase. Longer decay intervals, however, show only a slight decrease in performance, and the static energy shows more sensitivity to the decay interval than to extra latency, as seen in Figure 30e.

Since L2 accesses are relatively infrequent, program execution time is only mildly extended due to waiting for sleeping L2 cache lines to transition to the active mode. A zero-cycle decay interval leads to the largest IPC drop of 8%. With most lines in a low-leakage mode, additional processor cycles contribute only a small amount of extra leakage current. The largest static energy increase was 7% for the immediate-sleep policy. Figure 30e shows that as the decay interval increases, the effect of additional latency decreases. Since static energy is the largest component of the total energy in the level-2 cache, the effect of increased static energy is an overall energy increase of 5% for the immediate-sleep configuration.

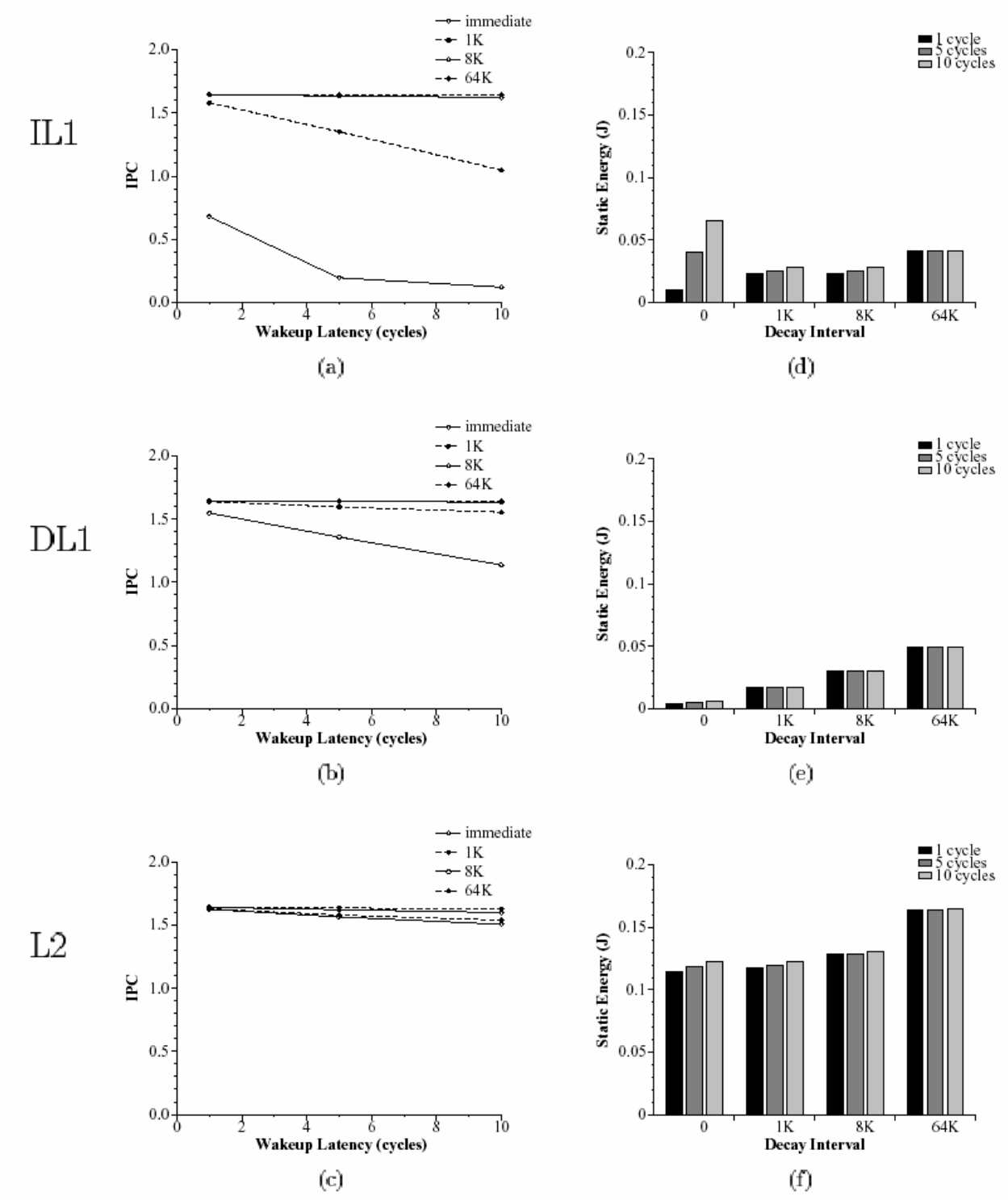


Figure 30: IPC and Energy Sensitivity to Access Delay for L1 and L2 MTCMOS Caches.

For reference, Tables 6 through 11 report the performance and energy-delay (E/IPC) results from the wakeup time sensitivity analysis.

Table 9 IPC Sensitivity to Wakeup Time Sensitivity for MTCMOS IL1 Cache

IPC	Benchmark	Wakeup Time (processor cycles)			
Decay Interval		**0**	**1**	**5**	**10**
Immediate Sleep	252.eon	1.75	0.81	0.20	0.13
	181.mcf	1.28	0.42	0.17	0.11
	183.equake	2.68	0.88	0.20	0.10
	176.gcc	1.38	0.71	0.20	0.14
	175.vpr	1.69	0.85	0.20	0.13
	harmonic mean	1.65	0.68	0.19	0.12
1K	252.eon	1.75	1.71	1.44	1.17
	181.mcf	1.28	1.14	0.89	0.67
	183.equake	2.68	2.70	2.21	1.21
	176.gcc	1.38	1.36	1.24	1.10
	175.vpr	1.69	1.69	1.60	1.46
	harmonic mean	1.65	1.58	1.35	1.05
8K	252.eon	1.75	1.75	1.74	1.73
	181.mcf	1.28	1.28	1.28	1.28
	183.equake	2.68	2.68	2.68	2.68
	176.gcc	1.38	1.37	1.35	1.32
	175.vpr	1.69	1.69	1.69	1.68
	harmonic mean	1.65	1.64	1.64	1.62
64K	252.eon	1.75	1.75	1.75	1.75
	181.mcf	1.28	1.28	1.28	1.28
	183.equake	2.68	2.68	2.68	2.68
	176.gcc	1.38	1.38	1.38	1.38
	175.vpr	1.69	1.69	1.69	1.69
	harmonic mean	1.65	1.65	1.64	1.64

Table 10: IPC Sensitivity to Wakeup Time Sensitivity for MTCMOS DL1 Cache

IPC	Benchmark	Wakeup Time (processor cycles)			
Decay Interval		**0**	**1**	**5**	**10**
Immediate Sleep	252.eon	1.75	1.71	1.58	1.37
	181.mcf	1.28	1.12	1.04	0.95
	183.equake	2.68	2.66	2.13	1.57
	176.gcc	1.38	1.34	1.19	1.02
	175.vpr	1.69	1.61	1.28	1.00
	harmonic mean	1.65	1.55	1.36	1.14
1K	252.eon	1.75	1.74	1.70	1.65
	181.mcf	1.28	1.28	1.22	1.18
	183.equake	2.68	2.68	2.71	2.70
	176.gcc	1.38	1.37	1.35	1.32
	175.vpr	1.69	1.69	1.64	1.57
	harmonic mean	1.65	1.64	1.60	1.55
8K	252.eon	1.75	1.75	1.75	1.75
	181.mcf	1.28	1.28	1.28	1.28
	183.equake	2.68	2.68	2.68	2.68
	176.gcc	1.38	1.38	1.37	1.36
	175.vpr	1.69	1.69	1.68	1.66
	harmonic mean	1.65	1.64	1.64	1.64
64K	252.eon	1.75	1.75	1.75	1.75
	181.mcf	1.28	1.28	1.28	1.28
	183.equake	2.68	2.68	2.68	2.68
	176.gcc	1.38	1.38	1.38	1.38
	175.vpr	1.69	1.69	1.69	1.69
	harmonic mean	1.65	1.65	1.65	1.64

Table 11: IPC Sensitivity to Wakeup Time Sensitivity for MTCMOS L2 Cache

IPC	Benchmark	Wakeup Time (processor cycles)			
Decay Interval		**0**	**1**	**5**	**10**
Immediate Sleep	252.eon	1.75	1.73	1.65	1.56
	181.mcf	1.28	1.26	1.20	1.18
	183.equake	2.68	2.68	2.71	2.71
	176.gcc	1.38	1.36	1.28	1.19
	175.vpr	1.69	1.69	1.65	1.61
	harmonic mean	1.65	1.63	1.57	1.51
1K	252.eon	1.75	1.74	1.67	1.60
	181.mcf	1.28	1.26	1.20	1.20
	183.equake	2.68	2.68	2.71	2.71
	176.gcc	1.38	1.36	1.30	1.24
	175.vpr	1.69	1.69	1.66	1.63
	harmonic mean	1.65	1.63	1.58	1.54
8K	252.eon	1.75	1.75	1.74	1.73
	181.mcf	1.28	1.28	1.28	1.28
	183.equake	2.68	2.68	2.68	2.68
	176.gcc	1.38	1.37	1.32	1.27
	175.vpr	1.69	1.69	1.68	1.66
	harmonic mean	1.65	1.64	1.62	1.60
64K	252.eon	1.75	1.75	1.75	1.74
	181.mcf	1.28	1.28	1.28	1.28
	183.equake	2.68	2.68	2.68	2.68
	176.gcc	1.38	1.37	1.36	1.34
	175.vpr	1.69	1.69	1.69	1.68
	harmonic mean	1.65	1.64	1.64	1.63

Table 12: Energy-Delay (E/IPC) Sensitivity to Wakeup Delay: MTCMOS IL1

E/IPC	Benchmark	Wakeup Time (processor cycles)			
Decay Interval		**0**	**1**	**5**	**10**
Immediate Sleep	252.eon	2.62	5.67	22.95	35.95
	181.mcf	3.48	10.98	27.27	42.92
	183.equake	1.70	5.25	23.30	44.91
	176.gcc	3.37	6.61	23.35	34.89
	175.vpr	2.73	5.47	23.09	36.41
	harmonic mean	2.78	6.80	23.99	39.03
1K	252.eon	2.62	2.69	3.18	3.94
	181.mcf	3.47	3.91	5.03	6.76
	183.equake	1.70	1.69	2.06	3.76
	176.gcc	3.36	3.40	3.73	4.21
	175.vpr	2.71	2.72	2.86	3.14
	harmonic mean	2.77	2.88	3.38	4.37
8K	252.eon	2.64	2.64	2.64	2.66
	181.mcf	3.47	3.47	3.47	3.47
	183.equake	1.70	1.70	1.70	1.70
	176.gcc	3.39	3.40	3.46	3.53
	175.vpr	2.72	2.72	2.73	2.75
	harmonic mean	2.78	2.79	2.80	2.82
64K	252.eon	2.64	2.64	2.64	2.64
	181.mcf	3.47	3.47	3.47	3.47
	183.equake	1.70	1.70	1.70	1.70
	176.gcc	3.42	3.42	3.43	3.43
	175.vpr	2.73	2.73	2.73	2.73
	harmonic mean	2.79	2.79	2.79	2.79

Table 13: Energy-Delay (E/IPC) Sensitivity to Wakeup Delay: MTCMOS DL1 Cache

E/IPC	Benchmark	Wakeup Time (processor cycles)			
Decay Interval		**0**	**1**	**5**	**10**
Immediate Sleep	252.eon	1.00	1.00	1.08	1.23
	181.mcf	1.12	1.29	1.33	1.45
	183.equake	0.53	0.53	0.64	0.86
	176.gcc	1.23	1.25	1.36	1.56
	175.vpr	0.93	0.98	1.21	1.55
	harmonic mean	0.95	1.00	1.12	1.32
1K	252.eon	0.99	0.99	1.01	1.05
	181.mcf	1.11	1.11	1.17	1.22
	183.equake	0.52	0.52	0.52	0.52
	176.gcc	1.22	1.22	1.24	1.26
	175.vpr	0.92	0.93	0.95	1.00
	harmonic mean	0.94	0.94	0.97	1.00
8K	252.eon	1.01	1.01	1.01	1.01
	181.mcf	1.12	1.12	1.12	1.12
	183.equake	0.52	0.52	0.52	0.52
	176.gcc	1.25	1.25	1.25	1.26
	175.vpr	0.94	0.94	0.94	0.95
	harmonic mean	0.95	0.95	0.96	0.96
64K	252.eon	1.02	1.02	1.02	1.02
	181.mcf	1.18	1.18	1.18	1.18
	183.equake	0.52	0.52	0.52	0.52
	176.gcc	1.29	1.29	1.29	1.29
	175.vpr	0.98	0.98	0.98	0.98
	harmonic mean	0.98	0.98	0.98	0.98

Table 14: Energy-Delay (E/IPC) Sensitivity to Wakeup Delay: MTMCOS L2 Cache

E/IPC	Benchmark	Wakeup Time (processor cycles)			
Decay Interval		**0**	**1**	**5**	**10**
Immediate Sleep	252.eon	0.08	0.08	0.09	0.10
	181.mcf	0.16	0.17	0.18	0.19
	183.equake	0.03	0.03	0.03	0.03
	176.gcc	0.14	0.15	0.16	0.18
	175.vpr	0.09	0.09	0.09	0.10
	harmonic mean	0.08	0.09	0.09	0.10
1K	252.eon	0.08	0.08	0.09	0.10
	181.mcf	0.16	0.17	0.19	0.19
	183.equake	0.03	0.03	0.03	0.03
	176.gcc	0.15	0.15	0.16	0.17
	175.vpr	0.09	0.09	0.09	0.10
	harmonic mean	0.09	0.09	0.09	0.10
8K	252.eon	0.09	0.09	0.09	0.09
	181.mcf	0.17	0.17	0.17	0.17
	183.equake	0.03	0.03	0.03	0.03
	176.gcc	0.17	0.17	0.18	0.20
	175.vpr	0.10	0.10	0.10	0.10
	harmonic mean	0.09	0.09	0.09	0.09
64K	252.eon	0.10	0.10	0.10	0.10
	181.mcf	0.26	0.26	0.26	0.26
	183.equake	0.03	0.03	0.03	0.03
	176.gcc	0.28	0.28	0.28	0.29
	175.vpr	0.14	0.14	0.14	0.14
	harmonic mean	0.11	0.11	0.11	0.11

Chapter 5 Related Work

The challenge of designing microprocessors within a tight power budget is compounded by a dramatic increase in static power in emerging technology generations. Several research and industrial groups are investigating means of reducing static power. One technique tailors transistor types in the circuit design to use high-speed transistors though critical paths and instantiates low-leakage transistors in areas that have more slack in their timing budgets [9]. This technique may be applied to combinational and sequential logic, as well as memory structures. The dual-V_T SRAM circuit used in this study is an implementation of this technique.

Another method of reducing leakage current is raising the effective threshold voltage with the "body effect" of transistors in series. Ye, et al. employ this technique with logic blocks by arranging the input signals of combinational logic to maximize the maximum number of series transistors that are "off" while the logic block is idle [20][3].

The gated-V_{DD} technique benefits from the body effect, as the gating transistor raises the effective V_T of other transistors in the memory cells, though the majority of leakage-current reduction in the gated-V_{DD} approach is due to the gating transistor that disconnects transistors from power supplies. Powell, et al. found that the gated-V_{DD} technique reduced leakage current by 97% in a 180nm technology generation for a level-1 instruction cache [4].

Several researchers have suggested methods for dynamically adjusting transistor threshold voltages [21][22][23][24][25][26] by changing the substrate bias, including the auto-backgate-controlled method we implemented as the MTCMOS technique in this study. In our research, we use the MTCMOS circuit to control the sleep mode for individual cache lines, rather than large logic blocks or areas of the chip as proposed in prior studies.

Another technique that reduces leakage current in memories is reducing the supply voltage to SRAM cells during periods of inactivity, and pulsing the power and ground nodes to higher and lower voltages, for a row in the memory array to temporarily give a larger supply voltage to SRAM cells during accesses [27].

These circuit techniques can be incorporated into architectural solutions, which rely on programs use of system resources to reduce static power and energy. One example of an architectural technique employs the gated-V_{DD} circuit to selectively disable cache lines based on miss rates, dynamically resizing the instruction cache (DRI I-cache) to a size appropriate for the currently executing program. Yang, et al. found that a 64K DRI I-cache reduced the energy-delay product by 62% with a 4% increase in execution time with SPEC95 benchmarks, compared to a standard cache [28]. Kaxiras, et al. are continuing development of the gated-V_{DD} technique with an adaptive control on the gating transistor, and have shown that their technique can reduce leakage energy in level-1 caches by a factor of 5 [14]. A static power model has recently been proposed by Butts [29], to allow comparison of architectural techniques' static power requirements.

Chapter 6 Conclusion

Power and energy consumption have become critical constraints for microprocessor design as both dynamic and static power components are increasing with each technology generation. Dynamic power and energy reduction have been active areas of research; managing static power and energy will also require innovative techniques to effectively control leakage current.

Leakage current has an exponential dependence on the operating temperature and the threshold voltage. Threshold voltages are scaling to lower values each technology generation; leakage current is increasing dramatically with each fabrication process. Circuit techniques and architectural solutions are under investigation to meet the need of microprocessor power management, including aggressive static power management. We evaluated three techniques to reduce static energy by reducing leakage current: dual-V_T, gated-V_{DD}, and MTCMOS. Each technique is effective in reducing leakage energy, and each has unique tradeoffs in energy and performance.

6.1 DUAL-V_T

The dual-V_T caches are effective at reducing leakage; however, with an extra cycle of delay, the technique has a negative effect on performance for level-1 caches. The dual-V_T technique reduced the static energy consumed by the IL1 cache by 96%, at the expense of degrading the IPC to less than half of the baseline case. The energy-delay product of the dual-V_T technique was 56% higher (worse) than the baseline case for the IL1. Although the leakage current and therefore static power and energy are reduced, the performance penalty may be unacceptable for a dual-$_{VT}$ method applied to an instruction cache, or other structures that rely on fast access times.

The dual-$_{VT}$ DL1 cache reduced static energy by 98%, with an energy-delay product that is 4% better than the baseline case. In the level-2 cache experiment, the dual-$_{VT}$ technique improved both static energy and energy-delay product. The static energy is reduced by 98% with negligible performance degradation and the energy-delay product improved by a factor of 50.

6.2 GATED-V_{DD}

In the gated-V_{DD} cache, static energy savings are offset by the dynamic energy and time required to service additional misses to prematurely disabled cache lines. The gated-V_{DD} technique implemented with a 64K decay interval produced a 72% static energy savings, with a 1.9% improvement in energy-delay for the IL1 cache compared with the baseline. In the DL1 cache, the technique had similar results: 79% reduction in static energy, with a 6.4% improvement in the energy-delay product. The level-2 cache is infrequently accessed and thus the penalties for additional time and dynamic energy are small. The gated-V_{DD} technique reduced energy by 95% in the level-2 cache, and improved the energy-delay factor by a factor of 20.

6.3 MTCMOS

The MTCMOS IL1 cache with an 8K decay interval reduced static energy by 75%, an improvement in energy-delay of 2%. In the DL1 cache, the MTCMOS technique and a 1K-decay interval reduced static energy by 87.95%, while improving the energy-delay product by 7.7%. The modest increases in energy-delay are due to the performance degradation of additional latency to wake up idle cache lines offsetting most of the energy savings. The level-2 cache with MTCMOS circuitry and an immediate sleep mode had a 97% reduction in static energy, and reduced the energy-delay product by a factor of 34.

6.4 SUMMARY

In this paper we have explored energy and performance trade-offs associated with three techniques for reducing static energy consumption in on-chip caches: high-V_T transistors in memory arrays, power supply switching, and dynamic transistor threshold modulation. Each of the techniques is effective in reducing energy consumption in primary and secondary caches. We found that with careful selection of decay intervals, the MTCMOS and gated-V_{DD} techniques yielded better energy-delay products than the dual-V_T technique in the primary caches, due to their overall lower access time. With our assumptions, both the gated-V_{DD} and MTCMOS techniques improve the energy-delay product by 2% in the IL1 cache, and yield an improvement of 6% and 7%, respectively, in the DL1 cache compared to the experimental baseline. The dual-V_T technique improves the energy-delay product of the DL1 by 4%, and degrades energy-delay product in the IL1. For the secondary cache, the dual-V_T technique has the best energy-delay characteristics, with a 50-fold improvement compared to the baseline case. The gated-V_{DD} and MTCMOS techniques were also effective at improving the energy-delay of L2 caches, with overall reductions of factors of 20 and 34, respectively. However, additional latency and energy penalties contributed by the leakage reduction strategy can extend program execution time and increase static energy consumption, especially when applied to the primary instruction cache. Increasing the dual-V_T IL1 cache access by two extra cycles results in performance degradation of 74%, and a 387% increase in static energy expenditure. For an MTCMOS IL1 with a zero-cycle decay interval, performance drops by 93% and static energy increases by a factor of 18 when the wakeup latency is ten cycles rather than one. In the level-1 data cache, the effect of additional access time was less detrimental. A dual-V_T DL1 with two additional cycles of access time reduces performance by 4% and increases static energy by 9%. An MTCMOS DL1 with a ten-cycle wakeup latency causes performance to drop by 31% with the shortest decay interval; longer decay intervals do not suffer such performance degradation. The unified level-2 cache is the least sensitive to additional delays, with a 2% dip in IPC for the dual-V_T L2 cache accompanied by a 2% increase in static energy; an MTCMOS L2 cache with the worst-case of immediate sleep policy caused 8% reduction in IPC and 7% increase in static energy consumed.

This paper has emphasized static energy reduction in cache memories while considering the effect on processor performance and total energy. The same principles may be applied to other hardware structures, as well. For example, the static energy required to maintain the state of branch predictor table entries may be balanced against the dynamic energy required to execute with fewer correct predictions. Future work will include static energy analysis of other microarchitectural features and their impact on microprocessor performance and total energy.

References

1 F. Pollack, New Microarchitecture Challenges in the Coming Generations of CMOS Process Technologies. Keynote speech, 32nd Annual International Symposium on Microarchitecture.

2 Brooks, D.M.; Bose, P.; Schuster, S.E.; Jacobson, H.; Kudva, P.N.; Buyuktosunoglu, A.; Wellman, J.; Zyuban, V.; Gupta, M.; Cook, P.W. Power-aware microarchitecture: design and modeling challenges for next-generation microprocessors. *Micro* 20(6):26-44, November/December 2000.

3 A. Chandrakasan. Low power circuit and system design, 2000. International Electron Device Meeting short course.

4 M. Powell, S.-H. Yang, B. Falsafi, K. Roy, and T. Vijaykumar. Gated-vdd: A circuit technique to reduce leakage in deep-submicron cache memories. In *International Symposium on Low Power Electronics and Desig*n, 2000, pages 90-95.

5 K. Nii, H. Makino, Y. Tujihashi, C. Morishima, Y. Hayakawa, H. Nunogami, T. Arakawa, and H. Hamano. A low power SRAM using auto-backgate-controlled MT-CMOS. In *International Symposium on Low Power Electronics and Desig*n, pages 293–298, 1998.

6 T. Mudge. Power: A first-class architectural design constraint. *Computer*, 34(4):52 -58, April 2001.

7 S. M. Kang and Y Leblebici. *CMOS Digital Integrated Circuits Analysis and Design*, McGraw-Hill Companies, Inc., page 68.

8 C.C. Wu, C. H. Diaz, B.L. Lin, S.Z. Chang, C. C. Wang, J. J. Liaw, C.H. Wang, K.K. Young, K. H. Lee, B. K. Liew, J.YC. Sun. Ultra-Low Leakage 0.16 um CMOS for Low-Standby Power Applications. In *International Electron Devices Meeting. IEDM Technical Digest.* pages 671-674, 1999.

9 T. McPherson, R. Averill, D. Balazich, K. Barkley, S. Carey, Y. Chan, Y. Chan, R. Crea, A. Dansky, R. Dwyer, A. Haen, D. Hoffman, A. Jatkowski, M. Mayo, D. Merrill, T. McNamara, G. Northrop, J. Rawlins, L. Sigal, T. Slegel, and D. Webber. 760 mhz g6 s/390 microprocessor exploiting multiple vt and copper interconnects. In *International Solid-State Circuits Conference*, pages 96–97, 2000.

10 S. Borkar. Design challenges of technology scaling. *IEEE Micr*o, 19(4):23–29, July-August 1999.

11 The international technology roadmap for semiconductors. Semiconductor Induatry Association, 1999.

12 S. M. Kang and Y Leblebici. *CMOS Digital Integrated Circuits Analysis and Design*, McGraw-Hill Companies, Inc., page 124.

13 G. Reinman and N. Jouppi. An integrated cache timing and power model, 1999. Unpublished document. research.Compaq.com/wrl/people/jouppi/cacti2.ps

14 S. Kaxiras, Z. Hu and M. Martonosi. Cache Decay: Exploiting Generational Behavior to Reduce Cache Leakage Power. To appear in *International Symposium on Computer Architecture*, 2001.

15 M. Horowitz, R. Ho, and K. Mai. The future of wires. In *Semiconductor Research Corporation Workshop on Interconnects for Systems on a Chi*p, May 1999.

16 D. Liu and C. Svensson. Power consumption estimation in CMOS VLSI chips. *IEEE Journal of Solid-State Circuit*s, 29(6):663–660, June 1994.

17 Pentium III processor for the sc242 at 450MHz to 1.13GHz. Intel Corporation, June 2000. Order Number 244452-008.

18 D. Burger and T. M. Austin. The simplescalar tool set version 2.0. Technical Report 1342, Computer Sciences Department, University of Wisconsin, June 1997.

19 R. Kessler. The alpha 21264 microprocessor. *IEEE Micr*o, 19(2):24–36, March/April 1999.

20 Y. Ye, S. Borkar, and V. De. A new technique for standby leakage reduction in high performance circuits. In *Symposium on VLSI Circuit*s, pages 40–41, 1998.

21 T. Hiramoto and M. Takamiya. Low power and low voltage MOSFETs with variable threshold voltage controlled by back-bias. *IEICE Transactions on Electronic*s, E83-C(2):663–660, February 2000.

22 M. Horiuchi. A new dynamic-threshold SOI device having an embedded resistor and a merged body-bias-control transistor. In *Proceedings of the IEEE International Electron Devices Meetin*g, pages 419–22, December 1998.

23 H. Makino, Y. Tujihashi, K. Nii, C. Morishima, Y. Hayakawa, T. Shimizu, and T. Arakawa. An auto-backgate-controlled MT-CMOS circuit. In *Symposium on VLSI Circuit*s, pages 42–43, 1998.

24 C. Vieri, I. Yang, A. Chandrakasan, and D. Antoniadis. SOIAS: Dynamically variable threshold SOI with active substrate. In *International Symposium on Low Power Electronic*s, pages 86–87, 1995.

25 T. Kuroda, T. Fujita, S. Mita, T. Nagamatsu, S. Yoshioka, K. Suzuki, F. Sano, M. Norishima, M. Murota, M. Kako, M. Kinugawa, M. Kakumu, and T. Sakurai. A 0.9-v, 150mhz, 10mw 4mm, 2-d discrete cosine transform core processor with variable threshold-voltage (vt) scheme. *IEEE Journal of Solid-State Circuit*s, 31(11):1770–1779, November 1996.

26 K. Nii, H. Makino, Y. Tujihashi, C. Morishima, Y. Hayakawa, H. Nunogami, T. Arakawa, and H. Hamano. A low power SRAM using auto-backgate-controlled MT-CMOS. In *International Symposium on Low Power Electronics and Desig*n, pages 293–298, 1998.

27 A. J. Bhavnagarwala, A. Kapoor, J. D. Meindl. Dynamic-Threshold CMOS SRAM Cells for Fast, Portable Applications. *In 13th Annual IEEE International ASIC/SOC Conference*, pages 359 –363, 2000.

28 S.-H. Yang, M. D. Powell, B. Falsafi, K. Roy, and T. Vijaykumar. An integrated circuit/architecture approach to reducing leakage in deep-submicron high-performance caches. In *International Symposium on High-Performance Computer Architecture*,pages 147-157, 2001.

29 J. A. Butts and G. S. Sohi. A Static Power Model for Architects. In *International Symposium on Microarchitecture*, pages 191-201, 2000.