Power-Optimal Pipelining in Deep Submicron Technology *

Seongmoo Heo and Krste Asanović
MIT Computer Science and Artificial Intelligence Laboratory
32 Vassar Street, Cambridge, MA 02139
{heomoo,krste}@csail.mit.edu

ABSTRACT

This paper explores the effectiveness of pipelining as a power saving tool, where the reduction in logic depth per stage is used to reduce supply voltage at a fixed clock frequency. We examine power-optimal pipelining in deep submicron technology, both analytically and by simulation. Simulation uses a 70 nm predictive process with a fanout-of-four inverter chain model including input/output flip-flops, and results are shown to match theory well. The simulation results show that power-optimal logic depth is 6 to 8 FO4 and optimal power saving varies from 55 to 80% compared to a 24 FO4 logic depth, depending on threshold voltage, activity factor, and presence of clock-gating.

We decompose the power consumption of a circuit into three components, switching power, leakage power, and idle power, and present the following insights into power-optimal pipelining. First, power-optimal logic depth decreases and optimal power savings increase for larger activity factors, where switching power dominates over leakage and idle power. Second, pipelining is more effective with lower threshold voltages at high activity factors, but higher threshold voltages give better results at lower activity factors where leakage current dominates. Lastly, clock-gating enables deeper pipelining and more power saving because it reduces timing element overhead when the activity factor is low.

Categories and Subject Descriptors: B.7.1 [Integrated Circuits]: Types and Design Styles–*Advanced Technologies, Microprocessors and Microcomputers, VLSI*

General Terms: Performance, Design, Theory

Keywords: Pipelining, Supply Voltage Reduction, Power Scaling

1. INTRODUCTION

Pipelining reduces the number of logic levels between registers and is usually employed by digital systems designers to increase achievable clock frequency. But the time slack obtained from pipelining can also be used to reduce power consumption by lowering supply voltage at a fixed clock frequency. This technique can be very effective for digital systems with fixed throughput requirements and highly parallel computations. Supply voltage scaling is by far one of the most effective techniques for trading time slack for power. Supply voltage reduction leads to a quadratic reduction in active power and also a super-linear reduction in leakage power, as leakage current has a strong dependency on drain voltage in deep submicron processes. A parallel architecture could also be used to provide excess performance to trade for lower power, but pipelining has the advantage of a lower area penalty. Power reductions from pipelining are eventually limited by the power overhead of the additional pipeline latches or flip-flops required for each additional pipe stage, leading to a power-optimal level of pipelining.

In this paper, we show how power-optimal pipelining varies for different operating regimes in deep submicron technology. We examine the tradeoffs between pipeline depth, supply voltage, threshold voltage, and total power using circuit-level simulations and analytical models. We also explore the effect of activity factor and clock gating.

2. RELATED WORK

The trend towards deeper pipelines in microprocessors is clearly seen in the evolution of Intel x86 family, with a factor of 7 reduction in logic depth per stage over the last decade [9]. This reduction in logic depth has combined with improvements in transistor speed from technology scaling to yield an even larger increase in processor clock frequency. Increasing the number of pipeline stages for an operation increases its latency in clock cycles, which in turn increases the number of pipeline stalls experienced by dependent operations. The resulting reduction in instructions completed per cycle (IPC) reduces the performance advantage from greater clock frequency, with greater impact on codes with lower instruction-level parallelism (ILP).

Processor architects have explored this tradeoff between increased clock frequency and reduced IPC to determine performance-optimal pipelining depth. Early work by Kunkel and Smith [10] considered pipelining in vector supercomputers and found that 8–10 ECL gate levels was performance-optimal for scalar code, and as little as 4 gate levels for more parallel vector code. Recently, several authors have investigated the performance-optimal pipeline depth for superscalar microprocessors [5, 9, 11], with a consensus in the range of 8–11 FO4 delays for SPEC integer codes and around 6 FO4 delays for SPEC floating-point codes, which generally have higher ILP. These performance-optimal numbers ignore power as well as the design and verification complexity that would accompany such high-frequency designs (roughly twice the clock rate of existing systems [11]).

Several authors have extended superscalar performance mod-

*This work was partly funded by NSF CAREER award CCR-0093354, NSF ITR award CCR-0219545, and a donation from Intel Corporation.

ISLPED'04, August 9–11, 2004, Newport Beach, CA

els with power models that include the power overhead of additional pipeline latches [12, 6]. Srinivasan et al. [12] found that power-performance optimal logic depth increases to about 18 FO4 for SPEC benchmarks and around 24–28 FO4 for TPC-C, a commercial application. Hartstein and Puzak [6] found 22.5 FO4 is the power-performance optimum according to their power-performance metric. They also found that clock gating pushes the optimum back to deeper pipelines [6] which agrees with our results.

This previous work focuses on processor performance, where limited instruction parallelism reduces the benefits of deep pipelines, and these studies limit power optimization to the selection of the correct number of additional pipeline stages. Other types of digital system, including digital signal processors, network processors, and graphics engines, have much greater levels of parallelism and often have fixed throughput requirements. For these systems, pipelining can be used together with voltage and threshold scaling to reduce total energy consumption while maintaining a fixed clock rate. The use of pipelining for power reduction was proposed by Chandrakasan et al.[2] but without an attempt to determine the power-optimal pipelining strategy.

3. METHODOLOGY

In this paper, our main target is a logic-dominant pipeline stage. We make several simplifying assumptions in our analysis. We are interested in fixed-throughput designs for highly parallel computations and so do not include any performance loss from an increased frequency of pipeline stalls as pipeline depths increase. Global wire delay does not scale as fast as gate delay as feature size is reduced, and some modern microprocessors have so-called drive stages which include only wires and repeaters [8]. We leave wire-dominant pipeline stages for future work but note that wire RC delay become relatively less important as supply voltage is scaled down in a fixed technology, because wire resistance remains constant while effective transistor resistance increases. We do not include local wire capacitance due to the absence of detailed circuit layouts, but note that wire cap can be an important component of total load in deep submicron technology even for a logic-dominant stage.

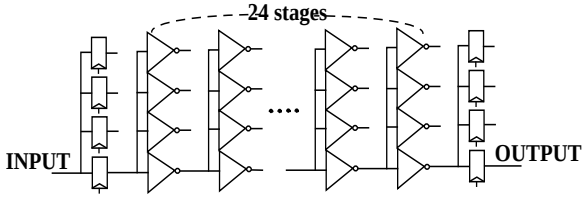

Figure 1: Baseline pipeline stage model. Input and clock buffers are not shown.

Figure 1 shows the baseline pipeline stage model assumed in our study. To model a well-designed path in a circuit, we use a simple static inverter chain with each inverter driving four copies of itself to yield a FO4 load. We use 24 FO4 delays as a baseline clock period, representing a current high-performance processor circuit (the high-frequency Pentium-4 has a 20 FO4 cycle time [11], and most other designs have somewhat shallower pipelines).

Even though different circuit styles and logic gates might lead to different power-optimal pipelining results, we assume that our FO4 inverter chain model is fairly representative and insights gathered from our simulation results can be applied to other cases. Flip-flops were chosen as the timing elements rather than latches due to their simplicity of usage, and the PowerPC transmission-gate flip-flop was chosen because it is a popular choice due to its robustness and energy-efficiency [7]. While the transistor sizes of inverters and flip-flops were fixed, the sizes of clock buffers were varied to ensure the appropriate clock rise and fall times when varying the depth of pipelining.

We used the BPTM 70 nm transistor models with different threshold voltages [4] and HSPICE for circuit simulation. Throughout the paper, clock frequency was fixed at 2 GHz and temperature was constant at 100 ^{o}C. We only considered subthreshold leakage; although gate leakage might become significant at some point in these technology generations, it is also likely that new gate dielectrics will make gate leakage insignificant again.

4. PIPELINING AND SUPPLY VOLTAGE

We begin by showing the effect of pipeline depth on supply voltage. With delay fixed, supply voltage scales down as pipelining deepens because the logic amount per pipeline stage decreases. Synchronous circuit delay is approximately given by

$$delay \quad \propto \quad (N+k) \times \frac{V_{dd}}{(V_{dd}-V_{th})^{\alpha}} \tag{1}$$

where N is the logic depth per pipeline stage in term of FO4 delay (or the number of FO4 inverters per pipeline stage), k is the timing element delay normalized by FO4 delay, α is a velocity saturation effect factor, V_{dd} and V_{th} are supply and threshold voltages respectively. Assuming α is 2 (actual value of α in deep submicron technology is close to 1.5 due to the short-channel effect),

$$N+k \quad \propto \quad V_{dd} - 2V_{th} + \frac{V_{th}^2}{V_{dd}} \tag{2}$$

Now assuming $\frac{V_{th}^2}{V_{dd}}$ is close to zero, we get a simple linear equation between V_{dd} and N, where a_0 is a constant:

$$V_{dd} \quad = \quad a_0 N + a_1 \tag{3}$$

$$(\frac{a_1}{a_0} \quad = \quad k + \frac{2}{a_0} V_{th}) \tag{4}$$

Figure 2 shows the simulated supply voltages when varying the number of FO4 inverters per stage for different threshold voltages. LVT, MVT, and HVT represent low, medium, and high threshold voltages respectively and their values are shown in Table 1. Low threshold voltage results in low supply voltage for the same delay. a_0 and a_1 were calculated using least square method and shown in Table 1. We can see that a_0 is proportional to V_{th} as well as a_1 (our simplified equations fail to explain the effect).

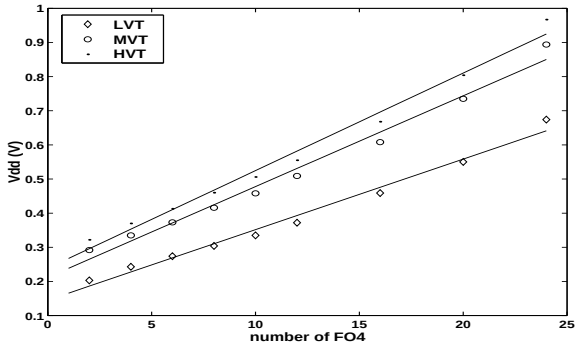

Figure 2: Supply voltage scaling shown as voltage required to achieve 2 GHz with given number of FO4 logic levels per pipeline stage.

V_{th} (V) NMOS(PMOS)	name	a_0	a_1
0.17 (-0.20)	LVT	0.0207	0.1450
0.19 (-0.22)	MVT	0.0266	0.2119
0.21 (-0.24)	HVT	0.0286	0.2389

Table 1: Threshold voltages and supply voltage scaling coefficients.

5. PIPELINING POWER COMPONENTS

In this section, we explore the impact of pipelining on the components of total power consumption when delay is fixed. We use the supply voltage scaling results shown above in Section 4 and investigate switching, leakage, and idle components of power consumption assuming no clock-gating mechanism.

5.1 Pipelining and Switching Power

Switching power remains the dominant component of total power consumption when the activity factor is high, even in leaky deep submicron technology. Switching power is the power consumed while charging and discharging load capacitances. The load capacitances include transistor parasitic and wire capacitances. Because we assume our pipeline stage is logic-dominant, wire capacitances are not included in our simulation.

The switching power of a pipelined logic stage can be divided between that due to logic gates and that due to timing elements, and can be modeled as:

$$P_{switching} = (b_0 + \frac{b_1}{N})V_{dd}^2 \quad (5)$$

$$= b_0 a_0^2 (1 + \frac{b_1}{b_0}\frac{1}{N})(N + \frac{a_1}{a_0})^2 \quad (6)$$

The overhead includes clock and switching power of timing elements and it is inversely proportional to, N, the number of logic gates per stage. We assume that the number of latches increases linearly with the number of pipeline stages. All the switching power components are proportional to V_{dd}^2. The ratio of switching power coefficients $\frac{b_1}{b_0}$ is approximately the ratio of the parasitic capacitances of one FO4 inverter and one timing element.

When N is much greater than $\frac{a_1}{a_0}$ and $\frac{b_1}{b_0}$, $P_{switching}$ becomes quadratic to N as shown in Eq. 7, which represents the dominance of logic gate switching power.

$$P_{switching} \approx b_0 a_0^2 N^2 \quad (7)$$

On the other hand, if N gets much smaller than $\frac{a_1}{a_0}$ and $\frac{b_1}{b_0}$, $P_{switching}$ becomes inversely proportional to N, as shown in Eq. 8, which represents the dominance of timing element switching power:

$$P_{switching} \approx b_1 a_1^2 \frac{1}{N} \quad (8)$$

Note that the $(N + \frac{a_1}{a_0})^2$ term in Eq. 6 makes relative $P_{switching}$ scale down slowly when $\frac{a_1}{a_0}$ is large. Since a higher V_{th} process has a higher $\frac{a_1}{a_0}$, as shown in Table 1, a higher V_{th} process gets less switching power saving from pipelining.

The optimal logic depth N^* is given by:

$$N^* = \frac{1}{4}(\sqrt{\frac{b_1}{b_0}^2 + 8\frac{a_1}{a_0}\frac{b_1}{b_0}} - \frac{b_1}{b_0}) \quad (9)$$

The equation indicates that the capacitance ratio of a timing element and an FO4 inverter $\frac{b_1}{b_0}$ is positively correlated to N^*. That is, larger timing element parasitics lead to less deep pipelining. However, N^* is not as sensitive to $\frac{b_1}{b_0}$ as it is to $\frac{a_1}{a_0}$. This means that V_{th} and timing element delay k affect N^* and correspondingly optimal power saving more significantly (Eq. 4).

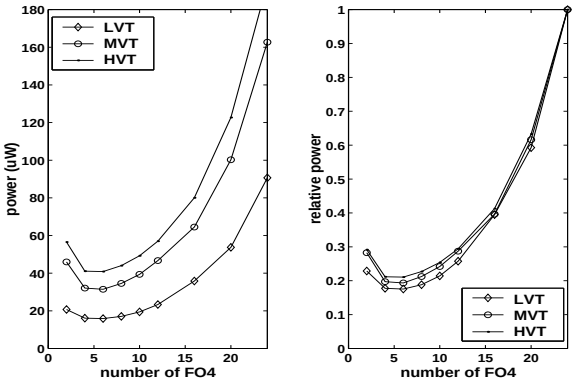

Figure 3: Switching power scaling.

Figure 3 shows the switching power obtained through HSPICE simulation of our model pipeline stage. Optimal logic depth N^* was 6 and optimal power saving was from 79 to 82% compared to the baseline of $N = 24$. The graphs show that lower threshold voltages gives slightly lower optimal logic depth and also slightly greater switching power saving. The variances are quite small since the variance of $\frac{a_1}{a_0}$ is small.

5.2 Pipelining and Leakage Power

The rapid reduction in gate length and accompanying downscaling of threshold voltages over the last few process generations has led to an exponential growth in leakage power. Within a few process generations, it is predicted power dissipation from static leakage current could be comparable to dynamic switching energy [1, 3].

The leakage power of our pipelined circuit can be given by the following equations:

$$P_{leak} = (c_0 + \frac{c_1}{N})V_{dd}\, e^{\frac{-V_{th}+\eta V_{dd}}{n v_T}} \quad (10)$$

$$= c_0 a_0\, e^{\frac{-V_{th}}{n v_T}}(1 + \frac{c_1}{c_0}\frac{1}{N})(N + \frac{a_1}{a_0})\, e^{\frac{\eta a_0}{n v_T}(N+\frac{a_1}{a_0})} \quad (11)$$

where nv_T is a constant representing leakage current slope, η is a Drain-Induced-Barrier-Lowering (DIBL) coefficient, and $\frac{c_1}{c_0}$ is the ratio of leakage power of one FO4 inverter versus one timing element. As in the switching power model, the leakage power in a stage can be divided into logic gate leakage and timing element leakage, with timing element leakage inversely proportional to N.

When N is much greater than $\frac{a_1}{a_0}$ and $\frac{c_1}{c_0}$, P_{leak} becomes proportional to the product of N and the exponential term $e^{\frac{\eta a_0}{n v_T}N}$:

$$P_{leak} \approx c_0 a_0 e^{\frac{-V_{th}}{n v_T}} N e^{\frac{\eta a_0}{n v_T}N} \quad (12)$$

The exponential term, $e^{\frac{\eta a_0}{n v_T}(N+\frac{a_1}{a_0})}$ represents the dependence of leakage current on the drain voltage (from DIBL). In modern deep submicron technology, for an appropriate supply voltage range, this term is larger than $O(1)$ but smaller than $O(N)$, therefore leakage power is reduced in a super-linear fashion as N decreases, though less than the quadratic reduction for switching power. Also, it is noted that the exponential term scales down faster as N decreases when a_0 is larger. A higher V_{th} process has higher a_0 as shown in Table 1, and so it is expected that higher V_{th} process will see greater leakage power saving from pipelining, which is the

opposite to the switching power case, but higher V_{th} processes have less absolute leakage to begin with.

On the other hand, if N becomes much smaller than $\frac{a_1}{a_0}$ and $\frac{c_1}{c_0}$, P_{leak} becomes inversely proportional to N just as in the $P_{switching}$ case:

$$P_{leak} \approx c_1 a_1 e^{\frac{-V_{th}}{n v_T}} \frac{1}{N} e^{\frac{\eta a_1}{n v_T}} \quad (13)$$

Figure 4 shows the simulated leakage power while varying the number of logic gates per stage. Optimal logic depth N^* was around six and optimal power saving was around 70–75%. The graphs show that lower threshold voltages gives less leakage power saving and slightly greater optimal logic depth.

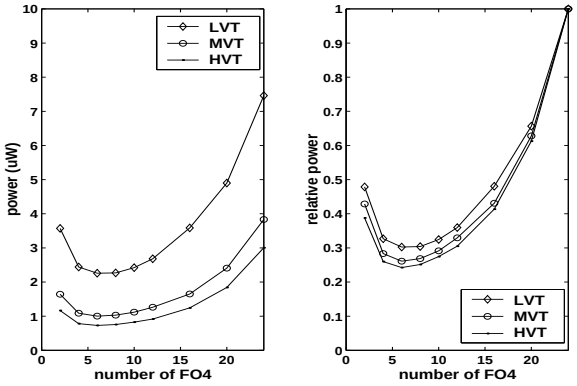

Figure 4: Leakage power versus logic depth per stage.

5.3 Idle Power without Clock-Gating

Clock-gating is a popular switching power reduction technique which inactivates the clock signal to timing elements within an inactive block when a circuit block is idle. But clock gating is not always possible due to the increase control complexity or the insufficient setup time of the clock enable signal. This section focuses on the impact of pipelining on an idle pipeline stage without clock-gating. The following section discusses the effects of clock-gating.

The following equations model idle power with no clock-gating mechanism as simply the sum of the switching power of the timing elements and the total leakage power. Because of the exponential dependency of leakage current on V_{th} as represented in the $e^{\frac{-V_{th}}{n v_T}}$ term, P_{idle} approximately follows the switching power of the timing elements when V_{th} is high and follows the total leakage power when V_{th} is low.

$$P_{idle} = \frac{b_1}{N} V_{dd}^2 + (c_0 + \frac{c_1}{N}) V_{dd} e^{\frac{-V_{th} + \eta V_{dd}}{n v_T}} \quad (14)$$

$$P_{idle} = b_1 a_0^2 \frac{1}{N} (N + \frac{a_1}{a_0})^2 + \quad (15)$$

$$c_0 a_0 e^{\frac{-V_{th}}{n v_T}} (1 + \frac{c_1}{c_0} \frac{1}{N})(N + \frac{a_1}{a_0}) e^{\frac{\eta_0 a_0}{n v_T}(N + \frac{a_1}{a_0})} \quad (16)$$

When N is much greater than $\frac{a_1}{a_0}$ and $\frac{c_1}{c_0}$, P_{idle} becomes proportional to the product of N and the exponential function of N or just proportional to N, depending on V_{th}:

$$P_{idle}(HighV_{th}) \approx b_1 a_0^2 N \quad (17)$$

$$P_{idle}(LowV_{th}) \approx c_0 a_0 e^{\frac{-V_{th}}{n v_T}} N e^{\frac{\eta a_0}{n v_T} N} \quad (18)$$

When V_{th} is high, P_{idle} shows a linear reduction as N decreases, which is slower than a quadratic reduction as in switching power or a super-linear reduction as in leakage power. Thus, we can expect that idle power savings from pipelining are lower than those of switching and leakage power saving when V_{th} is high.

On the other hand, if N is much smaller than $\frac{a_1}{a_0}$ and $\frac{c_1}{c_0}$, P_{idle} becomes inversely proportional to N:

$$P_{idle}(HighV_{th}) \approx b_1 a_1^2 \frac{1}{N} \quad (19)$$

$$P_{idle}(LowV_{th}) \approx c_1 a_1 e^{\frac{-V_{th}}{n v_T}} \frac{1}{N} e^{\frac{\eta a_1}{n v_T}} \quad (20)$$

Figure 4 shows the simulated idle power without clock-gating, varying the number of FO4 inverters per pipeline stage. Optimal logic depth N^* was 8, which is greater than the optimal logic depths for switching and leakage power. Also, optimal power saving was smaller (50 to 70%) compared to the switching and leakage power cases. For idle stages, the overhead of timing elements is more significant compared to active stages. The graphs show that lower threshold voltages gives more idle power saving and slightly lower optimal logic depth.

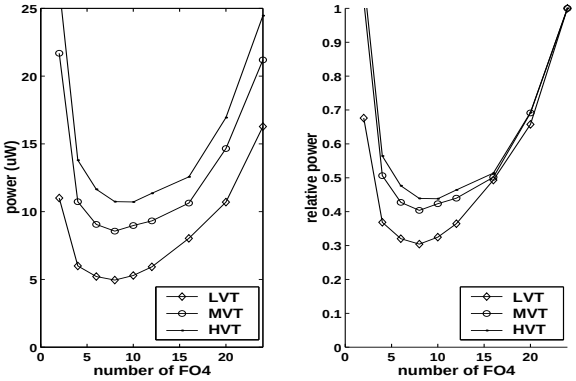

Figure 5: Idle power scaling.

6. RESULTS

In this section, we combine the results for the individual power components to calculate optimal logic depths and optimal power savings for different operating regimes including threshold voltage, activity factor, and presence of clock-gating. Power-optimal pipelining varies depending on activity factor and V_{th} because these change the proportion of switching power and leakage power (or idle power with no clock-gating mechanism), and each impacts pipelining power differently as seen in Section 5. This section is divided into two parts: the first part details the case when there is a clock-gating mechanism for pipeline stages and the second part considers the case without clock-gating.

6.1 Case 1: Clock-Gating Present

Figure 6 shows the simulated total power when a clock-gating mechanism is present for different activity factors. With a low activity factor, total power curves follow leakage power curves and high V_{th} leads to more power saving by pipelining. As the activity factor increases, total power curves follow switching power curves and high V_{th} leads to less power saving by pipelining.

Figure 7 shows the simulated optimal total power saving when a clock-gating mechanism is present. With zero activity factor, optimal power savings compared to a 24 FO4 design vary from 70

to 75% depending on V_{th}. Since switching power savings from pipelining are less dependent upon V_{th}, optimal power savings reach around 80% regardless of V_{th} as activity factor increases.

Because both switching power and leakage power are minimized when N is 6 as seen in Section 5.1 and Section 5.2, optimal logic depth was found to be 6 regardless of activity factor or threshold voltage when a clock-gating mechanism is present. However, as seen in Figure 3 and Figure 4, both switching and leakage power curves are quite flat around the optimum and power saving by pipelining is quite insensitive to modest deviations from the optimum. Therefore, 8 FO4 delays per stage might be a better choice since it simplifies design complexity with a small loss of power saving.

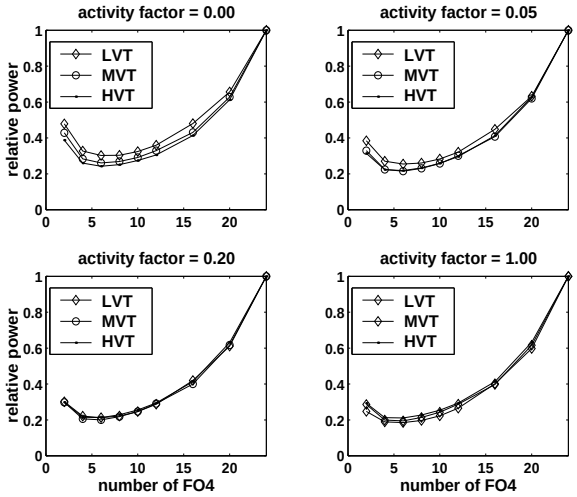

Figure 6: Total power scaling with a clock-gating mechanism.

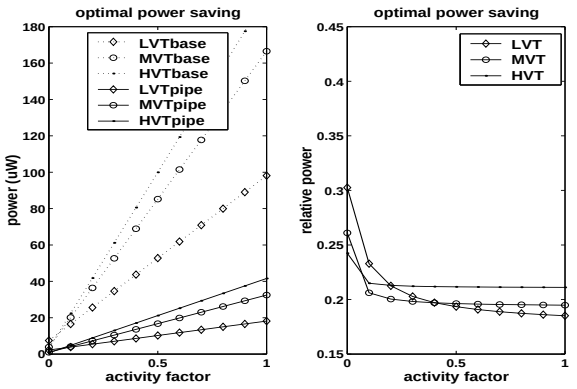

Figure 7: Optimal power saving with a clock-gating mechanism.

6.2 Case 2: No Clock-Gating Present

Figure 8 shows the simulated total power without clock-gating for different activity factors. With a low activity factor, total power curves follow idle power curves and low V_{th} leads to more power saving (Section 5.3). As the activity factor increases, total power curves follow switching power curves.

Figure 9 shows the simulated optimal total power saving when there is no clock-gating mechanism. With zero activity factor, optimal power savings are around 5 to 15% less than the clock-gating present case because of the timing element switching power overhead which is not present when there is a clock-gating scheme. Optimal power savings reach 80% slowly as activity factor increases compared to the clock-gated case. It is noted that low V_{th} gets the most power saving regardless of activity factor.

Figure 10 shows the optimal logic depths when clock is not gated for different threshold voltages. Because the idle power is minimized when N is 8 (Section 5.3), optimal logic depths remain at 8 until activity factor reaches around 0.2 (0.3 at high V_{th}) and after 0.2 (0.3 at high V_{th}), it falls to 6.

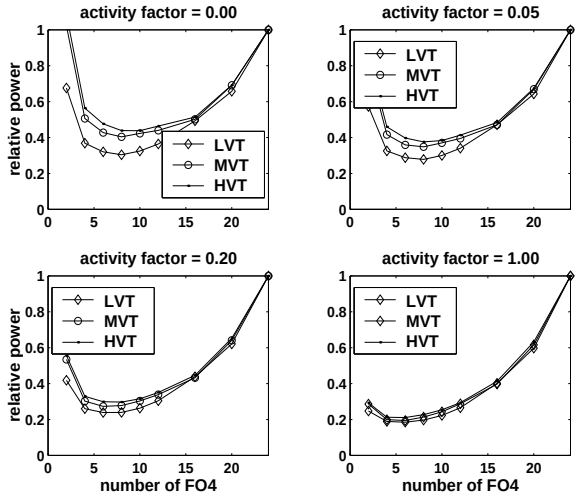

Figure 8: Total power scaling with no clock-gating mechanism.

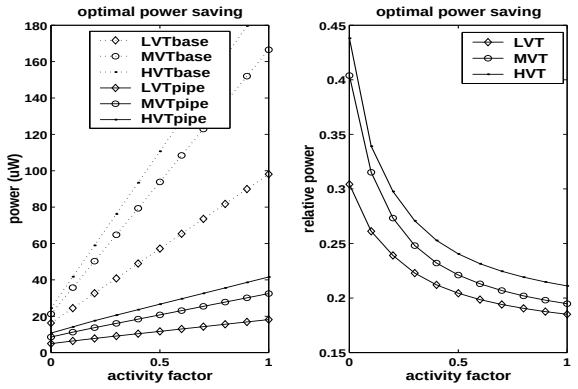

Figure 9: Optimal power saving with no clock-gating mechanism.

7. DISCUSSION

Our study has a number of limitations. The number of latches was assumed to grow linearly with the number of pipeline stages, whereas previous authors have used a superlinear latch count scaling formula of the form N^{η}, with an exponent $\eta \approx 1.1$ [12, 6]. It is not clear how latch counts scale in highly parallel architectures, but larger values of η would increase the optimal logic depth.

Depending on the computation being parallelized, additional state in the form of larger memory arrays might be required to track the increased number of operations in flight. A growth in the size of these memory structures would tend to increase energy per operation and hence increase optimal logic depth per stage, though

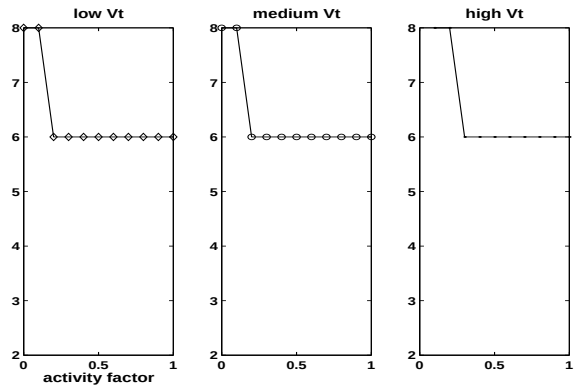

Figure 10: Optimal logic depth with no clock-gating mechanism.

we expect this effect to be minor as memories are generally lower power than processing units.

Our study did not include the effects of glitching on power. Others have noted that glitching activity reduces linearly with pipeline depth as it becomes less likely that inputs to a gate would have very different path lengths [12]. This effect would tend to push the optimum towards shallower pipeline stages.

We did not include parasitic wire capacitance. Adding wire load capacitance to our model will increase total switching power, and so will again push the optimum towards shallower pipeline stages.

For deeply pipelined circuits, fast path problems are more likely, as there will be an increase in the number of short logic paths between timing elements and an decrease in the relative wire delay. Because clock frequency is not increased, clock skew and jitter problems are not as apparent as in a frequency-scaled design, but clock jitter might increase as power supply to the clock drivers is reduced.

One benefit of supply scale-down is that wire delay becomes relatively less significant as gates slow down. This helps reduce some of the design effort of building a highly pipelined circuit compared with pipelining for increased clock frequency.

8. CONCLUSIONS

Pipelining can be an effective power-reduction tool when used to support voltage scaling in digital systems implementing highly parallel computations. Simulation results show that power-optimal logic depth is 6 to 8 FO4 and optimal power saving varies from 55 to 80% compared with a 24 FO4 design depending on threshold voltage, activity factor, and the presence of clock-gating.

Even though the exact power-optimal pipelining is technology-dependent, we can gain some important insights from the simulation results. First, higher activity factors decrease the power-optimal logic depth and increase the optimal power saving because pipelining is most effective at saving the additional switching power. Second, pipelining is more effective with lower threshold voltages, resulting in lower logic depths and lowest power, except for low activity factors when leakage power is dominant. Third, clock-gating enables deeper pipelining and more power saving because it reduces timing element overhead when activity factor is low.

Therefore, power-optimal pipelining with clock gating should be an efficient low-power technique for high throughput blocks in systems implementing highly parallel computations.

9. REFERENCES

[1] A. Chandrakasan, W. J. Bowhill, and F. Fox. *Design of High Performance Microprocessor Circuits*. IEEE Press, 2000.

[2] A. Chandrakasan et al. Low-power CMOS digital design. *IEEE JSSC*, 27(4):473–484, Apr. 1992.

[3] V. De and S. Borkar. Technology and design challenges for low power and high performance. In *ISLPED*, pages 163–168, 1999.

[4] Device Group at UC Berkeley. Predictive technology model. Technical report, UC Berkeley, 2001. http://www-device.eecs.berkely.edu/ ptm/.

[5] A. Hartstein and T. Puzak. The optimum pipeline depth for a microprocessor. In *ISCA 29*, pages 7–13, May 2002.

[6] A. Hartstein and T. Puzak. Optimum power/performance pipeline depth. In *MICRO*, Dec. 2003.

[7] S. Heo, R. Krashinsky, and K. Asanović. Activity-sensitive flip-flop and latch selection for reduced energy. In *19th Conference on Advanced Research in VLSI*, Salt Lake City,UT USA, March 2001.

[8] G. Hinton et al. A 0.18 μm CMOS IA-32 processor with a 4-GHz integer execution unit. *IEEE JSSC*, 36(11):1617–1627, Nov. 2001.

[9] M. Hrishikesh et al. The optimal logic depth per pipeline stage is 6 to 8 FO4 inverter delays. In *ISCA 29*, pages 14–24, May 2002.

[10] S. R. Kunkel and J. E. Smith. Optimal pipelining in supercomputers. In *Proceedings 13th Symposium on Computer Architecture*, pages 404–414, Tokyo, Japan, June 1986.

[11] E. Sprangle and D. Carmean. Increasing processor performance by implementing deeper pipelines. In *ISCA 29*, pages 25–36, May 2002.

[12] V. Srinivasan et al. Optimizing pipelines for power and performance. In *MICRO*, Nov. 2002.