Closing the Power Gap between ASIC and Custom: An ASIC Perspective

D. G. Chinnery and K. Keutzer

Department of Electrical Engineering and Computer Sciences
University of California at Berkeley

{chinnery,keutzer}@eecs.berkeley.edu

ABSTRACT

We investigate differences in power between application-specific integrated circuits (ASICs) and custom integrated circuits, with examples from 0.6um to 0.13um CMOS. A variety of factors cause synthesizable designs to consume ×3 to ×7 more power. We discuss the shortcomings of typical synthesis flows, and changes to tools and standard cell libraries needed to reduce power. Using these methods, we believe that the power gap between ASICs and custom circuits can be closed to within ×2.

Categories and Subject Descriptors

B.7.0 [**Integrated Circuits**]: General.

General Terms

Design, performance.

Keywords

ASIC, comparison, custom, energy, power, standard cell.

1. INTRODUCTION

Here we use ASIC to refer to a circuit produced by an ASIC design flow including register transfer level (RTL) synthesis and automated place and route. Automation reduces design time, but the resulting circuitry and fabrication process may not be optimal. Custom designers can optimize the individual logic cells, the layout and wiring between the cells, and other aspects of the design.

In the same technology generation, custom designs can be ×3 to ×8 faster than ASICs generated from RTL [5]. Many of the same custom techniques used to achieve high speed can also be used to achieve low power [16].

Low power consumption is essential for embedded applications. Power affects battery life, and power dissipation is limited by the packaging. Passive cooling is often required, as using a heat sink and/or fan is larger and more expensive. Power is also becoming a design constraint for high end applications due to reliability, and electricity and cooling costs. As technology scales, power density has increased with transistor density, and leakage power is a significant issue even for high end processors.

In Section 2, we illustrate the power gap of ×3 to ×7 between ASIC and custom designs. To date the contribution of various factors to this gap has been unclear. While automated design flows are often blamed for poor speed and energy efficiency (throughput/unit power), process technology is also significant. Section 3 discusses the components of power consumption. Section 4 outlines factors contributing to the power gap. We then examine each factor, describing the differences between custom and ASIC design methodologies, and account for a factor's impact on power. Finally, we detail approaches that can reduce this power gap.

2. ASIC AND CUSTOM COMPARISON

To illustrate the power gap, we examine custom and ASIC implementations of ARM processors and dedicated hardware to implement discrete cosine transform (DCT) and its inverse (IDCT). ARM processors are general purpose processors for embedded applications. ASICs often have dedicated functional blocks to achieve low power and high performance on specific applications. Media processing is a typical example where high speed and low power is required. JPEG and MPEG compression of pictures and video use DCT and IDCT. We discuss synthesizable and custom DCT and IDCT blocks, and show that a similar power gap exists.

2.1 ARM processors from 0.6 to 0.13um

We compare chips with full custom ARM processors,. soft, and hard ARM cores. Soft macros of RTL code may be sold as individual IP (intellectual property) blocks and are portable between fabrication processes. A hard macro is a design which has been optimized then fixed in a fabrication process. A hard macro may be custom, or it may be "hardened" from a soft core. A complete chip includes additional memory, I/O logic, etc.

Table 1 lists hard macro ASIC and custom implementations of ARM chips. Compared to the other designs, the three custom chips in bold achieved ×2 to ×3 millions of instructions per second per milliwatt (MIPS/mW) at similar MIPS. (The inverse, mW/MIPS, is energy per operation.) Dhrystone 2.1 MIPS benchmark is the performance metric. It fits in the cache of these designs, so there are no performance hits for cache misses or additional power to read off-chip memory.

Lower power was achieved in several ways. The DEC StrongARM used clock-gating and cache sub-banking to substantially reduce dynamic power [16]. The Intel XScale and DEC StrongARM used high speed logic styles to reduce critical path delay, at the price of higher power consumption on these paths. To reduce pipeline register delay, the StrongARM used pulse-triggered flip-flops [16] and the XScale used clock pulsed latches [6]. Shorter critical paths allow the same performance to be achieved with a lower supply voltage (V_{DD}), which can lower the total power consumption.

DAC 2005, June 13–17, 2005, Anaheim, California, USA.

Table 1. Full custom ARMs (in **bold**) have ×2 to ×3 MIPS/mW at similar MIPS, versus hard macro ARMs [3][10][11][14][15][22].

ARM	Process	Voltage	Frequency	MIPS	MIPS/mW
ARM710	0.60 um	5.0 V	40 MHz	36	0.08
Burd	0.60 um	1.2 V	5 MHz	6	1.85
Burd	0.60 um	3.8 V	80 MHz	85	0.18
ARM810	0.50 um	3.3 V	72 MHz	86	0.17
ARM910T	0.35 um	3.3 V	120 MHz	133	0.22
StrongARM	0.35 um	1.5 V	175 MHz	210	0.63
StrongARM	0.35 um	2.0 V	233 MHz	360	0.38
ARM920T	0.25 um	2.5 V	200 MHz	220	0.39
ARM1020E	0.18 um	1.5 V	400 MHz	500	1.25
XScale	0.18 um	1.0 V	400 MHz	510	3.39
XScale	0.18 um	1.8 V	1000 MHz	1250	0.78
ARM1020E	0.13 um	1.1 V	400 MHz	500	2.08

Table 2. ARM7TDMI hard cores are ×1.3 to ×1.4 MIPS/mW versus synthesizable ARM7TDMI-S soft cores. [1].

ARM Core	0.25 um		0.18 um		0.13 um	
(no cache, etc.)	MHz	MIPS/mW	MHz	MIPS/mW	MHz	MIPS/mW
ARM7TDMI	66	1.17	100	3.00	130	11.06
ARM7TDMI-S	60	0.83	90	2.28	120	8.33

Table 3. Comparison of ASIC and custom DCT/IDCT core power consumption at 30 frames/s for MPEG2. [9][32][33]

Design	Technology (um)	Voltage (V)	DCT (mW)	IDCT (mW)
ASIC	0.18	1.60	8.70	7.20
custom DCT	0.6 (L_{eff} 0.6)	1.56	4.38	
custom IDCT	0.7 (L_{eff} 0.5)	1.32		4.65

For the same technology and MIPS, the V_{DD} of full custom chips is lower than hard macros. The full custom chips can also operate at higher frequency with higher V_{DD}. If high performance wasn't required, the MIPS/mW would be even higher.

Energy consumption can be substantially reduced if performance is sacrificed. In Burd's 0.6um ARM8, the supply voltage was dynamically scaled with the processor load, in the range shown in Table 1. For MPEG and audio benchmarks, voltage scaling increased the energy efficiency by ×1.1 and ×4.5 respectively [3].

There is an additional factor of ×1.3 to ×1.4 between hard macro and synthesizable ARM7 soft cores, as shown in Table 2. These MIPS/mW are higher than those in Table 1, as they exclude caches and other essential units. The ARM7TDMI cores are also lower performance, and thus can achieve higher energy efficiency. Overall, there is a factor of ×3 to ×4 between synthesizable ARMs and the best custom ARM implementations.

2.2 A Comparison of IDCT/DCT cores

Application-specific circuits can reduce power by an order of magnitude compared to using general purpose hardware [24]. Two 0.18um ARM9 cores were required to decode 30 frames/s for MPEG2. They consumed 15× the power of a synthesizable DCT/IDCT design [9]. However, the synthesizable DCT/IDCT significantly lags its custom counterparts in energy efficiency.

Fanucci and Saponara designed a low power synthesizable DCT/IDCT core, using similar techniques to prior custom designs. Despite being three technology generations ahead, the synthesizable core was ×1.5 to ×2.0 higher power [9] (Table 3). Accounting for the technology difference by conservatively assuming power scales linearly with device dimensions, the gap is a factor of ×4.3 to ×6.6.

Table 4. Factors contributing to ASICs being higher power than custom. The excellent column is what ASICs may achieve using low power and high performance techniques. This table focuses on the total power when a circuit is active.

Contributing Factor	Typical	Excellent
microarchitecture	×2.6	×1.3
clock gating	×1.6	×1.0
logic design	×1.2	×1.0
high speed logic styles	×1.3	×1.3
technology mapping	×1.4	×1.0
cell sizing, wire sizing	×1.6	×1.1
voltage scaling, multi-vth, multi-vdd	×4.0	×1.0
floorplanning and placement	×1.5	×1.1
process variation and technology	×2.6	×1.2

3. COMPONENTS OF POWER

Designers typically focus on reducing both the total power when a circuit is active and its standby power. There is usually a minimum performance target, e.g. 30 frames/s for MPEG. When speed is less important, the energy per operation can be minimized. The active power is when logic evaluates. Static power is due to current leakage. In today's processes, leakage can account for 10% to 30% of the total power when a chip is active, and is dominant in standby.

Active power is due to switching capacitances (dynamic power), and short circuit power when there is a current path from supply to ground. Dynamic power increases quadratically with V_{DD}, and linearly with capacitance, switching activity and clock frequency. Short circuit power is typically about 10% of the active power, and increases with increasing V_{DD}, and with decreasing transistor threshold voltage V_{th}. Short circuit power can be reduced by matching input and output rise and fall times [30]. As dynamic power depends quadratically on V_{DD}, methods for reducing active power often focus on reducing V_{DD}. Reducing the capacitance by downsizing gates and reducing wire lengths is also important.

Static power in static CMOS logic is primarily due to subthreshold leakage, which increases exponentially with decreases in V_{th} and increases in temperature. It can also be strongly dependent on transistor channel length. Gate tunneling leakage is becoming significant as gate oxide thickness reduces with device dimensions.

4. FACTORS CAUSING THE POWER GAP

Various parts of the circuit design and fabrication process contribute to the gap between ASIC and custom power. Table 4 outlines our analysis of the most significant design factors and their impact on the total power when a chip is active. The "typical" column shows the maximum contribution of the factors. In total these factors can make power an order of magnitude worse. In practice, custom designs can't fully exploit all these factors simultaneously. Most low power EDA tools focus on reducing dynamic power in control logic, datapath logic, and the clock tree. The power consumed by memory is application dependent. The design cost for custom memory is low, because of the high regularity. Several companies provide custom memory for ASIC processes. Thus we do not focus on memory further.

Voltage scaling gives the largest potential for power reduction. If supply voltage can be halved, the dynamic power is reduced by ×4 (e.g. compare the two XScale's MIPS/mW in Table 1). Process technology can reduce leakage by more than an order of magnitude, and it also has a large impact on dynamic power.

Microarchitectural techniques such as pipelining and parallelism increase throughput, allowing gate downsizing and voltage scaling. The overheads for these techniques must be considered. Other factors in Table 4 have smaller contributions to the gap.

In the following sections, we examine the three largest factors in detail and overview the smaller factors. ASICs using the low power techniques that we recommend in these sections may close the gap to a factor of ×2 (the "excellent" column of Table 4).

5. MICROARCHITECTURE

Algorithmic and architectural choices can reduce the power by an order of magnitude [24]. ASIC and custom designers make similar algorithmic and architectural choices to find a low power implementation that is appropriate for the required performance and target application. With similar microarchitectures, how do ASIC and custom pipelining and parallelism compare?

On their own, pipelining and parallelism do not reduce power. Pipelining reduces the critical path delay, inserting registers between combinational logic. Glitches may not propagate through registers, but switching activity of combinational logic is otherwise unchanged. However, the clock signal to registers has high activity. Pipelining may reduce the IPC (instructions per cycle), due to branch misprediction and other hazards; in turn this reduces the energy efficiency. Parallelism trades off area for increased throughput, with overheads for multiplexing and more wiring. Both techniques enable the same performance to be met at lower V_{DD} with smaller gate sizes, reducing the power.

Overheads for pipelining include register delay, register setup time, clock skew, clock jitter, and any imbalance in pipeline stage delays that cannot be compensated for by slack passing or cycle stealing. This overhead reduces the clock frequency and the energy efficiency. For a given delay constraint, it reduces the slack available to perform for downsizing and voltage scaling.

5.1 What's the problem?

In the IDCT, the cost of pipelining was about a 20% increase in total power, but pipelining reduced the critical path length by ×4. For the same performance without pipelining, V_{DD} would have to be increased from 1.32V to 2.20V. Thus pipelining increased energy efficiency by about ×2 [33].

Most ASICs use slow D-type flip-flops for pipeline registers. The StrongARM used fast pulse-triggered flip-flops [16]. The XScale used clock-pulsed transparent latches. A clock-pulsed latch has smaller clock load and is faster than a D-type flip-flop. This reduced the clock power by 33%. Clock-pulsed latches have increased hold time and thus more problems with races. The pulse width had to be carefully controlled and buffers were inserted to prevent races. The clock duty cycle also needs to be carefully balanced [6]. Distribution of a duty cycle balanced clock signal with clock pulse generation requires manual clock tree design.

Comparing ASIC and custom microarchitecure, ASICs may lag custom speed by up to ×1.8 [5]. If the delay constraint is tight, a little extra slack can provide substantial power savings from downsizing gates. To estimate the impact of ASIC pipelining overhead and worse IPC[1], we used a general model for the pipeline delay and power consumption versus the number of pipeline stages [13]. We augmented this with models of power reduction achieved by downsizing and voltage scaling versus slack. From these models, ASICs can consume ×2.6 the energy per operation compared to custom designs at a tight delay constraint. Of this, a factor of ×1.2 is due to worse IPC for a typical ASIC. The remaining ×2.2 increase in energy per operation is because less timing slack is available for gate downsizing and voltage scaling.

5.2 What can we do about it?

Bhavnagarwala et al. predict a ×2 to ×4 reduction in power with voltage scaling by using 2 to 4 parallel datapaths. As the ratio of V_{DD} to V_{th} decreases, the performance penalty for low V_{DD} is higher, which reduces the energy savings [2]. Generally, ASICs can make full use of parallelism, but careful layout is required to minimize additional wiring and control overheads.

High-speed flip-flops are available in some standard cell libraries. ASICs can also use cycle stealing or latches to reduce the pipelining overhead per stage to as low as 5 FO4 delays [5]. This enables more slack to be used for downsizing, voltage scaling, or increasing the clock frequency. From our pipeline model, ASICs can close the gap for this factor to within ×1.3 of custom.

6. CLOCK GATING AND SLEEP MODE

There are tools for analyzing clock-tree power. These tools help designers identify architectural signals to gate (cut off) the clock signal to logic when it is not in use. Some tools also support automated clock gating. Clock gating can substantially reduce the dynamic power in the clock tree and registers. In the synthesizable DCT/IDCT, clock gating and data driven switching activity reduction increased the energy efficiency by ×1.4 for DCT and ×1.6 for IDCT [9].

Similar signals can be used with techniques to reduce leakage power in idle units by an order of magnitude. Sleep transistors can "power gate" the supply to ground leakage path with a high resistance. The substrate voltage can be changed to reverse bias leaky transistors. Input states can be assigned to limit the number of high leakage current paths. Using sleep transistors and low leakage state assignment are not currently supported by EDA tools. Standard cell libraries need to have leakage characterized at different supply and substrate voltages.

7. LOGIC DESIGN

Logic design refers to the topology and logic structure used to implement datapath elements such as adders and multipliers. Arithmetic structures have different power and delay trade-offs for different logic styles, technologies, and input probabilities.

Specifying the logic design requires carefully structured RTL and tight synthesis constraints. Hierarchical synthesis may be needed to avoid the structure being changed by synthesis "optimizations". Synthesis tools can also compile to arithmetic modules, with power and delay on par with tightly structured RTL.

Careful analysis is needed to compare alternate algorithmic implementations for different speed constraints. High-level activity analysis showed that a 32-bit carry lookahead adder had 43% lower energy than carry bypass or carry select adders, and there was a 15% energy difference between 32-bit multipliers [4].

8. LOGIC STYLE

ASICs almost exclusively use static CMOS for combinational logic, because it is more robust to noise and V_{DD} variation. However, pass transistor logic, dynamic domino logic and

[1] Pipelining overheads were: timing overhead of 10 FO4 delays and imbalance of 10 FO4 delays for ASICs (15% of clock period); vs. 2.6 FO4 delays total for custom designs with slack passing. The CPI (1/IPC) penalty was 0.025 per pipeline stage for custom, and 0.05 per stage for ASICs. From data in [5].

differential cascode voltage switch logic are faster than static CMOS logic. These high speed logic styles can increase the speed of combinational logic by ×1.5 [5]. From our pipeline models, we estimate that this can increase energy efficiency by ×1.3 at high performance targets. Static CMOS is lower energy than other logic styles when high performance is not required.

Using these high speed logic styles requires careful cell design and layout, but a typical EDA flow gives poor control over the layout. It is not viable to use high speed logic styles in ASICs.

9. TECHNOLOGY MAPPING

In technology mapping a logical netlist is mapped to a standard cell library in a given technology. Different combinations of cells can be used to implement a gate with different activity, capacitance, power and delay. For example, an AO22 with inverters can be used to implement a smaller and lower power XOR2, but it is slower. Power minimization subject to delay constraints is not yet supported in the initial technology mapping phase. Minimizing total cell area minimizes capacitance, but it can increase activity. For a 0.13um 32-bit multiplier, we found that the power was ×1.32 higher when using minimum area mapping instead of minimum delay. This was due to more (small) cells being used, increasing activity. Given switching activity information, technology mapping for low power should achieve better results, and it is not otherwise substantially more difficult than minimum area mapping. After the initial technology mapping, power minimization tools can do limited remapping and pin reassignment, along with clock gating and gate sizing [27].

At a given delay constraint, technology mapping can reduce power by 10% to 20%, for about a 10% to 20% increase in area [17][24]. Logic transformations based on controllability and observability relationships, common sub-expression elimination, and technology decomposition can give additional power savings of 10% to 20% [20][24]. Overall, automated technology mapping techniques for low power may be able to increase energy efficiency by up to ×1.4.

10. CELL SIZING AND WIRE SIZING

Wires and transistors should be sized optimally to meet timing constraints and reduce switching capacitance. ASICs must choose cell sizes from the range in the standard cell library. ASIC wire widths are usually fixed. To balance rise and fall delays, standard cells have P:N width ratio of about 2:1. To reduce power with smaller PMOS transistor capacitances, a ratio of as low as 1.5:1 may be better. Moreover, sometimes the rise and fall drive strengths needed are different. Custom libraries may be finer grained, which avoids over-sizing gates, and have skewed drive strengths. Specific cell instances can be optimized. Cells connecting nearby don't need buffering to guard band long wires. For synthesizable DSP (digital signal processor) modules, a fine grained library improved energy efficiency by ×1.4 [21]. In place cell optimization increased energy efficiency by ×1.4 for a design that had used a rich library [7].

Wire sizing can be automated, but is not currently supported by EDA tools, except for the clock tree. Gong et al. optimized clock buffers and wire sizes to reduce clock tree power by 63% [12].

Reducing the performance target can provide energy savings by gate downsizing. We synthesized a small embedded processor in 0.13um. The power/MHz was 43% lower at 100MHz than 400MHz, due to sizing. At 325MHz, power minimization with Design Compiler [27] was able to increase energy efficiency by ×1.35 with no delay penalty. Our sizing optimization results, with linear programming on combinational gate-level net lists, indicate that it may be possible to further reduce power 10% to 16% on average compared to Design Compiler.

11. VOLTAGE SCALING

Reducing the supply voltage V_{DD} quadratically reduces dynamic power. Short circuit power also decreases with V_{DD}. As V_{DD} decreases, a gate's delay increases. To reduce delay, threshold voltage V_{th} must also be scaled down. As V_{th} decreases, leakage increases exponentially. Thus there is a tradeoff between performance, dynamic power and leakage power.

11.1 What's the problem?

Custom designs can achieve at least ×2 speed compared to ASICs [5]. At the same performance target, custom designs can reach lower V_{DD} using the additional slack. Compare V_{DD} of the Burd, StrongARM and XScale chips to other ARMs in Table 1 – with lower V_{DD}, they save between 40% and 80% dynamic power. This is the primary reason for their higher energy efficiency. To use lower V_{DD}, ASICs must settle for lower performance or use high speed techniques to maintain performance.

Using low V_{DD} requires low V_{th}. The process technology determines V_{th}. A foundry has typically two or three libraries with different V_{th}: high V_{th} for low power; and low V_{th} for high speed at the expense of significant leakage power. Most ASIC designers cannot ask to fine tune V_{th} for their particular design. V_{DD} can be optimized for ASICs, but typical ASIC libraries are characterized at only two nominal supply voltages – say 1.2V and 0.9V in 0.13um. To use V_{DD} of 0.6V, the library must be re-characterized

11.2 What can we do about it?

Library characterization tools exist. Characterization can take several days or more for a large library. Standard cell library vendors can help by providing more V_{DD} characterization points.

Foundries often support high and low V_{th} cells being used on the same chip. Power minimization tools can reduce power by using low V_{th} cells on the critical path, with high V_{th} cells elsewhere to reduce leakage. Combining dual V_{th} with sizing reduces leakage by ×3 to ×6 versus using only low V_{th} [25].

Dual supply voltages can also be used. High V_{DD} is used on the critical path for performance, with low V_{DD} elsewhere to reduce active power. Dual V_{DD} requires tool support to cluster cells of the same V_{DD} to achieve reasonable layout density. Commercial tools do not adequately support dual V_{DD} assignment or layout, but separate voltage islands are possible. Voltage level converters are also required to prevent static current when a low V_{DD} cell drives a high V_{DD} cell. Level converters are not available in ASIC libraries. Usami et al. implemented automated tools to assign dual V_{DD} and place dual V_{DD} cells, with substrate biasing to lower V_{th} in active mode. They achieved total power reduction of 58% (×2.4 energy efficiency), with only a 5% increase in area [29].

12. FLOORPLANNING AND PLACEMENT

The power consumption due to interconnect has increased from about 20% in 0.25um to 40% in 0.09um [26]. Wire lengths depend on cell placement and congestion. Larger cells and additional buffers are needed to drive long wires.

Custom chips are partitioned into small, manually placed blocks of logic, reducing the wiring. Automatic place and route tools are not good at recognizing layout regularity in datapaths. An ASIC designer can generate bit slices from carefully coded RTL with tight aspect ratio placement constraints. Bit slices of layout may then be composed. We used BACPAC [26] to compare partitioning designs into blocks of 50,000 or 200,000 gates in 0.13um, 0.18um, and 0.25um. Using larger partitions increased average wire length by about 42% and delay by 20%, corresponding to about a 20% increase in total power and ×1.4 worse energy overall.

A conservative wire load model is required to meet delay constraints, but the result is gates being over sized [5], increasing the power. Physical synthesis should be used to refine wire length estimates and cell placement in an iterative manner. In our experience, physical synthesis can increase speed by 15% to 25%. The cell density increases, reducing wire lengths, and then cells may be downsized, which reduces power by 10% to 20%.

13. PROCESS VARIATION AND TECHNOLOGY

Within the same nominal technology generation, the active power, leakage power, and speed of a chip differ substantially depending on the actual process technology. Furthermore, the fabricated chips vary in power and speed due to process variation.

There are a number of sources of process variation within a plant, such as optical proximity effects, and wafer defects. The channel length L, transistor width, wire width and wire height have about 25% to 35% variation from nominal at three standard deviations (3σ). Threshold voltage V_{th} and oxide thickness have about 10% variation at 3σ. [19] A decrease in V_{th} or L can cause a large increase in leakage current, though such transistors are faster. Dynamic power scales linearly with transistor and wire dimensions, as capacitances increase.

To ensure high yield accounting for process variation, libraries are usually characterized at two points. To meet the target speed, the process' worst case speed corner is used – typically 125°C, 90% of nominal V_{DD}, with slow transistors. To prevent excessive power, the active power may be characterized at a worst case power corner, e.g. -40°C, 110% of nominal V_{DD}, and fast transistors. Leakage is worse at high temperature. Due to V_{DD} alone, the active power is 50% higher at the worst case power corner than at the worst case speed corner. These process corners are quite conservative and limit a design. The fastest chips fabricated in a typical process may be 60% faster than estimated from the worst case speed corner [5]. Similarly, when we examine the distribution of power of fabricated 0.3um MPEG4 codecs [28], the worst case power may be 50% to 75% higher than the lowest power chips produced.

We analyzed data from Intel and AMD chips [8]. After accounting for clock frequency and V_{DD}, the chips in high speed bins have about 10% to 20% lower energy than those in low speed bins. We estimate that the worst case power corner is ×1.2 to ×1.3 higher in power than a point with reasonable yield. Overall, high speed bin chips may have up to ×1.6 higher energy efficiency than ASICs at the worst case process corner estimates.

Within a technology generation, available processes can differ by up to 25% in speed [5]. We compared several gates in Virtual Silicon's IBM 8SF and UMC L130HS 0.13um libraries. 8SF has about 5% less delay and only 5% of the leakage compared to L130HS, but it has ×1.6 higher dynamic power [31]. Our study of TSMC 0.13um libraries with an embedded processor showed that their high V_{th}, low-k library was 20% lower power/MHz (66% less leakage, 14% less active power) than the low V_{th}, low-k library.

Low-k inter-layer dielectric insulators reduce wiring capacitance. Low-k dielectrics of 2.7 to 3.6 electrical permittivity (k) are used in different processes. Using low k dielectric reduces interconnect capacitance by 25%, reducing total power by about 5% to 10%.

Narendra et al. showed that silicon-on-insulator (SOI) was 14% to 28% faster than bulk CMOS for some 0.18um gates. The total power was 30% lower at the same delay, but the leakage power was ×1.2 to ×20 larger [18]. A 0.5um DSP study showed that SOI was 35% lower power at the same delay as bulk CMOS [23]. Double-gated fully depleted SOI is less leaky than bulk CMOS.

In the StrongARM, caches occupied 90% of the chip area and were primarily responsible for leakage. A 12% increase in the NMOS channel length L reduced worst case leakage by a factor of 20. Lengthening transistors in the cache and other devices reduced total leakage by ×5 [16]. This approach can be applied to ASICs, if such library cells are available.

We estimate that different process choices may give up to a factor of ×1.6 difference in power. Combined with the impact of process variation, process can contribute a power gap of ×2.6.

13.1 What's the problem?

ASICs must be characterized under worst case process conditions to guarantee good yield. ASIC parts are often sold for a few dollars per chip, which makes additional testing for speed binning too expensive. Thus ASIC power and speed are limited by the worst case parts. Without binning, there is an energy efficiency gap of ×1.2 versus custom chips that are binned.

Standard cells are characterized in a specific process. The cells must be modified and libraries updated for ASIC customers to take advantage of process improvements. Finding the lowest power for an ASIC requires synthesis with several different libraries comparing power at performance targets of interest. The lowest power library and process may be too expensive.

13.2 What can we do about it?

Generally, it requires little extra work to re-target an ASIC EDA flow to a different library. ASICs can be migrated quickly to different technology generations, and updated for process improvements. In contrast, the design time to migrate custom chips is large. ASICs should be able to take full advantage of process improvements.

To account for process variation, ASIC power may be characterized after fabrication. Parts may then be advertised with longer battery life. However, post-fabrication characterization of chip samples does not solve the problem if there is a maximum power constraint on a design. In this case, ASICs may be characterized at a less conservative power corner, which requires better characterization of yield for the standard cell library in that process. For typical applications, the power consumption is substantially less than peak power at the worst case power corner. Additional steps may be taken to limit peak power, such as monitoring chip temperature and powering down if it is excessive.

14. SUMMARY AND CONCLUSIONS

We compared synthesizable and custom ARM processors from 0.6um to 0.13um. We also examined discrete cosine transform cores, as an example of low power functional units. There was a power gap of ×3 to ×7 between these custom and ASIC designs.

We have given a top-down view of the factors contributing to the power gap between ASIC and custom designs. From our analysis, the most significant combination of factors is using micro-architectural techniques with voltage scaling. Reducing the register delays and using pipelining to increase slack can enable substantial power savings by reducing the supply voltage and downsizing gates. Multiple threshold voltages may be used to limit leakage while enabling a lower V_{DD}. Choosing a low power process technology and limiting the impact of process variation reduces power by a large factor. In summary, we believe that the power gap can be closed to within a factor of ×2 by using these techniques together with fine granularity standard cell libraries, careful RTL design and EDA tools targeting low power. The remaining gap is mostly from custom designs having lower pipelining overhead and using high speed logic on critical paths.

We have focused on circuit design and synthesis as a whole, with energy efficiency as a design driver. ASICs may be unable to meet the performance requirements for some high speed applications. However, as technology continues to scale down, ASICs can achieve higher speeds at lower power. We hope to encourage EDA tool developers to enable this path: to help ASICs achieve low power, and to help low power custom designers reduce design time.

15. REFERENCES

[1] ARM, ARM Processor Cores. http://www.armdevzone.com/open.nsf/htmlall/A944EB65693A4EB180256A440051457A/$File/ARM+cores+111-1.pdf

[2] A. Bhavnagarwala, et al., "A Minimum Total Power Methodology for Projecting Limits on CMOS GSI," *IEEE Trans. VLSI Systems*, vol. 8, no. 3, June 2000, pp. 235-251.

[3] T. Burd, et al., "A Dynamic Voltage Scaled Microprocessor System," in *Proc. Int. Solid-State Circuits Conf.*, vol. 35, no. 11, 2000, pp. 1571-80.

[4] T. Callaway, and E. Swartzlander, "Optimizing Arithmetic Elements for Signal Processing," *IEEE VLSI Signal Processing Workshop*, 1992, pp. 91-100.

[5] D. Chinnery, and K. Keutzer, *Closing the Gap Between ASIC & Custom*, Kluwer, 2002.

[6] L. Clark, et al., "An Embedded 32-b Microprocessor Core for Low-Power and High-Performance Applications," *J. Solid-State Circuits*, vol. 36, no. 11, Nov. 2001, pp. 1599-1608.

[7] M. Cote, and P. Hurat, "Faster and Lower Power Cell-Based Designs with Transistor-Level Cell Sizing," chapter 9 in *Closing the Gap Between ASIC & Custom*, Kluwer, 2002.

[8] CPU Scorecard, Intel CPU Roster and AMD CPU Roster. http://www.cpuscorecard.com/cpuprices/

[9] L. Fanucci, and S. Saponara, "Data driven VLSI computation for low power DCT-based video coding," in *Proc. Int. Conf. Electronics, Circuits and Systems*, vol.2, 2002, pp. 541-4.

[10] S. Furber, *ARM System-on-Chip Architecture*. 2nd Ed. Addison-Wesley, 2000.

[11] J. Ganswijk, Chip Directory: ARM Processor family. http://www.xs4all.nl/~ganswijk/chipdir/fam/arm/

[12] J. Gong, et al., "Simultaneous buffer and wire sizing for performance and power optimization," in *Proc. Int. Symp. on Low Power Electronics and Design*, 1996, pp. 271-6.

[13] A. Harstein, and T. Puzak, "Optimum Power/Performance Pipeline Depth," in *Proc. Int. Symp. on Microarchitecture*, 2003, pp. 117-126.

[14] Intel, Intel XScale Microarchitecture: Benchmarks. http://developer.intel.com/design/intelxscale/benchmarks.htm

[15] M. Levy, "Samsung Twists ARM Past 1GHz," *Microprocessor Report*, Oct. 16, 2002.

[16] J. Montanaro, et al., "A 160MHz, 32-b, 0.5W, CMOS RISC Microprocessor*," J. Solid-State Circuits*, vol. 31, no. 11, 1996, pp. 1703-14.

[17] B. Moyer, "Low-Power Design for Embedded Processors," *Proc. IEEE*, vol. 89, no. 11, Nov. 2001, 1576-1587.

[18] S. Narendra, et al., "Comparative Performance, Leakage Power and Switching Power of Circuits in 150 nm PD-SOI and Bulk Technologies Including Impact of SOI History Effect," *Int. Symp. on VLSI Circuits*, 2001, pp. 217-8.

[19] S. Nassif, "Delay Variability: Sources, Impact and Trends," in *Proc. Int. Solid-State Circuits Conf.*, 2000.

[20] D. Pradhan, et al., "Gate-Level Synthesis for Low-Power Using New Transformations," in *Proc. Int. Symp. on Low Power Electronics and Design*, 1996, pp. 297-300.

[21] R. Puri et al., "Pushing ASIC Performance in a Power Envelope," in *Proc. Design Automation Conf.*, 2003, pp. 788-793.

[22] J. Quinn, *Processor98: A Study of the MPU, CPU and DSP Markets*, Micrologic Research, 1998.

[23] P. Simonen, et al., "Comparison of bulk and SOI CMOS Technologies in a DSP Processor Circuit Implementation," in *Proc. Int. Conf. Microelectronics*, 2001.

[24] D. Singh, et al., "Power Conscious CAD Tools and Methodologies: a Perspective," *Proc. IEEE*, vol. 83, no. 4, April 1995, pp. 570-94.

[25] S. Sirichotiyakul, et al., "Stand-by Power Minimization through Simultaneous Threshold Voltage Selection and Circuit Sizing," in *Proc. Design Automation Conf.*, 1999, pp. 436-41.

[26] D. Sylvester, and K. Keutzer, "Getting to the Bottom of Deep Submicron," *in Proc. Int. Conf. on Computer-Aided Design*, 1998, pp. 203-11.

[27] Synopsys, *Design Compiler User Guide*, 2003.

[28] M. Takahashi, et al., "A 60-mW MPEG4 Video Codec Using Clustered Voltage Scaling with Variable Supply-Voltage Scheme," *J. Solid-State Circuits*, vol. 33, no. 11, 1998, pp. 1772-1780.

[29] K. Usami, and M. Igarishi, "Low-Power Design Methodology and Applications Utilizing Dual Supply Voltages," *in Proc. ASP Design Automation Conf.*, 2000, pp. 123-8.

[30] H. Veendrick, "Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits," *J. Solid-State Circuits*, vol. SC-19, August 1984, pp. 468-73.

[31] Virtual Silicon. http://www.virtual-silicon.com/

[32] T. Xanthopoulos, and A. Chandrakasan, "A Low-Power DCT Core Using Adaptive Bitwidth and Arithmetic Activity Exploiting Signal Correlations and Quantization," *J. Solid-State Circuits*, vol. 35, no. 5, May 2000, pp. 740-50.

[33] T. Xanthopolous, and A. Chandrakasan, "A Low-Power IDCT Macrocell for MPEG-2 MP@ML Exploiting Data Distribution Properties for Minimal Activity," *J. Solid-State Circuits*, vol. 34, May 1999, pp. 693-703.