sponsored by

Gabe on EDA • EDAmarket

Assembling the Future

A Newsletter About the Design and Production of Electronics

ISSUE 019 • SEPTEMBER 2012

SUBSCRIBE

PDF and Archives

twitter

In this issue:

Optimizing for Low Power Prior to Silicon Availability

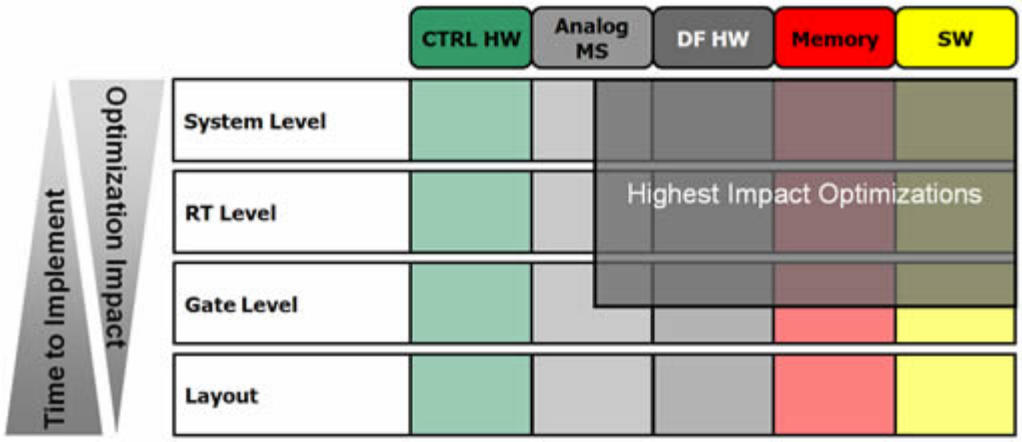
Frank Schirrmeister, Cadence Design Systems

Over the last decade, low power has become increasingly important for electronic design. It's a key driver for pure semiconductor hardware design as well as for software optimization and early hardware/software co-design. Design terminology itself has been updated with the term "PPA optimization," where PPA represents power, performance, and area. While performance (getting the design to work) and area (translating to cost) are still the main drivers, power has become a critical differentiator, especially in mobile designs and cost-sensitive designs where it is necessary to optimize packaging cost.

However, early optimization for low power is facing the classic system-level conundrum: the earlier a decision can be made, the higher the potential effect of the decision will be and the easier it is to implement, but because the decision is made early it has to be made based on limited or less reliable data. To address this dilemma, modern design flows have already changed to allow more frequent confirmation of decisions based on refined data once more detail becomes available. Decisions are made earlier and checked more often.

Figure 1 outlines the effect of early optimizations on the right-side of the graphic—the earlier that optimizations can be made, the bigger typically their impact will be. It shows the different areas of

chip-related optimizations based on their impact. It also illustrates different components of a hardware/software (HW/SW) design: analog/mixed-signal, control hardware, dataflow hardware, memories, and the software controlling a fair amount of functionality. Optimizations to the dataflow hardware—impacting the inherent switching in a datapath, for example—as well as memory accesses and software execution can have the biggest impact on power consumption in a design.



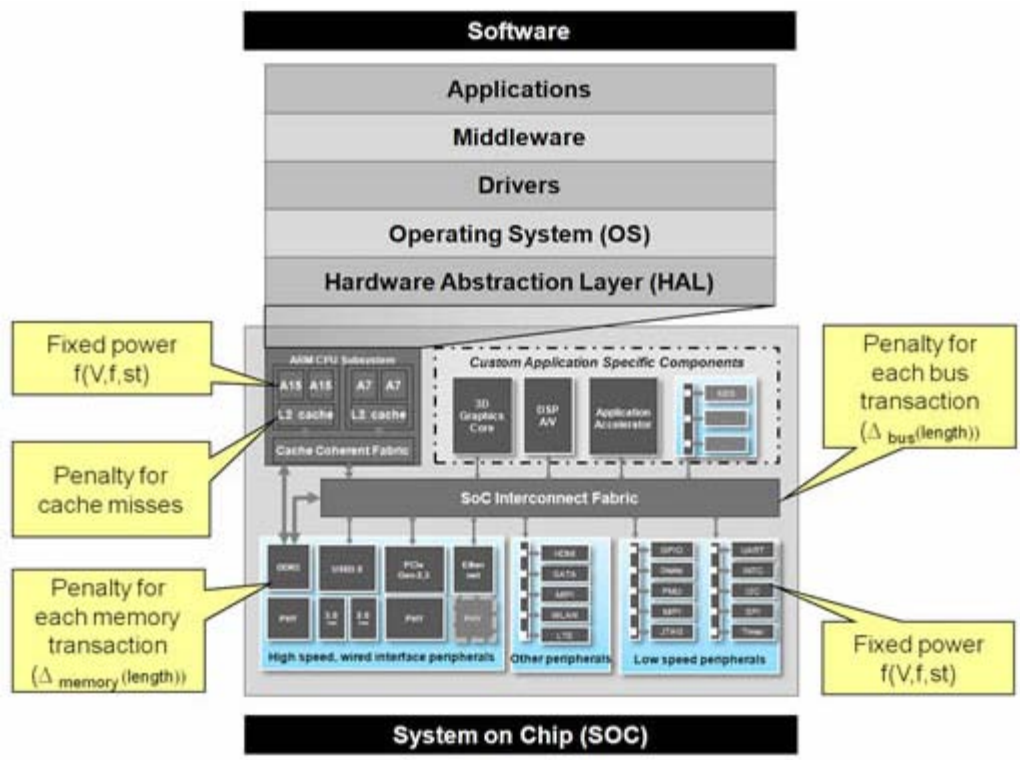
[Figure 1: Power optimization potential]

In the past, optimization focused on the lower levels of implementation by trying to optimize at the gate or register-transfer levels. While the gains have been considerable, attacking the problem at the system level can lead to much greater gains. In exchange, however, this creates new demands for modeling and analysis.

Most of today's attempts to reduce power consumption come from clock gating. This means turning off the clocks that are fed into components that are not actively being used. Given that large quantities of activity are associated with clocks, this activity is a waste of power if the only thing being accomplished is saving the state of registers that are not going to be read. With leakage power becoming a larger percentage of total power as chip geometries decrease, clock gating by itself is becoming a less effective strategy. Designers are now looking at completely powering down specific power domains and using variable voltages to tradeoff between performance and power consumption. Both of these schemes require a lot of additional analysis and are often controlled by software, which has to be written, verified, and debugged.

Power Analysis and Power-Aware Software Development at the Transaction Level

Performing system-level power analysis using virtual prototypes with transaction-level models (TLMs) early in the development cycle can allow power management policies to be tested, and enables software power optimizations to be assessed in a relative sense. Power figures will not be accurate enough to predict the exact amount of power that will be consumed, but will allow users to find out if a change would result in a relative savings. This is particularly true of a loosely timed (LT) TLM virtual prototype—at this level of timing detail, tools can only provide figures for the estimated power consumed per function. When the timing is refined down to the approximately timed (AT) level, tradeoffs are then possible between performance and power since the virtual prototype will now be able to provide profiles of power over time.



[Figure 2: Hardware/software system and example areas for which transaction-level power models can be used]

As indicated in Figure 2, components can be characterized by a set of power parameters that are then used in localized power equations to calculate the component power. These are then accumulated by a central logging module for analysis after the simulation run. Different types of components will have different parameters. For example, the parameters defined for a processor at the LT level are active power, dormant power, power inactive, and power shutdown. For a memory they would be power clock, power, idle, power read, and power write. More accurate models would have additional parameters such as voltage and frequency as well as those representing state.

The use of a virtual platform allows not only the relative assessment of low-power effects for HW/SW optimization, but also the early development of the required power management drivers that will help decide when certain portions of the design can be switched to a different power state (i.e. the Power Management Modeule). Failures of the software in this aspect of functionality tend to be critical; if a block is not powered-up in time for when it is expected to be used, a total failure could result.

Due to the high level of modeling of TLM-based virtual platforms, the mechanical aspects of annotating the parameters (as indicated in Figure 2) is in itself not difficult, despite the lack of standardized APIs today. The challenge lies in getting the right data to be annotated. Users today will simply run analysis based on power estimates, measurements from previous designs, or measurements based on refined data as it becomes available while the design flow progresses.

Power Analysis at the Register Transfer and Gate Levels

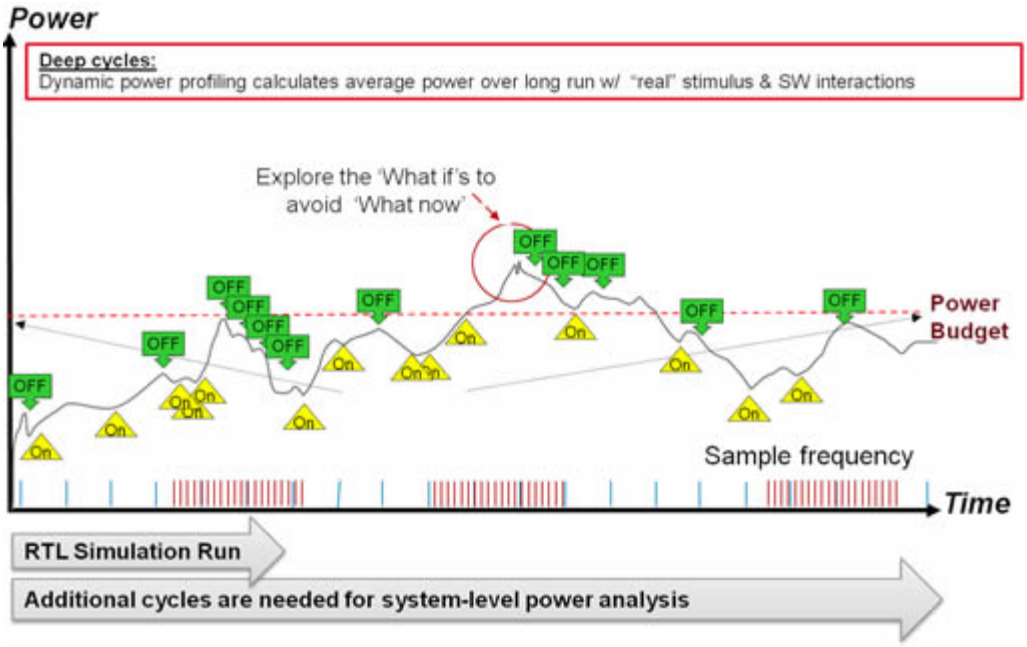
Once a design has been refined to the register-transfer level (RTL), more detailed power analysis becomes possible, which can be even further refined at the gate level. Two big challenges have to

be addressed: efficiently articulating the actual low-power intent to drive analysis, and effectively collecting the right amount of toggle information during execution to understand low-power effects.

Figure 3 illustrates that system-on-chip (SoC) power analysis requires deep cycles to identify and analyze peak and average power at the system level. RTL simulation alone is not capable of doing that—it needs to be efficiently augmented by hardware-assisted techniques such as those offered in the Cadence® Palladium® XP Verification Computing Platform. Using hardware-assisted verification, users are provided with a real test environment that allows the software impact to be factored in. Users expect specific capabilities such as full insight into and debug of the hardware, effective scripting, and dynamic force/release operations.

For power-aware functional verification, the low-power intent can be modeled in the Common Power Format (CPF). Palladium XP then reads the CPF file and automatically creates the instrumentation logic for low-power operations. Low-power activities can be captured and shown in waveforms for debugging.

As indicated in Figure 3, long execution cycles are required to avoid limiting the analysis to potential local maximum power consumption cases. The Palladium Dynamic Power Analysis (DPA) Option allows engineers to use the hardware's real test environment, thus providing more realistic stimulus to the device under test (DUT) and enabling more accurate low-power analysis results than those derived from simulation, especially with designs that are refined to the gate-level. Palladium XP contains a high-speed low-power calculation engine and enables tradeoff analysis between speed and accuracy. This engine can be combined with specific power debug functionality supporting DPA and low power functional verification, allowing users to combine power profiles, low power control signals, and design signals in the same waveform window to efficiently analyze power consumption scenarios.



[Figure 3: The need for deep cycles to enable efficient low-power analysis]

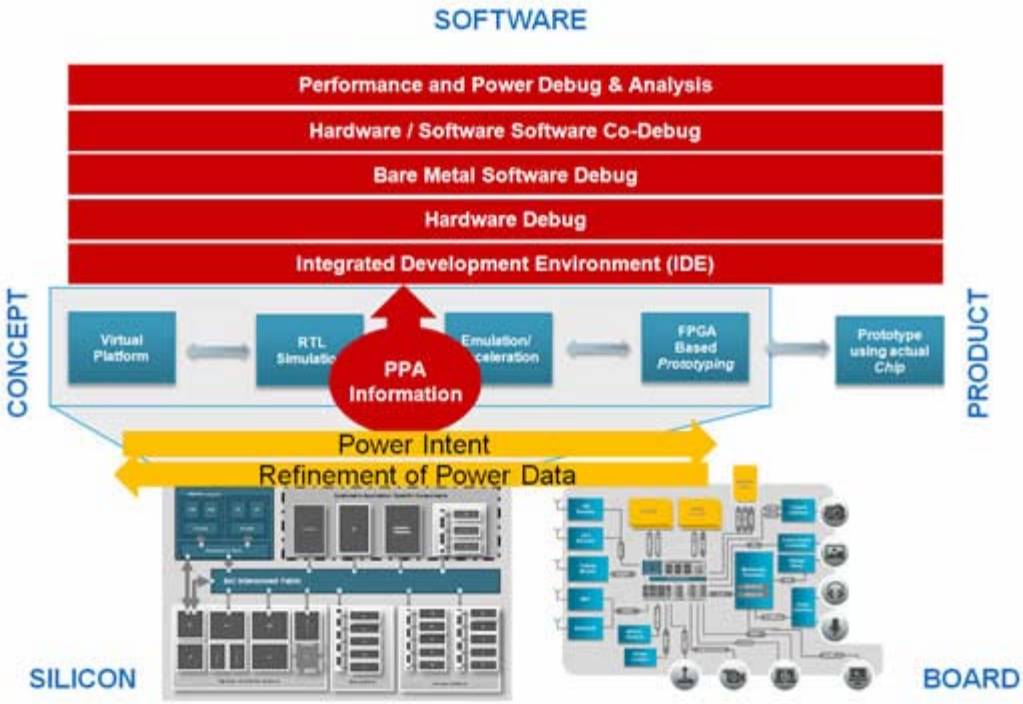
Specifically, different types of toggle counts—representing design activity that translates into power consumption—can be collected. A fast, coarse-grain analysis is provided using native toggle counts

(NTC). It is fastest to create and provides a raw toggle count for the design with each toggle counting as one. One toggle count is provided for each cycle for the partial design or the whole design. If peak identification with higher accuracy is the objective, then weighted toggle count (WTC) can be collected. It provides weights on different nets; for example, a memory write-enable toggle is weighted heavier than NAND gate input toggles. For highest accuracy, a fine-grain analysis can use full toggle count format (TCF) reports.

Palladium DPA is aware of the power intent captured using CPF, which allows the DPA report to be complete with power mode and power domain information.

How it All Fits Together

The techniques outlined here at the TLM and RTL levels are all working well for their specific target use models. Going back to the first section of this article—following the paradigm of deciding early and verifying often with refined data—next-generation power optimization flows will combine the techniques at the different levels of abstraction, as indicated in Figure 4.



[Figure 4: Power intent and refinement of power data in a connected power optimization flow]

Starting from TLM-based virtual platforms, power intent can be declared and forwarded to implementation in RTL. Once refined, more detailed data becomes available using the dynamic power analysis techniques described earlier, and the data can be back-annotated into the earlier phases of the design. That linkage can go all the way back into the static data analysis as it is provided for IP blocks to users of the Cadence Chip Planning System, which is used even prior to TLM-based models.

With a connected flow from TLM though RTL to implementation, users will be able to further optimize designs for power and performance, meeting the ever-growing demands that consumers have for the latest electronic gadgets.

2x, 5x, 10x a case for improved power/thermal modeling flows

Docea Power

Breakthroughs in technology are most noticeable when the user sees a substantial improvement in how they get things done. In computing systems speed up of task completion is the heartbeat of improvement. MIP's, IPS, Spec Mark and LINPAC rule the roost for many.

The user has grown accustomed to seeing improvements year over year, but often it is incremental and may not be that compelling. A 2x, 5x or 10 improvement in task completion gets the users attention. This happened in CPU designs when caches were integrated, combined with re-pipelining and integration of the floating point unit. It happened again when SIMD was introduced combined with superscalar and multiprocessing. Rather than just increasing clock speed we see advancements using multi-core and heterogeneous processing and fixed function offload engines.

In the world of power/thermal and performance modeling, the significant breakthroughs may come from a focus on improved energy efficiency and the use of tools for explicit power and thermal exploration. They achieve faster model creation and simulation speeds by providing the right level of abstraction. We continue to see the design cycles shrinking that compresses the time allowed for architectural exploration. Re-use in design often precludes changes in legacy blocks. Design automation improves but it feels like the incrementalism we see in performance improvements.

Energy and thermals have unique properties. They are best optimized in the hardware implementation as a function of a mixed set of behavior over a fairly long passage of time. To get huge improvements, you have to factor in an entire user sessions, almost like a day in the life of a user, or the duration of ownership of a devices and the services they use.

To get improvements in the modeling and simulation, doing concurrent engineering, parallelizing tasks is analogous to SIMD and multi-processing. It does require a new way to organize the tasks, scheduling and managing dependencies. The decoupled processes like power estimation and thermal management cannot be addressed in time or early enough. This is a case of separation of concerns where power and thermal models must be provided early, yet match, adjust and remain consistent with the functional and performance models.

This requires a new look at automation to provide the right abstract power and thermal models which are compact, automated to develop from a variety of sources and fast to simulate. They also must retain the key determinants of power and thermal heat dissipation from the implementation. It would be wonderful if you could re-abstract the RTL/Gate level design back into VHDL or System C for all blocks (including legacy) for functional and performance simulation. Likewise it would be great to re-abstract the full chip model into a black box of power sources and power consumers.

To get to the next breakthroughs in power, performance and thermal/mechanical designs, we will

need to re-look at our flows in exploration, development and test/validation. Automation downward to produce designs at the structural and physical level is great. We also have to look at ways to also automate upward to produce compact, accurate and up to date models which can speed up power/thermal model generation and significantly improve simulation times.

To learn more about the tools for power analysis and modeling please visit: www.doceapower.com