

RTL Architect

Simply Better RTL

Author

Jim Schultz

Product Marketing Manager,
Design Group, Synopsys, Inc.

Introduction

Electronic devices play a key role in society. They connect us to one another through voice, video and chat. They entertain, educate, protect and heal us in new and ever-expanding ways. They have changed the way we work, live and play. Silicon chips are the fast beating heart (2 to 3 billion beats per second) of these devices. For decades, the relentless advancements in semiconductor process technology have directly enabled these chips, and the devices they run, to do more and have a larger impact on society. However, for the latest process nodes, advancements in ASIC design tools and techniques are required to unlock these benefits.

Chip design benefits are typically measured in terms of power, performance and area (PPA). Traditionally, semiconductor manufacturers were regularly able to reduce area by 50% (guided by Moore's Law) by introducing new planar silicon process nodes that shrank transistors and wires. This was important because it doubled the density of logic cells and allowed more functionality to be packed onto a single chip. In addition, the area reduction decreased the circuit delay which increased the switching frequency by roughly 40%, providing a boost in performance (1.4X). Finally, the smaller gate area required less voltage to maintain the same electric field as the previous node (captured by Dennard's scaling). Voltage was reduced by 30% and power was reduced by approximately 50% (at 1.4X frequency).

However, simple planar process node shrinks could not continue to deliver these benefits indefinitely, as seen in Figure 1 Transistor Count and Frequency Scaling by Year. Shrinking transistor gates eventually encountered two critical limitations. First, leakage current's impact on overall power (previously a minor effect) increased due to a short-channel effect of the bulk CMOS process. Second, voltage scaling hit a physical lower limit which caused power density to increase in the smaller gates and led to chip overheating. This has roughly limited the max operating frequency to 4GHz. To continue shrinking, semiconductor manufacturers collaborated with design tool vendors to overcome the first issue. At 130nm, process manufacturers introduced body biasing to control the transistor threshold voltage (V_{th}) which allowed them to offer a selection of low, medium and high leakage cells which conversely affected the switching speed. The design tools contributed by analyzing timing paths and deciding which V_{th} cell to use, trading low static power (high V_{th} cell) vs. performance (low V_{th} cell).

As bulk CMOS processes continued to shrink, body biasing lost its efficacy around 28nm. Today, improving PPA is achieved through a combination of new process technologies and tool innovations. The non-planar, bent-gate, FinFET process was introduced at 22nm to dramatically reduce leakage current and therefore static power. Alternatively, the planar FD-SOI process, introduced at 28nm, offers a lower cost option to reduce leakage current. With static power under control by process innovation, design tools are used to lower the dominant dynamic power. Design techniques, such as, clock gating dynamic voltage and frequency scaling (DVFS), and glitch removal are used to control dynamic power. To improve performance, the problem shifted to the left to the logic architectural stage. Parallelization was introduced by logic design teams to improve the operations per second by dividing the work between multiple processing engines, such as, multi-core CPUs, GPUs, and AI chips. Sophisticated communication architectures, such as a Network on Chip (NoC), are used to connect the multiple engines in a power efficient manner and the performance scales well for large System on Chip (SoC) designs.

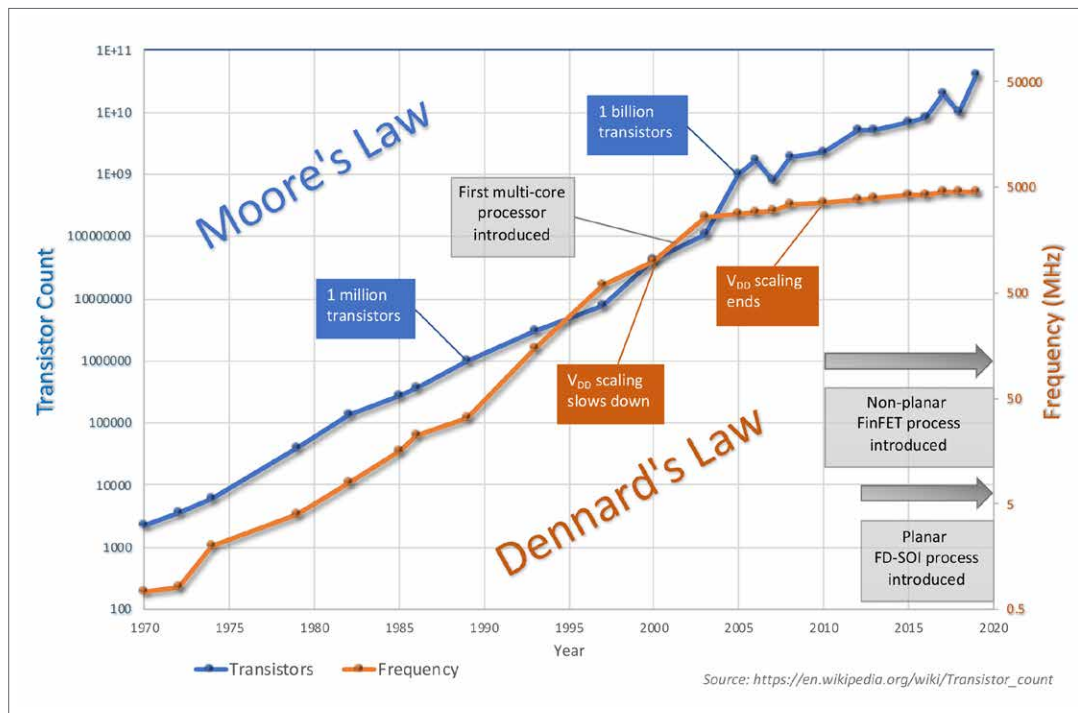


Figure 1: Processor Transistor Count and Frequency Scaling by Year

Design Challenges

Traditional design flows are not enough to keep pace with the advancements in process technology and market demands for better power, performance, and area (PPA). It has become increasingly difficult to extract the full potential out of these costly technology nodes through physical implementation techniques alone.

The advancements in process technology is a double-edge sword. It can be a difficult task for a product team to choose an implementation technology and metal stack. It typically requires synthesizing some representative RTL into gates and handing it off to the implementation team to perform some early PPA characterization.

In a traditional design flow, RTL developers architect circuit behavior with little visibility into how it is physically implemented. In many cases, module owners run a quick synthesis to check for syntax errors then hand off the blocks to an integrator who oversees consolidation and final synthesis. The integrator is primarily concerned that the block interface signals connect with the top-level, and that the timing and power constraints are consistent. The synthesized gate level netlist is then handed off to the physical implementation team to floor plan, place and route the design. In most cases, the RTL designer is separated from implementation by nearly a week.

We observe several issues with this process. The first is that the implementation flow is expected to achieve the full PPA potential of the technology node through place and route optimization techniques. However, at advanced nodes, there are many more requirements on cell placement and wire routing that limit the degrees of freedom and thus the effectiveness of optimization. The additional rules also cause the implementation run time to increase.

A second important issue is the disconnect between the RTL development and the physical implementation tasks. The RTL developer typically has little to no insight into how the logic will be physically implemented in layout. Their logic block might communicate to another block that is physically placed on the other side of the chip. In this scenario, implementation might require the addition of pipeline registers and feedthrough nets to the RTL to meet timing. However, it could take days from the RTL handoff to feedback from the implementation. In some cases, a new version of the RTL is ready before the previous version is synthesized and implemented. Besides timing, other useful implementation insights are power, congestion and area.

The third and final issue we will explore is Intellectual Property (IP) integration. IP can refer to external, third-party, blocks or internal logic blocks provided as a service to other groups. As previously mentioned, the IP provider delivers the RTL to an integrator who is responsible for incorporating the block into the overall design. The difficulty is that the integrator is unfamiliar with the IP and can only verify that the IP conforms to a set of pre-defined sanity checks, such as, clock consistency, timing constraints, and power domain definition.

Introduction to RTL Architect

The RTL Architect™ solution, as seen in Figure 2 RTL Cross-Probing, is the industry's first physically aware RTL design system that significantly reduces the development cycle and delivers superior quality-of-results. It continues the “shift-left” strategy introduced in the Fusion Design Platform to address PPA challenges earlier in the design cycle. The complexities of advanced process nodes have made it more difficult to meet PPA targets through physical implementation techniques alone, so RTL designers are tasked with exploring domain-specific architectures to dramatically improve PPA. RTL Architect provides a logical/physical workbench that can accurately predict the PPA impact of architectural changes without waiting for feedback from the physical design team. This section will introduce how the tool's core technologies (Predictive Engine, Hierarchical Design, RTL Power Analysis, and Constraints Management) provide insight and solutions to address the traditional design flow issues.

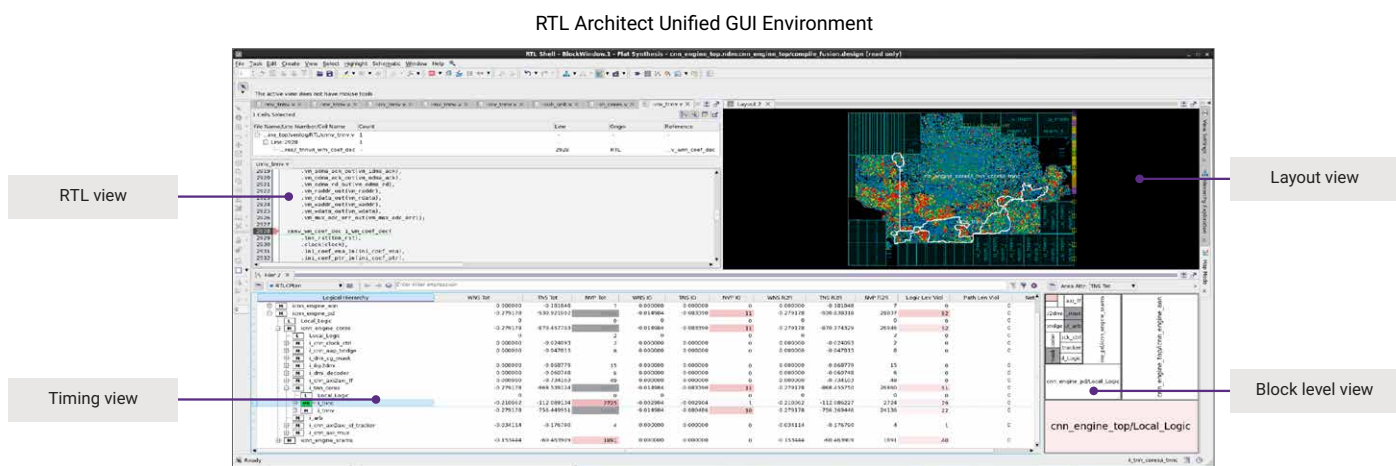


Figure 2: RTL Cross-Probing

At the heart of RTL Architect is the new, multi-dimensional, Predictive Engine (PE) built on the Fusion Design Platform. It can accurately predict and model synthesis results since it is built with the same technology that drives the dominant, industry-standard, synthesis tool. The concurrent optimization algorithms provide highly correlated results an average 3X, and up to 5x, faster than a traditional synthesis-implementation flow. This enables the RTL designer to run multiple experiments per day to evaluate the impact their code will have on implementation.

RTL Architect is built on the same unified data model used by Fusion Compiler and IC Compiler II. The enterprise-class infrastructure scales to meet the needs of the largest SoCs. RTL Architect was designed to provide timely results for multi-million instance block designs and sub-systems with tens of millions of instances. It is fully hierarchical, supporting Multiple levels of Physical Hierarchy (MPH) and it supports replication of Multiple Instantiated Blocks (MIB). Additionally, it meets the advanced node accuracy requirements with the integration of the signoff technologies found in PrimeTime, StarRC, and PrimePower.

A key feature of the RTL Architect solution is hierarchical floor planning. Similar to traditional floor planning, it will perform macro and standard cell placement using the predictive gate models. Block shaping can be used to experiment with different block layouts. The global router will route the design and provide congestion information that can be used to identify routing hotspots. Pin assignment on fully abutted or narrow channel designs indicate the need for purely physical feedthrough nets to be added to a logic block. Likewise, pipeline register planning can pinpoint which blocks need additional registers to meet timing requirements. Unlike traditional floor planning solutions, RTL Architect can cross-probe back to the source RTL. For example, as seen in Figure 3 Congestion Analysis and Repair, cells in a layout congestion hotspot can be traced back to the line in the RTL that produced the logic. This enables the designer to explore different architectures to avoid creating the congestion, such as, replacing muxes with combinational logic.

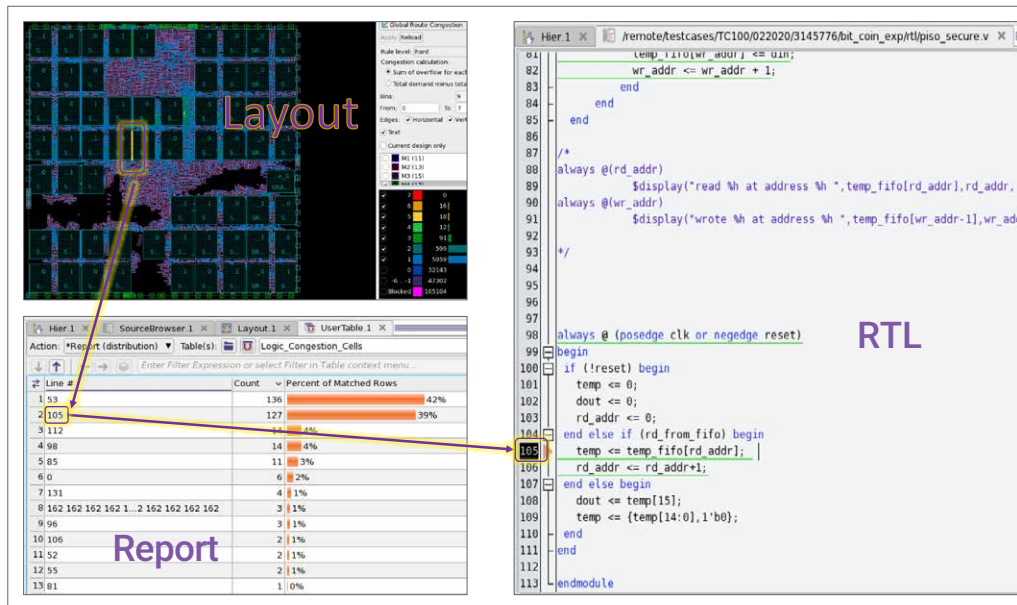


Figure 3: Congestion Analysis and Repair

Since traditional RTL designers are usually inexperienced with layout and schematic views, the default view consists of summary reports. The reports are active tables that can be sorted, filtered, and colorized to indicate severity of standard metrics, such as, timing, power, utilization, and congestion. Additional, non-standard, metrics help identify whether issues are caused by the logic or the floor plan, such as, logic induced congestion or floorplan induced congestion. By default, the reports are based on an automatic floor plan generated by RTL Architect so that the RTL designer does not need physical implementation experience to get results. Alternatively, an implementation expert can set project default values or provide a floor plan DEF if available.

Another important feature of RTL Architect is the ability to perform design sweeps over various parameters. The capability is referred to as Parallel Design Exploration. The user can create a base run and perform multiple parallel runs to experimentally determine optimal design configurations. For example, a design sweep using various technology libraries can be performed to explore the benefit of using one technology node vs. another. Another example, as shown in Figure 4 Block Aspect Ratio and Utilization Sweep Results, is to sweep a block using various combinations of aspect ratio and utilization to measure the impact on congestion. The tool analyzes the various runs of the sweep and creates a summary report to identify the optimal results.

Explore Design Summary

- Design : /remote/testcases/TC100/TCNN/ 184245/top/kan.ndm
- Total 4 experiments, 4 finished

Summary Table [Comparator](#)

Run	Timing			Power			Image	Congestion					Instance Count				
	WNS	TNS	NVE	Total	Leakage	Switching		#Cells in Congested Areas	Overflow		#GRC's has	Standard	Register	Repeaters	ICG's	Mbit Registers	
									Total	Max	Overflow(%)						max overflow
util90_aspect1-2	-0.58	-15.08	64	4.61e-03	2.93e-03	6.68e-04		25	44	4	34 (0.44%)	1	4413	148	638	1	148
util60_aspect1-2	-0.61	-15.72	64	4.20e-03	2.80e-03	4.53e-04		0	8	3	4 (0.04%)	2	4422	148	629	1	148
util90_aspect2-1	-0.59	-15.11	64	4.42e-03	2.79e-03	6.37e-04		673	758	3	681 (9.10%)	3	4396	148	642	1	148
util60_aspect2-1	-0.64	-16.45	64	4.61e-03	2.93e-03	6.68e-04		35	131	3	126 (1.11%)	1	4431	148	647	1	148

Figure 4: Block Aspect Ratio and Utilization Sweep Results

Power is one of the most important and difficult requirements to meet on advanced node designs. Architectural choices made during RTL design and synthesis have a much greater impact on overall power consumption than physical implementation optimization techniques and are also less disruptive. By providing early visibility into physically-aware RTL power estimates, RTL Architect enables designers to analyze, explore, and optimize PPA for their block, subsystem, and full-SoC RTL designs with confidence, achieving the right balance of performance and energy efficiency. To achieve this with signoff-consistent results, RTL Architect directly integrates the Synopsys PrimePower golden signoff power analysis engine. Supported RTL power analysis includes average power, peak power, glitch power, clock network power, dynamic and leakage power, and multi-voltage power; with activity from RTL vectors generated via simulation or emulation, or even in a vectorless manner. The tool provides an RTL power analysis dashboard, seen in Figure 5 Power Analysis Dashboard, to conveniently review power metrics and debug any issues. The reports and graphs provide guidance to improve clock gating efficiency, memory access rate and data path power to meet average and peak power budgets sooner.

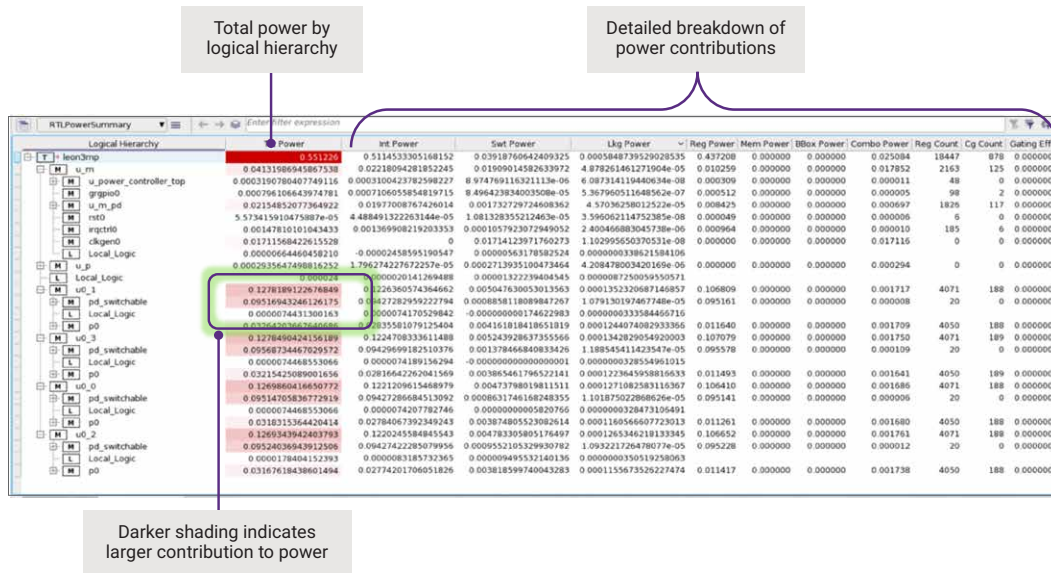


Figure 5: Power Analysis Dashboard

Lastly, RTL Architect includes a robust constraints management system. During block integration the system checks timing constraints to make sure they are consistent. Constraints can be pushed down from the top to the blocks through a demotion process. Reciprocally, block constraints can be promoted from the bottom to the top. This is particularly important when integrating 3rd party IP. Constraints can be promoted and used to drive top-level constraints, such as, clock mapping, case analysis and others. This core technology is important for RTL Architect's logical re-structuring feature. As designer's experiment with logical hierarchy, such as re-parenting, the constraints management system will track and record the impact on the constraints, which is typically a tedious and error-prone process.

Conclusion

From 5G wireless communication to AI automation, electronic devices will deliver the next wave of benefits to society through low power, high performance, massively parallel ASICs. Innovative architectures and design tools are the key to unlocking the benefits of advanced process nodes. The Synopsys Fusion Design Platform (Figure 6 Fusion Design Platform) is not a collection of point tools, instead it is an integrated solution developed to enable a new “shift-left” approach to solving PPA challenges at advanced process nodes. RTL Architect identifies and corrects physical implementation issues early in the design cycle to achieve aggressive PPA targets at advanced nodes through better RTL.

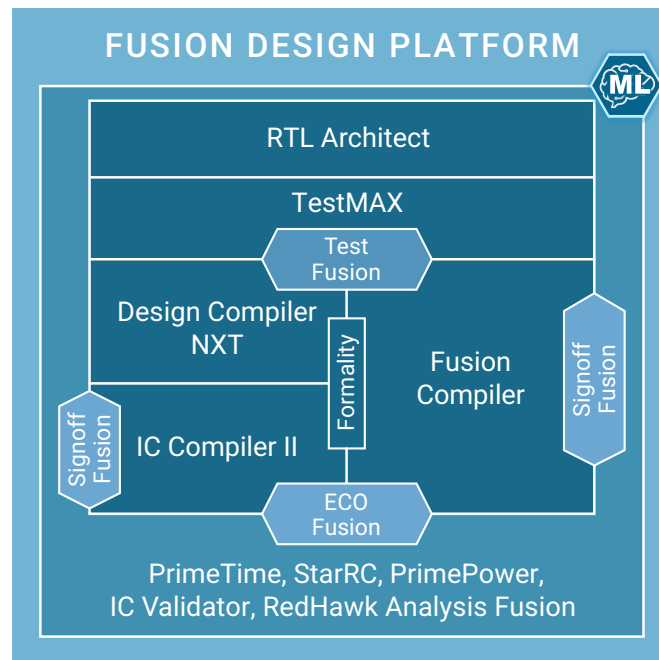


Figure 6: Fusion Design Platform