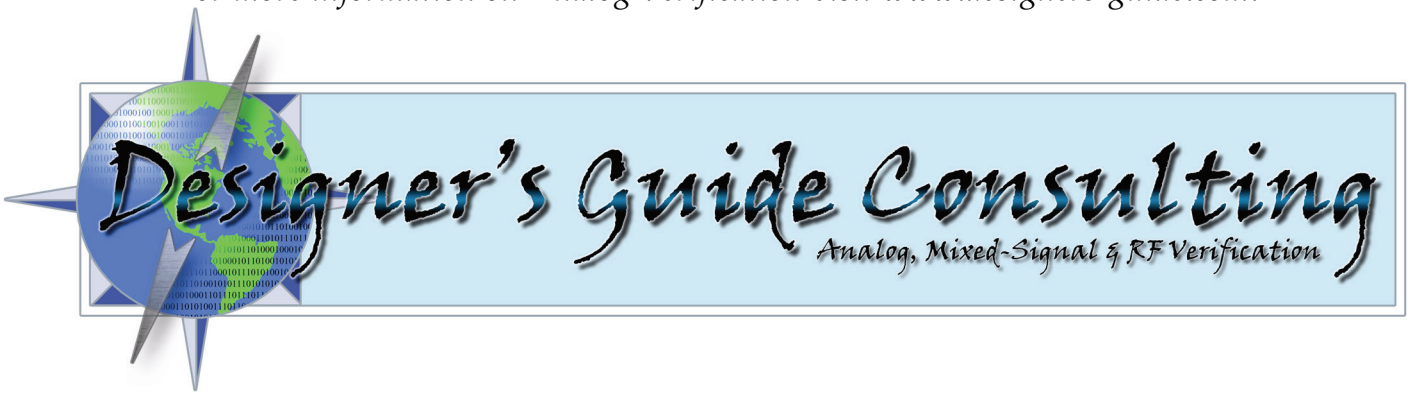


Top-Down Design and Verification of Mixed-Signal Circuits

Ken Kundert
Henry Chang

Version 1a, 20 June 2005

Provides and introduction and motivation for a top-down verification. Top-down verification is a refinement of the top-down design methodology that adds considerable rigor and greatly reduces the chance of a functional failure.

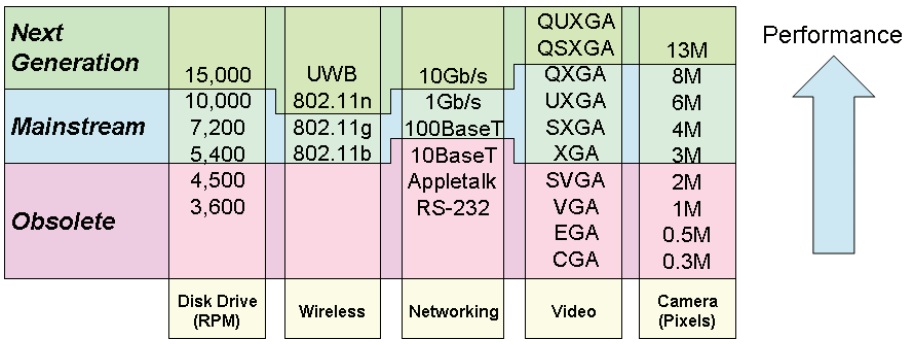
The is paper was first presented in the 20 June 2005 issue of "The Planet Analog EE Times Magazine Supplement". It was last updated on November 24, 2009. You can find the most recent version at www.designers-guide.com. Contact the authors via e-mail at consulting@designersguide.com.

1 The Challenge of Mixed-Signal Design

The challenge of mixed-signal design is different in character from the challenge of digital design. In digital design, the overwhelming challenge stems from the large size of the circuits. At the level of individual gates, the challenges are not demanding. It's just that there are so many gates that the aggregate challenge is huge. This is the reason why design automation is so important. It is only by automating the process of digital design does it become tractable on the very large designs that are common today. This issue has given rise to the infamous *EDA gap* — an ever widening gap between the number of components that can fit on a chip, and the capacity of the design tools.

With mixed-signal design the challenge is different. It is not so much the size of the design that creates the challenge, though it is contributing more and more each day. Rather, it is the sheer difficulty of achieving the needed performance. As shown in Figure 1, the pace that performance increases in mixed-signal systems is relentless. Competitive pressures always push mixed-signal performance requirements right to the edge of what is possible. In the few cases where performance is capped by external factors, such as with voiceline modems, the mixed-signal circuitry eventually becomes a low-value commodity or is replaced by digital. After all, digital circuitry is easier to design, more flexible, cheaper, and less risk.

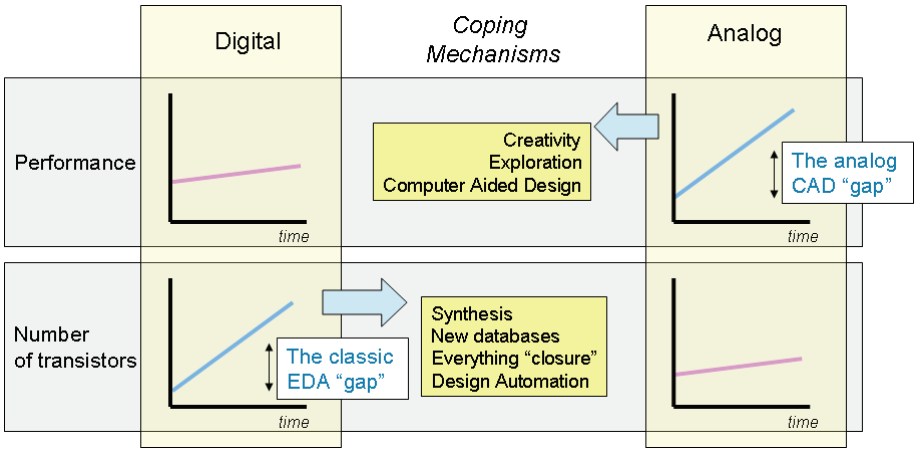
FIGURE 1 *Analog pushes performance boundaries.*



In a mixed-signal design, each piece of the design presents unique and difficult design problems. They can only be solved by producing a custom circuit that fits the requirements of its particular situation. The challenge is the extreme performance requirements, which is addressed using creativity. Creativity is virtually impossible to automate. This represents a different type of gap, a gap between the needed performance and what can be efficiently designed given the available tools. Both of these gaps are illustrated in Figure 2.

The EDA industry has missed this point. For the last 10 years companies like Barcelona, Neolinear, and ADA, and now Cadence and Synopsys, have been trying to automate the mixed-signal design process with little success. With

FIGURE 2 *The classic EDA gap versus the analog EDA gap.*



mixed-signal circuits becoming quite large, automation is important, but providing automation that does not allow designers to inject substantial creativity as an inherent part of the process is a waste of everyone's time.

Rather than Electronic Design *Automation*, what is needed is Computer-*Aided* Design. The idea must be to aid the designer rather than replace the designer. What's needed is a methodology (and supporting tools) that allows designers to be very creative, highly productive, and capable of building large complex systems with a high likelihood of success. The only known such methodology is described in Sections 3 and 4.

2 Bottom-Up Mixed-Signal Design

The traditional approach to design is referred to as bottom-up design. In it, the design process starts with the design of the individual blocks, which are then combined to form the system. The design of the blocks starts with a set of specifications and ends with a transistor level implementation. Each block is verified as a stand-alone unit against specifications and not in the context of the overall system. Once verified individually, the blocks are then combined and verified together, but at this point the entire system is represented at the transistor level.

While the bottom-up design style continues to be effective for small designs, large designs expose several important problems in this approach.

1. For complex designs, the greatest impact on the performance, cost and functionality is typically found at the architectural level. With a bottom-up design style, little if any architectural exploration is performed and so these types of improvements are often missed.
2. Once the blocks are combined, simulation takes a long time and verification becomes difficult and perhaps impossible.

3. Any errors or problems found when assembling the system are expensive to fix because they involve redesign of the blocks.
4. Communication between designers is critical, yet an informal and error prone approach to communication is employed. With the limited ability to verify the system, any failure in communication likely results in the need of a silicon respin.
5. Several important and expensive steps in the bottom-up design process must be performed serially, which stretches the time required to complete the design.

3 Top-Down Mixed-Signal Design

In order to address these challenges, many design teams are either looking to, or else have already implemented, a top-down design methodology. In a top-down approach, the architecture of the chip is defined as a block diagram and simulated and optimized using a system simulator such as Matlab or Simulink. From the high-level simulation the performance requirements for the individual circuit blocks are derived. Circuits are then designed individually to meet these specifications. Finally, the entire chip is laid out and verified against the original requirements.

Top-down design only addresses one of the issues with bottom-up design, that of Point 1 in Section 2. In essence, these design groups have not fundamentally changed their design process; they have simply added an architectural exploration step to the front. The issue is that the focus of top-down design is performance, and of the 5 points, only Point 1 involves performance. In Points 2-4 the issue is verification and in Point 5 the issue is time-to-market.

In addition, a traditional top-down design style adds a discontinuity in the design flow that results because the representation used during the architectural exploration phase is incompatible with the representation used during implementation. This discontinuity leaves the block designers without an efficient way of assuring that the blocks all work together as expected. As with bottom-up design, one could assemble transistor-level representations of the blocks and run simulations, but the simulations are too slow to be effective. The first time the blocks can be thoroughly tested together is first silicon, and at that point any errors found trigger a respin. The discontinuity also makes communications more difficult and *ad hoc* and so acts to separate the system designers from the circuit designers, and the circuit designers from each other. Without a reliable communication channel, designers resort to using verbal or written specifications, which are often incomplete, poorly communicated, and forgotten half way through the project. It is the poor communication process that creates many of the errors that force respins, and the separation that allows the errors to hide until the design is available as silicon.

To overcome these issues, one needs a design methodology that

1. Improves communications between all designers.
2. Eliminates the discontinuity in design representation that acts to hide errors and separate the system designers from the block designers.
3. Improves verification so that it finds the errors that cause respins, and finds them earlier so that they are less disruptive and easier to fix.
4. Reorganize the design tasks, making them more parallel and eliminating long serial dependencies.
5. Reduce the need for extensive transistor-level final verification.
6. Eliminate respins!

Notice that the focus of all these requirements is functional verification.

4 Top-Down Design and Verification

A top-down design and verification process layers a formal modeling and verification methodology onto a top-down design process. A top-down design and verification methodology systematically proceeds from architecture- to transistor-level design. Each level is fully verified before proceeding to the next and each level is leveraged in verification of the next. A top-down verification process also formalizes and improves communications between designers. This reduces the number of flaws that creep into a design because of miscommunication. The formal nature of the communication also allows designers to be located at different sites and still be effective.

4.1 Principles of Top-Down Verification

An effective top-down verification process follows a set of basic principles.

1. A shared design representation is used for the entire length of the project that allows the design be simulated by all members of the design team and in which all types of descriptions (behavioral, circuit, layout) can be co-simulated.
2. During the design process each change to the design is verified in the context of the entire, previously verified, design as dictated by the verification plan.
3. A design process that includes careful verification planning where risks are identified up-front and simulation and modeling plans are developed that act to mitigate the risks.
4. A design process that involves multiple passes, starting with high level abstractions and refining as the detail becomes available. In effect, running through the entire process very quickly at the beginning with rough estimates and guesses to get a better understanding and better estimates, and then refining the design as the process progresses.

5. To the degree possible, specifications and plans should be manifested as executable models and scripts, things that are used in the design process on a daily basis, rather than as written documents.

5 Mixed-Level Simulation

Without analog synthesis, analog design is done the old fashioned way, with designers manually converting specifications to circuits. While this allows for more creativity and gives higher performance, it also results in more errors, particularly those that stem from miscommunication. These miscommunications result in errors that prevent the system from operating properly when the blocks are assembled even though the blocks were thought to be correct when tested individually.

To overcome this problem, mixed-level simulation is employed as a critical component of top-down design verification for analog and mixed-signal circuits. This represents a significant but essential departure from a digital design methodology. Mixed-level simulation is required to establish that the blocks function as designed in the overall system.

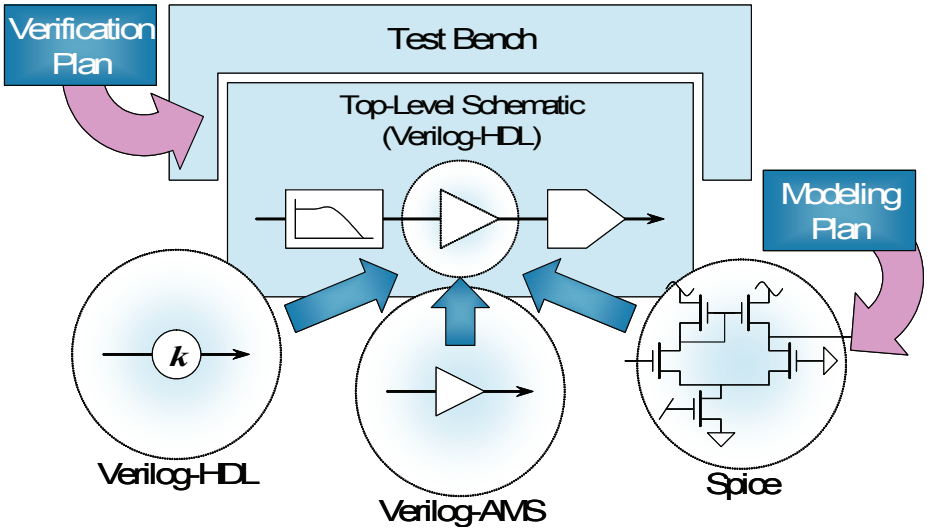
To verify a block with mixed-level simulation, the model of the block in the top-level schematic is replaced with the transistor level schematic of the block before running the simulation, as shown in Figure 3. For this reason, the simulator used for the top- or architectural-level of the design must be capable of co-simulating with transistor-level components. In addition all of the blocks in the architectural description of the system must be “pin-accurate”, meaning that they must have the right number of pins and characteristics of each pin must be representative of the expected signal and bias levels, polarities, impedances, etc. so they can be replaced with their transistor-level equivalents with no other changes.

The pin-accurate system description, described at a high level, acts as a test bench for the block, which is described at the transistor level. Thus, the block is verified in the context of the system, and it is easy to see the effect of imperfections in the block on the performance of the system.

Typically, the top-level model would be described in Verilog-HDL. In this way the model can be shared with the people designing larger digital portion of a chip. It can also be simulated within the mixed-signal design team with a Verilog-AMS simulator, which enables the mixed-level simulation because the individual blocks can be replaced with Verilog-AMS models, Verilog-A models, or transistor level netlists.

Mixed-level simulation is the only feasible approach currently available for verifying large complex mixed-signal systems. Some propose to use either timing simulators (sometimes referred to as fast or reduced accuracy circuit simulators) or circuit simulators running on parallel processors. However, both approaches

FIGURE 3 *Mixed-level simulation as controlled by the verification and modeling plans.*



defer system-level verification until the whole system is available at transistor level, and neither provides the performance nor the generality needed to thoroughly verify most mixed-signal systems. They do, however, have roles to play both within the mixed-level simulation process and during final verification.

Successful use of mixed-level simulation requires careful planning and forethought (provided during the verification planning process). And even then, there is no guarantee that it will find all the problems with a design. However, it will find many problems, and it will find them much earlier in the design process, before full-chip simulations, when they are much less costly to fix. And with mixed-level simulation, it is possible to run tests that are much too expensive to run with full-chip simulation.

6 Conclusion

Employing a rigorous top-down design and verification methodology greatly reduces the chance of respins and redesigns. It also has the somewhat surprising benefit that it dramatically increases the effectiveness and productivity of a design team and reduces time to market. It is not something that is only theoretical or impractical; it is being employed by a small number of design teams today, with great success. It was not possible to fully describe an effective top-down verification methodology here. To get a much more complete description of this methodology, visit the *Documents* page of *www.designers-guide.com* and download “Principles of top-down mixed-signal design”. Comments and questions should be directed to the “Top-Down Design and Reuse” forum at *www.designers-guide.org/Forum.*

About The Authors

Ken Kundert. Ken is an IEEE Fellow and co-founded Designer's Guide Consulting in 2005. From 1989 to 2005, Ken worked at Cadence Design Systems as a Fellow. Ken created Spectre and was the principal architect of the Spectre circuit simulation family. As such, he has led the development of Spectre, SpectreHDL, and SpectreRF. He also played a key role in the development of Cadence's AMS Designer and made substantial contributions to both the Verilog-AMS and VHDL-AMS languages. While in school he authored *Sparse*, an industry standard sparse linear equation solver and created Agilent's harmonic balance simulator. Before that Ken was a circuit designer at Tektronix and Hewlett-Packard, and contributed to the design of the HP 8510 microwave network analyzer. He has written three books on circuit simulation: *The Designer's Guide to Verilog-AMS* in 2004, *The Designer's Guide to SPICE and Spectre* in 1995, and *Steady-State Methods for Simulating Analog and Microwave Circuits* in 1990; and created *The Designer's Guide Community* website. He has also authored eleven patents and over two-dozen papers published in refereed conferences and journals.

Ken received his Ph. D. in Electrical Engineering from the University of California at Berkeley in 1989, his M. Eng. degree in 1983 and his B. S. in 1979. Ken was elevated to the status of *IEEE Fellow* in January 2007 for contributions to simulation and modeling of analog, RF, and mixed-signal circuits.

Henry Chang. Henry co-founded Designer's Guide Consulting in 2005. From 1995 to 2005, Henry worked at Cadence Design Systems, Inc. in research and development, methodology services, product marketing, corporate strategy, and in the office of the Chief Technology Officer. He has also worked at Micro Linear and GE Lighting. He is the author of three books: *Winning the SoC Revolution: Experiences in Real Design* in 2003, *Surviving the SoC Revolution: A Guide to Platform Based Design* in 1999, and *A Top-Down, Constraint-Driven Design Methodology for Analog Integrated Circuits* in 1997. He is on the steering committee of the IEEE Custom Integrated Circuits Conference, serving presently as the general chairman. He holds 10 US patents, has authored 14 technical papers, and has participated at conferences giving tutorials, sitting on panels, and giving keynote addresses.

Henry received his Ph. D. and M. S. in Electrical Engineering from the University of California at Berkeley in 1994 and 1992 respectively. He received his Sc. B. degree in Electrical Engineering from Brown University in 1989.

About Designer's Guide Consulting

We help design teams overcome difficult verification challenges. Those challenges can involve either functional or performance verification. A difficult functional verification problem might be assuring that a mixed-signal circuit consisting of tens or hundreds of thousands of transistors with multiple ΔΣ converters operates correctly in each of its thousands of distinct operating modes. A difficult performance verification might be assuring that a large behaviorally complex circuit meets some demanding specification, such as a SerDes operating with a bit-error rate of less than 10^{-18}.

Designer's Guide Consulting
101 First Street, #150
Los Altos, CA 94022
+1 650-968-8291
consulting@designers-guide.com
www.designers-guide.com