BY JERRY TWOMEY • EFFECTIVE ELECTRONS

Efficient simulation and validation for mixed-signal SOCs

ADDING ANALOG CONTENT TO DIGITAL-IC DESIGNS CAN BE A NIGHTMARE. SIMULATION AND VALIDATION TOOLS CAN HELP WARD OFF THE TERROR.

The IC-design groups tackling complex digital-IC projects often have problems attaining complete system coverage of their designs. But when they add analog content into their designs, attaining coverage becomes a nightmare because no comprehensive method exists for directly validating an entire design. The ever-rising popularity of cell phones and assorted wireless interconnects has reinvigorated the use of RF and analog circuits. Consequently, mixed-signal SOCs (systems on chips) are here to stay. Unfortunately, that means the complexity of top-level validation is also here to stay.

Designers now attribute the most common errors in SOCs to a mix of human error and a lack of comprehensive validation. The most commonplace errors in this vein are errors in interconnection, failure to test all possible modes of operation, polarity inversion of control signals, and errors with transposed digital buses. You can also attribute a lot of these problems to a lack of a single comprehensive validation method.

IC design has traditionally been in two camps: analog and digital. Designers build most digital circuits behaviorally in RTL code with automated synthesis and most analog circuits using TLD (transistor-level-design) tools—including schematics and a Spice tool. Neither method is completely ideal for full system validation. Design validation in Spice is painstakingly slow or simply doesn't converge, whereas digital simulators have no graceful way of dealing with analog- and mixed-signal functions.

You can approach the simulation and validation problem in a number of ways, all having strengths and weaknesses. However, design complexity and the amount of analog versus digital components make some methods impractical.

ALL ANALOG

In an all-analog approach, you use a Spice simulation, which defines everything as transistors. The all-analog approach works well for small designs, sub-blocks of a larger system, or individuals with lots of time to waste. Low-dropout regulators, op amps, comparators, and other commodity ICs come to mind here. Professional versions of Spice, including Cadence's Spectre, Mentor Graphics' Eldo, Synopsys' HSpice, and Simucad's SmartSpice (www.cadence.com, www.mentor.com, www.synopsys.com, www.simucad.com), all attempt to speed things up. However, most SOC designs require a lot of simulation. Design and transistor-model complexity are outpacing computer speeds for simulation. Consider that the BSIM (Berkeley Short-Channel IGFET Model) 4.2 transistor model has more than 230 parameters, and designs with 1 million transistors are part of the challenge. Spice's ability to simultaneously solve equations and do floating-point math prevents quick simulations of these complex structures. When you add corner testing and multimode operations into the mix, the all-analog methodology is simply not viable.

However, the analog methodology is still viable for developing the small analog sub-blocks in an SOC. Also, any necessary frequency-domain analysis must remain all-analog due to the fact that designers simulate digital systems only in the time domain.

ALL-ANALOG, FAST SPICE

To help speed analog development, a number of companies have developed "fast-Spice" tools. These tools expedite

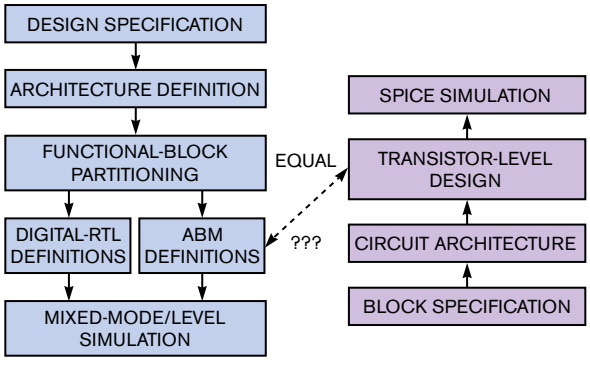

Figure 1 Designers today typically employ a top-down (left) or a bottom-up (right) design method.

simulation but sacrifice accuracy. Common fast-Spice techniques include: model simplification, relaxing error tolerances, lookup-table methods, event-driven methods that ignore inactive circuits, hierarchy, better simultaneous equation-solving methods, variable time-steps, and design partitioning. Most tools interactively trade accuracy for execution time. Designers can make adjustments depending on the accuracy they require for a given portion of the design they are simulating.

However, at the heart of the method, the tools define all the circuits at the transistor level. This approach might improve simulation time, but any significant digital content or large amounts of analog circuitry make this method quickly nonviable. Anything with digital gates should use RTL-simulation methods. Thus, fast-Spice tools are more valuable as part of mixed-mode-simulation methods.

MIXED MODE

In mixed-mode simulation, designers define analog circuits as transistors and digital circuits as RTL. This method is viable in limited situations but often chokes if your design contains a significant amount of analog content. Virtually any transistor-level content greatly slows the process. Early analog/digital co-simulation products comprised two simulators with a method of transferring information between them. Some simulators attempt to improve speed and ease of use, but, no matter how you configured them, number crunching with Spice makes things run slowly. Designs employing a mix of ADCs, DACs, and PLLs do not simulate efficiently with this method, but this approach is valuable when you are developing smaller, mixed-signal blocks. Designing an ADC with digital and analog parts is a good example.

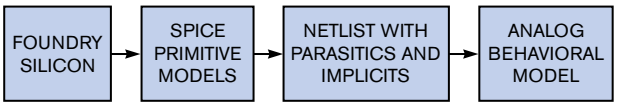

Figure 2 Designers need to ensure that silicon correlates to Spice models, Spice models correlate to a simulation netlist, and Spice simulation correlates to an ABM.

With no efficient way to simulate larger designs with transistor-level tools, the design industry developed ABMs (analog behavioral models), such as Verilog-A, Verilog-AMS, and VHDL-AMS, to speed simulation. Generally, for a large SOC, this approach is the only one in which execution time is viable. Minimizing the need for floating-point math, removing the use of complex-transistor models, and providing functional definitions that sidestep the details of transistor-level implementation all combine to expedite simulations. Accuracy of the behavioral model is a manageable concern. In this method, each ABM needs to have a validation path back to the TLD. System-level designers perform most first-pass ABMs to a mathematical ideal.

Indeed, behavioral modeling and simulation have become new disciplines within the design community with conferences and working groups addressing the needs and issues associated with these disciplines (**references 1** and **2**). Design entry is currently language-driven, with Verilog-A, Verilog-AMS, and VHDL-AMS being the dominant tools. The use of HDL-based tools implies that the orientation is to the digital designer who writes code. Successful analog simulation requires knowledge of the subtle aspects of circuits, and many analog designers don't want to write code. The solution for this problem is simple although not readily available: a schematic-entry tool for ABM development. Conceptually, this tool would be similar to The MathWorks' (www.mathworks.com) Simulink environment, which serves as a tool for the creation of Matlab code (**Reference 3**). One start-up (**Reference 4**) is attempting to develop such a tool, so options are becoming available. Designers, depending on their preference, will most likely use either the HDL or the schematic methods. For now, however, designers wishing to use ABMs need coding skills. The skills are easy to acquire, and numerous examples are available to help designers get started (**Reference 5**).

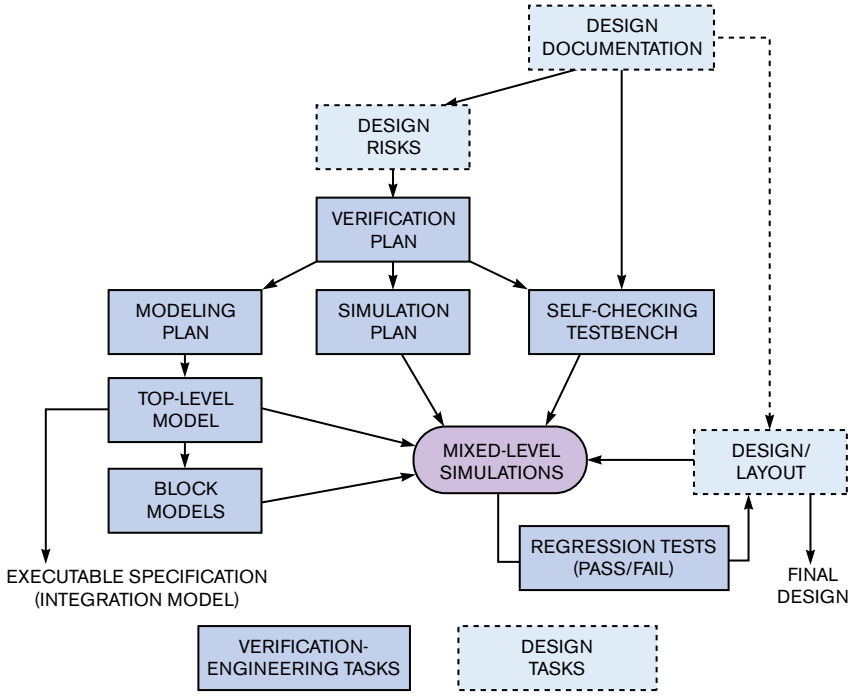

Figure 3 Designer's Guide Consulting's Ken Kundert has an approach that includes use of automation and a complete set of ABMs, which results in a more efficient process.

MIXED LEVEL AND MODE

Using a mix of ABMs, TLD, and RTL digital becomes both mixed-mode and

mixed-level simulation. This type of simulation is a useful approach because it allows you to behaviorally expedite things while optimizing TLD. The behavioral blocks help create the testbench for the transistor-level section. For example, when designing a charge pump in a PLL, using an ABM voltage-controlled oscillator and an RTL-feedback divider considerably expedites simulation. Being able to close the feedback loop allows easier optimization of the device.

The selective use of ABMs during this process implies that a full set of these models will be available as the block-by-block design nears completion. With these models in place, designers can do a fully behavioral top-level simulation. With complex designs, designers should avoid using transistor-level design at the top level.

MORE AT EDN.COM

Go to www.edn.com/ms4220 and click on Feedback Loop to post a comment on this article.

CURRENT-MODE SIMULATION

A technique for determining timing delay in digital designs, current-based modeling, does not apply to analog design, because simulation tools need to solve all fundamentals, including Ohm's Law, Kirchoff's Voltage Law, Kirchoff's Current Law, V=LdI/dt, I=CdV/dt, I=dq/dt, and q=CV, for full Spice accuracy. According to CK Kumar, product-marketing manager at Nascentric (www.nascentric.com), current-based modeling is "valuable for analog behavior of digital circuits" but does not provide a complete analog-design-tool set, as most Spice tools do.

TOP DOWN VERSUS BOTTOM UP

Designers today typically employ a top-down- or a bottom-up-design method (**Figure 1**). The two approaches usually come from different types of designers. Top-down design gets a lot of buzz about being the "correct" way to develop chips. In this method, designers develop a block-level architecture with RTL or ESL (electronic-system-level) functional definitions and first-order ABMs for analog- and mixed-signal structures. The typical top-down designer works from either a system or a digital perspective. Designers have been successful using the method for large SOCs, but the weakness in top-down design is that designers must validate the ABM-to-analog-TLD correlation and frequently are unaware of the nonideal nature of the TLD.

Bottom-up design suits TLD, in which designers manually piece together each functional block to create amplifiers, ADCs, and PLLs. This method far predates the SOC era, when transistor counts and area were small enough that designers could simulate them in a reasonable time with Spice. All mixed-signal chips combine top-down and bottom-up methods. Even if designers use a top-down approach and purchase their mixed-signal IP (intellectual property) from a third-party vendor, someone did TLD for those boxes. So both approaches are valuable and needed. The important thing is getting them to properly meet in the middle with an accurate ABM to represent the TLD.

ABM-TO-TLD CORRELATION

Designers need to ensure that silicon correlates to Spice models, Spice models correlate to a simulation netlist, and Spice simulation correlates to an ABM (**Figure 2**). All of these items are issues because lack of correlation breaks the chain necessary for maintaining accuracy. Many problems arise from inaccurate foundry models, and designers frequently omit implicit items from simulation netlists, such as package and bonding models, component mismatch, noise parameters, and parasitic coupling, and the Spice model, consequently, is either incomplete or inaccurate.

The step that is most often troublesome, especially when digital designers are using a top-down approach and analog designers are using a bottom-up approach, is Spice simulation to ABM correlation. Designers often overlook a mismatch in this step. An all-too-common example of this omission occurs when an "ideal" DAC in a top-level design runs well through simulation. But when the design undergoes manufacturing, the DAC causes large current spikes on the power when it is clocking, or it takes too long to power up.

Much debate exists in the industry about the amount of design detail and quality designers need to make sufficiently accurate ABMs. For example, an ABM modeling a synchronous interface to the digital core—an ADC or a DAC, for example—may require only a simple model that validates the interconnect, the response to digital controls, and ideal operation. Designers can initially use Spice analysis to gain data on the detailed performance of the block. Designers then must use the ABM to validate control polarity and proper interconnect within the SOC. Designers can derive ABM definitions directly from the bottom-up-design process of multimode and multilevel simulation. Running a testbench with TLD and ABM while developing subsections of a design can give designers an interactive comparison of the two models. Top-down design implies that design teams have created an ABM before TLD takes place, so designers must ensure that the final ABM and TLD plug and play in the same way.

Designers can add even more characteristics to ABMs to quantify the interaction of blocks and ensure that their ABMs thoroughly validate their designs. Some industry participants think that design teams should even create transistor-level ABMs, but others see transistor-level ABMs as redundant. Going to this level of detail puts designers in the "too-complex-to-simulate" cage from which they are trying to escape. They need an accurate, functional black-box model. Beyond digital interconnect and ideal functioning, designers can add other components to their black-box ABMs to improve their quality. For example, they can add loading and source impedances—both entering and leaving the box—to ensure that the model provides a reactive impedance similar to the TLD. Latency is another factor, because a delay always occurs from the input to the output of a circuit. A delay similar to that of the TLD is appropriate. Undetermined state periods, including mode switching, PLL-acquisition times, and the like can lead to "output-not-stable" scenarios. A complex model could mimic these conditions, but conditional error flagging during those periods should suffice. Designers need to check power cycling in the TLD for suitable behavior. After that, the ABM can use error flagging for a predetermined period whenever the power supply

or a power-down control changes state. Designers need to create conditional statements that monitor the acceptable range of power and ground voltages. These languages support the concepts of "analog events" and "event monitoring." In addition, conditional statements monitor the stimulus and loading of the pins of the ABM block. The intent is to provide enough information to validate interconnection and proper functions. Designers need to check each small block in Spice, as well, and to balance the model's complexity with how much simulation time they need to run the more complex models.

SYSTEMATIC TESTING AND VERIFICATION

If designers use top-level verification, they must keep track of multiple models of the same blocks, model correlation, multiple modes of operation, multiple testing scenarios, revision control, and the associated database management. Depending on the complexity of the design, designers may find that manually performing these tasks can be error-prone, cumbersome, or simply impossible to track. Consulting with an expert on how to either streamline or fully automate the process seems a wise course of action. Ken Kundert, one of the founders of Designer's Guide Consulting (www.designersguide.com), specializes in the verification of large mixed-signal chips. While at Cadence, Kundert led the development of Spectre, SpectreHDL, and SpectreRF. He has also been involved in the definition of Verilog-AMS, Verilog-A, and VHDL-AMS modeling languages. Due to the complexity of the process, Kundert's approach includes use of automation

DESIGNERS NEED TO CHECK EACH SMALL BLOCK IN SPICE AND TO BALANCE THE MODEL'S COMPLEXITY WITH HOW MUCH SIMULATION TIME THEY NEED TO RUN THE MORE COMPLEX MODELS.

and a complete set of ABMs, which results in a more efficient process (**Figure 3**). Kundert's approach maximizes the use of software automation, because validation coexists with, rather than is merely a part of, the design flow.

Key concepts in the methodology include a verification plan that defines the models, modes of operation, input stimulus and acceptable outputs; a modeling plan that defines the necessary HDL and ABM models; and a top-level model that becomes an executable specification, which demonstrates the functions and features of the final chip. The methodology also includes a simulation plan that itemizes the tests to be run, the appropriate configuration in each test run; a self-checking testbench for model selection, providing stimulus and response monitoring and parametric pass/fail decisions; and regression testing to ensure that design progress and modifications do not corrupt design integrity and to swap small TLDs with ABMs so that model correlation remains unbroken.

You can find a comprehensive description of these concepts

on the Designer's Guide Web site, but note that it provides only top-level validation. If you use the approach, you need to run TLDs through a complete set of tests before bringing them into the top-level system. The approach does not consider process, voltage, temperature, statistical variance, mismatch, noise, and linearity. Using the approach requires designers to broaden their skills, write models, and understand control and test and verification definition. Designers who lack these skills can hire validation specialists; doing so is more cost-effective than multiple design re-spins.

INSUFFICIENT SYSTEM VALIDATION AND TOP-LEVEL CHIP VALIDATION NOW CAUSE MOST PROBLEMS IN SOCs.

In conclusion, insufficient system validation and top-level chip validation now cause most problems in SOCs. Spice-level validation is more accurate than using simplified ABMs, but Spice simulation is too slow, and most design groups lack the computational power necessary to make it practical to boost Spice simulation and make transistor-level validation of large SOCs. Instead, designers are now developing methods to validate their large designs. A popular choice is to use ABMs, which range in complexity from simple, ideal structures to highly detailed Spice equivalents. Of these ABM approaches, perhaps the most effective is the black-box equivalent in which the ABM and TLD have similar stimulus-response characteristics.

Designers can effectively perform ABM-to-TLD validation on much smaller blocks. For SOCs, however, they should bring the system together with ABMs and multimode, multilevel methods. Doing so ensures viable simulation times and the ability to run multiple simulations at the top of the design. Automating the validation as a separate procedure outside the design flow is a valuable way to expedite the process, reduce errors, and provide a greater probability of success. EDN

REFERENCES

1. IEEE International Behavioral Modeling and Simulation Conference, www.bmas-conf.org.
2. www.verilog.org/verilog-ams.
3. Lynguent, www.lynguent.com.
4. Designer's Guide Consulting, www.designersguide.com.
5. Effective Electrons, www.effectiveelectrons.com/foundry-models.html.

AUTHOR'S BIOGRAPHY

Jerry Twomey is the founder of Effective Electrons (San Diego CA), where he is responsible for IC-product development, including consulting, training, seminars, design services, and technical support. Twomey received bachelor's and master's degrees from the Worcester Polytechnic Institute (Worcester, MA). His personal interests include sailing, photography, hiking, camping, and inline skating. You can reach him at jerry@effectiveelectrons.com.