



# Designer's Guide Consulting

Analog, Mixed-Signal & RF Verification

ANALOG VERIFICATION NEWSLETTER

NUMBER 7, NOVEMBER 2008

## Contents

- 1 - Announcements
- 2 - Analog Verification Benefits IP Re-Use, Top-Down Design, and Platform-Based Design

## Greetings

Dear friends and colleagues,

Beyond analog verification (AV), we are often asked our thoughts on approaches to improving analog and mixed-signal *design* efficiency, such as IP re-use, top-down design, and platform based design. Analog verification does not require any of these approaches be used, but what we have found is that AV forms a symbiotic relationship with these approaches. When not in place, AV can kick-start the use of these design efficiency methods; and once adopted, their use feeds back to help AV. This is the subject of this month's newsletter.

After a successful AV class in September, we've decided to offer our 4 day AV training class again 27 January to 30 January, 2009. Details can be found below. What you can expect to learn from our class can be found in our July, 2008 newsletter in an article entitled, *Introducing Analog Verification* ([www.designers-guide.com/newsletters/0807](http://www.designers-guide.com/newsletters/0807)).

We'd also like remind our readers, that besides consulting services to help your company transition to applying analog verification, we now also offer modeling services. Details can also be found below.

As always, feedback is greatly appreciated.

Sincerely,

*Henry Chang and Ken Kundert*

## Announcements

### *Verification Training Class*

*27 January — 30 January, 2009 in San Jose, California*

This challenging four day course provides participants with the tools they need to take on the task of verifying complex analog, RF, and mixed-signal integrated circuits. It combines lecture with a substantial amount of time in the lab to teach the overall analog verification process. You will learn how to develop a verification plan, functional models of analog blocks, regression tests for those models, and a fully verified Verilog model for the entire analog portion of the design for use in chip-level verification.

*Target Audience.* The class is intended for anyone who would benefit from a working knowledge of analog verification. These include: analog verification engineers, analog designers, analog design leads, and digital verification engineers and CAD engineers who meet the prerequisites.

*Instructors.* Ken Kundert and Henry Chang.

*Prerequisites.* Students should have a working knowledge of Verilog-A, analog circuits, and the Cadence design environment. It is also helpful to have gone through Verilog-AMS training. The better prepared you are, the more you will get from the class.

*Cost.* The class will be held from 27 Jan. — 30 Jan. in San Jose, CA. The price is \$2250 until Jan. 7, at which time it becomes \$2600.

For more information or to sign up, visit [www.designers-guide.com/classes](http://www.designers-guide.com/classes).

### ***Modeling Services***

Again, we'd like remind our readers, that besides consulting services to help your company transition to using analog verification, we now also offer modeling services. We follow strict guidelines and best practices to ensure that high quality and efficient models and test benches for analog and mixed-signal blocks are written. Please contact us if you have such a need.

E-mail [consulting@designers-guide.com](mailto:consulting@designers-guide.com).

## ***Analog Verification Benefits IP Re-Use, Top-Down Design, and Platform-Based Design***

*By Henry Chang and Ken Kundert*

Analog, mixed-signal, and RF design companies have always sought to improve design efficiency. For over a decade, several promising approaches to increase efficiency have been discussed, and to some extent, adopted by some companies. These approaches are IP re-use, top-down design, and platform based design. Analog verification (AV) is a different enhancement to the design process. Analog verification focuses on finding bugs in the design and is ideally performed *independently* of the design effort. Specifically, AV focuses on finding bugs in complex mixed-signal chips at the chip level, the interface between digital and analog, in the RTL that works with the analog, at the analog subsystem level, and within the analog blocks. As part of AV, validated models of the analog blocks are created. In the overall verification effort, these models can be used by other verification groups or the designers to search for more bugs or design issues. It is these models, the byproducts of AV, that can kick-start or enhance IP re-use, top-down design, and platform based design. The cost of these approaches is reduced because models are developed as part of AV. The quality of the approaches are improved as the models of the analog, mixed-signal, and RF are validated to be functionally consistent with the design implementation, i.e. the schematics. Platform based design also benefits from the regression tests created. AV benefits from the use of all of these techniques, and thus, AV forms a symbiotic relationship to these design approaches.

Analog/mixed-signal/RF IP re-use is applied in two basic forms. The first is re-using blocks from other chips within a company. The second is buying IP from a third party vendor. In both cases, there is limited time to understand the IP that is being integrated. When buying IP, there might also be limited expertise in using analog IP. As a result, a black box approach is most preferred when re-using IP, where there is a well defined set of data being delivered from the IP supplier to the IP integrator. When buying IP, data is usually provided in a more standardized form and is likely to be more complete, as it is often difficult to discuss issues with the designers of that IP or



have them help directly with the integration. Although, it would be preferable that all IP is well packaged, companies seldom have the time or the money to invest in packaging their own IP for internal re-use. A reasonable list of the types of data needed was developed by the VSI Alliance [1]. A subset is listed here:

- User guide / specifications
- Process definition (requirements, tolerances, sensitivities, etc.)
- Functional / timing digital simulation model
- Block level HDL simulation model
- Digital placeholder (“dummy”) model
- Digital timing model
- Detailed transistor/gate level schematics (optional)
- Circuit level simulation netlist (optional)
- Test support information
- Physical block implementation (gdsII, placement information, etc.)

Generating most of the listed items is straightforward. For example, it is well understood how to deliver layout information. Writing a user guide or specifications may be cumbersome, but it can be done. However, in practice and while the VSI specifications were being developed, one thing that was not well understood was how to develop a validated functional “digital” model of the IP. The purpose of which is so that the user of that IP can successfully design to the IP, and verify that the IP block will work with the rest of the chip. The digital designation was meant to indicate that the model should be written in a digital high-level description language (HDL) such as Verilog or VHDL allowing maximum compatibility and speed with the verification environment of the IP integrator. Ideally, analog behavior is included and needs to be behaviorally modeled with the digital HDL. Another deliverable that was also not well understood in terms of how to develop was the block level HDL simulation model. The intent of this model is to model the complete functional behavior of the IP and possibly some of its performance at a more detailed level. This was specified to be written in Verilog-AMS or VHDL-AMS, languages more amenable to modeling analog and mixed-signal behavior. The uncertainty wasn’t so much in how to write functional models, but how to make certain that the models written are consistent with the design. Analog verification solves this for both models. A basic part of AV is to write models that represent the design implementation or schematics. This is shown in Figure 1. To validate a model, we write a self-checking regression test that is applied to both model and schematic. The regression test returns a “pass” or “fail” result. If the regression test passes on both, then we claim that the model and schematic are functionally equivalent. If, in this case, more than one model is required, such as one in Verilog and one in Verilog-AMS, then the regression test is applied to both of the models and the schematic. If the regression test passes on all three, we claim that the models and schematic are functionally equivalent.

Thus, if when designing the IP, AV is applied, validated models are produced that can be delivered to the IP integrator. For legacy IP, one would need to prioritize, write models and regression tests as needed. In this situation, AV provides the rigor to validate the models. Certainly, AV benefits from having validated models of the IP. As the IP is integrated, the validated models can be used as part of the chip level verification effort. When using any model that is not validated, there is no guarantee that we are verifying the implementation.

Some choose to integrate IP without a model. In this case, one is relying purely on the specifications of the IP to design and verify to it. This is error prone. We’ve also seen cases where a non-validated model was used. In this case, one of the modes of the circuit was not modeled correctly.



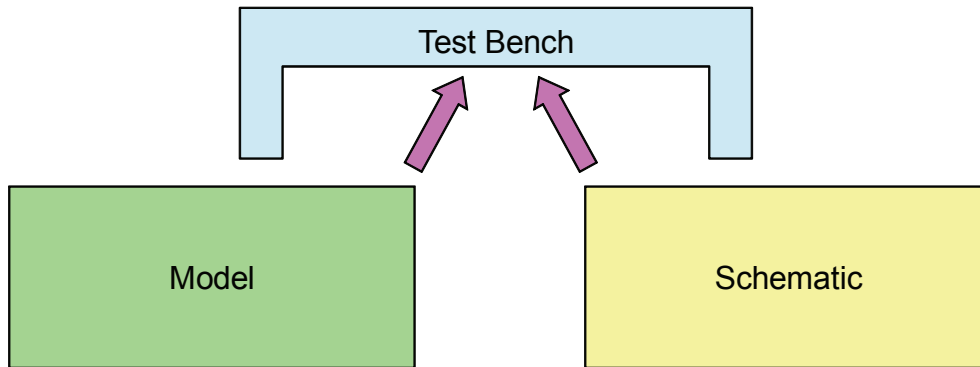


Figure 1: Validating models and schematics as being equivalent.

As a result, the RTL was designed to work properly with the model, but not to the actual implementation.

Applying top-down design (TDD) in analog is problematic for many reasons. First, there is no one size fits all solution. Each design group has to figure out what will work and what will not for their team and the designs they are doing. It is also not an all or nothing methodology. Some design groups will find a certain level of “top-down” is sufficient for their needs. In beginning to adopt TDD, these issues create significant unknowns. There is never a good time to start. There is seldom time to try a new approach on a production design. Fortunately, AV does not require a top-down design flow. We’ve applied it to many design types employing many design methodologies. AV layers over an existing flow with little disturbance. AV can be applied and can itself justify the return on investment (ROI) in terms of finding bugs to prevent respins. AV does, however, tend to naturally foster top-down design. AV provides models that can aid TDD. Since few, if any, designs start from scratch, some part of a prior design will be reused. If AV had been applied on that prior design, models will be available for the next chip and one can easily explore the use of TDD. An example of a model that is natural for a TDD flow is the one published in our previous newsletter ([www.designers-guide.com/newsletters/0807](http://www.designers-guide.com/newsletters/0807)). In system design, a system designer can quickly explore different coefficients or see how the equalizer works in a feedback system. The vast majority of models produced as part of AV are functional models rather than performance models. Performance models are only used on an as needed basis. Functional models generally are simple to modify for design exploration purposes. In a TDD flow, these models serve to close the gap between system design and circuit design. Because AV naturally produces models, the library of models associated with the library of blocks to be re-used can significantly lower the barrier to trying TDD. Thus, without investing in a full blown effort to move to TDD, design groups can move to a top-down approach opportunistically. With each design, more models are created and with each subsequent design, more TDD techniques can be applied. TDD also aids AV. TDD helps to provide top-level schematics earlier with a focus on defining the interfaces earlier. This allows AV to focus on the chip level verification earlier during the design cycle. This results in significantly better chip level regression tests. AV can help to kick-start TDD, and TDD can significantly improve AV coverage, especially at the chip level.

In platform based design or where a large product family or many derivative designs are planned based on a set of key components and an architecture, AV serves as a strong tool in making certain that each design is functionally correct. Further AV serves to significantly improve the efficiency of verification as the knowledge gained in verification is captured and leveraged from one product to the next.



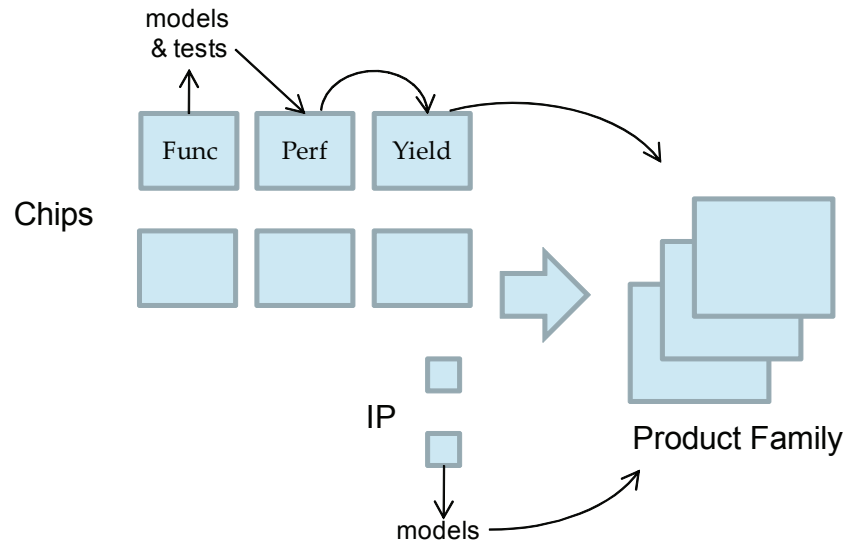


Figure 2: Chip design cycle and derivative designs

First, we discuss how AV aids in the chip design cycle itself for a chip that is to go into high volume production. In Figure 2 in the upper left, a typical design cycle is shown where several revisions (respins) are planned as the chip goes into high volume production. The goal of the first revision is for the chip to be fully functional. A good faith attempt is made toward achieving performance and yield goals, but seldom is only one revision required. The intention of this first revision may be to ship the part to the system customer so that they can begin to bring up the system, putting it in context with the other hardware and writing embedded software for the chip. The next step is to tune the design for performance and yield. Once the performance specifications are fully satisfied, a third revision may be required to improve yield to squeeze the last bit of cost out of the chip. In this scenario, it is critical that the first revision be functional, since the system team is depending on using the chip for development. AV helps achieve first silicon success. As the chip goes from one revision to the next, if only minor changes are made, only minor changes need to be made to the models and regression tests. However, one can still apply the full set of tests. The end result, is that with little additional effort, the same level of confidence for first silicon success can be systematically achieved in subsequent revisions. It is very undesirable that during a performance or yield tuning step to accidentally introduce a bug. AV helps to ensure that this does not happen. Additionally, through each of the revisions, the regression tests can be enhanced to increase the coverage of AV. Without a rigorous, systematic, and repeatable AV methodology, even if the number of design changes are few, a tremendous effort still has to be applied in verification. Since our regression tests are self-checking, we can be assured that the same or better level of tests have been applied in each subsequent revision. AV is always leveraging its past. This is especially important if the chip and derivative design is to last a number of years. The design and verification team often changes. The verification engineer assigned in the initial revision of the chip, may not be around to work on the final revision. AV, in its use of self checking regression tests, allows a different verification engineer to run the tests than the person who wrote them originally.

Usually, the reason for going to platform based designs or building derivative designs is to leverage prior work to minimize effort and reduce risk. In this case, AV provides models to increase design efficiency while the regression tests aid in risk reduction. As shown in the Figure 2, often the goal is to combine several chips and IP into a product family. In this case the models and





regression tests from previous design efforts serve as the verification building blocks for the next level of integration and for building more extensive regression tests. An example is the regression tests that talk to the RTL that affects and observes some analog behavior. As more analog is integrated, how one programs the RTL often remains the same with only additions to the register map. In this case, the original regression test can serve as a building block where additional tests are added to talk to and observe the new analog blocks being integrated. As the platform aids in design re-use, AV helps in verification re-use. Both the models and regression tests are leveraged from subsequent verification efforts building on the verification knowledge base.

There are many challenges to adopting IP re-use, top-down design, and platform based design. AV helps to overcome some of the challenges by providing validated models with no additional cost (since AV pays for itself). When it comes to platform based or derivative design approach, the models serve to accelerate the design process while the regression tests align with the risk reduction goals. AV directly benefits from all three approaches. Our recommendation for any design efficiency improvement, is start with AV. If IP re-use, top-down design, or a platform based approach is not being used, try using the models created as part of AV to kick-start the use of one of those approaches. If one or more of these approaches are already being used, we believe that AV will naturally see benefit and IP re-use, top-down design, and platform based design will also see benefit.

## ***References***

[1] *Analog/Mixed-Signal VSI Extension Specification*, AMS Specification 1, Version 2.2, VSI Alliance, [www.vsi.org](http://www.vsi.org), February 2001.

Disclaimer: We strive to provide information that is both helpful and accurate. Designer's Guide Consulting, Inc., the creators of this newsletter, makes no representation or guarantees on the newsletter contents and assume no liability in connection with the information contained on it.

Copyright, 2008 © Designer's Guide Consulting, Inc. No reproductions of this newsletter can be made without the expressed permission of Designer's Guide Consulting, Inc.

