Designer's Guide Consulting
Analog, Mixed-Signal & RF Verification

ANALOG VERIFICATION NEWSLETTER
NUMBER 4, OCTOBER 2007

Contents

Greetings

Dear friends and colleagues,

This month's article, "The Cost of Failure," discusses potential business and personal costs due to a failed chip. In "Responses from Our Readers," we introduce the topic of regression testing. Look forward to an entire article on analog/RF/mixed-signal regression testing in a future issue.

We have been asked why we go to the effort of creating this newsletter and providing information on our website. We firmly believe that for certain analog/RF/mixed-signal designs analog verification is critical, and that the number of designs for which this is true has hit a turning point and is rapidly growing. We want to help chip companies usher in this new approach. We hope that the materials provided help to serve as a starting point for those embarking on analog verification.

We also write this newsletter to remind our readers that Designer's Guide Consulting, Inc. is dedicated to helping this trend emerge by providing consulting, methodology assessments, training, and project work. As with any other discipline such as analog circuit design or digital verification, it is experience that makes endeavors successful. Whether you're just starting out or whether you want to refine the way you write your verification plan, models, or regression tests, having applied analog verification on a variety of projects including baseband analog, RF, and high speed digital communications, we can help. Please send us mail at *consulting@designers-guide.com*. We look forward to hearing from you.

As always, feedback is greatly appreciated.

Sincerely,

Henry Chang and *Ken Kundert*

Announcements

Analog Verification Seminars in Austin

13-14 November 2007 in Austin, Texas

Veriseo Engineering Services, a provider of high-end consulting/contracting for functional verification in the digital domain, in partnership with Designer's Guide Consulting would like to invite you to attend a very interesting presentation and discussion on new techniques in the verification of analog, RF, and mixed signal designs. We have planned for 4-5 presentation slots over

a 2 day period. Please contact Veriseo ASAP if you have executives, senior management, or technical leads who would be interested in hearing this talk.

For more information visit *www.designers-guide.com/partners/lawell.html*.

Ken to Speak at Dallas Circuits and Systems Workshop (DCAS-07)

16 November 2007 at Southern Methodist University, Dallas Texas

Ken will be giving a keynote presentation on analog verification at the Sixth IEEE Dallas Circuits and System Workshop.

For more information visit *www.designers-guide.com/partners/dcas.html*

Designer's Guide Community Opportunities Page

Analog/RF/Mixed-Signal engineers are currently highly in demand. We receive at least one call a day from recruiters asking for engineers. We have decided to start an opportunities page that lists jobs for analog/RF/mixed-signal professionals. If you are interested in looking for a job, our goal is to give you a great starting point to see the wide variety of jobs available. We are just starting, so we don't have too many listings yet. We do charge a fee for the posting. This helps us support the website, and also ensures that the people who post are serious about looking. If you are interested, please don't hesitate to contact them. Even if you don't send in a resume, I think many of the people who post would be interested in just talking to you.

For more information visit *www.designers-guide.com/partners/dg-opportunities.html*

The Cost of Failure

By Henry Chang and Ken Kundert

As we have discussed in earlier articles and in our papers, failing to apply analog verification in complex analog/RF/mixed-signal design can lead to fatal errors that can stall the development cycle. The chip either has to be re-spun or costly and time consuming focused ion beam (FIB) repair work needs to be conducted before the chip can go into production or be given to the end customer for system testing and embedded software development. What is often surprising is the lack of understanding of all of the consequences of a failed chip, and the unwillingness to change behavior to avoid these consequences. In this article, we will attempt to cover these consequences in quantifiable terms. We will discuss the cost to the company in terms of:

- Additional non recurring engineering (NRE) cost
- Revenue loss
- Market loss

We will talk about the personal cost to the analog, mixed-signal/RF design engineer, design lead, or manger/VP, and founder of a startup in terms of:

- Financial loss
- Personal reputation loss

Before we begin, we want to make an important disclaimer about the numbers we provide. We believe it is important to use numbers to give a sense of the order of magnitude and to allow for relative comparisons. We have attempted to use representative numbers, but particular circumstances can significantly alter these numbers.

The most well understood cost is the additional NRE cost in terms of a re-spin. For a few functional errors, the re-spin costs of changing 3 or 4 mask layers in a 0.18u or 0.13u process is around $50k-$100k. This cost depends on the number of mask layers that needs to be changed, the mask grade levels, and of course, the cost only becomes higher at 90nm, 65nm, and beyond. More significant is the additional engineering time required. Bugs in silicon take much longer to find than in simulation by a factor of three or four. The typical time for a re-spin is 2 to 3 months. Suppose it takes two months for two engineers including the design lead to debug the chip, and suppose their fully loaded cost is $200k/year. This translates to an additional $70k. There will be an additional cost for re-implementation if there are significant changes. For example, if the change impacts the digital portion, then additional place and route, timing verification, and physical verification need to occur. On top of debug and re-design, there are often six to eight people sitting idle who had expected a working chip. These include system designers, embedded software developers, product engineers, field applications engineers, marketing people, and production engineers. With eight support engineers idle, there is an additional $270k expense. This cost goes up as follow-on projects are delayed. If the lead designer is stuck debugging, a new project could be stalled with the other new project engineers sitting idle or worse, starting the design without guidance. With four engineers idle, this translates to an additional $130k. The total additional NRE cost is now up to $570k. In some situations where the entire company is relying on a first chip, a failed chip could stall the entire company. New projects are postponed while the entire company focuses on fixing the first chip since, failure of the first one means death for the company. The total cost could now go as high as $1M-$1.5M in additional NRE.

On the revenue side, suppose the ultimate goal for this chip is to be the number one supplier in its market segment shipping 2 million units per year at $2.00 per chip. At best, a 3 month delay translates to only a delay of $1M in sales. But even a delay of $333k/month is significant. Growth plans could be put on hold or additional sources of funding may need to be obtained to sustain the existing run-rate. What's worse is if the delay causes the company to drop from the leading supplier position to second place. In the semiconductor business, being second results in significant market share loss and significant reduction is per unit price. Suppose the market leader takes 50% and the second place player takes 25%, and the per unit price drops to $1.50 per chip. Rather than annual sales of $4M / year, annual sales are now $1.5M/year, representing a loss of $2.5M/year.

In the consumer marketplace, the system business is brutal in terms of cutthroat competition. The extreme is in the cellular handset business where handsets are given away for free to secure long term service agreements from the end customer. Tremendous time-to-market and price pressures exist for these systems. To play in this space, a typical system provider will mitigate their risk of having their chip supplier fail by designing three or four virtually identical products but each using chips from the different chip supplier. This scenario is extremely problematic for a chip company who has only one or two initial customers. If the chip company has significant product delays over their competition, the system company is already well positioned to cancel the chip order. So, for the chip company, rather than a graceful loss in sales as explained previously, the loss could now be catastrophic to the design group or the company. Projects could be cancelled. Venture capitalists may stop future funding. In the least, having a fatal error such as having an extra inverter or a miss-wired bus leaves a very poor impression with the end customer which could result in the loss of future business. Design delays are often explained by saying that extra time is being taken to assure quality. Even more credibility is lost if the chip does not function after making such a claim.

To look at personal cost, let's look at the situation where designers have a significant stake in the chip they are designing. This is usually true in a startup where the engineer is either one of the

founders or one of the first employees. To calculate cost of failure, we'll first look at what a founder or engineer might nominally make in a startup. It is important to understand that in order to make the nominal amount, all of the investor expectations need to be met. A set of criteria might be:

- By the end of round A, the technology needs to have been validated by two to three beta customers.
- By the end of round B, production needs to be stable and the company needs to have 10 beta customers.
- By the end of round C, there should be broad product proliferation, and the focus is on scalability.

We'll use a simple model and some assumptions to calculate how much a founder and an engineer might make. Suppose the founders are serial entrepreneurs. They know the right venture capitalists and manage to get good financing terms. They understand what the market wants and focus on a great chip. They decide that rather than making an initial public offering (IPO), they will sell the company after three rounds of funding. Suppose there are 4 founders who raise a total of $50M. They sell the company for $100M. We'll assume each founder owns 3.0% of the common stock, and on average, each engineer owns 0.32%. The result is that the founders each make $1.5M and the engineers each make $160k.

Suppose something goes wrong, and there's a fatal chip error. According to Rich Simone, president of Simone Consulting Services, the unfortunate issue with a chip error is that it inevitably occurs when the company has ramped-up to prepare for production. The company burn-rate is at its peak while the chip is being debugged and re-spun. Unfortunately, when this happens other factors come into play besides the NRE cost. Morale at the company may drop. Employees may leave and time has to be spent hiring and retraining people. And as mentioned before customers may lose confidence and reduce their order size. The competition may take the lead. Suppose an additional funding round is required, assuming the company still manages to sell for $100M, making the assumptions shown in Table 1, this leaves the founders making only $700k and the engineers $77k. If failure continues, the common shareholders will run into another issue, which is that if the company sells for less than it raised, the common shares typically become worthless. On a more positive note, if instead of 3 rounds of funding the company only requires 2, the founders can make $3.2M and the employees, $280k. We have known fabless semiconductor startups to succeed in 1.5 rounds. They had near perfect execution and an exceptional management team.

Case	Amount Invested	Selling Price	Value of Common Shares	Founder's Ownership	Engineer's Ownership	Founder's Return	Engineer's Return
2 rounds	$30M	$100M	$70M	4.5%	0.4%	$3.2M	$280K
3 rounds (nominal)	$50M	$100M	$50M	3%	0.32%	$1.5M	$160K
4 rounds	$70M	$100M	$30M	2.4%	0.26%	$700K	$77K

Table 1: Founder's and engineer's return vs. rounds of funding.

Another key variable to how much the founders and engineers make is the selling price of the company. Table 2 shows a comparison. If because of market share loss, the company can only sell for $75M, the founder's and engineer's return can drop dramatically as seen in the table. Again,

on the plus side, if the company can be sold for more, the founder and engineer stand to make significantly more, especially if the "dot-com" days ever return.

Case	Amount Invested	Selling Price	Value of Common Shares	Founder's Ownership	Engineer's Ownership	Founder's Return	Engineer's Return
Lower price	$50M	$75M	$25M	3%	0.32%	$800K	$80K
Nominal	$50M	$100M	$50M	3%	0.32%	$1.5M	$160K
Higher price	$50M	$125M	$75M	3%	0.32%	$2.3M	$240K
"Dot-com" days	$50M	$500M	$450M	3%	0.32%	$13.5M	$1.44M

Table 2: Founder's and engineer's return vs. selling price.

In a large company, the downside of a failed chip for the analog/RF/mixed-signal engineer is probably not that large. Design jobs are currently plentiful. It likely just translates to more stress.

The final cost is personal reputation cost. This tends to have the greatest impact to the design lead or the VP/manager of the analog group since they are the ones in charge of the project, and because many times functional errors occur at the interfaces which are their responsibility. It's difficult to quantify this, but here are two stories that get at some of the reputation cost. As is often the case in a company, there are many digital design teams while there are only one or two centralized analog design teams. Team leaders meet for planning and status purposes. It was at one of these team meetings that the head of the analog division had to explain why a chip they had all worked on did not power up because of misplaced inverter in the analog section. Normally, the digital designers can't question the analog designers because they don't understand analog. If the chip fails because of some missed temperature or substrate coupling noise effect, it's hard for the digital team to voice any criticism. However in the case of a functional error, they understand. They asked the analog manager, "Don't you have a verification methodology?" After being asked this several times after several failed chips due to functional errors, the analog design manager decided he needed to bring in analog verification. In another story, a design team had just gotten a chip back, but it was dead due to functional errors. But because schedules were tight, the system company executives had already flown in from overseas to plan next steps. With the chip company's customers and the company's own executives waiting outside the lab, the analog chip lead had to scramble to find a way to make the chip do something. In terms of quantification, a VP of engineering once told us that he only hires engineers that have been on successful projects. So, for personal cost, it's not so much that an engineer might lose his/her job, but an opportunity might be lost.

For the company, a failed chip could cost $1M + $2.5M per year. For the founders and engineers in a start-up, it could cost them close to $1M and $90k respectively. Certainly, it will likely cause the entire team stress. For a manager or design lead, it could limit opportunities in the future. In the least there is an "embarrassment" factor. There are many things that can cause fatal errors. The functional errors that analog verification catches are only a subset of these errors. How much to invest should be weighed against the possible downside of not making that investment. Often designers place more value on what is difficult rather than on what is easy. An extra inverter may be easy to fix, but the cost could be tremendous. As they say in digital verification, if it's easy, then check it, and also if it hasn't been verified then it will be broken.

Acknowledgements. The authors would like to thank Rakesh Kumar, President of TCX Inc. for discussions on manufacturing related costs and issues; James Hogan, angel investor and venture capitalist, for discussions on expectations from fabless semiconductor start-ups; and Rich Simone, president of Simone Consulting Services (*rbsimone@pacbell.net*), for his discussions on best practices in operations at semiconductor companies.

Responses from Our Readers

Writing regression tests seems difficult

This reader writes that writing regression tests seems like a difficult task. Using a script-based approach to post process simulation output is cumbersome, especially if the use of a proprietary scripting language not taught in schools is required. He adds that assuring that all inputs and modes have been checked also seems difficult as some type of coverage metric may be needed. Finally, he asks, what happens if there is function modeled but overlooked during the implementation? If the model is solely used for system design, the fact that the implementation is not complete may be missed.

Our approach is that we write regression tests in an AMS behavioral modeling language (e.g. Verilog-AMS). We check results as we go and print pass/fail messages. Since we're primary focused on function and not so much on performance, rudimentary checking of spec parameters isn't too difficult. If for function, more complex metrics such as frequency, amplitude, and phase, need to be checked, this can be done. Our regressions tests require that no nodes be saved requiring no post processing. We only save nodes for easier debugging if something goes wrong.

On the testing all possible inputs, we follow some guidelines. We make sure all the functional aspects in the specifications are checked. We go through all modes/settings and all sequences (e.g. power-up, sleep mode, etc.). We talk to the designers and design lead to understand what they worry about and add more tests based on what they tell us. We also have verification reviews where we try to get more input. There is always a chance we could miss something. That's why it's important to start the regression test development early in the design cycle and continue to refine it as the project continues.

To validate that the model functionally matches the implementation, we run the regression test first on the model, and then on the transistor level implementation. If both pass, we assume that the two are functionally equivalent.

If a model implements more than what is in the implementation, and if the feature is in the specifications, then the regression test will pass on the model, but fail on the implementation. This is how we'll know if the implementation is incomplete.

Stay tuned. We'll be writing a full article on regression testing. We'll also answer the question why we prefer an AMS modeling language versus using a SPICE netlisting language which was another question from a reader.

Is there a conference where the topic of analog verification is discussed?

The IEEE Custom Integrated Circuits conference just had an Education Session partially on this topic. Three of the four speakers in the "Mixed-Signal SOC Design Methodology" session talked on this topic. I'm not aware of any other venue on this topic going forward. If one of our readers knows of one, please let us know and we'll publish it in our next newsletter.

We're considering the idea of starting an analog verification workshop. It'll be a lot of work, and we'll need help. If anyone is interested in volunteering, please send us mail. As with any workshop, we'll need to fill key positions, such as the technical program chair, publicity chair, etc.

In digital verification, we check everything. Why in analog do you only focus on function and not performance? That makes no sense to me.

Being from the analog/RF/mixed-signal design world, this question never even occurred to us. In the analog world, verification has traditionally not been a separate task. The thought of even pulling out functional verification out of the designer's hands is a big methodological change, although one that is definitely beginning to happen. To separate both functional and performance verification would be a bigger change than we think most of industry would consider and it's not clear to us right now that this even makes sense in most circumstances.

We focus on function, because this is where the traditional analog approach is failing. For performance analysis, the traditional analog approach mostly still works. We focus on the practical, justifying the approximate cost of 1 analog verification engineer per 5 design engineers required for analog functional verification. Recently, a design manager explained to us how they needed the verification engineers to validate performance. His design group implemented a unique architecture where traditional analog and RF analyses and tools failed. Their verification engineers built highly accurate models to predict performance. The cost, however, was that they had a 1 to 1 ratio between verification engineers and designers. He volunteered that if he were to do the project again, he would definitely keep the 1 to 1 ratio. So, it's not that we're against modeling performance or verifying performance, the question is, is the cost justified?

What else is there in analog verification besides writing the models?

A circuit designer asked me how is it that our consulting business can make money if one can simply download example models from our website, *www.designers-guide.org*.

Analog verification consists of a lot more than just writing models. We estimate model writing to be 25% to 30% of the effort. Included in this, is making the models pin accurate. The models downloaded from our website are not pin accurate, and do not model all of the modes and settings found in found in today's analog/RF/mixed-signal blocks. The remaining 70% to 75% of the effort consists of verification planning, writing regression tests, running mixed-level simulations, and debugging. There are also critical design dependent decisions that need to be made on what to model and what not to. We see too often that the focus of verification is all on model writing. Without using the entire methodology, there is no assurance that the models are correct. Unfortunately, today's often used approach of validating models is to visually inspect the model and simulation results with the circuit designer. This can easily lead to errors in the model, because it's difficult to go through all the waveforms and it's an error prone process when trying to keep up with last minutes changes in the design. Without this critical step, model writing could actually cause an error, since the system designer is assuming that the model is accurate. As all the steps of analog verification are not well known, our consulting company provides help to companies embarking on analog verification.