

DESIGN CHAIN DISASTERS

Four ways they're caused—and three ways to avoid them.

By Geoffrey James

<http://www.designchain.com/featurearticle.asp?id=2&issue=fall04>

March 9, 2006

Scenario: You're responsible for building a chip that's the future of your fledgling firm. Over a six-month period, you watch helplessly as design glitches and organizational confusion clobber your schedule. You hope, with increasing desperation, that your design will somehow hit the dwindling market window, only to discover, when your chip comes back from the foundry, that it's so full of noise and timing errors that you're back to square one. Suddenly your product rollout plans are looking like the course headings for the Titanic, because in some competitive markets, according to experts, each day's delay can mean a loss of up to \$1 million in revenue. If that's the kind of scenario that keeps you awake at night, you're not alone.

According to Semico analyst Rich Wawrzyniak, 35 percent of ASIC and COT design chains never result in a usable product. That's pretty scary when you consider that failed design chains waste up to \$30 million in development cost, not to mention the lost-opportunity cost from a missed market window. To make matters even more challenging, design chain disasters become more difficult to avoid every day. Today's processes are so interrelated that even a minor problem can snowball. "The issue today is that any one design mistake is more likely to end up as a disaster," says Jordan Selburn, a principal analyst at market research firm iSuppli.

Not to worry. You can mitigate design disasters and perhaps avoid them altogether, first by understanding their root causes and second by ensuring that those root causes don't develop within your chain. Here are the four root problems that cause design chain disasters and what you can do to make sure that you're not the one going back to square one.

Root Problem #1: Bad Partnerships

The quickest way to create a design chain disaster is to have a weak partner at one of the critical links.

For example, one of the most important partnerships in a design chain is the one between the design engineers and third-party IP suppliers. Because of this, the selection of an IP partner can be a "huge minefield," according to Selburn. "I know of one design team that picked a small IP supplier whose RTL did great in simulations, but when they actually laid the IP onto the silicon, the resulting chip was rife with timing problems," he says. And those kind of timing problems can be major bad news, according to Mark Templeton, CEO of IC vendor Artisan Components. "Timing errors may appear only at certain temperatures and voltage points and may not show up consistently on all parts of the chip," he says. "This makes them very difficult to debug."

Unfortunately, although there are thousands of reliable firms in the semiconductor industry, there are also firms that overpromise and can't deliver, says David Wyland, a 35-year industry veteran who has held technical and management positions at Advanced Micro Devices and SRI International. "There are a lot of firms out there that give good PowerPoint presentations and promise technology that does everything except toast bread," he warns. "But when it comes to actually delivering silicon, you can basically write them off."

Kamalesh Ruparel, director of engineering for the Internet Switching Business Unit at Cisco Systems, recounts that he once relied on a small IP supplier for an auxiliary chip that Cisco needed in order to complete a system design. "Although the demonstration chips tested OK, we saw a 15 percent

failure rate when the chip went into production," Ruparel explains. Unfortunately, Ruparel was stuck with a partner that had presented itself as being fully capable of delivering but lacked the personnel to deliver on those promises. "The company didn't have the right people to fix the problem," he says. "It was basically in this business just to make a quick buck."

The best way to avoid bad partnerships is to do complete due diligence on all partners prior to the launch of the design, according to iSuppli's Selburn. "The penalty for picking a bad partner is simply too large to be tolerated," he says. "You absolutely must spend the time to make sure your partners are top-rate and can deliver what they say they can."

Root Problem #2: Communications Breakdowns

Partnership is only part of the equation. Communication is vital to the health of any partnership, and a design chain can quickly go awry if the partners in the chain don't communicate frequently and frankly.

For example, when Nvidia tried to create its NV30 next-generation graphics chip, manufacturing problems delayed the chip's release. This, in turn, caused Nvidia to miss a product cycle and lose market share to its competitors, according to Erach Desai, an EDA analyst at American Technology Research. "Nvidia came up with a chip that wasn't very manufacturable, very reliable, or a killer on performance," he says.

The Nvidia debacle provides an excellent illustration of the potential financial impact of a design chain disaster. From 2000 until the end of 2002, Nvidia's revenues were growing relative to those of its major competitor, ATI Technologies. However, after the NV30 ran into trouble at the end of 2002, Nvidia's growth flattened while ATI's revenue spiked upward as ATI gained market share. If the NV30 had been successful, ATI's growth would have likely gone to Nvidia, in which case Nvidia would have booked an additional \$363 million or so for the year.

The source of the Nvidia's problems appears to be a communications breakdown between the design team and the foundry, according to independent semiconductor industry analyst Peter Glaskowsky. "Nvidia was moving to a new node at a time for which Taiwan Semiconductor Manufacturing Company (TSMC) wasn't completely ready to deliver a fully debugged process," he says. Glaskowsky insists that "it would be a mistake to blame the entire problem on either TSMC or Nvidia," but there's no question that the disaster would have been less likely had there been better communications about the limitations of the evolving process technology and the full scope of what Nvidia's designers were attempting to accomplish. Without effective communication, problems are inevitable with projects that target a new process technology, according to Patrick Lin, chief SoC architect at foundry United Microelectronics Corporation. "The more advanced the technology, the more a foundry needs to work in partnership with the designers," he says.

Communications breakdowns anywhere in the design chain create a potential for disaster, according to A. J. Sen, CEO of Astro Semiconductor, a supplier of analog IP. He describes a customer that was so secretive that it ignored Astro's offer to help integrate the IP Astro had supplied. "The result was a chip with a completely unacceptable amount of noise," he says. Fortunately, Astro was able to fix the customer's problem before it became a full-blown disaster, but Sen insists that it was a close call. "If the IP vendor does not participate fully in the integration and the customer does not have enough depth to ensure an optimal integration, you can end up with a chip that simply doesn't work," warns Sen.

"We added a new feature a month before tapeout, and that was the only area where we ended up with a functional bug." Kamallesh Ruparel, Cisco Systems

Cisco's Ruparel remembers a situation in which a lack of communication resulted in a double disaster. "The design engineers passed a design to the test engineers that contained a pretty significant bug," he explains. "By the time the test engineers discovered the bug, the design engineers were already

at work on the next generation of the chip." The first chip had to be delayed while the problem was fixed, but the designers were treating the test lab as a separate organization rather than as part of the team. Because of this, word of the bug never got back to the designers. Because of this communications breakdown, the designer engineers put the exact same bug into the second version of the chip. "It was a total waste of time and effort to fix that bug a second time," complains Ruparel.

The key to avoiding communications problems is to make sure that every design chain includes a generalist, an engineer who may not be deeply skilled in any particular discipline but who understands the basics of all the processes in the chain. Generalists act as an early warning system, because they have just enough knowledge to know when one of the teams in the chain is beginning to have problems. "The presence of someone who has a working knowledge of the entire process, from Verilog to place/route, is a good predictor of project success," says Gary Smith, an EDA analyst at research firm Gartner.

Root Problem #3: Feature Creep

Even with strong partners and great communication, a design chain can falter if the engineers keep adding features to the design.

Industry veteran Wyland, for example, once worked on a microprocessor project in which the engineers were encouraged to cram as many features as possible into the design. "The attitude was that if a few features are good, then more features are even better," he says. Unfortunately, all those new features kept raising the design complexity, which resulted in schedule slips, which, in turn, resulted in the assignment of additional engineering resources. After all the delays and all the extra engineering costs, the end result was "a chip that was too big, too hot, and too slow," says Wyland.

Even a single feature added at the wrong time can result in a disaster, according to Ruparel. One of his teams once tried to leapfrog the competition at the last minute. "We added a new feature a month to the ASIC just before tapeout," he says. To add the feature and still make the deadline, Ruparel was forced to assign additional engineering resources, adding to the overall cost of the project. To make matters worse, "that was the only area where we ended up with a functional bug," says Ruparel. He then had to assign even more engineering resources in a last-minute attempt to fix the problem. In the end, all the extra work was for naught, because "we ended up not shipping the feature at all, given that we simply couldn't get it to work," he says.

Feature creep clobbers design chains, because it inevitably results in schedule slips, which, in turn, cascade down the design chain, affecting the schedules of all the downstream partners. For example, Gartner's Smith recalls a situation at a chip design firm that slipped a key milestone, thereby throwing an entire project into jeopardy. The company needed to demonstrate a chip at a major industry trade show, but the schedule slip made it impossible to come up with the product in time. In order to get something ready for the show, the company had to completely redo the design, at enormous cost. "It did it by moving to a bigger die size and by utilizing only 34 percent of the die," says Smith.

Because the demo chip was not suitable for production, the company was then forced to return to the original design to create the actual production chip. "It had to get down to the original estimated die size to shrink the cost to the original estimate," says Smith. As a result, the firm ended up paying for additional packaging design work and incurred extra nonrecurring engineering costs. Although the project met its product launch date at the industry trade show, the production of the chip was delayed by about a year.

The best defense against feature creep (and resulting schedule-slippage cascades) is a project manager with the clout to say "no" and make it stick, according to Astro's Sen. He defines the ideal project manager as somebody who has been with a company for six to seven years, has successfully

taped out chips, understands the methodology, and also has strong management skills. "Poor project management is the kiss of death," Sen insists.

Root Problem #4: Management Meddling

Management, of course, is supposed to help design chains avoid disasters, but alas, that's not always the case.

In fact, management interference is a major source of design chain disasters, according to Jerry Worchel, senior analyst at market research firm Instat. He tells of a CEO who mandated the use of IP from a particular vendor with which he had had a prior relationship. "The order came down: 'You're going to use a core from this company,' without the CEO's having bothered to consult the design team," says Worchel. Unfortunately, the IP never worked right, so the chip was full of problems and never got to market. "It's a big mistake to dictate design decisions from on high," warns Worchel.

Management meddling can even prevent a good design chain from ever getting a chip out the door, according to Nick Tredennick, a venture capitalist who has provided seed capital for several semiconductor firms. One of those firms is Quicksilver Technology. According to Tredennick, Quicksilver was originally mandated to create a configurable chip set for cell phones and, in order to accomplish this, hired a star-studded staff of engineers, including (coincidentally) David Wyland.

However, before the chip was completed, the firm's top management decided to franchise the company's technology, much as IP vendors license their designs. Supporting two wildly different business models stretched the firm's engineering resources so thin that the design chain completely broke down, according to Tredennick. "It ended up with not enough designers and too many verifiers," he says.

In the end, neither business model was successful, according to Andy Krumel, Quicksilver's director of tools engineering. "No, we never came out with that chip," he admits, adding that the franchise scheme is now "an artifact left over after a change in business model." Straddling two business models burned through \$82 million. The company, founded in 1998, is now running on what is likely a final \$5 million from its investors, has laid off all but around 20 employees (down from 120 at its peak), and is now desperately trying to sell off its technology, according to Tredennick. "It's a sad failure for a company that held some real promise," he says.

When ill-considered management meddling comes from the very top, there's probably no way to avoid a disaster, even with a product manager who can say "no" to top management. Most firms, however, don't give project managers that level of authority, according to Sen. "In most design firms, the project managers should probably be making more money than the CEO, but that's seldom, if ever, the case," he says. Avoiding design chain disasters isn't brain surgery. "It's really a matter of paying attention to good management practice," says Ruparel. "It sounds trite, but successful design projects really do depend on the basics, such as team building, training, and keeping communications channels open." Something as complex as today's design chains will always have the potential for disaster, but if you're willing to confront and neutralize the root problems, there's no reason why you should be losing sleep worrying about them.

ABOUT THE AUTHOR

Geoffrey James writes frequently on EDA for Electronic Business. His work has also appeared in Computerworld, CIO, Red Herring, Upside, Business 2.0, and Men's Health.