The real costs of failed design collaboration

By Ben Jordan, Altium

This article explains in detail the need to change the engineering world of our fathers, so as to keep pace with the challenges of the future and be able to design the next generation of connected electronic products.

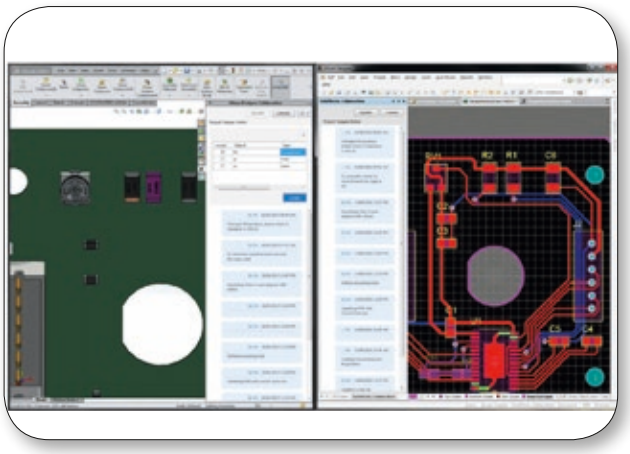

Figure 1. Example of a parallel design workflow

All over the world, engineers are tasked with designing ever denser, smaller, and smarter products. To design something that is small and powerful requires a design process that is tightly aligned across all engineering domains. There are a number of product design trends underway – below are just a few that are directly impacting how engineers work together:

Look at the news any day of the week, and you're bound to see a new story about self-driving vehicle technology. This surge in sophisticated electronics introduces a need for a workflow that tightly integrates the electrical and mechanical sides of the design process. Not only do engineers have to deliver a finished product, but they need to deliver one that adheres to the strict safety requirements for human transportation.

We're nearing a future with growing dependency on connected products where every product we interact with will transmit some form of data. What was once considered a dumb product, like a light bulb, will transmit data about its current state. Forgot to lock your house? Do it from your smartphone, while you're at work. The way we interact with the basic physical objects in our world is going to change in a big way as electronics becomes integrated into everyday household products.

With the release of the Apple Watch, wearable technology has finally entered the realm of mainstream technology. And with this surge in interest comes an even greater demand on design processes. Not only are these devices small, they also are flexible, and PCBs need to accommodate this with rigid-flex sections.

Despite the reality of the changing product experience, the same tired design practices that focus not on collaboration, but on treating everyone involved in the design process as an isolated specialist, are employed day in and day out. In many ways, companies are still clinging to the ideals of the industrial era of engineering, when what is needed is quite the opposite. We're still relying on the same technology that was introduced decades ago, such as the following.

Interchange file formats. We know these well because many packages of data are tossed back and forth between electrical and mechanical design teams. In goes the intelligent and complex design intent of a PCB into a black box, and out comes a basic board shape that communicates zero design intent to the mechanical designer. This process occurs every single day, in every engineering workflow. STEP files, IDF, DXF – the design intent gets crammed into these boxes and sent out to other engineering teams. And what happens when they get it? They have to tweak the data before it ever works. Spending hours trying to adjust that PCB or mechanical data before it's in a usable state.

Linear design processes. Companies are still trying to survive on the industrial engineering era's design methods. Assembly lines worked great, back in the days when products were static and standalone, and thus each part of the design process was easily replaceable. We wanted more, and the design processes delivered. Now engineers are building intelligent products that communicate, yet design teams don't. We design our particular aspect of a design with our specialized knowledge, pass it down the line, and forget about it until there's a problem.

Unmanaged communication methods. Trying to explain the complexity of a PCB in a disconnected series of one-way communications like an email thread is wildly inefficient. How has this ever worked? Even worse, sometimes an email isn't received on time. Maybe that needed revision to the placement of a component was buried somewhere in our inbox. All the while, we keep on designing, with our communication methods trying to catch up, but more often holding us back.

The true costs of failed design collaboration.

Everyone involved in the electronics design industry is familiar with these approaches, and we're all guilty of relying on them every single day. Every week, countless hours are spent patching the holes in design workflows, fixing mistakes, working overtime and maybe even weekends because of them. Yet the engineering industry as a whole has done little to respond. Next year there might be a better interchange file format, but that's entirely missing the point. We're all part of this problem, and we are becoming all too familiar with the real costs of our failed design collaboration processes. First: missed time to market and budgets, with design revisions slipping through the cracks and prototype costs skyrocketing from failed communication processes. Second: wasted time and productivity with designers having to manage multiple revisions that could have been solved the first time with a properly implemented collaboration system. Third: product experiences that are compromised during the design phase based on budget and time constraints vs. being iterated to perfection. The pain is clear, the wasted time is ever-present, and the one thing we all want to know is: what are we going to do about it?

I'll say it again: We don't need another interchange file format. If you've been raised on old-school engineering tools like most of us have, then you've been forced to accept interchange file formats as a part of life. But this method of design is flawed, and it completely ignores the real needs of true design collaboration. What is needed are intelligent design tools that allow for communication between one another. Intelligent tools that don't require engineers to shove data into a box and pass it along. With these tools, there are no boxes – there is no data translation. In this collaborative engineering process, data is being shared and transmitted instantly between design platforms, across a diverse range of engineering domains. To achieve this, our design tools have to change first, in big ways. Here's what's needed: Bi-directional data synchronization. These tools n eed to share data seamlessly, without requiring any kind of interchange file formats. What does this mean in a practical scenario? Being able to commit changes between design environments, and have those changes instantly transmitted to our fellow engineers. It can be as simple as this if I move a component on a board it's going to possibly affect t he mechanical e nclosure t hat my MCAD engineer designed. The only way to efficiently keep the MCAD engineer in the loop about this change is to push the change to his design environment. Allow him to see the revision in his own workflow, so he can adjust his design accordingly.

Not only does data need to be shared, we need to be able to add the human element of communication into the mix outside of the unmanaged channels that we rely on. Within our design platforms, we need a connected and universal communication environment that allows us to clearly articulate the design revisions that have been made and share those details with others involved in the design process. What does this look like in a practical application? Like the example, an engineer makes a change to the component placement on a PCB, and not only is this change pushed to the mechanical designer environment, but the electronics engineer is also able to include a detailed note of exactly what was changed, and most importantly, why it was changed. This is the way forward for effectively communicating design intent.

As our tools evolve to provide us with the data synchronization and commenting abilities that are needed, it's also necessary to work on the design workflows. As engineers, we need to develop a holistic understanding of the entire design process, and understand how our specialized applications will affect the entire product workflow. Simply put, we need to begin working in parallel on our designs. Not only will this enhance our ability to communicate, but it will make our design process that much more efficient.

We need to start developing these solutions now. It won't happen overnight, and it's going to be a slow transition. But the reality is that technology is just going to become more complex. Products are going to keep getting smaller, thinner, and faster than we could have ever imagined. Do we really want to be relying on interchange file formats 10 years from now? Trying to communicate layers upon layers of complexity in a chain of emails and translated file formats? We don't. We need something more intelligent, and we need our design tools to finally catch up to this information-rich era of technology in which we all live. ■