CREATING CONNECTIONS

between electronics design and manufacturing

PCB DESIGN

IS JUST NOT WHAT IT USED TO BE.

Even though it has dramatically evolved over the last couple of decades, it has always tended to be a singular, focused task that a board designer can get stuck into while surfacing periodically for coffee and new design information.

Perhaps the first challenge in tackling today's advanced designs is losing that comfortable concept of an isolated process that that is not directly dependent on other parts of the product design process. On the contrary, the board design in today's leading electronic products is now inextricably locked to the mechanical and 'soft' embedded parts of the design, so all design disciplines need to work together in a collaborative way. The hard facts are that design information, and perhaps even your coffee, need to be shared in a cooperative product design environment.

This is not a sudden change. The evidence has been progressively appearing in the way board designs and the components they hold have evolved. Along with the expected changes in scale that are so familiar in the electronics domain – boards becoming smaller and more complex, component pin counts increasing, signal clock speeds rising, and the number of board layers growing – a number of other changes have crept in that don't necessarily follow the linear, 'smaller-faster-denser' evolution sequence.

These are differences in board design that are less directly coupled to semiconductor technology advances, and more to how product design itself is evolving from the sum of several independent design silos to a single collective process.

The clues to this change are the other shifts in board design such as complex PCB shapes, the introduction of flexible board substrates, the adoption of large-scale programmable devices and, in an increasing number of current designs, an actual decrease in component count.

The drivers of these developments can be grouped in two influential trends: the wholesale move towards a 'soft-centric' design approach, and the increasing influence of a product's mechanical design in the board development process. Both of these changes are driven by significant developments within the electronics design industry that are here to stay.

To remain competitive while creating tomorrow's products, designers from all disciplines need to respond to these developments by considering the overall product design task and how the domains share design data – particularly between the electrical and mechanical design environments.

THE DRIVERS OF CHANGE

From a business standpoint the obvious shift in the electronics industry is globalization of the market form both a sales and product development perspective. This is represented by a world-scale revolution in the way many products, or key elements of those products, are now designed, manufactured and distributed. The tangible result of this change can be seen in the form of low-cost electronic products that have now become 'commodity' items – look no further than the DVD players at your local retailer. Irrespective of the development origins, these devices have been transformed from a unique product to a universal, low-cost product that applies to a worldwide market.

BOARD DESIGN...

A SINGULAR, FOCUSED TASK THAT YOU GET STUCK INTO WHILE SURFACING PERIODICALLY FOR COFFEE AND NEW DESIGN INFORMATION.

Whether it's a consumer product, an industrial controller or display sub-assembly, globalization – the move to a single, highly competitive world marketplace – creates an environment where commodity products pour onto the world market from the most cost-effective manufacturing regions. For both big and small companies worldwide this also opens the opportunity to outsource cost-sensitive processes, typically manufacturing and distribution, on a world scale. The move to 'off-shoring' and the general trend to global product commoditization have generated entire categories of products that can only be differentiated from each other on price.

This change has challenged the fundamental thinking on what makes a product unique, desirable and ultimately successful in today's highly competitive market. Thanks to global competitiveness, creating a product for a slightly lower price might gain a temporary market advantage, but it's only a matter of time before this is undercut by an equivalent product from somewhere on the globe.

Similarly, getting a product to market first is, by definition, only a temporary competitive advantage since others will quickly follow to dissolve that gain. Both these approaches can only create brief windows of competitive opportunity, and should be regarded as target 'survival drivers' rather than a path to achieving sustained product differentiation.

The hard facts are that while there are essential criteria for today's board designs – meeting cost, quality, deadline and performance goals – the physical electronics hardware itself will not deliver sustainable market differentiation to the final product. Any unique intellectual property in the physical hardware is relatively easy to reproduce and therefore cannot remain unique. What's more, virtually all components and sub-assemblies within the design are universal 'commodity' items in themselves and are available to all designers – one USB sub-block or display interface is pretty much like any other.

Along with market globalization and the proliferation of commodity electronics, the electronics industry is undergoing an era of unprecedented technological change. This has been driven by factors such as the increasing 'connectivity' of electronics products and, in particular, the advent of low-cost, large-scale programmable devices.

Reprogrammable devices such as FPGAs have created a revolution in the way products are designed by offering an open-ended platform for creating complex 'soft' hardware in the programmable design space. For some time the competitive factors of a design have largely been defined by software rather than physical hardware, but the introduction of programmable hardware allows that 'soft' influence to enter into the electronics design itself. The functional intelligence that determines a product's competitive edge can now be defined in both software and hardware.

The impact of this is significant from a board design perspective, since an ever-increasing number of PCB designs now contain one or more programmable devices. These are likely to be large, high pin-count (or BGA) FPGA devices which present substantial board routing challenges, but tend to reduce the total number of components on the board. The most obvious reason for that is that the majority of logic devices required by the design can be transferred into the fabric of an FPGA, saving considerable board real estate and reducing the layout complexity.

The less immediately obvious, but far more potent, impact of FPGAs is the potential for virtually an entire design – including microprocessors, memory, peripheral bocks and interfaces – to be contained within the FPGA. With this approach the remaining on-board electronics and hardware simply provide the interface to the outside world. Routing the board also presents unique challenges when FPGAs are involved, since the pin configuration (the functional position and electrical characteristic of each pin) is fully programmable and determined within the FPGA – rather than the board – design domain. Once programmed, the pin configuration of every FPGA is unique, and likely to change during the course of the design development.

Taking the higher-level view, the changes that are affecting today's board designs have moved away from the better-smaller electronics device path and more to an approach that achieves sustainable product differentiation in the market: namely the move to a soft-centric design within programmable devices, with less emphasis on IP contained within the physical hardware design.

Ultimately, real and sustainable product differentiation lies in the way a product looks, feels and functions. Today's competitive products – those that exhibit a differentiating edge among competitors – are more than ever defined by the user experience, which can be described by that product's form and function. This critical connection between the product and user is defined by factors such as aesthetics, ergonomics and its functional behavior, which are in turn established by the mechanical and software (but not the board) design of that product.

The mechanical aspects of a product design now directly and profoundly influence the electronic design by determining the board shape, size and positioning of its components, and in many cases by also defining the type of components used and how the software should behave. Complex board shapes and flexible board materials are physical evidence of the intimate ties between the mechanical case design and the board assembly it houses. This trend makes the interaction between the design domains more important than ever, since the competitive success of a product can now hinge on the effectiveness of that electrical and mechanical design cooperation.

YOU ARE NO LONGER ALONE

The overall message here is that the task of creating today's successful products must involve the close interaction between all the elements – electrical, mechanical and software – of a design. This represents the broader and pervasive unification of the product design processes, including PCB design, which is needed to meet the challenge of creating the next generation of electronic products.

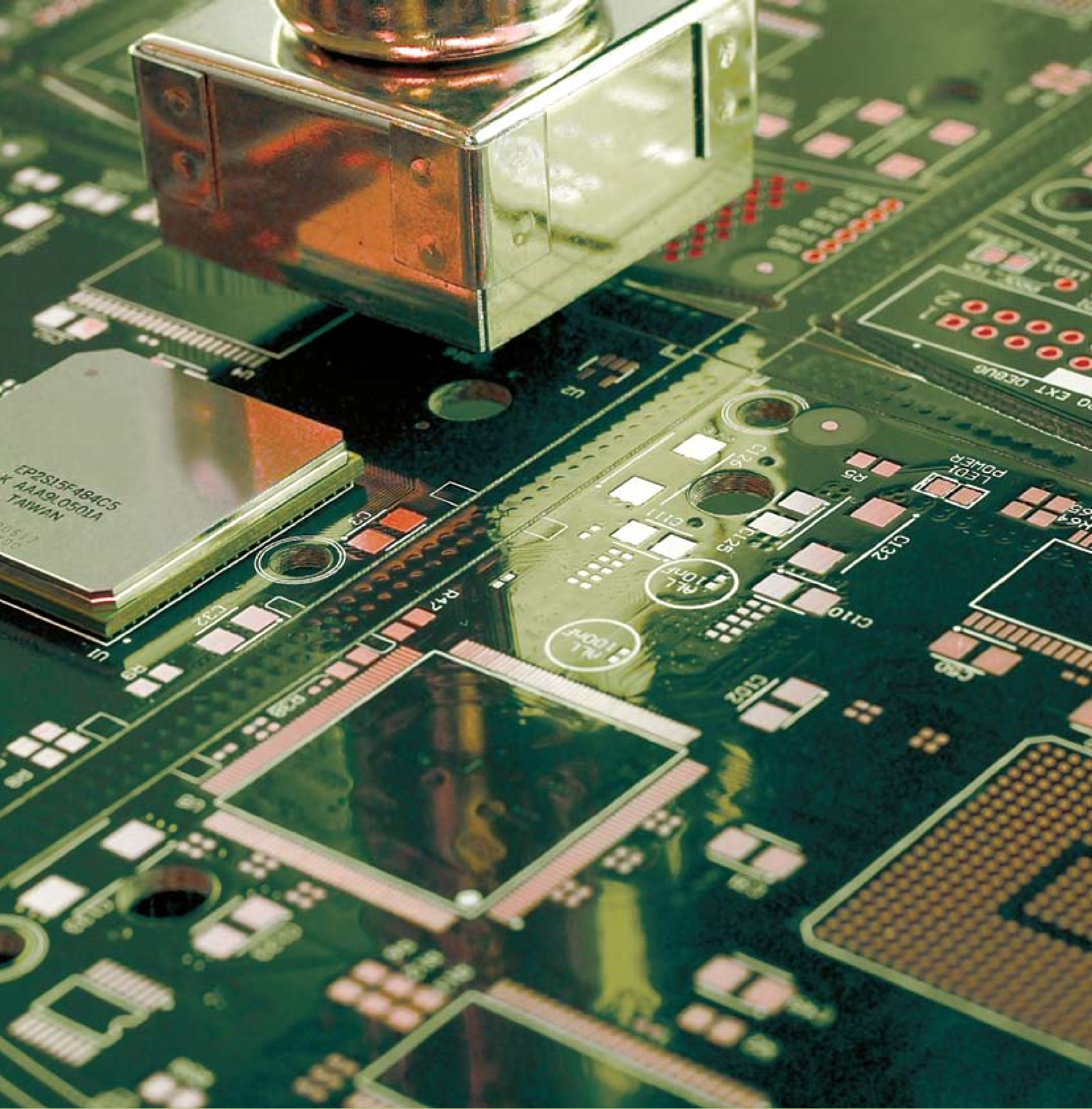
So where does this leave the process of PCB design for today's products? The first certainty is that board design can no longer exist in isolation from the soft (software and embedded hardware) and mechanical design processes. As the design disciplines converge, the processes become interdependent and at the least must successfully share design data.

From a practical board design perspective, the embedded hardware design information needs to be incorporated into the layout as FPGA pin data and its associated electrical constraints, as these fundamentally influence the routing process. Changes in the FPGA design generate disruptive routing challenges and a further exchange of design data between the FPGA-PCB domains, creating a potentially cumbersome and error-prone design workflow.

In a similar way, the mechanical parts of the design – in the simplest sense, housing or enclosure – determine the dimensional properties of the board, creating the need for another data flow between the domains. Changes in the mechanical design will trigger board design revisions, imposing further delays and complexity in the product development workflow.

When design systems involved exist as a traditional collection of separate process 'silos', effectively sharing design data information becomes a significant problem in its own right. If the file exchange systems exist to successfully translate and propagate the data – and this is not guaranteed – the information can be passed 'over the wall' to the next design domain and process. A conventional product design system that exists as a disparate group of design tools silos already suffers from the inefficiencies of a sequential workflow, and adding further layers of data exchange – such as that between MCAD and ECAD – serves to compound those problems.

The struggle to bring the mechanical (MCAD) and electrical (ECAD) design worlds together has been hampered by very different nature of the two design disciples. Unlike the common electronic relationship between embedded development and



board design, mechanical design traditionally exists in a very different head space to electronic design. The fundamental differences between mechanical and electrical CAD tools tend to reflect this gap, which in the past has negatively impacted the common ground of design data exchange formats.

INFORMATION PATHS

In practice the need for ECAD and MCAD design data transfer has been addressed at a simple level by the use of common file formats that pass basic dimensional information between the design applications in each domain.

The development of 3D MCAD design during the 1970s, and then solid modeling in the '80s, has been paralleled by the introduction of a bewildering range of data exchange formats. These vary widely in capability and accuracy, with few formats ideally suited to exchanging comprehensive 3D data to the ECAD domain. The result is an ECAD-MCAD design flow that tends to exist at a basic level and relies on a range of different file exchange formats, depending on the MCAD and ECAD applications in use.

Traditionally this means the dimensional and object positioning data from one application are processed and transferred to the other via a range of 2D and 3D file formats, as 'milestone' events. In workflow terms, for example, basic PCB shape details might pass from MCAD to ECAD, then at a later stage, a simple 3D model file of the board assembly is passed from ECAD to MCAD to check the mechanical fit between the board and enclosure. With each of these steps suitable design modifications are made, and another data exchange is usually instigated to confirm those modifications, resulting in a protracted and iterative process that does little to encourage MCAD-ECAD design collaboration.

Another approach to the problem has been the use of separate, third-party design translators to ease file compatibility issues (for example IDF, the Intermediate Data Format, is sparsely supported in the MCAD world) and make the process more flexible. These often provide import/export options in the native format of the ECAD-MCAD applications, and in some cases connect directly into those programs using object linking (OLEs) or programming interfaces (APIs).

To date both these approaches have fallen short of the ideal. With the basic file exchange setup, data translation errors are frequent due to limitations and inconsistencies in the exchange format itself and there is little control over the degree of data sent – too much or too little is equally problematic. But above all the process is generally clunky and marred by the incompatibilities of weak data exchange formats.

Dedicated CAD translation programs generally offer a better outcome, thanks to more rigidly defined formats and data filtering options that allow you to specify what objects are included for transfer. Unfortunately, however, it is often a case of two steps forward and two steps back due to the added layer of translation complexity inserted into the process.

The approach can, for example, make the translation process version-sensitive because of its intimate ties to the MCAD-ECAD applications, and it certainly adds another licensing cost to the overall design system. The linked (OLE, API) version of the translator programs can offer a more integrated solution by bolting itself into the MCAD or ECAD application, but the trade-off is that it then becomes 'version-critical' and the MCAD-ECAD applications must be loaded on the same PC platform so the OLE/API interconnections can be established.

TACKLING THE FUNDAMENTALS

The first step in creating a design environment that promotes real design collaboration between the ECAD and MCAD domains is taking a higher level view of the electronic to mechanical design relationship. A core concept here is the current changes in the electronics design industry mean that product design must now be thought of as a single task, rather than a collection of processes that are ultimately brought together.

From a board design standpoint, this means embracing a shared and collaborative approach to design that keeps a firm eye on the final result – a complete electronic product that meets market goals. In turn, this means recognizing developments such as the often dominating influence of a product's mechanical design in the board design process, and the need for design data exchange systems that work together rather than connect together.

What's needed at a fundamental level then is a reliable, comprehensive and convenient way to transfer that data between the domains. Existing solutions attempt to bridge the MCAD-ECAD gap though a maze of file formats and applications designed to stitch processes together. These systems have evolved over time to meet the growing need for design data exchange, but in the process have adopted proprietary formats or pressed existing but inadequate ones into service.

However, 3D data transfer protocols have now moved to the next level with the relatively new STEP format, which is a data-rich and extremely robust protocol designed for 3D design and manufacturing processes. STEP is now supported by most MCAD systems, so an ECAD solution that supports bi-directional STEP transfer will significantly reduce 3D data translation problems through this feature alone.

STEP files can be large, but this can be easily constrained if the ECAD system also offers an intelligent range of object filtering options in the translation interface. Along with file compatibility benefits, this approach can also remove the complications and expense of third-party applications, and does not suffer from MCAD-ECAD application version issues.

Beyond robust file systems for design data exchange, the ECAD-MCAD workflow needs to be considered from a productivity standpoint. For example, introducing separate third-party translation and processing applications adds more sequential stages to the process, leading to an increase in workflow complexity and the likelihood of recursive errors in critical design data. In short, any solution that introduces multiple file formats and sequential data translations must, by definition, increase the risk of slowing and complicating the product development process.

Another point to consider is how the 3D data models are created and applied for viewing in the MCAD space. Performing accurate judgments of how the electro-mechanical parts fit together – in practice, object clearance and interference checking – relies on the availability of accurate 3D object models. At a practical level this means that assembly information passed from ECAD to MCAD must include accurate component models, or those electrical models must be available within the MCAD application where they can be inserted as required.

Systems that rely on IDF file transfer are an example where the included 3D model information is inadequate for accurate clearance checking in the MCAD space. If IDF transfer is used in a stand-alone translation application that also performs clearance checking, critical models must be replaced with more accurate ones from its own 3D library. Data exported from this application as IDF files will then lose this more accurate data due to the limitations of the file format. At the least, this adds yet another layer of translation complexity to the process.

In terms of data integrity and workflow efficiency, MCAD-ECAD connectivity at its basic level is best served by a straightforward approach of passing STEP models directly between the two domains. While this seems simple enough, it relies on an ECAD system that includes STEP import/export capabilities, comprehensive 3D modeling data, and filter options to control the 3D content of exported files.

THE NEXT LEVEL

When considering the size and application of STEP file exchange in MCAD-ECAD systems, it's interesting to note the differences in content for each direction flow – and what this implies. In a typical workflow, 3D data representing the product's housing, a component or a new board shape will be transferred from the MCAD to the ECAD space, while a model of the complete PCB assembly is usually transferred from the ECAD to MCAD domain for clearance checking purposes.

In terms of data flow and file complexity, the MCAD models passed to the ECAD domain are relatively simple (say, an enclosure) while those passed from ECAD to MCAD are usually complex (such as a complete PCB assembly, including components). Board assemblies are object-rich and create complex 3D data files which must be loaded then rendered ▸

in the MCAD space for clearance checking purposes. Any corrections for the board layout or shape are passed back to the ECAD space, where revisions are made and data exchange processes repeated.

The point of note here is that checking and revising the board assembly to fit the mechanical housing constraints is largely an ECAD problem, but much of the process occurs in the MCAD space using complex 3D board assembly data. When you consider the fundamental needs of that workflow, it becomes clear that ideally, a significant part of the mechanical fit problem needs to be solved in the ECAD domain.

To make ECAD clearance checking a possibility what's needed are real-time 3D capabilities within the PCB editor, plus the ability to import MCAD assemblies into that space. Using the STEP format to bring, say, an enclosure model into the ECAD domain, practical interference checking would then be a reality in the 3D PCB design environment. If the system is then coupled to user-defined clearance rules and 3D object transparency options, a large part of the mechanical fit task can be resolved in real time within the ECAD domain.

File exchange processes can be reduced or even eliminated by introducing a further refinement: linking to 3D data files, rather than embedding that data within the ECAD design files.

With a linked setup the ECAD application would simply load data from an external 3D STEP file that has been generated by the MCAD application. The PCB editor can then alert the user when that external file changes, in response to an update from the MCAD domain, then refresh the object in the PCB work space and ECAD design files. This would occur in a real-time 3D design environment, allowing mechanical clearance errors to be resolved on the fly rather than through a protracted series of MCAD-ECAD design iterations.

Ultimately, the increasing importance of the physical properties of today's designs means that the interdependency of the ECAD and MCAD design environments needs to be catered for by systems that deal with the core problem directly. Most existing systems that attempt to provide an MCAD-ECAD solution take a piecemeal or add-on approach, and in the process fall short of the mark or, at worst, create counterproductive and error-prone workflow. As a result, intermittent design concurrency between the domains is the best possible result.

What's needed is a more unified view of the overall product development process where the entire design is treated as a single entity, with a single design data model. In this way the solutions to the growing need for intimate MCAD-ECAD connectivity come from a higher level view that considers the final aim, rather that a closely focused approach of dealing with the file exchange systems alone.

By implementing the groundwork of a robust 3D data exchange format (STEP) and direct data transfer, the process is simplified and can even be transferred to the ECAD domain where it needs to be solved. In this way designers from both domains can interact in a highly connected product development environment that promotes concurrent MCAD-ECAD design.

As the electronics design industry continues to evolve and design disciplines converge, it's now crucial that all of the design domains interconnect to a level where cooperative, concurrent design is a reality. When previously disconnected worlds join and work together the benefits are invariably profound and far reaching – the world of electronics design is no exception.

Published: Circuitree, US, August '08

EMBRACE

A SHARED AND COLLABORATIVE APPROACH TO DESIGN THAT KEEPS A FIRM EYE ON THE FINAL RESULT.