EMBEDDED DESIGN WITHOUT HARD BARRIERS
The availability of high-capacity, high-performance FPGAs and more feature-rich, these components becoming more capable, higher in capacity part of the design to move into the soft domain. Not only are FPGAs over the last few years has allowed an even greater behavior of today's electronic products.

The explosion of programmable hardware components such as FPGAs over the last few years has allowed an even greater part of the design to move into the soft domain. Not only are these components becoming more capable, higher in capacity and more feature-rich, they are becoming significantly cheaper. The availability of high-capacity, high-performance FPGAs at relatively low cost has opened up the possibility of using this technology to change the way designers interact with hardware and software by extending the concept of programmed device intelligence from just software, to encompass both software and hardware.

With this approach the system hardware itself can be defined in the soft realm – from mass logic through to high-performance microprocessors and matching system memory – allowing developers to create whole systems within the reprogrammable fabric of an FPGA. A much greater portion of the design process is done in a soft domain, meaning that the defining value of that product has moved away from the unique properties of the physical platform. This hardware platform nevertheless supports the soft elements and forms the interface to the outside world, and therefore remains a large and crucial part of the electronic development process – and in the process, takes a proportionally large share of the product development time and cost.

For all the revolutionary advances in electronics devices and how electronic products are designed, the process we use to develop and complete those products has generally failed to progress at the same pace. We still treat the design of the board-level hardware separately from the development of the software that runs on it, and programmable hardware design ends at the pins of the device – a separate discipline in its own right.

As more of a design is moved into a soft platform, the lines between the traditional design disciplines such as hardware, software and FPGA design begin to blur. Dealing with these design elements independently and with separate tools becomes increasing difficult and inefficient as design complexity increases and time-to-market cycles shrink.

The move to higher levels of abstraction within individual processes helps to cope with specific complexities but at the same time but increases the specialization required within each domain. Ultimately, of course, these individual elements of the design must be brought together to create a final product, but the increased specialization of each piece is making the final puzzle much harder to assemble. This blows out design times and is ultimately to the detriment of product innovation.

A COHESIVE SOLUTION

Rather than a traditional ‘point-tool’ solution of approaching the design problem as a collection of independent processes that are eventually linked together, a unified methodology offers a solution to this complexity barrier by treating the product design process as a single problem.

Unifying the design processes at a platform level creates a product development system that can manage design complexity while harnessing the potential of soft design within the programmable device realm. Bringing together all the hardware and software elements of the design process within one intrinsically connected environment creates a singular design flow and data model that dramatically simplifies the process.

For example, by exploiting the mutual-awareness and shared data of the hardware, software and programmable hardware design processes, a unified design system can seamlessly automate many of the inter-discipline tasks such as synchronizing the pin definitions of programmable devices across both the chip and board-level design spaces, and automatically managing hardware dependencies between the software and its execution platform. These things simply can’t be done effectively if the parameters and constraints controlling the different elements of a design exist in independent applications.

Importantly, unifying design processes creates an environment where the abstraction level of those processes can be raised as a whole, rather than within each separate piece of the traditional point-tool collection. In this way the complexity of today’s designs is managed as a whole, creating a new approach to electronics design where the overall process can be dealt with as a single higher-level problem.

Through a cohesive abstraction of the processes, unified design reduces the need for expertise in multiple design disciplines and the use of multiple different tools. It also reduces the level of low-level hardware design and system software needed by
The combination of a unified design system and a reconfigurable hardware platform opens the possibility for embedded developers to create an entire electronic product without custom hardware design.
lifted from the development environment and programmed into a tailored, off-the-shelf hardware platform. This is possible within a unified design environment because the design system can manage the low-level hardware dependencies to ensure the software and programmable hardware make the right connections to the targeted physical environment.

Using this approach a designer could choose from a range of targeted basic hardware platforms – say a handheld application, an industrial device, a consumer application or a rack-mount piece of equipment – customize that platform by plugging in a range of special purpose modules that contain fixed and programmable components, then simply download the custom design intelligence into it to provide a product solution ready to go to market. It could be considered as Commercial Off-The-Shelf (COTS) hardware, but thanks to the potential of high-capacity FPGAs the capabilities effectively cover software, hardware and programmable hardware – sufficient to develop then create a complete and viable product.

This concept enables someone who doesn’t have the necessary hardware skills for custom PCB design – let’s say a hardware-savvy embedded developer – to build a production-ready...
At any time, without time or cost penalties. This, reinforced by the inherent portability of designs created within such an environment, has a profound effect on product design cycles. It streamlines hardware design, opens the door to concurrent software and hardware development, and raises design abstraction to a level where existing design skills can be used beyond traditional design boundaries.

Ultimately though it has the potential to commoditize the concept of commercial reconfigurable off-the-shelf hardware as a means to bring a wide range of products to market, and to allow designers and companies to create innovative products without having to design – or pay someone to design – custom hardware or application-specific circuit boards. As designs move further into the soft realm, a unified approach to off-the-shelf hardware will allow designers, regardless of their hardware development expertise, to quickly implement systems that deliver the device intelligence needed to achieve true product differentiation in the market.

Published: Embedded.com, US, October ‘07 and Programmable DesignLine, US, October ‘07