# The soft option?

Will fpgas be the pcb of the future? By **Nancy Eastman**.

Whilst pcb manufacturing technology has evolved, the fundamental concept of board level electronics system design has remained the same – take a bunch of components and connect them together to form a circuit.

High capacity, low cost programmable devices have the potential to alter this paradigm. The question for board level system designers is how their skill set will fit into an increasingly 'soft' future.

The biggest recent upheaval came with low cost microprocessors, which brought 'embedded intelligence' to previously 'dumb' devices. The microprocessor's success was down to its ability to allow a significant part of the design to be moved from a hard wired environment into a 'soft' domain. Design at the board level continued to use the 'hard wired components' paradigm, but product behaviour – or 'intelligence' – could be added as software.

This platform based approach allowed the same physical hardware to be manufactured with lower risk because it was possible to modify the behaviour after constructing the hardware.

## The fpga revolution

It's clear that fpgas represent the 'next big thing'. Their recent sales growth and increasing pervasiveness has captured everyone's attention.

The availability of high capacity, high performance fpgas at relatively low cost is allowing previously fixed design elements – such as the processor and its peripheral components and logic blocks – to be moved into a soft domain. What makes this compelling is the potential to bring the same level of freedom to hardware design that moving functions into software gave developers when microprocessors became available.

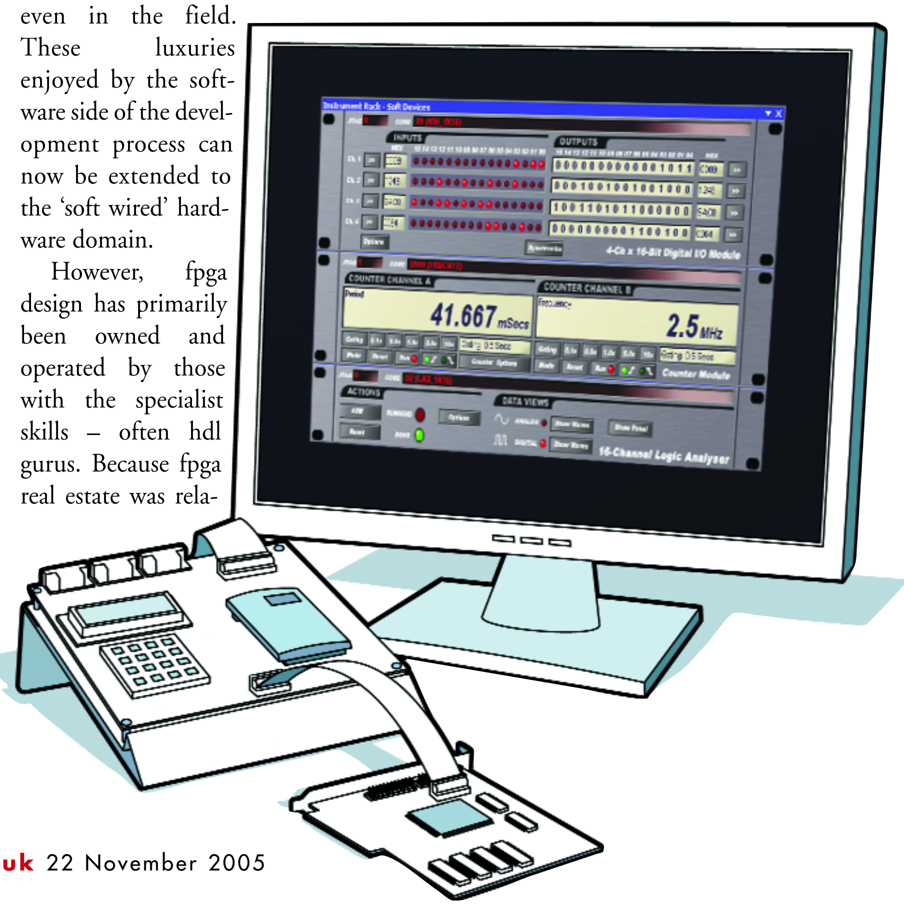
The reprogrammable nature of fpgas is a boon, allowing hardware changes to be made without the inherent time and cost penalties of a board respin. What's more, changes to the underlying system hardware can be made right up until the time the product goes out the door – or even in the field. These luxuries enjoyed by the software side of the development process can now be extended to the 'soft wired' hardware domain.

However, fpga design has primarily been owned and operated by those with the specialist skills – often hdl gurus. Because fpga real estate was relatively expensive, the focus has been on extracting the nth degree of optimisation from a given device. In essence, the programmable device was viewed as a high value component within a larger system defined at the board level.

As the price of large capacity programmable devices plummets, the potential to use them as a complete system platform changes that view. Defining the device's RTL functionality gives way to a need to develop fpga functionality at a higher abstraction level – a design process accessible to mainstream engineers.

## Familiar processes

When casting around for design methodologies, it's worth looking at board level techniques. Board level embedded systems designers often create designs that contain




The use of virtual instruments in conjunction with a suitable fpga development board allows an interactive approach to be taken to system test and debug.

many millions of underlying gates, but they don't need to deal with this complexity at the register or gate level. The complexity is contained in the components used to construct the circuitry. Designers simply treat these components as 'black boxes' during system development.

While large scale IP blocks are available off the shelf for fpgas, this is mainly supplied as hdl source files. The fpga designer must instantiate the IP in the target source code, which usually entails a working knowledge of the way the logic block is put together. Once included in the circuit description, the whole system – including the large scale IP block – must be synthesised for the target.

"Simulation can command as much as much time as the initial design process."

Nancy Eastman, **Altium**

Far from being ready to use, current IP distribution methods tend to add to design process complexity.

For system level design within an fpga to be efficient and accessible to the mainstream of engineers, the delivery of IP for fpga implementation must be streamlined to make it useable 'off the shelf'. What's more, the design environment for using this IP must allow the same flexibility and high level of abstraction as that enjoyed by system designers at the board level.

Altium has taken on this challenge with Altium Designer. This seeks to take a unified approach to the development of digital systems that integrate board and fpga level system design, allowing board level designers to create systems on fpgas with the same tools with which they construct systems for board implementation.

The approach is to package the functionality required to build soft systems into large scale, soft components (such as microprocessors and peripheral devices) and then provide an environment that allows the designer to assemble these using a schematic based approach. The design can then be implemented automatically in a target fpga.

Systems for fpga implementation can be created in Altium Designer in the same way as those for board implementation – fpga components are taken from libraries, dropped onto a schematic and wired together to form the system hardware.

## Interactive testing

Equally important is the ability to test and debug during development. Conventional fpga design flows rely heavily on simulation at the hdl level to resolve potential timing and functional issues – and simulation can command as much time as the initial design process.

At the board level, simulation plays a supporting role to direct 'hands on' testing. This is because the components used are assumed to work as the manufacturer specifies – there is no need to verify the performance of the devices themselves.

Streamlining the fpga capture process by using 'off the shelf', presynthesised components raises the question of how to test the system implemented inside the fabric of an fpga. One possibility is to build test instruments into the programmable device as part of the capture process.

Altium Designer provides a library of virtual instruments that includes devices such as logic analysers and frequency counters/generators. As with the other fpga components in the system, these instruments are presynthesised blocks that can be incorporated into the design at the schematic level. Once the design is processed, the hardware portions of the instruments are downloaded to the target fpga and Altium Designer then provides soft front panels that are used to set and read signals within the device.

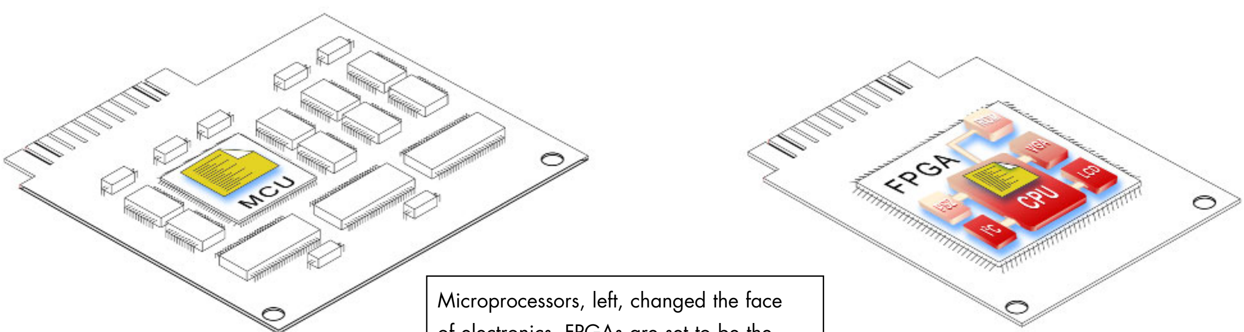
The use of these virtual instruments in conjunction with a suitable fpga development board – such as Altium's NanoBoard – allows an interactive approach to system test and debug. This provides a 'nano level' breadboarding environment that facilitates system development and verification without the need for hdl simulation.

## The future is soft

The extensive penetration of fpgas into broad electronics segments is inevitable. The key to unlocking the potential of this 'soft' design paradigm is to open up fpga based system development to the mainstream of system engineers, who already have the skills to create products using 'black box' discrete devices containing millions of transistors.

## Author profile:

Nancy Eastman is director, Altium Designer Applied Technologies.




Microprocessors, left, changed the face of electronics. FPGAs are set to be the next 'big thing'.