Designing Signal Processing Systems for FPGAs

John Heighton
Xilinx, Weybridge, UK
John.heighton@xilinx.com

1. The Need to Increase Productivity

DSP System designers have no shortage of great ideas, and are forever finding new, powerful, and creative algorithms, now this is what they are good at and it is this skill they generally use to secure their paycheck. So there doesn't appear to be any problems with this situation, you have ideas people being paid to come up with ideas, everyone is happy.

Most of the companies who pay system designers have a product to complete, and it is getting this product to market, and selling the product, that is the challenge. This is not just a business problem, but also a technical issue as designers are being driven to consider more efficient ways of verifying and implementing their designs.

For a long time software tools have helped the designer develop and simulate their designs, thus helping to improve productivity. However as designs become more complex, such as Wireless Broadband, Software Defined Radio, and Video/Imaging systems, the task of converting algorithms into code for hardware implementation becomes exponentially more difficult.

One commonly used sub-function of these high-performance DSP systems is Sample Rate Conversion, this function could be implemented into a design with greater ease and at a faster rate if the building blocks such as up/down sampling, clock domains, FIFOs, and clock enables were pre-verified and available to just drop into a design.

Another factor is the increasingly aggressive demands being placed on performance and productivity. Therefore designers now have to find new ways to speed up design development, and new ways to improve the methodology of getting designs to product.

Designers have always sought to overcome these challenges but until recently the options were extremely limited. It was possible to buy a faster computer that could run your simulation a little quicker, or to work at improving the communication between system designers and the hardware team, however these solutions only had a very limited impact on productivity.

A tool was needed that would allow the designer to continue working in a familiar environment, but which allowed designs to be verified in "hardware in the loop", meaning that the designer could run their design on any board fitted with the appropriate FPGA.

To resolve the "getting to product" challenge there was a need for a tool that could link the environment provided by The Mathworks tools to the Hardware Description Language (HDL) environment utilised by the hardware people.

2. Tools Meet the Challenges

Xilinx® looked at these two challenges several years ago and saw an opportunity to resolve both issues with the one software tool. The tool was developed by working closely with The Mathworks to create a solution that closed the loop between the algorithm and hardware tools.

The tool known as System Generator™ for DSP is revolutionising the way DSP system designers are working on their designs. The designer can now develop their design in The Mathworks Simulink™ environment, and making use of the "hardware in the loop" function provided in the System Generator for DSP, straight forward designs can be verified in minutes, while extremely complex designs still only take several hours to verify. "Hardware in the loop" allows a simulation to run in the FPGA dramatically reducing the run time. The designer doesn't need to wait days for their simulation to run, they can now perform this task in hours if not minutes, dramatically increasing their productivity.

The other major impact of this tool is that it allows the DSP system designer to very quickly provide the hardware team with a version of the design in a format they are familiar with. All of this can be done without the DSP designer ever having to become a hardware expert. The hardware team can then do their implementation, and carry out any optimisation that may be required. This process is proving to be far more efficient, and reduces the time spent debating just exactly what the design specification means and the best way of implementing it, thus vastly improving productivity.

So with verification times hugely reduced, algorithm development time can be a fraction of what it was. Add to this the ability to improve the process of implementing the design in hardware, and the DSP system designer is now able to significantly improve their own productivity and thus the productivity of their company.