Design Reuse Strategy for FPGAs

The design productivity gap creates an opportunity for competitive advantage.

by Carol Fields
Methodology Manager, Design Solutions Group, Xilinx
carol@xilinx.com

With today's technology, you are faced with more usable gates then ever before; that's both a blessing and a curse. Moore's law, stating that "the number of transistors per square inch on an integrated circuit doubles every two years", has been holding true since 1965, allowing a higher level of integration to occur on one piece of silicon. However, you can only take advantage of this exponential growth in densities if it is matched by a similar growth in design productivity.

Why worry about this gap in productivity? History has shown that on average, for a given product market, the first entrant takes 70% of all sales over the lifetime of the product, the second takes 15%, and the later ones share the remaining 15%, as illustrated in Figure 1. It pays to be first.

Design Reuse Fills the Productivity Gap

The most promising way to fill this productivity gap, and be the first to the market with your new product, is to reuse existing designs or Virtual Components (VCs). Design reuse is not a new idea; designers have always reused code, scripts, testbenches, and so on. What is new (from two years ago) is the growth in intellectual property (IP) infrastructure and the formalization of Design Reuse methodologies.

Design Reuse exists in several levels from "ad hoc" reuse of a previous project to purchasing a design from a third-party vendor. If an in-house VC has the potential of being reused more than twice, it is wise to put some effort into developing a Design Reuse methodology. This methodology addresses all aspects of the process including legal and business practices, VC specification, designing and coding, testing strategies, design storage and retrieval, and design metrics.

The amount of effort or lack of effort has a direct effect on the VC's reusability, especially if the person reusing the module is not the original designer. If a third-party VC is to be purchased, it is important to understand the total cost and actual usability of the VC.

The key to Design Reuse is trust. For an internally generated VC you need to build-in trust, and when purchasing a third party VC you need to define what makes the VC trustworthy.

Design Reuse Strategy Today

The most popular design reuse methods are for ASIC designs. This makes sense because the abundance of gates provided by Moore's Law is allowing such a high level of integration as to place an entire System-on-a-Chip (SoC). In 1995 the Dataquest definition of SoC included a compute engine (microprocessor, microcontroller, or digital signal processor), at least 100K of user gates, and significant on-chip memory. Five years later we still do not find an abundance of systems on an ASIC chip. However, what we are

seeing is an increased use of System Level Integration (SLI). What we are also finding is more SLI occurring on FPGA devices with many of the same design issues as SoC.

The predictions are that by 2003 the bulk of the industry revenue growth will be because of SLI. The existing Design Reuse methodologies today are closely linked to SoC. In reality it makes more sense to tie Design Reuse methodology to both SoC and SLI since a formalized Design Reuse methodology would benefit both SoC and SLI designers.
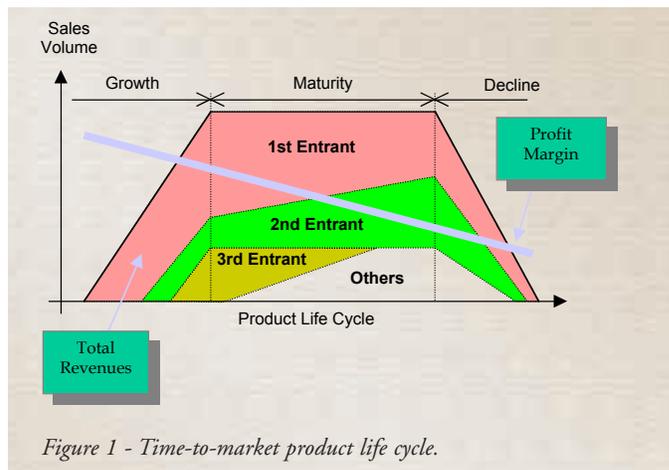

Figure 1 - Time-to-market product life cycle.

Systems-On-a-Reprogrammable-Chip

FPGAs have changed dramatically since they were first introduced and used as glue logic just 15 years ago. In the late 1980 and early 1990s, FPGAs were primarily used for prototyping and lower volume applications while custom ASICs were used for high volume, cost sensitive designs. FPGAs had been too expensive and too slow for many applications, let alone for Systems-on-a-Chip. Plus, the development tools were often difficult to learn and lagged the features found in ASIC development systems.

Silicon technology now allows us to build FPGAs consisting of tens of millions of transistors allowing for more features and capabilities in programmable technology. With today's deep sub-micron technology, it is possible to deliver over three million usable system gates in a FPGA. Today's average ASIC design operating at 30 to 50MHz can be implemented in an FPGA using the same RTL synthesis design methodology. By the year 2003, a state-of-the-art FPGA will exceed 10 million system gates, and will

operate at internal speeds far surpassing 200 MHz. Many designs that once could only be implemented in an ASIC due to speed, density, or pricing are converting to a much more flexible and productive FPGA solution.

It is a safe bet that more systems will be implemented in FPGAs in the future, especially given Moore's law and the ingenuity of FPGA R&D engineers. The industry analysts predict that in 2003 FPGAs will begin to replace standard cell ASICs in all but very high volume applications. However, there are many cases where the volume is not high enough for Standard Cell technology, such as the adoption of gigabit Ethernet, and areas where there is a demand for higher performance products to meet increasing traffic flow from the Internet.

Because Design Reuse is about planning for the future, the term Systems-on-a-Reprogrammable-Chip (SoRC) is used for SoC implemented in an FPGA. This term is used to define both full and partial system-level designs since the challenges facing these FPGA designers are almost identical. Although it is rare to find entire systems on an FPGA today, there is an increasing amount of SLI designs occurring due to the newer FPGA architectures (such as the Virtex family containing system-level features).

Planning for the Future

Today most major digital design companies are in the process of defining (or redefining) their Design Reuse strategy. This generally includes the creation of an internal Design Reuse Department, similar in structure to the existing EDA Department used to manage CAD tools. In the late 1980's many major companies had already formed their EDA Department to support the multitude of ASIC EDA tools. At this point FPGA technology was just emerging with designers using either a proprietary EDA tool or a simple schematic capture tool; these tools hardly required a department to manage them. By the mid-1990s FPGAs were being widely used by these same companies for higher density glue logic and SLI, and FPGA support was retrofitted into the EDA Departments.
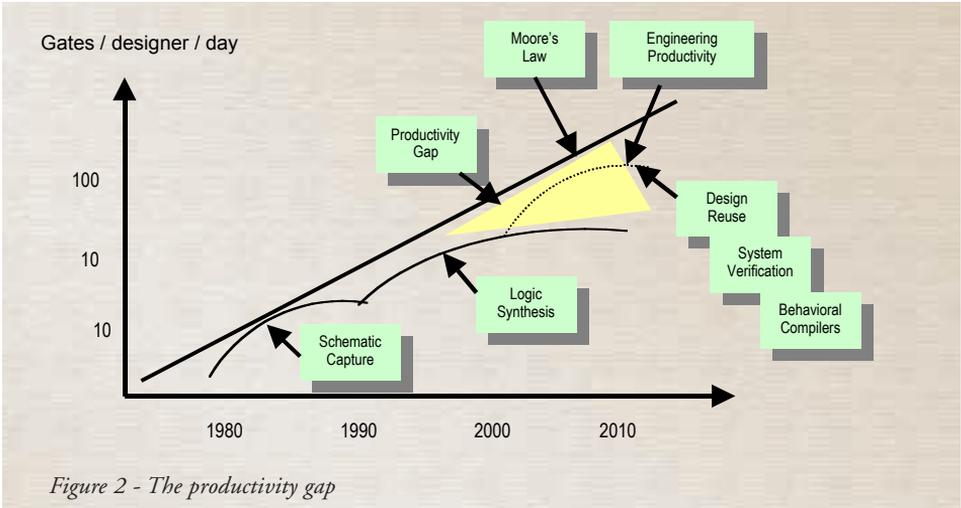
Figure 2 - The productivity gap

Today, we have the opportunity to define a reuse strategy that can not only co-exist for FPGAs and ASICs but can also work seamlessly between the two technologies. The decision to include FPGAs in a Design Reuse strategy must be made up-front because it affects almost all phases of the Design Reuse process, from design specification to verification planning.

Sharing RTL Design Methods

One of the most exciting outcomes of the dramatic improvements in FPGA architectures, pricing, and design tools is that this technology advancement has made it possible for ASIC and FPGA designers to share a common RTL design methodology. A common RTL design methodology is the basis for a common design reuse methodology.

Though ASICs will continue to provide higher levels of design integration, higher speeds, and new EDA environments, FPGAs are never far behind. The major FPGA and EDA companies have made a conscious decision to keep their design environments the same, from the end users point of view, to make it easy for users to move from one technology to the another. This was illustrated by the wide adoption of RTL synthesis tools and verification tools in the mid-1990s. In the case of RTL synthesis, existing ASIC methodology was kept the same and the synthesis algorithms where changed to target specific FPGA devices. Today we are seeing higher-level EDA tools such as Floorplanners and team-based design tools using the Internet.

Conclusion

In 1999, the number of ASIC design starts peaked at only 1000 designs, and despite all the publicity over the multimillion gates designs, most of these design starts were under 200K transistors. The average FPGA design start in 1999 was between 10K and 50K gates, with the fastest growing size range between 50K and 100K gates. Considering that FPGAs are more widely used than ASICs in digital designs today, it makes sense to include FPGAs in a design reuse strategy.

There are many benefits of sharing a common design reuse strategy; one of the most compelling is the flexibility it gives the designer to choose the IC technology late in the design cycle. It provides the flexibility to choose the best method to implement an SLI design without the overhead of retraining the design teams. In this fast pace market it is difficult to predict what features your product will need and what technology you should use.

A Design Reuse strategy is more than RTL code and synthesis. Many companies reusing designs have found more value in the design and test specifications than the actual RTL design. If you are currently an ASIC user, the good news is that many of the elements of a good design reuse methodology can easily incorporate FPGAs with minimal modifications.

Xilinx has joined efforts with Qualis Design Corporation to create the first Reuse Design Guide for FPGA users. This new FPGA Reuse Field Guide will walk you through the elements of building a design reuse strategy and is available, free of charge, from the Xilinx website at: www.xiliinx.com/ipcenter.