Platform Based Design: Does it Answer the Entire SoC Challenge?

Gary Smith
Gartner Dataquest
251 River Oaks Parkway, San Jose, CA 95134
+01 408 468 8271
gary.smith@gartner.com

ABSTRACT
During the mid-1990s, design fads emerged as the design challenge was becoming significant and the EDA vendors were entered the methodology business. Trying to market methodologies helped produce today's unfortunate design fad phenomenon.

1. INTRODUCTION
Platform-based design can be traced back to the reference designs during the era of vacuum tubes. As with many of these new ideas, this one has been with us for three-quarters of a century. The idea was to show engineers how a vendor's products, in this case tubes, could be used to solve engineering problems. Reference designs have been with us ever since. They gain popularity whenever there is a major breakthrough in technology. During the 1970s, there was a minor shift in the use of reference designs. Engineers were starting to use these designs in production, at least for the first few systems. The reference designs became a fast way of getting a product to market. If you were out first, you could grab market share and, after the design became more familiar to your engineers, you could start exploring some of your own architectures using the reference design as a guide. As the microprocessor became part of these designs, engineers found that they could leave the reference design as is and gain competitive advantage using software rather than hardware. This was the start of the embedded board market, and the point where reference designs became platform-based designs.

2. What is Platform Based Design?
In essence, a platform is a frozen architecture. Once the architecture is frozen, you may standardize the interfaces and give the engineers some choice of building blocks. Actually, the choices range from none (where you make your modifications using software) to multiple choices on multiple blocks. Supposedly, all of the interfaces have been verified for all of the various block combinations. Obviously, the choices are very limited in the beginning of a specific platform and then grow as time goes by, giving the platform builder the time to verify more blocks. That is until the architecture becomes obsolete, which is the main problem with platform-based design. In today's world of technology, that comes all too quickly. That's why the word "flexible" is often attached to platform-based design. Unfortunately, flexibility is easier said than done. The best example is the PC platform. Keep in mind that once you change the architecture, you tend to disrupt your interfaces. That calls for a major effort in verification and in setting up the new standards required to keep the platform viable. The dollar cost of Intel's engineering effort to maintain architectural compatibility — a six-syllable word for disrupting as few interfaces as possible — has been considerable. In design engineering, just as in most everything else, you can't get something for free.

DAC 2004, June 7-11, 2004, San Diego, California, USA.
ACM 1-58113-828-8/04/0006.

3. Derivative Design
Unfortunately, the fad is in full bloom. To cover up the fact that platform-based design is not the answer to all design questions, marketing has broadened the definition of platform-based design until it has become a fairly meaningless buzzword or, in this case, buzzterm. One clever design technique that is starting to be called platform-based design is the derivative design. Derivative designs started coming out of the PC industry in the late 1980s. AST was one of the leaders in this technology. As the PC is a platform, it was a natural for the technique. First, a main design was done based, in those days, on a gate array design. This took nine months to one year. The difference was that the system design was "architectured" so that it could be modified fairly rapidly, using a field programmable gate array (FPGA) or software. These modifications could be done in three to four months, allowing the industry to introduce four PCs per year. This was the primary cause of the PC recession of the early 1990s. It is now known that if you introduce products too close together, you will disrupt consumer-buying patterns. Once the PC introductions opened up to two per year, the market stabilized.

The cellular phone industry started using derivative designs. But, instead of introducing cell phones multiple times per year into the same market, they introduced derivative designs into different segments of the cell phone market, in general, holding each introduction to one year or 18 months for each sub-segment. It is now possible to do derivative system on a chip (SoC) design, with a good architecture and a strict hierarchical design methodology, by simply replacing cores as you would integrated circuits (ICs) in a PC board design. Therefore, derivative design has become a powerful tool for the power user community, a far cry from the platform-based design of the mainstream community.

4. Where Platform-Based Design Works
In looking at a system design, you need to understand the three styles of design. The SoC design style is used when your competitive advantage is in silicon design. Embedded software design is where your competitive advantage is in software development. Component-based design is used where the markets are unable to support expensive design costs or where the real competitive advantage lies outside the electronic design itself. Embedded software design is the obvious place for platform-based design. It can also be used in mainstream SoC design. Keep in mind that the higher up on the design flow, the greater the impact on the completed designs. All silicon design breakthroughs happen at the architectural level. If your architecture is frozen, as it is in platform-based design, your ability to use silicon as a competitive advantage is severely limited.

5. Perspective
Platform-based design is a good fit for the embedded software design style, but it is only a temporary solution for SoC designers. As today's designs are targeted toward full systems, the style of design and the techniques used to compete using those styles can affect the entire business strategy of the companies involved.