

How to Reduce Time-to-Market for System-on-Chip Design

By Stefano Zammattio
Product Marketing
ARC International

Introduction

In today's fast-paced market there are many factors that influence the commercial success of communication and consumer devices. One of the most critical factors for success is the early availability of silicon, as this directly impacts when the device can be introduced and the volume and price that can be commanded in the marketplace.

A quick survey of any ASIC- or ASSP-dependent market will show that the price of any new product is subject to a steep exponential decay. For example, a multi-standard DVD player capable of supporting Dolby Digital, MP3 and VCD can easily be found in stores today for less than \$100—nearly three times lower than the price of a comparable product just one year ago!

As little as a six-month delay in availability means not only losing customer opportunities but also having to sell the device at a lower price to be able to compete. With profits hinging on early availability, pressure is really on the design teams to get the device designed, tested and manufactured as quickly as possible. But this is becoming more difficult as communication and consumer devices continue to integrate more features, often creating more complex and time-consuming design cycles.

So how can these complex designs be accomplished in less time? One way is through the use of licensed intellectual property (IP). A recent survey by Electronics Weekly shows that developers expect IP and improved tools to help them get the job done quicker. The same survey showed that 70% of all projects involve more than a single piece of IP and that the average development project currently lasts one year.

The aim of using IP is to reduce the development time for the standard parts of the system, enabling developers to spend more time on differentiating their product. Developers want to use IP as off-the-shelf black boxes that can be fit together like building blocks. Certainly the benefits of licensing complex IP such as processors and complex standards-based peripherals are clear; after all it is much easier to drop a pre-built engine into a car rather than build one yourself!

But is it really that easy? Does it really help reduce time-to-market? This depends on how developers address IP issues during the product design cycle. We will take a look at each stage of the cycle and explore ways that developers can successfully reduce time-to-market for their system-on-chip (SoC) designs. By following a few simple guidelines, developers can exploit the potential of IP and actually reduce their design costs, time-to-market and overall risk.

Step 1: Choosing and Evaluating IP

At the beginning of a design project, some of the most critical choices regarding IP and development tools are made. Once the product is in development, changing any of these decisions is costly both in dollars and time-to-market, so good choices at the start of the project help maximize the probability of success.

Many of today's designs are really about creating effective system solutions. This means that developers must consider the impact of all of the system components—including software development tools, operating systems and middleware—right at the beginning of the project.

For each IP component there are many products available, and the developer has the bewildering task of sorting through them to identify which ones are most suitable. Once each piece of IP has been individually evaluated, the developer must go through the possible system permutations to identify the combinations of IP that actually work together and at what engineering cost. Most systems contain several IP components and this task alone can take months. Licensing IP without validating that the component is up to the job can be a very expensive mistake.

A recent Electronics Weekly survey indicated that the top three criteria used by developers to make IP selection decisions are, in priority order, (1) IP performance, (2) single IP supplier and (3) supplier reputation. These criteria are born from the experiences of developers who are using IP "in the trenches".

It seems obvious that developers would seek IP that is able to meet their performance specifications, but why should they consider single source IP suppliers and their reputation? Certainly at this early stage it is far simpler to evaluate the products of a single company as opposed to looking at a different supplier for each part. Risk can be dramatically reduced by obtaining integrated, pre-verified, and "packaged" technologies from a single source, as opposed to have to expend the sometimes considerable amount of time, energy and money needed to accomplish this in-house. Having a single point of contact to answer questions on all of the IP is much simpler and it should be much easier for the supplier to demonstrate that the IP components work together.

This clarity should make the evaluation take much less effort and time. Other key criteria to be evaluated at this stage are the flexibility of the IP and the quality of the documentation, IP implementation and of the support provided by the supplier. This is where the reputation of the supplier usually comes into play. A supplier who has a proven customer track record is more likely to be able to meet these criteria.

Step 2: Licensing of IP

Once the IP has been evaluated and the final selection made, the next step is to license the IP and start developing the design. However, one factor that engineers often underestimate is the amount of time consumed by the commercial negotiation of the price and terms of an IP license. This can be further complicated by having to deal with several IP suppliers. What happens if one of them will not agree to your terms? You may have to re-think the entire design and go to plan B. Unfortunately, plan B often requires more evaluation and more time.

As company lawyers and executives develop the license agreement into something that is mutually acceptable, each new iteration of the agreement can take days and even weeks to review and approve. This is a critical part of the business that has to be right; and getting it right with several IP suppliers, each with different lawyers and agreements, can be a long, costly and tedious process. This is one of the reasons why buying from a single IP supplier is number two on the priority list.

Step 3: Integration and Module Testing

Once the IP modules are licensed and installed, the first task is to make sure that they work. Once this is done, the developer can start to plug them together and add their proprietary bits. The truth is that integration of IP is often much more difficult than it seems, and there are often unforeseen system issues. This is the first time that those crucial decisions made in Step 1 are tested. A hurried decision, or simply targeting the lowest cost products, can result in decisions to use IP with non-standard interfaces, poor documentation, poor support and with multiple IP suppliers. Often choices like these will start to present problems even at this early stage in the design process. Simply installing some IP and verifying that it does what was originally claimed can be difficult. This is why many developers make the effort to try before they buy.

In the *Electronics Weekly* survey nearly 60% of the developers who chose not to use IP said it was because they believed integration would take too long. Developers who decide to buy all or most of their IP from a single reputable supplier will immediately save a significant amount of time and effort at this stage, not only due to the ease of contractual negotiations, but also because of the quality and pre-verified compatibility of the IP. The coverage and quality of support from the IP supplier will also make a huge difference to the developers who are climbing up the learning curve required to use the IP. The combination of these two factors results in the developer being able to get a basic system up and running much more quickly.

Step 4: System Testing

For a larger project, the amount of effort required to integrate the IP is usually less than that involved in testing and verifying the IP modules as part of an integrated system. If the IP is sourced from different suppliers, this may well be the first time that these components have ever been tested together. If there is a problem (and there invariably is), the burden is typically on the developer to isolate and prove the problem to the IP supplier. This is where the quality of support is really tested. How quickly will the supplier acknowledge the problem and produce a workaround or fix?

With IP sourced from several suppliers, the system environment cannot usually be reproduced by the supplier without going on-site. This makes it difficult not only to identify the problem but also to fix it. The developer has to reproduce the problem in a single module system so that the supplier can identify and fix the issue; this is simply a waste of time and resources for the developer.

A single-source supplier can reproduce the problem more easily, given their ownership and knowledge of all of the IP components. This saves the developer valuable time and also makes the supplier more inclined to fix the problem, because other customers are likely to encounter the same problem.

Step 5: System Profiling

So, at last, the system is ready for manufacture. Or is it? If the system is complex and performance is critical, it is usually a wise idea to run performance profiles of the system using the actual application software (or something very close). If compatible simulation models of the IP are available then this can begin as soon as the IP is delivered.

In the event that all the simulation models are not available, then this work could begin a little later using co-simulation tools to co-verify the application software on a hybrid software model/RTL simulation. This would enable the developer to verify the system performance and functionality, thus providing a high degree of confidence that the system will meet the required specification when it comes back from the silicon foundry.

The ability to simulate the system also means that while the chip is being manufactured, the software developers can continue to develop the software, so that it is ready as soon as the chip is available. This capability can significantly reduce system verification time, ensuring that the real system can actually deliver what the customer wants, and as a result, help to bring the device to market much sooner

But what happens if the system does not meet performance requirements? After all, many of today's systems are highly complex and it is entirely possible that a limiting corner case may have been overlooked by the developer. These types of problems are usually created by a combination of several conditions, some of which may have been unknown in the initial design stages, while others may be reside within the licensed IP. Certainly ripping out an offending piece of IP and integrating a new one would be a very time consuming and costly option, particularly if it is an expensive piece of IP at the center of the system.

There are three key elements that combine to solve this kind of problem. First comes the skill and knowledge of the developer to identify and develop a workaround for the problem. Second is the level of support from the IP supplier, who should work closely with the developer to help solve the problem. Third is the design of the IP itself; for instance a hard macro would require significant effort to make an architectural change—and this is entirely in the hands of the IP supplier. Even if the IP supplier is willing to make the change, the amount of time taken to fix the problem can often be prohibitive.

When choosing the IP, one should look for well-supported, flexibly designed technology, preferably in soft format. This usually means that solutions can be readily identified, implemented and verified in a much shorter period of time.

Step 6: Layout and Tapeout

By the time system profiling is complete (or even before), the deadline date for generating the first mask set is usually fixed. This means that the race is on to turn the design into a functioning physical implementation. One of the most critical milestones at this stage is timing closure—in other words, getting the physical implementation up to the speed required by the device specification. This often involves re-synthesizing or hand tweaking gates and paths to speed up the slowest bits of the design.

Fortunately, there are ways to minimize physical implementation efforts. The obvious one is to build in some "headroom"—for example setting a higher speed requirement for the initial synthesis of the RTL design. But even this extra slack won't help if fundamental problems exist with the routability of the design. Routability issues can be solved only by better planning of the physical implementation and by using tools that have better facility with the physical issues. These tools will give much better estimates of the achievable speed of the design and produce netlists that are better suited to physical implementation from the outset. Modern development tools also allow much earlier and easier investigation of the effects of chip floorplanning. Once again the use of modern tools to evaluate the design early on in the design process saves time later on.

Step 7: Manufacture, Test and Prototyping

The time it takes to manufacture the chip is totally dependent on the chosen process and foundry. Once the chip has been manufactured, time has to be allocated for test and assembly.

While the chip is in manufacture, a suitable prototyping board is usually designed and manufactured. In the ideal scenario, software developers finalize the product and diagnostic test software on the system simulation. The chip is then mounted into the prototype board, the test software is run and, within a few days, the working system

can be passed over to the software developers to test their application software. Once this is complete the product can be demonstrated, sold and shipped to customers.

If system simulation work is done early in the project, developers can come close to this ideal scenario. Without system simulation, each step of this process can take weeks or even months, delaying the ability to demonstrate and sell the solution. Issues are normally addressed by modifying the software, and developers deliberately build in performance headroom for these last minute fixes. Sometimes however, the hardware can be found to be simply not up to the job, resulting in months of costly chip re-design and lost market opportunity.

Summary

It is clear that several common themes can be seen throughout the SoC design flow, from concept to manufactured chip. Purchasing from a single IP supplier, like ARC International, plays a major role in reducing time-to-market, design risks and development costs. In addition, using the latest technology in tools to evaluate the performance of the system as early as possible helps to avoid design errors that would require major system rework, and facilitates the back end of the flow. This ability to evaluate system performance can be extended to allow the development of application software in parallel to the hardware, once again reducing the time and effort taken at the end of the project.

As a supplier of a highly integrated, single tool-chain SoC development platform, ARC International has provided the backbone for a number of communication and consumer devices in today's market. ARC is able to supply processor, peripheral, RTOS and software IP, so customers can benefit greatly from this high levels of integration and support. For more information on how ARC can help with your SoC designs, visit us at www.arc.com.

NORTH AMERICA:
ARC International
2025 Gateway Place, Ste. 140
San Jose, CA 95110
Tel: +1 408 437 3400
Fax: +1 408 437 3401

EUROPE:
ARC International
The Waterfront, Elstree Road
Elstree, Herts. WD6 3BS, UK
Tel: +44 (0) 20 8236 2800
Fax: +44 (0) 20 8236 2801

ARC Worldwide Offices:
New Hampshire
Tel: +1 603 882 2282

Israel
Tel: +972 9 971 5616/7

Germany
Tel: +49 89 930 86 221

Scandinavia
Tel: +46 8 594 76 736

Ottawa
Tel: +1 613 271 8077

France
Tel: +33 (0) 1 64 63 25 20

The information contained in this white paper is subject to change without notice. ARC International assumes no responsibility for any errors contained within this white paper and makes no commitments to update or to keep current the information.

The ARC logo, ARChitect, ARClangent, Precise/MQX, Precise/RTCS, MetaDeveloper, and SeeCode are trademarks of ARC International. MetaWare is a registered trademark of ARC International. ARC International recognizes other brand and product names as trademarks or registered trademarks of their respective holders.

W1195.0