Building the IP Ecosystem

by
Thomas Harms

The term System-on-a-Chip (SoC) defines both a product and a process. As a product, SoC defines specific, targeted applications and contains an entire system. An SoC product will typically include processors (possibly with special functions) to control the system, embedded memory to store the system program and data, communication peripherals for interaction with other systems, and analog interfaces to the real world. SoC products face unique constraints including extensive voltage and temperature requirements for automotive applications and low power for portability in wireless applications. Every SoC product, to be complete, includes embedded software. In fact, an SoC could be a general purpose IC differentiated only by software, potentially at multiple levels including device drivers, real-time operating systems, application programming interfaces, and application software.

As a process, SoC defines system requirements that are modeled, analyzed, and partitioned into hardware and software specifications for design and implementation. Intellectual Property (IP) includes both hardware and software development, which are designed and verified in parallel, rather than serially. The objective of an SoC process is to define and design a silicon based system, create and integrate the necessary IP, and fabricate and qualify the finished silicon parts, all in as rapid a time-to-market as possible.

To achieve rapid time-to-market, chip manufacturers and system developers have both recognized the need to implement reuse of their semiconductor designs. The goal is for various industry participants to cooperate in implementing reuse standards, create IP specifically designed for ease of reuse, certify the IP, and incentivize IP providers to deposit the IP in a widely accessible repository. With all of these precursors in place, system developers can begin to effectively reuse IP to create true SoC devices in rapid time-to-market within the context of an SoC Ecosystem. (See chart xxx-PowerPoint Slide)

But industry participants face a difficult challenge: How to deliver system solutions in a highly competitive market in rapid time-to-market. This challenge is compounded by the fact that IC complexity is growing at a rate of 58% per year, yet design productivity is increasing at a rate of only 21% per year. (See Moore's Law and the Design Productivity Gap Chart). SoC developers have to find a way to bridge this gap.

To successfully compete as an SoC provider, a company must deliver four basic product development capabilities. First, they must have acquired and apply system design skills to define and optimize systems on silicon. This involves a shift of effort and focus to higher levels of design abstraction. This capability can increase productivity by focusing design exploration in the most efficient manner. Second, the system designers must have

access to a reusable IP portfolio to develop complex integrated systems within the desired competitive cycle times. Third, silicon implementation expertise ensures that the resulting SoC product meets all functional and performance requirements. And fourth, the manufacturing technology must be capable of processing, packaging, and testing complex high pin-count mixed signal SoCs within the cost constraints of the application.

Consumerization & Miniaturization

Two powerful interrelated industry forces working in conjunction are driving SoC product and process development. Consumerization of electronic products, the first of these forces, results in ever-faster design cycle times. Currently, SoC system designers are facing time-to-market design deadlines in the range of 3-6 months in current generation products. This is down from prior generation design cycle times of 12-18 months. It is crucial for system developers to be first-to-market. To do so, they must compete within these cycle times. Failure to do so means that companies risk loss of market share and credibility. One of the identifiers of the consumerization of electronic products is high volume. Consumers and consumer-like business-to-business markets form a large potential for market specific applications. These opportunities are enormous but only if a system developer can produce a product within tight (low) cost constraints in high volume. Because these market opportunities are substantial, intense competition forces system providers to accelerate time-to-market and uncover new methods to reduce costs. Consumers want more functionality and more availability at a constantly lower price, and have come to expect and demand it.

Miniaturization, the second and equally important of these forces, means that systems developers can utilize the new, smaller geometries available in silicon devices. Using these geometries, system producers can design and manufacture highly complex and high performance integrated SoC based products. The key for system providers is to differentiate products through application software. For the semiconductor manufacturer, the key is to recover fabrication costs through added system value.

Advances in process technology have enabled semiconductor manufacturers to support system providers with faster, better performing silicon products. As speed and performance have increased, designers are liberated to migrate functionality to software, which provides design flexibility and reduces cycle time. Thus, higher levels of on-chip integration reduce costs and result in creation of a true SoC. One of the critical strategies that silicon manufacturers and system developers can employ to optimize SoC development time is to implement a methodology for reusing Intellectual Property (IP).

Currently, industry efforts are under way to create and implement industry-wide IP reuse standards. An industry-wide plug-and-play standard, the Virtual Socket Interface Alliance (VSIA), has developed initial standards now ready for user adoption. Companies have also made an aggressive commitment to develop and implement reuse standards under programs, which follow the VSIA standards. Several large companies have implemented reuse standards internally and, in the process, have learned much about the practical obstacles to implementation.

As a company, Motorola has a truly pressing urgency to implement reuse standards as the critical foundation of an efficient reuse infrastructure. Implementation benefits both our customers and our operating units as creators and providers of IP. We must develop innovative ways to overcome the obstacles to IP reuse. As we do, we deliver a functional, far-reaching, all-inclusive standard that will slash our customers' time-to-market and improve IP return-on-investment for its developers. In short, a true SoC development ecosystem begins to emerge.

Many companies utilize their own internal set of semiconductor standards for developing and reusing IP. As the demands for uniformity among IP reusers increases, Motorola has chosen to release our internal standards to the public. This release is our contribution to an industry solution and cooperates with industry initiatives such as the VSIA.

The Need for Universal IP Standards

As noted above, the most compelling pressure for IP reuse standards arises from the design productivity gap combined with the various forces driving rapid time-to-market design cycles. With the forces of consumerization, high volume/low cost market opportunities, and miniaturization pushing the industry, an effective industry-wide IP reuse methodology is critical. The whole point of reuse is to provide a method for sharing best practices and best design blocks among IP and SoC developers. The methodology for reuse arises from the design flow. Design flows consist of a series of ordered steps that moves the design from higher levels of abstraction to a specific set of hierarchical, ordered design tasks. Each task when completed, in effect, becomes a module in the system.

Each module fulfills a purpose in the design and as each task or module is defined, the design flow becomes a checklist for completing modules. In overview, this approach serves as a checklist to complete the overall SoC design. The design flow dictates the tools required for each module and the verification necessary before integrating any design module into the system. The question for system developers becomes: "How much of our system could we implement with existing IP?" And perhaps more importantly, "How much of the design <u>can</u> we actually implement with existing IP?"

Naturally, some of the design modules that emerge from the design flow will require development of original design content. However, many of the design modules may be available as reusable IP from a number of different types of IP sources. System developers can turn to three different types of IP including star IP, standards-based or commodity IP, and market specific IP. Examples of star IP include vendor-specific embedded processors such as Motorola's M*Core, Star*Core, Coldfire, DSP, or PowerPC processors. Standards-based IP examples include Ethernet, Firewire, IRDA, USB Ports, and serial interfaces. Market specific IP examples include MPEG, VoCoder, FLEX (pager protocols), and CDMA/TDMA (cell phone protocols).

Design for Reuse

Although design reuse is a familiar and easily understood concept, it is not commonly practiced. When reuse has occurred, it has been applied on a small scale within local organizations. It has not been efficiently implemented in a way that optimizes a rapid SoC design cycle. Efficient reuse requires someone willing and able to design a block for reuse and someone willing and able to reuse it.

Design for reuse, or more accurately, creating IP specifically formatted for the reuse needs of the IP integrator, requires more initial design time than design for a single use. Reusable IP design requires a well-documented, parameterized general solution at a high level of abstraction. The reusable IP can then be easily adapted to meet various functional and process technology requirements in subsequent applications.

Once a reusable design has been created, it must be deposited in a widely and readily accessible IP repository and the IP creator must also be willing to provide technical support to subsequent IP integrators. With an effective industry wide IP repository in place, it's conceivable that in a hypothetical platform-based design that contains 20 different modules, existing IP could be reused in as many as 16 or 18 of the modules. Actual new design may involve only a few modules. To be effective in the quest to cut time-to-market, reusable IP must reliably conform to a Semiconductor Reuse Standard.

Design reuse, as a significant element in the design process, faces logistical, technical, and cultural barriers. When a designer is unwilling to consider IP reuse as an alternative to IP (re-)creation, it effectively implements a "not-invented-here" syndrome. Logistical barriers exist when the desired IP is difficult to obtain, either because it does not exist when needed, cannot be located, requires lengthy legal and financial negotiations to acquire it, or is too costly. Technical barriers arise because of difficulties integrating IP that is incompatible with the integration methodology and/or tools.

Most of the technical issues can be addressed by establishing design for reuse standards that ensure completeness and compatibility. But standards alone do not assure system designers that IP can in fact be reused. To actually realize the benefits of reuse, IP providers must make it efficient and attractive to reuse IP. Further, a compatible SoC development system needs to be deployed to ensure the plug-and-play integration of the compliant IP into an SoC. Lastly, the system design community must reorient itself to optimize IP reuse vs. recreating existing IP.

Reuse Studies

A design block requires additional design time and effort so that it can be reused and integrated in subsequent designs. A study conducted by Collett International, Inc. shows that the effort to reuse IP in a second use currently requires 41% of the original design effort. If it took 100 hours to create the original IP, it currently requires 41 hours to integrate the same IP block in a second use and subsequent uses. However, this study also suggests that with optimized original design for reuse, future reuse and integration of

the same IP block must be reduced to as little as 7% of the original effort. Thus, an efficiently reusable IP block that originally required 100 hours, requires only 7 hours to effectively integrate in second and subsequent reuse applications.

Looking at a 5M-gate design under these reuse conditions, current engineering productivity is estimated at 50K gates per person per year for new designs. The following table lists the required effort and cost for the design with or without reuse. The associated reuse effort is 41% and 7% respectively with an assumed staffing cost of $125K per person year. Note that with no reuse on a strictly new design, the effort required is 100 manyears and the cost is $12.5M to implement a 5M-gate design. For a platform-based design it is assumed that 90% of the SoC content is being reused and only 10% is designed from scratch. Reuse scenario 1, where a high level of integration is required (41% effort), reduces effort to 47 manyears and costs to $5.9M. Reuse under optimized (7%) conditions results in the same end result produced with only 16.3 manyears effort for only $2M. (Show chart of costs, possibly treat as sidebar.)

As system developers reuse IP more and more, integrating reusable IP comes to dominate the design cycle. For system developers to achieve the 7% reuse efficiencies noted in Table xxx, clear, verifiable reuse standards must be in place and adhered to by the IP creators. To realize the required rapid cycle times for SoCs, IP reuse must be supported by an efficient reuse infrastructure as well as an associated SoC development ecosystem. With both in place, system designers can achieve efficient reuse not once, but many times over.

<INSERT SIDEBAR>

Two Reuse Case Studies

As part of Motorola's Semiconductor Reuse Standards (SRS) program, design teams evaluated a number of projects to assess the impact of IP reuse vs. recreating new IP. Two case summaries follow:

Project A:
Description: This project was developed for a Motorola customer using dual cores. In addition to the cores, Motorola provided several memory modules and a few small peripherals (including a PLL). The customer module is implemented in a sea of cells, while the cores and memories are placed as hard macros in the final design.

Size:	450K gates, of which 350K gates were developed by the customer. The cores comprise most of the remaining logic.
Project Scope & Notable Design Tasks	New Design using existing cores. ■ Operating frequency for the core: 66 MHz. ■ Motorola performed all integration and layout tasks.

Total Design Effort (With Reuse)	58 man-weeks. Reuse of core IP reduced required effort by 50 man-weeks.

Project B:
Description: This project was developed with a Motorola customer, using a modified version of the ColdFire microprocessor. In addition to the ColdFire core, Motorola provided libraries and RTL for several peripherals. Motorola, in turn, synthesized the customer module, integrated the modules, and performed all back-end tasks. The customer module and several of the Motorola-supplied peripherals were implemented in a sea of cells, while the core and one of the peripherals were placed as hard macros in the final design.

Size	560K gates, of which half were on the customer module and the remainder were the core peripherals and library modules supplied by Motorola.
Project Scope & Notable Design Tasks	New design using existing microprocessor core, consisting of RTL, netlist, SDF, and physical views: ■ the microprocessor core was only slightly modified 2 modules were reused: ■ USB interface ■ 10/100 BaseT Ethernet
Total Design Effort (With Reuse)	67 Man-weeks. Reuse of USB and Ethernet IP reduced effort by 78 weeks.

<END Sidebar>

The Peripheral Bus Interface and IP Deliverables

The first step in any reuse standard is to establish a list of required IP deliverables with a focus on the key architectural aspects such as bus standards. Perhaps the most enabling portion of an architectural reuse standard is the Peripheral Bus Interface standard. The Peripheral Bus Interface, a set of standard module interface definitions, allows IP modules to easily connect with other Peripheral Bus Interface compliant modules and various CPU architectures. Each standard module interface is defined by a “colored-line” (e.g., Blue-Line, Purple-Line, Tan-Line). Currently, several bus standards have been publicly announced including the VSIA VC Interface, Motorola's Peripheral Interface Standard, and IBM's CoreConnect.

However, the Peripheral Bus Interface standard is only one part of the overall IP Deliverables. IP Deliverables provide the user community with a standard list of dataviews required for any IP module. This list enables easy integration into an SoC and supports the standard set of tools necessary for the emergence of an SoC ecosystem.

SRS Standards

Motorola's SRS consists of 12 areas which are combined into 5 categories. The categories range from system level aspects, through architectural, implementation and verification issues to general IP related aspects such as documentation and the deliverables list. Each area addresses a specific aspect of the IP creation process. For example, the architectural category addresses the on-chip and off-chip interfaces. The implementation category focuses on IP creation and SoC integration aspects such as HDL coding standards, physical representation, cell libraries, and analog. Functional verification and testability standards are included in the verification category.

Summary

Systems developers, confronted by the twin forces of miniaturization and consumerization, have reduced their time-to-market design cycles to as low as three months. Additionally, rapidly increasing chip complexity has outstripped the rate of design productivity, creating a gap. One of the most crucial strategies that systems developers can pursue to close the gap is to leverage an efficient reuse infrastructure based on VSIA or proprietary reuse standards.

Reuse studies have shown that to fully leverage the benefits of IP reuse for rapid cycle times, IP reuse must be performed efficiently. The effort related to the integration of IP into an SoC must be less than 10% of the original IP development time. To achieve this level of reuse, Semiconductor Reuse Standards must be implemented and fully supported in an efficient and sophisticated reuse infrastructure. Furthermore, an SoC development ecosystem tightly linked with the reuse infrastructure enables plug-and-play integration of the IP and reduces the reuse effort to under 10%.

# # #

2908 Words