Networks on Chips for Consumer Electronics

Kees Goossens*

*Philips Research, Eindhoven, The Netherlands. E-mail: kees.goossens@philips.com

1 Introduction

Systems on a chip (SOC) are complex embedded systems consisting of many hardware and software blocks (also known as intellectual property, or IP). As the complexity of SOCs grows, the focus is less on the computation, and increasingly on communication. This results in a shift from design based on platforms [KMN+00] (design templates) to a design style that is *communication-centric* [SSM+01, NTIJ04]. In this new paradigm, on-chip interconnects must address a number of challenges. First, we list those related to the application domain:

- Formerly separate *application domains are converging* in today's mobile phones, personal digital assistants, set-top boxes, and so on. Applications with different characteristics in terms of computation and communication are being mixed within a single device, leading to many heterogeneous modes. Moreover, even single applications are becoming more dynamic, e.g. dynamic object detection and synthesis in MPEG4.
- Systems are increasingly *embedded*, which means that they often have a concrete interaction with the real world. Whereas a crash of a personal computer is annoying, it will not spoil your food, as a crash of a refrigerator may. When systems are embedded in heating systems, cars, and so on, the system behaviour becomes *real time and safety critical*.
- On a related topic, in the consumer electronics domain, *reliable and predictable behaviour* of devices is the norm. A television that crashes is not considered fit for purpose, and the user will return it to shop. This places high demands on system predictability.
- Finally, price (*low cost*) is a critical factor in consumer electronics.

Second, as the complexity of embedded systems grows, managing the design of the SOCs that implement them is of critical importance.

- One of the foremost problems of SOC design concerns the deep submicron aspect, in particular the local and global *physical wires* on a chip. To avoid congestion in the lay out, the number of global wires should be minimised. However, the local wires of an IP and the global wires connecting it to other IPs are often not distinguished. As a result, the timing correctness of different IPs is interdependent: correcting a timing violation in one IP may invalidate the timing of another. Hence, the process of verifying the timing of the SOC as a whole (*global timing closure*), does not necessarily converge to a solution, and is specific for each design (not reusable) [The00].
- Reducing the *time to design* a SOC is important. Profits decrease rapidly when the window for introducing a product on the market has passed. We should also lower the cost of designing a SOC, which is significant with large design teams.

In the communication-centric design paradigm, on-chip interconnects must address the problems described above. *Networks on chips* (NOC) have emerged as a new type of interconnect that can solve

these problems [DT01, BDM02, JT03]. NOCs apply and adapt packet-switching and routing concepts from computer networks to the context of SOCs. In particular:

- Heterogeneous applications will result in *diverse communication* types (e.g. streaming, control, event-based, message passing, memory-mapped) [SSM+01]. Protocols stacks such as OSI [Zim80] enable multiple application protocols on one underlying network protocol.
- Different communication levels, or *quality of service* (QOS) is an important concept that receives increasing attention. In particular, guaranteed communication services (GS), such as guaranteed delivery, minimum bandwidth, maximum latency and jitter, can help in designing robust real-time and safety-critical systems. Also, many communications have intrinsic real-time requirements (such as audio and video).
- Guaranteed services *decouple* the communication behaviours of different IPs. As a result, SOCs can become more robust, e.g. failure of one component does not affect the remainder of the system.
- NOCs *reduce the number of global wires* by sharing them. This is predominantly done by re-using packet-switching and routing concepts from computer networks. Furthermore, by using well-defined IP interfaces such as AXI [ARM03], the implementation and timing verification of an IP becomes independent of other IP. This is essential to reduce global timing closure to a number of simpler independent local timing closures. It also fits very well with a globally-asynchronous locally-synchronous timing style [MVF00].
- NOCs with guaranteed performance reduce the time to design a SOC because the constituent IPs can be designed and verified independently. This enables a *modular, compositional design style*.
- NOC components are re-usable IP themselves. However, as every SOC requires its own specific NOC, an *automated* NOC *design flow* is an essential part of any SOC platform [GDG+05].
- NOCs offering guaranteed services allow *analytical* reasoning about the performance of the application using the NOC. In this way, performance guarantees can be computed quickly and exactly (not statistical), without simulation [GRG+05]. This reduces the time to verify a SOC.
- Moreover, because a NOC is run-time *programmable* they can make the SOC as a whole more programmable. The SOC can be re-used for multiple applications. This amortises the design cost over a larger number of devices, and lowers the average cost.

We conclude that, while many research challenges remain, NOCs have great potential to be at the core of the emerging communication-centric paradigm.

2 Networks on Chip

In this section we briefly outline some of the challenges that arise in the design of a NOC with the requirements of the previous section. Networks in general consist of a number of routers that send packets to one another. Because the NOC is made up of a number of distributed arbiters (the routers), giving global (end-to-end) performance guarantees is challenging [Zha95, RHS98]. A number of different approaches to offering GS have emerged:

- *circuit switching*, whereby the links on path in the NOC are fully reserved for a single connection (Metro [CMd+94], SOCBus [WL03]). Router buffers are minimal, but offering GS requires many disjoint paths in the NOC (increasing wire cost). Latency is minimal.
- *virtual circuit switching*, whereby the links on a path are reserved but shared with others, using virtual channels [Fel04, WSRS05, BS05]. A one-flit buffer is required for every virtual channel in every router. Latency depends on the link scheduling, and falls between the pure circuit switching and contention-free approaches.

- *contention-free* approaches that include Æthereal [RGR+03], Nostrum [MNTJ04], aSOC [LLJ+01]. Although links are shared, contention is avoided, usually by use of time-division multiple access (TMDA) virtual circuit switching. No virtual channels are required, reducing router buffer sizes to one flit, at the cost of higher average latency.

[GvMPW02, BS05, RGR+03, MNTJ04] offer both best-effort (BE) and guaranteed services.

In all cases, *network interfaces* (NI) that convert the IP view on communication (such as AXI transactions) to the router view (packets), are complex components [RDGP+05]. NIs implement packetisation, admission control, and end-to-end flow control, etc. Routers and NI are parametrisable, the latter to a much higher degree.

The router and NI are the basic components from which SOC-specific NOC are to be constructed. A NOC design flow [PCSV03, BMC+04, HM04, MDM03, GDG+05] must aid in this complex task. First a NOC *topology* must be synthesised (easy if restricted to regular topologies). Simultaneously or subsequently, IP ports must be *mapped* to NI ports. This stage has great impact on the performance of the resulting SOC. At this point VHDL RTL implementations and/or SystemC models can be produced. Often NOCs can be programmed or *configured* at run-time, necessitating the generation of embedded software that configures the NOC. In all cases, the NOC and the IP models can be simulated.

As introduced in the previous section NOCs with guaranteed services have the major advantage that performance verification can be added as a step to the NOC design flow [GDG+05, GRG+05]. Without simulation, analytical performance guarantees can be quickly computed at design time.

3 Conclusions

Systems on chip for consumer applications are complex ICs, where the integration of IPs to a working system is the bottle neck. Networks on chip have great promise to lift existing platform-based design to communication-centric design. NOCs do so by 1) addressing deep submicron challenges (global wires, global timing closure, et.), and 2) offering a structured view on communication between IPs inspired by protocol stacks. In particular, NOCs that offer guaranteed communication services make SOCs more robust and easier to design.

References

[ARM03] ARM. *AMBA AXI Protocol Specification*, June 2003.

[BDM02] Luca Benini and Giovanni De Micheli. Networks on chips: A new SoC paradigm. *IEEE Computer*, 35(1):70–80, 2002.

[BMC+04] Evgeny Bolotin, et al. Automatic hardware-efficient SoC integration by QoS network on chip. In *ICECS*, December 2004.

[BS05] Tobias Bjerregaard and Jens Sparso. A router architecture for connection-oriented service guarantees in the MANGO clockless network-on-chip. In *DATE*, March 2005.

[CMd+94] Frederic Chong, et al. Metro: A router architecture for high-performance, short-haul routing networks. In *Int'l Symposium on Computer Architecture*, April 1994.

[DT01] William J. Dally and Brian Towles. Route packets, not wires: on-chip interconnection networks. In *DAC*, pages 684–689, 2001.

[Fel04] Tomaz Felicijan. *Quality of Service (QoS) for Asynchronous On-Chip Networks*. PhD thesis, Department of Computer Science, Faculty of Science and Engineering, University of Manchester, 2004.

[GDG+05] Kees Goossens, et al. A design flow for application-specific networks on chip with guaranteed performance to accelerate SOC design and verification. In *DATE*, pages 1182–1187, March 2005.

[GRG+05] Om Prakash Gangwal, et al. Building predictable systems on chip: An analysis of guaranteed communication in the Æthereal network on chip. In Peter van der Stok, editor, *Dynamic and Robust Streaming In And Between Connected Consumer-Electronics Devices*, volume 3 of *Philips Research Book Series*, chapter 1, pages 1–36. Springer, 2005.

[GvMPW02] K. Goossens, J. van Meerbergen, A. Peeters, and P. Wielage. Networks on silicon: Combining best-effort and guaranteed services. In *DATE*, pages 423–425, March 2002.

[HM04] Jingcao Hu and R. Marculescu. Energy-aware communication and task scheduling for network-on-chip architectures under real-time constraints. In *DATE*, 2004.

[JT03] Axel Jantsch and Hannu Tenhunen, editors. *Networks on Chip*. Kluwer, 2003.

[KMN+00] K. Keutzer, et al. System-level design: Orthogonalization of concerns and platform-based design. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 19(12), 2000.

[LLJ+01] Andrew Laffely, et al. Adaptive systems on a chip (aSoC) for low-power signal processing. In *Proc. of the Asilomar Conference on Signals, Systems, and Computers*, 2001.

[MDM03] Srinivasan Murali, et al. SUNMAP: A tool for automatic topology selection and generation for NOCs. In *DAC*, June 2003.

[MNTJ04] Mikael Millberg, et al. Guaranteed bandwidth using looped containers in temporally disjoint networks within the Nostrum network on chip. In *DATE*, 2004.

[MVF00] Jens Muttersbach, Thomas Villiger, and Wolfgang Fichtner. Practical design of globally-asynchronous locally-synchronous systems. In *Async*, April 2000.

[NTIJ04] Jari Nurmi, Hannu Tenhunen, Jouni Isoaho, and Axel Jantsch, editors. *Interconnect-Centric Design for Advanced SoC and NoC*. Kluwer, 2004.

[PCSV03] A. Pinto, L.P. Carloni, and A.L. Sangiovanni-Vincentelli. Efficient synthesis of networks on chip. In *ICCD*, 2003.

[RDGP+05] Andrei Rădulescu, et al. An efficient on-chip network interface offering guaranteed services, shared-memory abstraction, and flexible network programming. *IEEE Transactions on CAD of Integrated Circuits and Systems*, 24(1):4–17, January 2005.

[RGR+03] E. Rijpkema, et al. Trade offs in the design of a router with both guaranteed and best-effort services for networks on chip. *IEE Proceedings: Computers and Digital Technique*, 150(5):294–302, September 2003.

[RHS98] Jennifer Rexford, John Hall, and Kang G. Shin. A router architecture for real-time communication in multicomputer networks. *IEEE Transactions on Computers*, 47(10):1088–1101, October 1998.

[SSM+01] M. Sgroi, et al. Addressing the system-on-a-chip interconnect woes through communication-based design. In *DAC*, pages 667–672, June 2001.

[The00] T. N. Theis. The future of interconnection technology. *IBM journal of research development*, 44(3):379–390, May 2000.

[WL03] D. Wiklund and Dake Liu. Socbus: switched network on chip for hard real time embedded systems. In *IPDPS*, 2003.

[WSRS05] Pascal T. Wolkotte, et al. An energy-efficient reconfigurable circuit switched network-on-chip. In *RAW*, April 2005.

[Zha95] Hui Zhang. Service disciplines for guaranteed performance service in packet-switching networks. *Proceedings of the IEEE*, 83(10):1374–96, October 1995.

[Zim80] H. Zimmermann. OSI Reference Model - the ISO model of architecture for open systems interconnection. *IEEE Transactions on Communication*, COM-28(4):425–432, April 1980.