Chapter 1
Interconnect Issues in High-Performance Computing Architectures

Alberto Scandurra

Abstract Systems on chip (SoCs) are complex systems containing billions of transistors integrated in a unique silicon chip, implementing highly complex functionalities by means of a variety of modules communicating with the system memories and/or between them through a proper communication system. Integration density is now so high that many issues arise when a SoC has to be implemented, and the electrical limits of interconnect wires are a limiting factor for performance. The main SoC building-block to be affected by these problems is the on-chip communication system (or on-chip interconnect), whose task is to ensure effective and reliable communication between all the functional blocks of the SoC. A novel methodology aiming at solving the problems mentioned above consists of splitting a complex system over more dice, exploiting the so-called system in package (SiP) approach and opening the way to dedicated high-performance communication layers such as optical interconnect. This chapter deals with the SoC technology, describes current solutions for on-chip interconnect, illustrates the issues faced during the SoC design and integration phases and introduces the SiP concept and its benefits.

Keywords System on chip (SoC) • Interconnect • Bus • Network on chip (NoC) • Integration • System in package (SiP)

Outlook

Systems on chip (SoCs) are complex systems containing billions of transistors integrated in a single silicon-chip, implementing highly complex functionalities by means of a variety of modules communicating with the system memories and/or between

A. Scandurra (✉)
OCCS Group, STMicroelectronics,
Stradale Primosole 50, 95121, Catania, Italy
e-mail: Alberto.scandurra@st.com

I. O'Connor and G. Nicolescu (eds.), *Integrated Optical Interconnect Architectures for Embedded Systems*, Embedded Systems, DOI 10.1007/978-1-4419-6193-8_1,

them through a distinct and organized communication system. Ever-increasing integration density has led to the emergence of many issues in the implementation of systems on chip, not least the electrical limits of interconnect wires as a limiting factor for performance. In this context, a new technology is required for on-chip interconnect, in order to overcome current physical and performance issues.

In order to cover all the topics introduced above, this chapter is organized as follows:

- Section “Introduction to Systems on Chip” describes the SoC as the modern approach for designing and integrating complex systems.
- Section “On-Chip Communication Systems” deals with the SoC communication infrastructure, illustrating the concepts of the on-chip bus and network on chip.
- Section “SoC Performance and Integration Issues” describes physical and performance issues usually met during the SoC integration phase.
- Section “The Interconnect Bottleneck” describes how the interconnect, rather than logic gates, is now the major origin of performance and physical issues.
- Section “3D Interconnect” deals with Systems in Package and die to die communication.

Introduction to Systems on Chip

The system on chip (SoC) is now the essential solution for delivering competitive and cost-efficient performance in today’s challenging electronics market. Consumers using PCs, PDAs, cell-phones, games, toys and many other products demand more features, instant communications and massive data storage in ever smaller and more affordable products. The unstoppable drive in silicon fabrication has delivered technology to meet this demand—chips with hundreds of millions of gates using 130 nm processes are no more than the size of a thumbnail. These SoCs present one of the biggest challenges that engineers have ever faced; how to manage and integrate enormously complex designs that combine the richest imaginable mix of microprocessors, memories, buses, architectures, communication standards, protocol processors, interfaces and other intellectual property components where system level considerations of synchronization, testability, conformance and verification are crucial. Integrated circuit (IC) design has become a multi-million-gate challenge for which the demands on design teams are ever greater.

The techniques used in designing multi-million-gate SoCs employ the world’s most advanced electronic design automation (EDA), with a level of sophistication that requires highly trained and experienced engineers. Key issues to be managed in the design process include achieving timing closure that accounts for wire delays in the metal interconnects inside the chip, and designs for tests so that the chips can be manufactured economically. Early prediction of the right architecture, design-flow and best use of EDA solutions is required to achieve first silicon success and necessarily decrease the time-to-market from years to months.

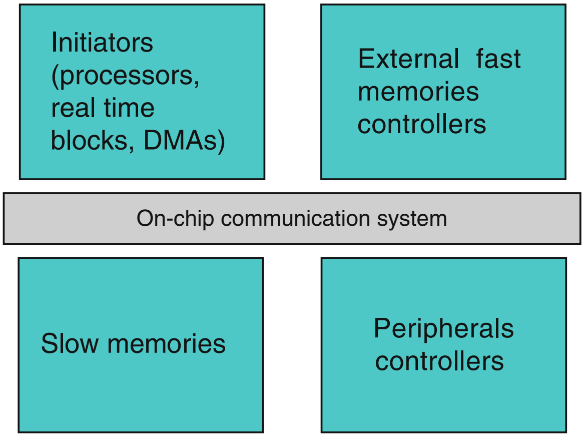

Initiators (processors, real time blocks, DMAs)
External fast memories controllers
On-chip communication system
Slow memories
Peripherals controllers

Fig. 1.1 Typical organization of a SoC

The building-blocks of a SoC can be distinguished as initiators or processing elements (PEs), targets or storage elements (SEs), and communication infrastructure blocks, composing as a whole the on-chip interconnect (see Fig. 1.1); *initiators* represent all blocks able to generate traffic, i.e., write data into a SE and read data from a SE; *targets* are blocks able to manage the traffic generated by the initiators.

Among the initiators of the system the following classes can be identified:

- Processors
- Real time initiators
- DMAs (direct memory access)

Processors, such as the ST20, ST40, ST50 and LX from STMicroelectronics, have strict requirements in terms of latency and bandwidth, and their bandwidth must further be in some way limited to allow the other initiators to be serviced.

Real time initiators, such as audio/video blocks, are more latency-tolerant than processors, but have strict requirements in terms of bandwidth.

DMAs do not have any particular requirements in terms of latency or bandwidth, and can normally work using the remaining bandwidth, i.e. the part of the bandwidth not used by the processors and real time initiators.

Among the targets the following classes can be identified:

- External fast memories
- Internal slow memories
- Peripherals

External fast memories comprise high performance memories such as SDRAM (synchronous dynamic random access memory) and DDR (dual data rate) SDRAM, used mainly for real time applications (e.g. video), and today operating at around 400 MHz. Their speed is limited by physical constraints imposed by pads.

Slow memories are usually low-performance memories such as SRAM and Flash, used for the storage of huge amounts of data, whose access is managed by caches, and operating at around 200 MHz. Their speed is limited by application requirements.

Peripherals are slow memories such as I^{2}C and Smartcard, used where no high performance is required, and operating at around 50/100 MHz.

Normally the CPUs run at the highest speed and the memory system represents the SoC bottleneck in terms of performance.

Hence within a single chip, different circuit “islands” run at different frequencies; this approach is called GALS (globally asynchronous locally synchronous) and is widely used today. The different clock frequencies required to operate the various subsystems are generated by the clock generator (clockgen), while the subsystems are linked together by the on-chip interconnect, such as the STBus/STNoC [1] in the case of STMicroelectronics products. Typically the on-chip interconnect optimizes the CPU path, i.e. the interconnect structure normally operates at the same frequency as the CPU. Since the other subsystems often operate at a different frequency, dedicated frequency converters have to be placed between the interconnect and the other subsystems to enable inter-block communication.

On-Chip Communication Systems

As already shown in Fig. 1.1, a SoC can be seen as a number of intellectual properties (IPs) properly connected by an on-chip communication architecture (OCCA), an infrastructure that interconnects the various IPs and provides the communication mechanisms necessary for distributed computation over a set of heterogeneous processing modules. The throughput and latency of the communication infrastructure, and also the relevant power consumption, often limit the overall SoC performance.

Until now the prominent type of OCCA has been the on-chip bus, such as the STBus from STMicroelectronics, the AMBA bus from ARM [2], CoreConnect from IBM [3], which represent the traditional shared-communication medium. This type of OCCA, while not at all scalable, has been able to fulfill SoC requirements because the performance bottleneck has always been the memory system. However, with the growing requirements of more modern SoCs and CMOS technology scaling, the performance bottleneck is moving from memories to interconnect, as detailed in Sect. 4.

In order to overcome this limit, a new generation architecture, called network on chip (NoC), has been deeply studied and proposed; it is an attempt to translate the networking and parallel computing domain experience into the SoC world, relying on a packet-switched micro-network backbone based on a well-defined protocol stack. Innovative NoC architectures include STNoC from STMicroelectronics [4], Æthereal from Philips Research Lab [5], and Xpipe from University of Bologna [6].

On-Chip Bus

On-chip buses are communication systems composed of intelligent logic, responsible for arbitration among the possible traffic flows injected by the different SoC initiators (PEs able to generate traffic), and a set of physical channels through which the traffic flows are routed from initiators to targets (PEs able to receive and process traffic) and vice versa.

The peculiarities of a bus, which are also the main drawbacks, are:

- *Limited available bandwidth*, given by the product of the bus size (width) by the bus operating frequency. To achieve a higher available bandwidth implies either widening the bus size, thereby amplifying physical issues such as wire congestion, or increasing the operating frequency, leading to increased power consumption, and which is moreover limited by physical issues such as capacitive load and capacitive coupling.
- *Lack of bandwidth scalability*, since connecting more IPs to the bus implies dividing the total available bandwidth among all the IPs, thereby allocating a lower bandwidth to each of them.
- *Limited system scalability*, since connecting more IPs to the bus results in an increase of the capacitive load, which leads to a drop in operating frequency.
- *Limited quality of service*, since there is no possibility to process different classes of traffic (such as low latency CPUs, high bandwidth video/audio processors, DMAs) in a different way.
- *High occupation area*, due to the large number of wires required to transport all the protocol information, i.e. data and control signals (STBus interfaces for example are characterized by hundreds of wires).
- *High power consumption*, which is determined by the switching activity and potentially affects all the wires of the bus.

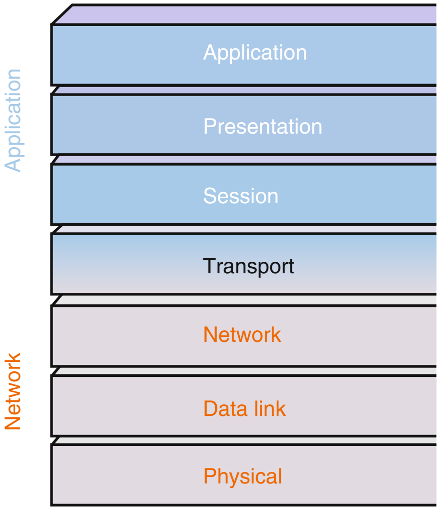
Network on Chip

The new requirements of modern applications impose the need for new solutions to overcome the previously mentioned drawbacks of on-chip buses, both for the classic shared-bus (such as AMBA AHB) and the more advanced communication systems supporting crossbar structures (such as the STBus). In conjunction with the most recent technology features, a novel on-chip communication architecture, called network on chip (NoC), has been proposed.

It is important to highlight that the NoC concept is not merely an adaptation to the SoC context of parallel computing or wide area network domains; many issues are in fact still open in this new field, and the highly complex design space requires detailed exploration. The key open points are, for instance, the choice of the network topology, the message format, the end-to-end services, the routing strategies, the flow control and the queuing management. Moreover, the type of quality of service (QoS) to be provided is another open issue, as is the most suitable software view to allow the applications to exploit NoC infrastructure peculiarities.

From lessons learned by the telecommunications community, the global on-chip communication model is decomposed into layers similar to the ISO–OSI reference model (see Fig. 1.2). The protocol stack enables different services and allows QoS,

Fig. 1.2 ISO–OSI protocol stack



providing to the programmer an abstraction of the communication framework. Layers interact through well-defined interfaces and they hide any low-level physical DSM (Deep SubMicron) issues.

The *Physical* layer refers to all that concerns the electronic details of wires, the circuits and techniques to drive information (drivers, repeaters, layout), while the *Data link* layer ensures reliable transfer despite the physical unreliability and deals with medium access (sharing/contention). At the *Network* level there are issues related to the topology and the consequent routing scheme, while the *Transport* layer manages the end-to-end services and the packet segmentation/re-assembly. The other levels, up to the *Application* layer, can be viewed as a sort of merged adaptation layer that implements (in hardware or through part of an operating system) services and exposes the NoC infrastructure according to a proper programming model [e.g. the message passing (MP) paradigm].

Despite the similarity discussed above, it is clear that the micro-network in the single chip domain differs from the wide-area network. Distinct features of NoCs include the spatial locality of connected modules, the reduced non-determinism of the on-chip traffic, the stringent energy and latency constraints, the possibility of application specific stack services, and the need for low cost solutions.

An open issue in NoC literature is the trade-off between the QoS provided by the network and the relevant implementation cost. QoS must be supported at all layers, and basic services are a fixed bandwidth, a maximum latency, the correctness (no errors) and the completion (no packet loss) of the transmission. Another approach consists of using a best effort service strategy, which allows for a better average utilization but cannot support a QoS. Since users demand application predictability, mixing both approaches could be a good solution.

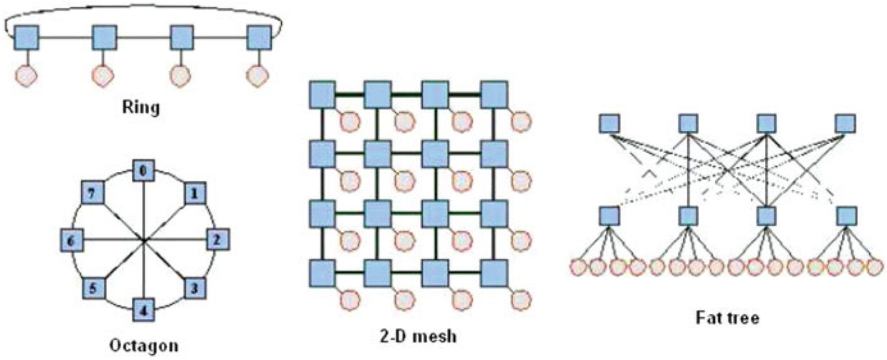


Fig. 1.3 Various NoC topologies

NoC communication is packet-based and the generally accepted forwarding scheme is a wormhole, because it allows for a deeper pipeline and a reduced buffering cost. Packets are divided into basic units called flits; the queues in each node have flit granularity and the physical node-to-node links are managed by a flow control that works on a flit per flit basis.

Another key point is the network topology, which has to be regular and simple. The literature points to hybrid solutions, with local clusters based on shared buses, and global communication using NoC. Some NoC state-of-the-art projects are based on the simple ring, two-dimensional mesh, fat tree [7], and Octagon [4] topologies, as shown in Fig. 1.3.

As far as the routing policy is concerned, it is possible to choose between deterministic, adaptive, source, arithmetic or table-driven schemes; deadlock handling is topology dependent. Input queues are suitable for a low cost implementation, but they show limited performance with respect to output buffering. In terms of control flow, many solutions select a simple request/grant scheme, others a more efficient credit-based one. Links can be noisy channels, so the literature begins to present work on error detection code or error correction code applied to on-chip interconnections, with distributed or end-to-end error recovery strategies.

Besides routers, a significant amount of area is consumed by the so-called network interface (NI) that is the "access" to the NoC, translating the connected IP transactions to packets that are exchanged in the network. The NI hides network dependent aspects to the PE, covering the transport layer (connection handling, de-assembling of messages, higher level services).

To summarize, the main benefits of the NoC approach are:

- *Modularity*, thanks to standard basic components, the NI and the Router
- *Abstraction* as an inherent property of the layered approach, fitting also the demands of QoS

- *Flexibility/scalability* of the network as a benefit of a packet-based communication
- *Regular* and well controlled structure to cope with DSM issues
- *Re-use* of the communication infrastructure viewed as a platform

Topology

A first parameter for the topology is its scalability; a topology is said to be scalable if it is possible to create larger networks of any size, by simply adding new nodes. Two different approaches can be followed for the specification of the topology of a NoC: topology-dependent and topology-independent. The former approach specifies the network architecture and its building blocks assuming a well defined topology. The latter aims at providing flexibility to the SoC architect in choosing the topology for the interconnect, depending on the application. This means that it is possible to build any kind of topology by plugging together the NoC building-blocks in the proper way. While this second approach is more versatile because of the higher configurability allowed, it also has the following drawbacks:

- A very wide design and verification space, which would require significant effort to ensure a high quality product to the NoC user.
- Exposure of the complexity of the network layer design (including issues such as deadlock) to the SoC architect, thus requiring novel specific competencies and a high effort in defining an effective (in terms of performance) and deadlock-free architecture.
- A need for high parametric building blocks, with few cost optimization possibilities.

Moreover, a NoC built on top of a specific topology still needs a high degree of flexibility (routing, flow control, queues, QoS) in order to properly configure the interconnect to support different application requirements.

Routing Algorithms

Routing algorithms are responsible for the selection of a path from a source node to a destination node in a particular topology of a network. A good routing algorithm balances the load across the various network channels even in the presence of non-uniform and heavy traffic patterns. A well designed routing algorithm also keeps path lengths as short as possible, thus reducing the overall latency of a message.

Another important aspect of a routing algorithm is its ability to operate in the presence of faults in the network. If a particular algorithm is hardwired into the routers and a link or node fails, the entire network fails. However, if the algorithm can be reprogrammed or adapted to bypass the failure, the system can continue to operate with only a slight loss in performance.

Routing algorithms are classified depending on how they select between the possible paths from a source node to a destination node. Three main categories are specified:

- Deterministic, where the same path is always chosen between a source and a destination node, even if multiple paths exist.
- Oblivious, where the path is chosen without taking into account the present state of the network; oblivious routing algorithms include deterministic routing algorithms as a subset.
- Adaptive, where the current state of the network is used to select the path.

Deadlock

A deadlock occurs in an interconnection network when a set of packets are unable to make any progress because they are waiting for one another to release network resources, such as buffers or channels.

Deadlock is a catastrophic event for the network. After a few resources are kept busy by deadlocked packets, other packets get blocked on these resources, thus paralyzing the network operation. In order to prevent such a problem, two solutions can be put into place:

- Deadlock avoidance, a method to guarantee that the network cannot become deadlocked.
- Deadlock recovery, a method consisting of detecting and correcting deadlock.

If deadlock is caused by dependencies external to the network, it is called high-level deadlock or protocol deadlock (hereafter we term low-level deadlock as that related to the dependencies of the topology plus the relevant routing algorithm). For instance a simple request/response protocol could lead to deadlock conditions when dependencies occur in target devices between the incoming requests and the outgoing responses.

A network must always be free of deadlock, livelock, and starvation. A livelock refers to packets circulating the network without making any progress towards their destination. Starvation refers to packets indefinitely waiting at a network buffer (due to an unfair queuing policy). Both livelock and starvation reflect problems of fairness in network routing or scheduling policies.

As far as deadlock is concerned, in the case of deterministic routing, deadlock is avoided by eliminating cycles in the resource dependency graph; this is a directed graph, which depends on the topology and the routing, where the vertices are the resources and the edges represent the relationships due to the routing function. In the case of wormhole packet switching, these resources are the virtual channels; so we talk about a virtual channel dependency graph. A virtual channel (VC) provides logical links over the same shared physical channels, by establishing a number of independently allocated flit buffers in the corresponding transmitter/receiver nodes. When the physical link is not multiplexed among different VCs, the resource dependency graph could be simply called a channel dependency graph.

Protocol (or high-level) deadlock refers to a deadlock condition due to resource dependencies external to the network. For instance, when a request-response protocol, such as STBus from STMicroelectronics or AMBA AXI from ARM, is adopted as end-to-end in the network, a node connected as target introduces dependencies between incoming requests and outgoing responses: the node does not perform as a sink for incoming packets, due to the finite size of the buffers and the dependencies between requests and responses.

In shared memory architectures, complex cache-coherent protocols could lead to a deeper level of dependencies. The effect of these protocol dependencies can be eliminated by using disjoint networks to handle requests and replies. The following two approaches are possible:

- Two physical networks, i.e., separated physical data buses for requests and responses.
- Two virtual networks, i.e., separated virtual channels for requests and responses.

Quality of Service

The set of services requested by the IPs connected to the network (called network clients) and the mechanisms used to provide these services are commonly referred to as QoS.

Generally, it is useful to classify the traffic across the network into a number of classes, in order to efficiently allocate network resources to packets. Different classes of packets usually have different requirements in terms of importance, tolerance to latency, bandwidth and packet loss.

Two main traffic categories are specified:

- Guaranteed service
- Best effort

Traffic classes belonging to the former category are guaranteed a certain level of performance as long as the injected traffic respect a well defined set of constraints. Traffic classes belonging to the latter category do not get any strong guarantee from the network; instead, it will simply make its best effort to deliver the packets to their destinations. Best effort packets may then have arbitrary delay, or even be dropped.

The key quality of service concern in implementing best effort services is providing fairness among all the best effort flows. Two alternative solutions exist in terms of fairness:

- Latency-based fairness, aiming at providing equal delays to flows competing for the same resource.
- Throughput-based fairness, aiming at providing equal bandwidth to flows competing for the same resource.

While latency-based fairness can be achieved implementing a fair arbitration scheme [such as round-robin or least recently used (LRU)], throughput-based fairness can be achieved in hardware by separating each flow requesting a resource into a separate queue, and then serving the queues in round-robin fashion. The implementation of such a separation can be expensive; in fact while physical channels (links) do not have to be replicated because of their dynamic allocation, virtual channels and buffers, requiring FIFOs, have to be replicated for each different class of traffic. So it is very important to choose the proper number of classes needing true isolation, keeping in mind that in many situations it may be possible to combine classes without a significant degradation of quality of service but gaining a reduction in hardware complexity.

Error Recovery

A high performance, reliable and energy efficient NoC architecture requires a good utilization of error-avoidance and error-tolerance techniques, at most levels of its layered organization. Using modern technologies to implement the present day systems (in order to improve performance and reduce power consumption), means adopting lower levels of power supply voltage, leading to lower margins of noise immunity for the signals transmitted over the communication network of the system. This leads to a noisy interconnect, which behaves as an unreliable transport medium, and introduces errors in the transmitted signals. So the communication process needs to be fault-tolerant to ensure correct information transfer. This can be achieved through the use of channel coding. Such schemes introduce a controlled amount of redundancy in the transmitted data, increasing its noise immunity.

Linear block codes are commonly used for channel encoding. Using an (n, k) linear block code, a data block of k bits length is mapped onto an n bit code word, which is transmitted over the channel. The receiver examines the received signal and declares an error if it is not a valid code word.

Once an error has been detected, it can be handled in one of two different ways:

- Forward error correction (FEC), where the properties of the code are used to correct the error.
- Retransmission, also called automatic repeat request (ARQ), where the receiver asks the sender to retransmit the code word affected by the error.

FEC schemes require a more complex decoder, while ARQ schemes require the existence of a reverse channel from the receiver to the transmitter, in order to ask for the retransmission.

SoC Performance and Integration Issues

In decananometric CMOS technologies, DSM effects are significant and the physical design of a SoC is increasingly faced with two types of issue:

- *Performance issues*, related mainly to the bandwidth requirements of the different IPs, that in order to be fulfilled, would require SoCs to run at very high speeds.
- *Integration issues*, related to the difficulties encountered mainly during the placement of the hard macros and the standard cells, and during the routing of clock nets and communication system wires.

Performance Issues

New generation systems will be composed of functional building blocks with a computation capability requiring a very high bandwidth (i.e. the number of bytes transferred per time unit) compared to those currently exploited. Bandwidth increase can be obtained in a variety of ways:

- Increasing the physical channel size
- Increasing the clock frequency

While this can be done with a few problems at IP level, for example with wider interfaces and/or faster transmission frequency, various problems affect the communication system to achieve the same target (the so called *offered throughput*), mainly in terms of congestion and crosstalk.

In fact, wider physical channels imply the need to route a higher number of wires between different points of the chip, resulting in routing and congestion issues. Increasing transmission frequency results in a higher level of energy coupling effects (crosstalk) between wires, leading to corruption of the transmitted signal. This is true for both bus-based interconnects and Networks on Chip, where the offered throughput is the aggregated throughput of all the links between different nodes.

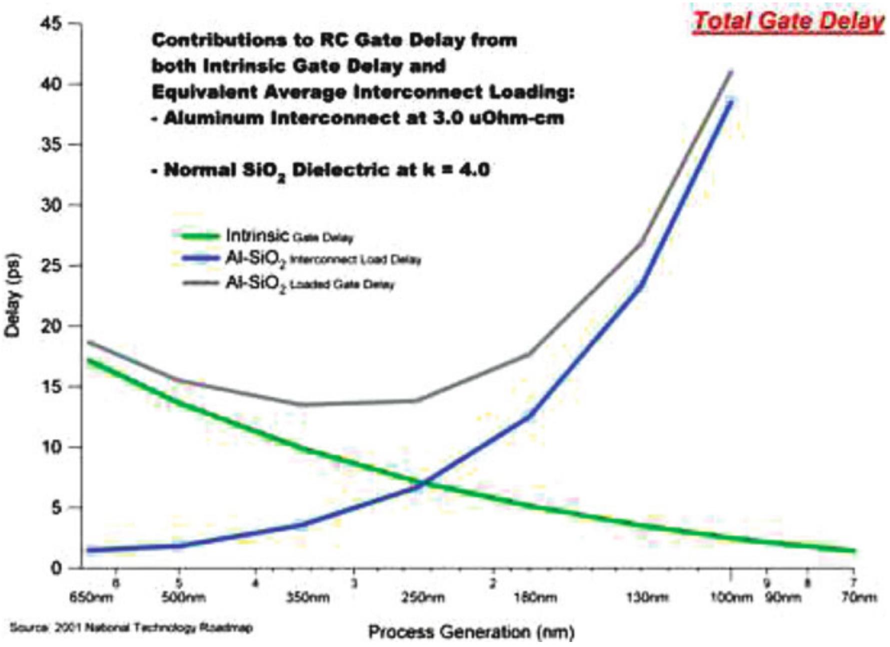
The throughput an on-chip interconnect can offer is also limited by physical implications. As far as the overall operating frequency of a SoC is concerned, two main factors influence it, namely the device switching times and the bandwidth offered by metallic wires. Current technologies can achieve unprecedented transistor transition frequencies due to short transistor lengths. However, the same is not true for interconnects. Indeed, continually shrinking feature sizes, higher clock frequencies, and growth in complexity are all negative factors as far as switching charges on metallic interconnect are concerned. This situation is shifting the IC design bottleneck from computing capability to communication.

Feature sizes on integrated circuits and also, therefore, circuit speed have followed Moore's law for over four decades and CMOS integration capability is still increasing. In this respect, according to the international technology roadmap for semiconductors (ITRS) [8], the RC time constants associated with metallic interconnects will not be able to decrease sufficiently for the high-bandwidth applications destined to appear in the next few years (see Fig. 1.4).

Internal data rates of processors fabricated in deep submicron CMOS technology have exceeded gigahertz rates. While processing proceeds at GHz internally, off chip wires have held inter-chip clock rates to hundreds of MHz.

Integration Issues

Figure 1.5 is an illustration of the physical issues; it shows the floorplan of an example CMOS chip for a consumer application.



Source: 2001 National Technology Roadmap

Fig. 1.4 Average interconnect delay as a function of process

In this figure the rectangles represent the various IPs of the chip (both initiators and targets); the space available for the communication system is the very irregular shape between all the different IPs. In such an area the Network Interfaces, representing the access points of the IPs to the on-chip network, the nodes, responsible for arbitration and propagation of information, and all the physical channels connecting the different NoC building-blocks have to be placed. Because of the shape, which is quite irregular and with thin regions, and the area size, it is evident that the placement of the interconnect standard cells can be difficult, and the routing of the wires that can be also very long will likely suffer congestion.

Electrical Interconnect Classification

From a technological point of view interconnects can be classified in the following categories (see Fig. 1.6):

- *Local interconnect*, used for short-distance communication, typically between logic units, and comprising the majority of on-chip wires; they have the smallest pitch and a delay of less than one clock cycle.
- *Global interconnect*, providing communication between large functional blocks (IPs); they are fewer than local interconnects, but are no less important. Improving the performance of a small number of critical global links can significantly

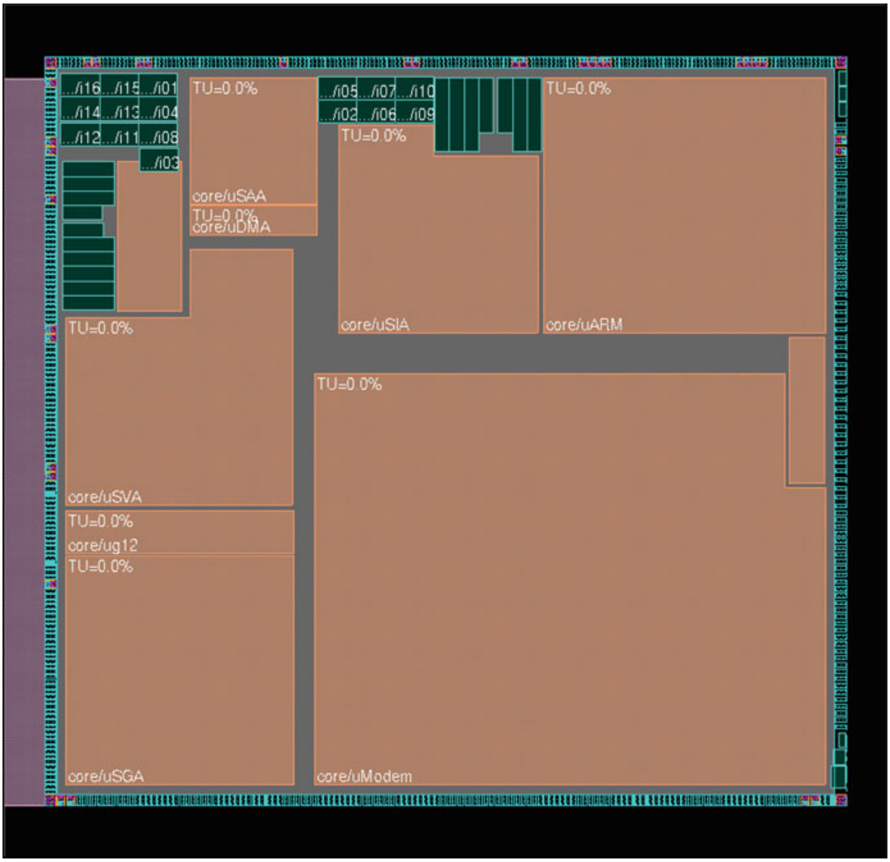


Fig. 1.5 Example CMOS chip floorplan

enhance the total system performance. Global interconnects have the largest pitch and a delay typically longer than one or two clock cycles.

- *Intermediate interconnect*, having dimensions that are between those of local and global interconnects.

A key difference between local and global interconnect is that the length of the former scales with the technology node, while for the latter the length is approximately constant.

From a functional point of view, the two main important and performance-demanding applications of interconnects in SoC are signaling (i.e. the communication of different logic units) and clock distribution. In this context they can be classified as:

- *Point-to-point links*, used for critical data-intensive links, such as CPU-memory buses in processor architectures.

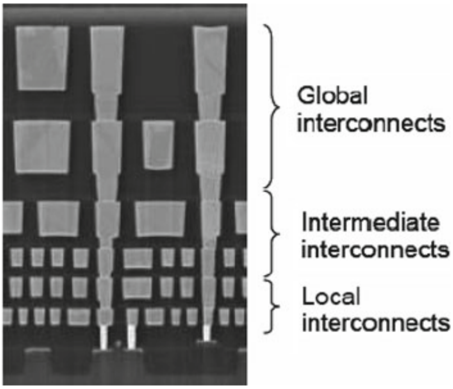


Fig. 1.6 Interconnect classification

- *Broadcast links*, representing physical channels where the number of receivers (and therefore repeaters) is high and switching activity is also high.
- *Network links*, targeted at system buses and reconfigurable networks, aiming at serving complete system architectures, whose typical communication is around several tens of GB/s.

The Interconnect Bottleneck

The continuous evolution and scaling down of CMOS technologies has been the basis of most of today's information technologies. It has allowed the improvement of the performance of electronic circuits, increasing their yield and lowering the cost per function on chip. Through this, the processing and storage of information (in particular digitally encoded information) has become a cheap commodity. Computing powers not imaginable only a few years ago have been brought to the desktops of every researcher and every engineer. Electronic ICs and their ever increasing degree of integration have been at the core of our current knowledge-based society and they have formed the basis of a large part of the growth of efficiency and competitiveness of large as well as small industries.

Continuing this evolution will however require a major effort. A further scaling down of feature sizes in microelectronic circuits will be necessary. To reach this goal, major challenges have to be overcome, and one of these is the *interconnect bottleneck*.

The rate of inter-chip communication is now the limiting factor in high performance systems. The function of an interconnect or wiring system is to distribute clock and other signals to and among the various circuits/systems on a chip. The fundamental development requirement for interconnect is to meet the high-speed transmission needs of chips despite further scaling of feature sizes. This scaling down however, has been shown to increase the signal runtime delays in the global interconnect layers severely. Indeed, while the reduction in transistor gate lengths increases the circuit speed, the signal delay time for global wires continues to increase with technology scaling, primarily due to the increasing resistance of the wires and their increasing lengths. Current trends to decrease the runtime delays, the power consumption and the crosstalk, focus on lowering the RC-product of the wires, by using metals with lower resistivity (e.g. Copper instead of Aluminum) and by the use of insulators with lower dielectric constant. Examples of the latter include nanoporous SiOC-like or organic (SilK type) materials, which have dielectric constants below 2.0 or air gap approaches, which reach values close to 1.8–1.7. Integration of these materials results in an increased complexity however, and they have inherent mechanical weaknesses. Moreover, introducing ultra low dielectric constant materials finds its fundamental physical limit when one considers that the film permittivity cannot be less than 1 (that of a vacuum).

Therefore, several researchers have come to the conclusion that "the global interconnect performance needed for future generations of ICs cannot be achieved even with the most optimistic values of metal resistivity and dielectric constants". Evolutionary solutions will not suffice to meet the performance roadmap and therefore radical new approaches are needed.

Several such possibilities are now envisaged, the most prominent of which are the use of RF or microwave interconnects, optical interconnects, 3D interconnects and cooled conductors. The ITRS roadmap suggests that research and evaluation is greatly needed for all these solutions for the next few years. Subsequently, a narrowing down of remaining solutions and start of an actual development effort is expected.

As has already been stated, the main limitations due to metallic interconnects are the crosstalk between lines and the noise on transmitted signals, the delay, the connection capability and the power consumption (due to repeaters). As a result, the Semiconductor Research Corporation has cited interconnect design and planning as a primary research thrust.

Electrical Interconnect Metrics

An ideal interconnect should be able to transmit any signal with no delay, no degradation (either inherent or induced by external causes), over any distance without consuming any power, requiring zero physical footprint and without disturbing the surrounding environment.

According to this, a number of metrics have been defined in order to characterize the performance and the quality of real interconnects, such as:

- Propagation delay
- Bandwidth density
- Power-delay product
- Bit error rate

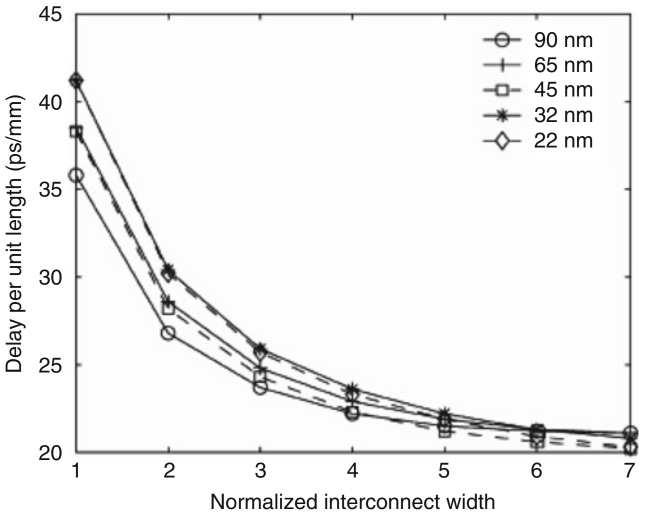


Fig. 1.7 Interconnect delay as function of interconnect width

Propagation Delay

The propagation delay is the time required by a signal to cross a wire. Pure interconnect delay depends on the link length and the speed of propagation of the wavefront (time of flight). Electrical regeneration introduces additional delay through buffers and transistor switching times. Additionally, delay can be induced by crosstalk.

It can be reduced by increasing the interconnect width at the expense of a smaller bandwidth density.

Technology scaling has insignificant effect on the delay of an interconnect with an optimal number of repeaters. The minimum achievable interconnect delay remains effectively fixed at approximately 20 ps/mm when technology scales from 90 to 22 nm, as shown in Fig. 1.7.

Bandwidth Density

Bandwidth density is a metric that characterizes information throughput through a unit cross section of an interconnect. Generally, it is defined by the pitch of the electrical wires.

Power-Delay Product

Signal transmission always requires power. In the simplest case, it is required to change the charge value on the equivalent capacitor of a metallic wire. In more realistic cases, power will also be required in emitter and receiver circuitry, and in regeneration circuits.

A distinction can also be made between static and dynamic power consumption by introducing a factor *a* representing the switching activity of the interconnect link ($0<a<1$).

The power-delay product (PDP) is routinely used in the technology design process to evaluate circuit performance.

Bit Error Rate

The bit error rate (BER) may be defined as the rate of error occurrences and is the main criterion in evaluating the performance of digital transmission systems.

For an on-chip communication system a BER of 10^{-15} is acceptable; electrical interconnects typically achieve BER figures better than 10^{-45}. That is why the BER is not commonly considered in integrated circuit design circles. However, future operation frequencies are likely to change this, since the combination of necessarily faster rise and fall times, lower supply voltages and higher crosstalk increases the probability of wrongly interpreting the signal that was sent.

Errors come from signal degradation. Real signals are characterized by their actual frequency content and by their voltage or current value limits. The frequency content will define the necessary channel bandwidth, according to Shannon–Hartley's theorem. Analogue signals are highly sensitive to degradation and the preferred mode of signal transmission over interconnect is digital.

Signal degradation can be classed as time-based, inherent and externally induced:

- Time-based: non-zero rise-time, overshoot, undershoot, and ringing time-based degradation can be incorporated into the delay term for digital signals. While the whole of these degradations can be assimilated into a quasi-deterministic behavior that does not exceed the noise margins of the digital circuits, a transformation in temporal space is possible (to contribute to the regeneration delay term). This assumption is however destined to disappear with nanometric technologies, because of a more probabilistic behavior and especially of weaker noise margins.
- Inherent: attenuation (dB/cm), skin effect, and reflections (dB).
- Externally induced: crosstalk (dB/cm) and sensitivity to ambient noise.

The allowable tolerance on signal degradation and delay for a given bandwidth and power budget forces a limit to the transmission distance. The maximum interconnect segment length can in fact be calculated, a segment being defined as a portion of interconnect not requiring regeneration at a receiver point spatially distant from its emission point.

Signal regeneration in turn leads to a further problem, i.e., the energy used to propagate the signal in the transmission medium can escape into the surrounding environment and perturb the operation of elements close to the transmission path.

3D Interconnect

The typical electronics product/system of the near future is expected to include all the following types of building-blocks:

- Digital processors (CPU)
- Digital signal processors (DSP)
- ASICs
- Memories
- Busses and NoC
- Peripheral and interface devices
- Analog baseband front-end
- RF and microwave processing stages
- Discrete components (R, L, C)
- Micro-electro-mechanical-systems (MEMS)
- Displays
- User interfaces

Several studies and technology roadmaps have highlighted that these electronics products of the future will be characterized by a high level of heterogeneity, in terms of the following mix:

- *Technology*: digital, analog, RF, optoelectronic, MEMS, embedded passives.
- *Frequency*: from hundreds of MHz in digital components domain till hundreds of GHz in RF, microwave and optical domains.
- *Signal*: digital circuits coexisting with ultra low-noise amplifier RF circuits.
- *Architecture*: heterogeneous architectures, i.e. event driven, data driven and time driven models of computation, regular versus irregular structures, tradeoffs required between function, form and fit over multiple domains of computational elements and multiple hierarchies of design abstraction.
- *Design*: electrical design to be unified with physical and thermal design across multiple levels of design abstraction.

In order to simplify the design and manufacturing of such complex and heterogeneous systems, relying on different technologies, an adequate approach would be to split them over a number of independent dice. Some, or even many, of the dice will need to be in communication with each other. This approach is known as system in package (SiP) [9], however many terms are in use that are almost synonymous: high density packaging (HDP), multi chip module (MCM), multi chip package (MCP), few chip package (FCP) [10]. In general the term SiP is used when a whole system, rather than a part, is placed into a single MCM.

The SiP paradigm moves packaging design to the early phases of system design including chip/package functionality partitioning and integration, which is a paradigm shift from the conventional design approach. Packaging has always played an important role in electronic products manufacturing; however in the early days its role was primarily structural in nature, while today and tomorrow it is playing

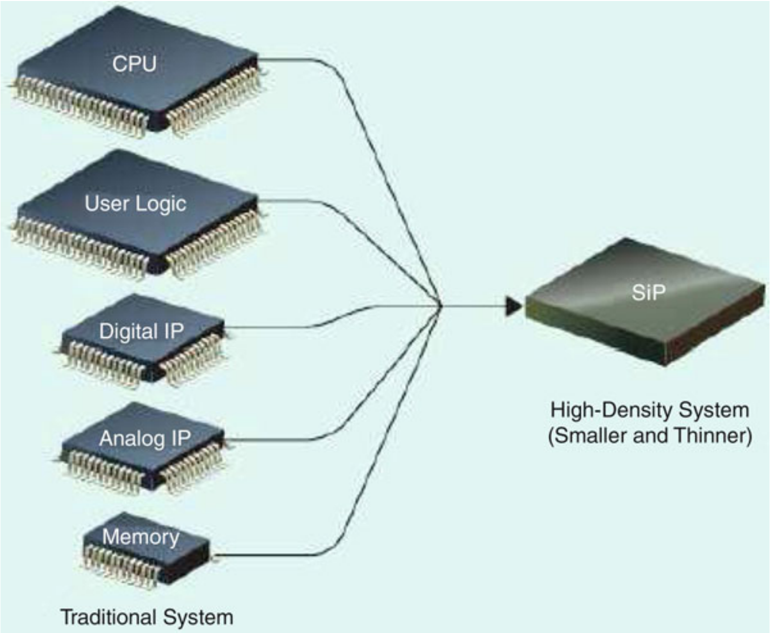


Fig. 1.8 Example of heterogeneous integration

increasingly important roles in carrying out the product's function and performance.

Such a technology offers many significant benefits, including:

- *Footprint*—more functionality fits into a small space. This extends Moore's Law and enables a new generation of tiny but powerful devices.
- *Heterogeneous integration*—circuit layers can be built with different processes, or even on different types of wafers. This means that components can be optimized to a much greater degree than if they were built together on a single wafer. Even more interesting, components with completely incompatible manufacturing could be combined in a single device (see Fig. 1.8). It is worth considering that non-digital functions (memory, analog) are best built in non-digital processes, that can be integrated in a low-noise and low-cost process by integrating them in a package, rather than in a chip with additional process steps.
- *Speed*—the average wire length becomes much shorter. Because propagation delay is proportional to the square of the wire length, overall performance increases.
- *Power*—keeping a signal on-chip reduces its power consumption by 10 to a 100 times. Shorter wires also reduce power consumption by producing less parasitic capacitance. Reducing the power budget leads to less heat generation, extended battery life, and lower cost of operation.

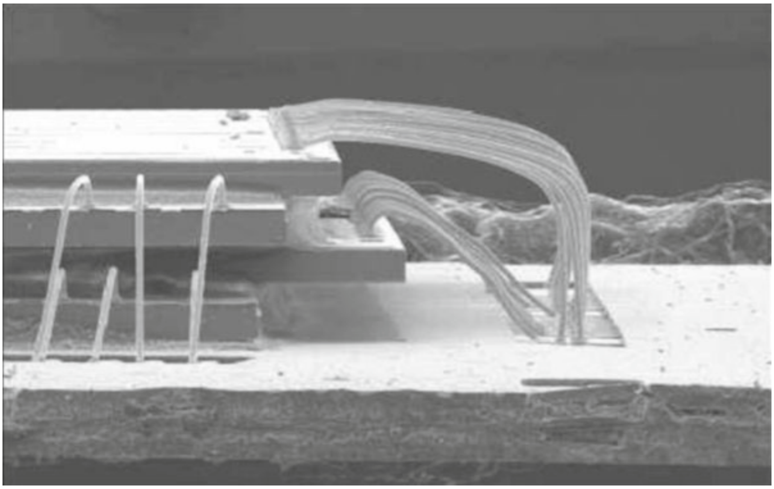


Fig. 1.9 Detail of electrical wires between dice

- *Design*—the vertical dimension adds a higher order of connectivity and opens a world of new design possibilities (see Fig. 1.9).
- *Circuit security*—the stacked structure hinders attempts to reverse engineer the circuitry. Sensitive circuits may also be divided among the layers in such a way as to obscure the function of each layer.
- *Bandwidth*—the lack of memory bandwidth is increasingly becoming the primary constraint for improved system performance, in particular in multimedia and data-intensive applications. Moreover, the random nature of memory accesses in many applications results in relatively ineffective caches and the memory bandwidth becoming strongly dependent on SDRAM accesses. 3D integration allows large numbers of vertical vias between the layers. This allows the construction of wide bandwidth buses between functional blocks in different layers. A typical example would be a processor plus memory 3D stack, with the cache memory stacked on top of the processor. This arrangement allows a bus much wider than the typical 128 or 256 bits between the cache and processor. Wide buses in turn alleviate the memory wall problem. Figure 1.10 highlights the communication wires between two dice, in both cross section view and top view.

Summarizing, the system in package technology offers the possibility to improve significantly the overall system performance when the system is too large to fit on a single chip, or when the system is a mixed-signal one and putting everything into a single chip is not possible from the technological point of view.

However, in spite of the significant advantages the SiP approach gives with respect to the more traditional SoC paradigm, the fact that chip count, clock speed and number of I/O per chip are growing rapidly in electronic systems is pushing the

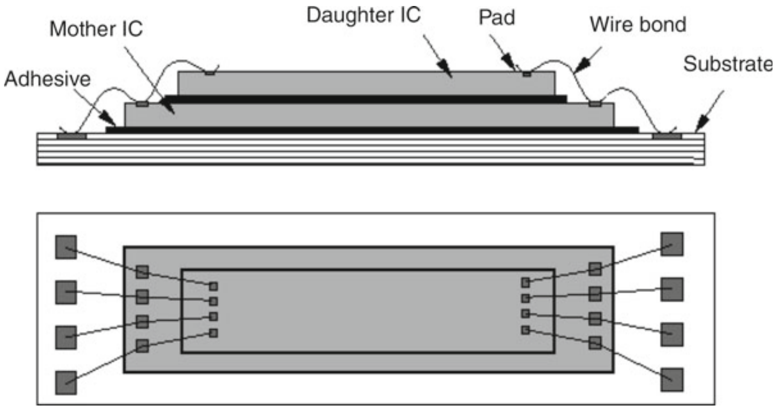


Fig. 1.10 Die to die physical link wires

limits of electrical I/O channels between dice. Using other interconnect technologies (as previously mentioned) within single chips or even a dedicated interconnect layer in a chip stack may alleviate these issues.

Conclusion

In this chapter the system on chip concept is introduced, and current SoC communication systems are described. The main features, as well as the limitations, of the various types of on-chip interconnect are illustrated. Some details are given about both performance issues and physical integration issues, highlighting why today interconnect, rather than logic gates, is seen as the system bottleneck.

The system in package approach is then introduced, seen as a possibility to relax the issues affecting SoC technology and allow the implementation of complex, heterogeneous and high performance systems.

However the increasing complexity and requirements in terms of computation capability of new generation systems will reach the limit of electrical interconnect quite soon, requesting novel solutions and different approaches for reliable and effective on-chip and die-to-die communication.

References

1. STMicroelectronics. UM0484 User manual: STBus communication system concepts and definitions. http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/USER_MANUAL/CD00176920.pdf. Last accessed on October 8, 2012

2. ARM Ltd. AMBA open specifications. http://www.arm.com/products/system-ip/amba/amba.open-specifications.php. Last accessed on October 8, 2012
3. IBM Microelectronics. CoreConnect Bus Architecture. https://www-01.ibm.com/chips/techlib/techlib.nsf/productfamilies/CoreConnect_Bus_Architecture. Last accessed on October 8, 2012
4. Coppola M, Locatelli R, Maruccia G, Pieralisi L, Scandurra A (2004) Spidergon: a novel on-chip communication network. In: SOC working conference, Tampere
5. Goossens K, Dielissen J, Radulescu A (2005) AEthereal network on chip: concepts, architectures, and implementations. In: Design & test of computers. IEEE, New York, NY, USA
6. Dall'Osso M, Biccari G, Giovannini L, Bertozzi D, Benini L (2003) Xpipes: a latency insensitive parameterized network-on-chip architecture for multiprocessor SoCs. In: 21st international conference on computer design, San Jose, CA, USA
7. Dally WJ, Towles B (2003) Principles and practices of interconnection networks. Morgan Kaufmann, San Francisco
8. ITRS web site, http://www.itrs.net. Last accessed on October 8, 2012
9. Madisetti VK. The System-on-Package (SOP) Thrust, NSF ERC on Packaging, Georgia Tech. http://users.ece.gatech.edu/~vkm/sop.html. Last accessed on October 8, 2012
10. Tummala R. High Density Packaging in 2010 and beyond. IEEE 4th International Symposium on Electronic Materials and Packaging, Taipei, Taiwan, December 4th–6th 2002