

Relieving physical issues in new NoC-based SoCs

D. Mangano, A. Scandurra and C. Pistrutto
IP&D - On Chip Communication Systems (OCCS)

STMicroelectronics

daniele.mangano@st.com

ABSTRACT

Many research activities in the area of Network on Chip (NoC) architectures have been performed. The results achieved up to now are quite attractive but often are not directly applicable because of technological reasons or implementation difficulty. In this paper an industrial experience is presented by introducing the approach followed to support the transition from the traditional interconnects to the NoC architectures. The paper mainly focuses on the strategy used to overcome physical issues and particularly the difficulty to perform system synchronization.

Categories and Subject Descriptors

B.4.3 [Input/Output and Data-Communications]: Interconnections; B.7.1 [Integrated Circuits]: Types and Design Styles; C.5.4 [Computer System Implementation]: VLSI Systems.

General Terms

Design.

Keywords

NoC, STNoC, VSTNoC, GALS, mesochronous, asynchronous, synchronization.

1. INTRODUCTION

Traditional interconnects suffer of problems such as scalability, flexibility, performance, wire congestion, difficulty to distribute clocks signals and balance wire delays. Thus, a new interconnect paradigm, leveraging on computer network technology, called Network on Chip (NoC), has been proposed to overcome many of these limitations [1, 2]. Such a paradigm is actually considered as a set of solutions to the different issues rising with the technology scaling [3]. However, being SoCs design based on the reuse, thinking that NoC paradigm can be suddenly introduced is an utopia. Indeed, when a new SoC is designed, its architecture is generally achieved by enhancing and refining older systems. This approach implies also that the interconnection system is reused, and the development of new products using a fully new interconnect would imply an intolerable effort. In addition, complete NoC solutions embed many features that aim at

overcoming limitations that currently are not introducing heavy limitations. Then, an extra price in terms of complexity, area and power should be paid without significant benefits.

STMicroelectronics developed its own internal network on chip solution, called STNoC™ [4, 5], an on-chip packet-switched micro network, flexible and scalable in terms of both performance and number of supported external IPs. STNoC™ consists of three different types of building blocks, appropriately interconnected to each other, and it is based on a patented network topology that promises to deliver the best price/performance trade-off for future MPSoC applications. The On Chip Communication System (OCCS) group at STMicroelectronics delivers to its customers a Network on Chip solution called Versatile-STNoC (VSTNoC), which aims at supporting the evolution from traditional (e.g. STBus and AMBA) to NoC-based interconnects. According to this approach, depending on the system requirements, just a subset of the STNoC features are tailored in VSTNoC in order to overcome specific issues and meet a suitable cost/performance trade-off.

Currently the most limiting issues rise at physical layer and mainly concern wire congestion and clock distribution. The complexity reached by the SoCs makes physical issues heavier and solutions able to mitigate them are more and more urgent. To have an idea of the reached complexity, in fig. 1 an example of SoC by STMicroelectronics is shown. This is the STi7200 [6], a heterogeneous system for high-definition set-top box/DVD decoder chip, providing very high performance for low-cost HD systems. The interconnection system of STi7200 is based on the STBus [7] and represents a key part which drastically affects the overall system performance and cost. STi7200 includes more than one hundred of clock signals and about twenty of them reach the STBus subsystem. Moreover, it is worth noting that in such systems the interconnect generally spans the whole chip.

VSTNoC provides a way to overcome both wire congestion and clock distribution problems. In particular, the number of wires is efficiently reduced by using a smart mechanism to compress the overhead due to the control information without paying meaningful bandwidth. The issues related to the clock distribution are overcome by means of the *Globally Asynchronous Locally Synchronous* (GALS) paradigm, which is the chosen solution to make deeply different IPs, usually operating at different clock frequencies, to coexist within the same chip [8, 9, 10]. In particular, VSTNoC uses some advanced techniques to implement mesochronous communication links [11, 12, 13, 14], which enable skew tolerant design. Some solutions to effectively implement delay-insensitive asynchronous links have been also deployed [12, 15, 16].

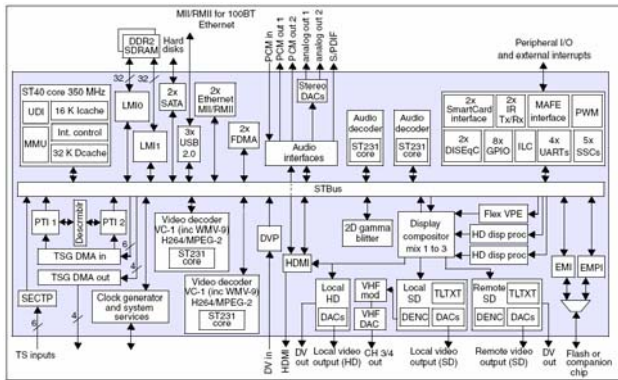


Figure 1. Example of SoC by STMicroelectronics.

This paper introduces some possible solutions to arrange the VSTNoC physical layer by using synchronizer and mesochronous or asynchronous links. The proposed physical layer aims at minimizing the back-end issues due to the clock distribution and wire-delay.

The rest of this paper is organized as follows. Section 2 introduces the rationale and some prior solutions to mitigate physical issues in deep-submicron technologies. Section 3 presents an overview of the possible choices for the implementation of the physical links in VSTNoC. Finally, in Section 4 an overall VSTNoC physical arrangement is proposed.

2. RATIONALE AND PRIOR SOLUTIONS

In the last years, many different GALS-based interconnection systems have been proposed. However, some different approaches can be used to implement the GALS paradigm inside a NoC system. A first solution is to build a clockless network that provides the transport service needed to interconnect the IP components running asynchronously, that is, with different clock signals locally generated. MANGO is an example of clockless Network on Chip [16]. Such an approach is quite attractive, but from an industrial point of view it is still not applicable due to some difficulties concerning the design flow: non-standard cells are used, design of self-timed circuits is not trivial and timing verification is not reliable. However, in some years clockless NoCs could become a valuable alternative. Another common approach is to employ synchronizers at clock domains boundaries, which are between the NoC running at its own clock speed and the IPs components, as in [17]. This solution, breaking the clock tree, introduces some benefits, but does not mitigate the physical issues due to the wire-delay effects inside the network.

A quite attractive solution to overcome the physical issues also inside the network is to employ mesochronous links, which relies on the assumption that clock phase is unknown but constant. This implies that interconnected units have their respective clocks derived from the same source and that an arbitrary skew may exist between them. Effective mesochronous solutions show less limitations w.r.t. the asynchronous approach, such as high latency, high area and wire overhead and synchronization failures. Nevertheless, asynchronous approach is useful to build delay-insensitive links, which enable to overcome the wire-delay

problem, particularly heavy when a long distance inside the chip has to be covered.

As it has been introduced in [12, 14], known mesochronous solutions suffer of some issues, which are mainly: difficulty in design and verification using standard design flow, hard trade-off between robustness and performance/complexity (risk of synchronization failure or poor performance) and, finally, lack of capability to manage flow control and full-duplex communication.

For what concerns asynchronous delay-insensitive links, it is known that in order to achieve asynchronous communication, expensive coding and decoding circuits are needed. Furthermore, an overhead in terms of wires has to be paid. Designing asynchronous communication systems, the main difficulty is just to meet effective solutions for minimizing the wire and logic overheads with satisfactory performance (i.e., latency, bandwidth, etc.).

A lot of work has been spent to develop suitable solutions to implement effective mesochronous and asynchronous links for VSTNoC. In next section a survey of all these solutions is presented. In section 4 the VSTNoC physical arrangement is presented analyzing the possible operation scenarios.

3. VSTNoC ADVANCED LINKS

Hereafter the techniques and the components developed in the last years to implement mesochronous and asynchronous links in VSTNoC are introduced. In particular, an efficient mesochronous communication technique, an effective full-duplex mesochronous link and two asynchronous delay-insensitive (DI) communication techniques are presented. Details are not provided here, but suitable references have been included. The purpose of this paper is just to provide a survey of the developed solutions and of how they are actually applied in VSTNoC.

3.1 Skew Insensitive Links (SKIL)

SKew Insensitive Link (SKIL) [11, 12] is an effective mesochronous communication technique that enables the interconnections of synchronous modules running with arbitrarily skewed clock signals. It can be implemented through standard cells using a standard design flow; just two requirements are needed: clock signals must be derived from the same source and interconnection delay has to be less than one clock period. SKIL enables to build unidirectional point-to-point links with no clock skew constraints providing maximum throughput with latency up to two clock cycles. Its operation is based on a particular mechanism which enables the communication between a transmitter (TX) and a receiver (RX) for any phase relationship between the two clock signals. Such a mechanism guarantees that no timing violations occur by assuring that RX reads data when they are stable. In particular, SKIL operation relies on a particular two-stage buffer structure that is written by the transmitter and read by the receiver. Fig. 2 shows a top-level view of the proposed scheme. It is mainly composed of two units, SKIL TX and SKIL RX. The former provides the strobe signal needed at the RX side for writing data in the buffer, while the latter includes the needed buffering capability, manages the mechanism to recover the synchronization at system start-up (through the strobe signal) and correctly reads data from the buffer.

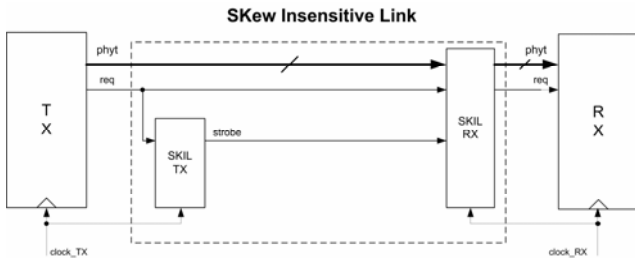


Figure 2. SKIL top-level architecture.

SKIL operation can be divided into two phases, start-up and steady-state. During the first phase, correct synchronization is recovered by means of a synchronizer circuit and a proper initialization phase. Thus, during steady-state operation, no further synchronization is needed and there is the guarantee that no synchronization failures occur. The start-up phase duration deterministically depends on the number of latches in the synchronizer. At the basis of SKIL steady-state operation there is a policy for writing and reading data into and from the buffer in a “ping pong” fashion.

3.2 Full-duplex Mesochronous Link

The proposed full-duplex mesochronous link exploits the physical service provided by the SKIL mesochronous link to achieve maximum throughput, very low latency and fully robustness against clock skew offering the capability to manage the flow control in a full-duplex communication [13, 14].

The starting idea to achieve full-duplex communication is to combine two different SKIL links. With such a solution the problem for managing the flow control is that at the target side there is no way to know when a request is granted (by means of an acknowledgment signal) at the opposite link end. For the same reason, at initiator interface there is no way to know when a response is granted at the target side. Thus, in the proposed solution initiator flow-control information are monitored at target and, vice versa, target flow-control information are monitored at initiator. Moreover, in order to avoid throughput degradation, buffering capability is embedded at both target and initiator sides and flow-control is properly managed by control logic.

Summarizing, the key points in the proposed solution are:

- Flow-control monitoring mechanism to make possible suitable operation.
- Buffering capability at both initiator and target sides to meet maximum throughput.
- Effective flow-control managing policy to meet low latency, minimum buffer sizes achieving at the same time maximum throughput.

Fig. 3 shows the top-level view of the proposed full-duplex mesochronous link. The main components are: the modified SKIL TX block (TX), SKIL RX, a 7-stage buffer structure and a Control Unit.

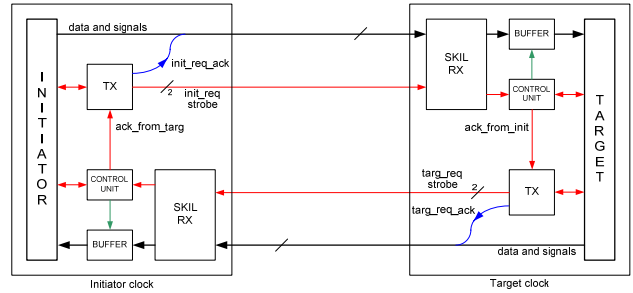


Figure 3. Full-duplex mesochronous link top-level architecture.

3.3 Asynchronous links

A first asynchronous delay-insensitive (DI) communication technique is based on the Berger code [16]. In general, DI codes have been used in many applications for error detection and delay-insensitive communication. Their main feature is the ability of allowing the correct interpretation of the code word independently of the delay of individual bits. Several delay-insensitive coding schemes have been proposed, but, effective CMOS implementations are needed in order to make feasible asynchronous DI on-chip communication [19, 20].

The Berger code is a systematic code which is purposely designed for error detection in data transmission [21]. It is composed of two parts: the information bits (D, data bits) and check bits (CR), i.e. the binary representation for the count of the information bits which are to 0. Therefore, when D increases, wire overhead decreases a greater advantage in using this coding scheme is achieved. Fig. 4 shows the top-level block diagram for the proposed architecture.

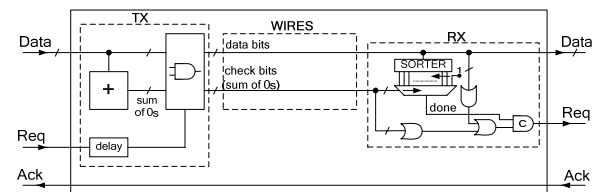


Figure 4. Top-level architecture for the link based on the Berger coding scheme.

The second asynchronous DI communication technique relies on the m-of-n coding scheme [20]. An effective implementation of the m-of-n coding scheme, with $n=8$ and $m=n/2=4$, has been developed [12]. Although this solution is related only to a particular case of m-of-n coding, a technique for building arbitrarily wide links by composing elementary 4-of-8 building blocks has been also proposed. These building blocks enable to implement the 4-of-8 coding scheme. These can be used to encode a 6 bit wide data-path. In particular, there are two basic building blocks: an elementary transmitter (4-of-8 TX) and an elementary receiver (4-of-8 RX). A full-duplex asynchronous link

can be obtained by using two different unidirectional links. Fig. 5 shows as the building are combined in order to build a full duplex n-bit link.

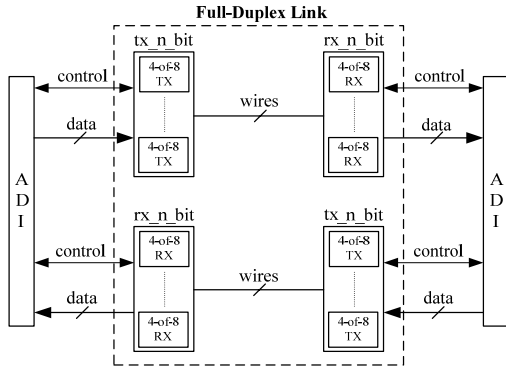


Figure 5. Top-level architecture of the 4-of-8 link.

A brief analysis shows that from a wire overhead point of view, when the data-path is larger than 12 bits, the Berger code appears more advantageous with respect to the modular 4-of-8 coding scheme. However, for a correct comparison of the two encoding techniques, also the required logic for coding, decoding and completion detection must be taken into account. Doing this, it appears that delay and complexity of the Berger solution increases much more than the 4-of-8 one. An evaluation shows that for data-path large up to 30-40 bits, Berger solution appears still more convenient.

For both the presented solutions, an asynchronous pipeline mechanism can be implemented by inserting barriers of c-element. This technique can be used to cover long distance over wires with arbitrary delay.

In [15] an STBus asynchronous decoupler has been presented. This enables to plug STBus interfaces on the above asynchronous links. This kind of decoupler can be employed as it is also for VSTNoC. In particular, the decoupler provides a way to perform synchronization at the boundary between synchronous and asynchronous regions, interfacing for the asynchronous link and managing of the flow-control according to the end-to-end protocol.

4. VSTNoC PHYSICAL LAYER

According to the components and techniques presented in the previous section, the possible applicable solutions are listed in the following.

1. **Full-duplex mesoch. Link:** This can be used inside the same clock domain each time that the constraints on clock skew have to be removed.
2. **Full-duplex mesoch. Link + Synchronizer:** As it is shown in fig. 6, this combination is useful to build links able to cross clock domains removing at same time constraints on clock skew. Such a kind of solution can be employed for example when the boundary between two clock domains has to be crossed guarantying maximum bandwidth (this could

be the case of a link between a Network Interface and a Node).

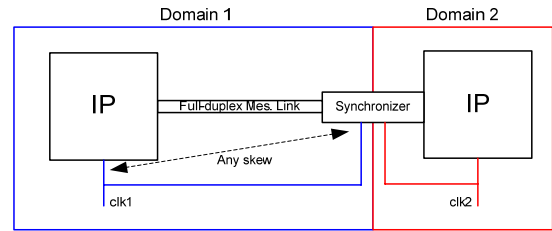


Figure 6. Full-duplex Mesochronous Link + Synchronizer

3. **Berger Asynch. Link + Decoupler:** This solution is suitable to interconnect far IPs belonging to different clock domains, with no heavy bandwidth requirements and with a bus interface smaller than 40 bits and larger than 12 bits.
4. **4-of-8 Asynch Link + Decoupler:** Similarly, this solution is suitable to interconnect far IPs belonging to different clock domains, with no heavy bandwidth requirements and with a bus interface larger than 40 bits or smaller than 12 bits.

Fig. 7 shows an example of VSTNoC physical arrangement built keeping in consideration the above guidelines.

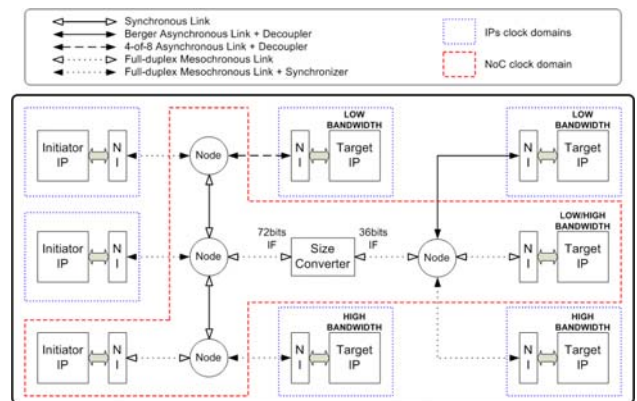


Figure 7. Example of VSTNoC physical arrangement.

Such a system is mainly composed by two clusters. In the first one the interfaces are 72bits wide and there are three initiators and two targets. Two of the three initiators and the two targets run with clocks asynchronous with respect to the interconnect clock, while the remaining initiator is synchronous. The second cluster is characterized by an interface size of 36bits and includes one synchronous and two targets. The inter-cluster communication is handled by a size converter. It could be convenient to merge the three nodes of the 72bits cluster in only one node. However, in this example three separated nodes have been used in order to better explain the VSTNoC physical arrangement. In the 72bits cluster, to cross the clocks boundaries the *Full-duplex Mesochronous Link + Synchronizer* solution is

used when high bandwidth is required. For low bandwidth, as said before the *4-of-8 Asynchronous Link + Decoupler* solution is best choice. When no clock boundary crossing is required either synchronous link or the *Full-Duplex Mesochronous link* solution can be used: the choice mainly depends on the fact that it is required or not to span long distances. In the 36bits cluster the same rules are applied except that when low-bandwidth is required, clock boundary crossing is performed by means of the *Berger Asynchronous Link + Decoupler* solution.

5. ACKNOWLEDGMENTS

Writing this topic has been possible thanks to the previous contribution of the overall ST OCCS team (On Chip Communication Systems) and AST Grenoble Lab, working at novel interconnect solutions for systems on chip (NoC). Thanks also to the Universities of Pisa and Messina (Italy).

6. REFERENCES

- [1] L. Benini and G. De Micheli, Networks on chips: A new SoC paradigm, *Computer*, 35(1): 70-78, Jan. 2002.
- [2] P. Guerrier, and A. Greiner, A generic architecture for on-chip packet-switched interconnections. In *Proceedings of the Design Automation and Test in Europe (DATE)*. IEEE, 250–256, 2000.
- [3] T. Bjerregaard and S. Mahadevan, A survey of Research and Practices of Network-on-Chip, *ACM Computing Surveys*, 38(1), 2006.
- [4] M. Coppola, C. Pistrutto, R. Locatelli and A. Scandurra STNoC™: An Evolution Towards MPSoC Era”, *NoC Workshop, Design Automation and Test in Europe (DATE)*, March 2006.
- [5] M. Coppola, R. Locatelli, G. Maruccia, L. Perialisi, A. Scandurra, Spidergon: a novel on-chip communication network, *SoC 2004*, Nov. 2004.
- [6] www.st.com/stonline/products/literature/bd/12876/sti7200.htm
- [7] Overview on the STBus: <http://www.stmcsu.st.com>
- [8] J. Muttersbach, T. Villiger, K. Kaeslin, N. Felber and W. Fichtner, Globally-asynchronous locally-synchronous architectures to simplify the design of on-chip systems, *Proc. 12th Int. ASIC/SOC Conf. (IEEE, 1999)*, pp. 317–321.
- [9] J. Sparsø, Future networks-on-chip: will they be Synchronous or Asynchronous, *SSoCC'04*, 13-14 Apr. 2004.
- [10] S. Moore, G. Taylor, R. Mullins, P. Robinson, Point to Point GALS Interconnect, *Proc. Of ASYNC'02*, Apr. 2002.
- [11] R. Locatelli, A.M. Coppola, D. Mangano, L. Fanucci, F. Vitullo, D. Zandri, N.E. L'Insalata., SKew Insensitive Link (SKIL): A Physical Link Architecture for STNoC on chip communication network, *Pending European Patent*, Number 06291440.3, 13 September 2006.
- [12] D. Mangano, R. Locatelli, A. Scandurra, C. Pistrutto, M. Coppola, L. Fanucci, F. Vitullo, D. Zandri, Skew Insensitive Physical Links for Network on Chip, *1st International Conference on Nano-Networks (NANO-NET 2006)*, Lausanne, Switzerland, 14-16 September 2006.
- [13] D. Mangano, A. Scandurra, G. Guarnaccia, Effective Full-Duplex Mesochronous Link Architecture with flow-control capability, *Pending European Patent*, 24 May 2007.
- [14] D. Mangano, G. Falconeri, C. Pistrutto and A. Scandurra, Effective full-duplex Mesochronous Link Architecture for Network-on-Chip Data-Link layer, *10th Euromicro Conference on Digital System Design, Architectures, Methods and Tools (DSD07)*, Lübeck, Germany, 29-31 August 2007.
- [15] A. Scandurra, S. Pisasale, D. Mangano, STBus Asynchronous Decoupler: an answer to the IP integration issues in future technologies, *IP-SOC 2004*, Dec. 2004.
- [16] G. Campobello, M. Castano, C. Ciofi, and D. Mangano, GALS networks on chip: a new solution for asynchronous delay-insensitive links, in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE '06)*, pp. 160–165, Munich, Germany, March 2006.
- [17] T. Bjerregaard and J. Sparsø, Implementation of guaranteed services in the MANGO clockless network-on-chip, *IEE Proc.-Comput. Digit. Tech.*, Vol. 153, No. 4, July 2006.
- [18] A. Radulescu, J. Dielissen, K. Goossens, E. Rijpkema, and P. Wielage, An efficient on-chip network interface offering guaranteed services shared-memory abstraction, and flexible network configuration. *Proc. 2004 Design, Automation and Test in Europe Conf. (DATE 2004)*, (IEEE, 2004), pp. 4–17.
- [19] A. Davis and S.M. Nowick, An Introduction to Asynchronous Circuit Design, *Technical Report UUCS-97-013*, Computer Science Department, University of Utah, Sep. 1997
- [20] W.J. Bainbridge, W.B. Toms, D.A. Edwards, S.B. Furber, Delay-Insensitive, Point-to-Point Interconnect using m-of-n Codes, *Proc. IEEE Ninth International Symposium on Asynchronous Circuits and Systems (ASYNC'03)*.
- [21] J.M. Berger, A note on error detection codes for asymmetric binary channels, *Inform. Contr.*, vol. 4, pp. 68-73, Mar. 1961.