Systematic Comparison between the Asynchronous and the Multi-Synchronous Implementations of a Network on Chip Architecture

A. Sheibanyrad [1], I. Miro Panades [2] and A. Greiner [1]

[1] The University of Pierre et Marie Curie, Paris, France
[2] STMicroelectronics, Grenoble, France

Abstract

In this paper we present a systematic comparison between two different implementations of a distributed Network on Chip: fully asynchronous and multi-synchronous. The NoC architecture has been designed to be used in a Globally Asynchronous Locally Synchronous clusterized Multi Processors System on Chip. The 5 relevant parameters are Silicon Area, Network Saturation Threshold, Communication Throughput, Packet Latency and Power Consumption. Both architectures have been physically implemented and simulated by SystemC/VHDL co-simulation. The electrical parameters have also been evaluated by post layout SPICE simulation for a 90nm CMOS fabrication process, taking into account the long wire effects.

1. Introduction

NoCs (Networks on Chip) are a new design paradigm [1] for scalable, high throughput communication infrastructure, in Multi-Processor System on Chip (MP-SoC) with billions of transistors. The idea of NoC is dividing a chip into several independent subsystems (clusters) connected together by a global communication architecture which spreads on the entire chip.

Because of physical issues in nanometer fabrication processes, it is not anymore possible to distribute a synchronous clock signal on the entire wide chip area. The NoC using Globally Asynchronous Locally Synchronous (GALS) [2] techniques address this difficulty.

Cost-Performance tradeoff [3] is a major issue in NoC design, and determines whether NoCs are blessing or nightmare [4]. We believe that the answer to this question could be found by analyzing five key features: Silicon Area, Network Saturation Threshold, Communication Throughput, Packet Latency and Power Consumption. The main goal of this paper is presenting a systematic comparison between these performance parameters for two different implementations of a NoC respecting the GALS paradigm.

The first implementation (DSPIN: Distributed Scalable Predictable Interconnect Network) has a multi-synchronous architecture. The second implementation (ASPIN: Asynchronous Scalable Predictable Interconnect Network) is fully asynchronous. As the general NoC architecture and the provided services are totally identical, the performance comparison between DSPIN and ASPIN may help to answer this question: Will future Networks on Chips, be synchronous or asynchronous? [5]

The SPIN Micro Network [6, 7] was the first published attempt to solve the bandwidth bottleneck, when interconnecting a large number of IP cores in Multi Processors SoCs. After this, a large number of NoC architectures have been published such as Dally's NoC [8], AETHEREAL [9], XPIPES [10] and NOSTRUM [11] which have synchronous architecture. The proposed asynchronous NoCs are CHAIN [12], MANGO [13], QNOC [14], ANOC [15] and QoS [16].

The DSPIN architecture is exhaustively presented in [17], but section 2 contains a brief description of the DSPIN and ASPIN general principles. Section 3 presents silicon area comparison. Section 4 presents the bandwidth analysis. Section 5 presents the latency comparisons. Section 6 analyzes the power consumption. Section 7 contains the system level simulations for both implementations.

2. DSPIN/ASPIN Architecture

In MP-SoC design, a fundamental challenge is the capability of operating under totally independent timing assumptions for each subsystem. Such a multi synchronous system contains several synchronous subsystems clocked with completely independent clocks. They are connected together with a global interconnect (Micro-Network).

Each subsystem (or cluster) may contain one or several processors, one or several physical memory banks, optional dedicated IP cores (Hardware Coprocessors, I/O Controllers …) and a local interconnect. Even if the architecture is physically clusterized, all processors in all clusters share the same flat address space and any processor in the system can address any target or peripheral.
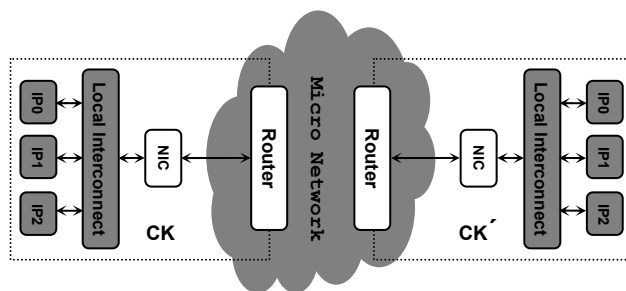

Fig. 1. Cluster Architecture

The switching module of the network is named Router. As it is demonstrated in Fig. 1, in a generic subsystem, the network is connected to the subsystem by a Network Interface Controller (NIC) which is the only access to the network. The NIC translates the local interconnect protocol to the network protocol. It provides services at the transport layer on ISO-OSI reference

model, offering to the subsystem independency versus the network implementation. The IPs are connected to the Network Interface Controller through the local interconnect.

2.1. Topology

For Both DSPIN and ASPIN, the network topology is a two dimensional mesh, with routers physically distributed in each cluster. As there are two independent networks for requests and responses (in order to avoid deadlocks), there are two routers per cluster. In each cluster, the routers are connected to the north, south, east and west neighbors by means of point-to-point, asynchronous links. The size and shape of the clusters have no constraints, but the mesh topology has to be respected.

2.2. Synchronization

The possibility of synchronization failure (Metastability) between two different clock domains is the main issue of GALS architectures. In DSPIN, this difficulty is solved by bi-synchronous FIFOs and in ASPIN by Synchronous ⇔ Asynchronous Converters.
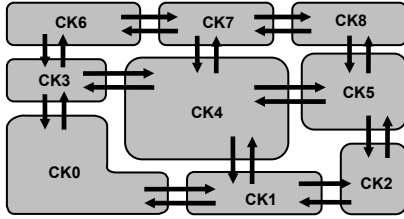

Fig. 2. DSPIN

In DSPIN, the physical links between routers are implemented as bi-synchronous FIFOs [18] (black arrows in Fig. 2) which carry out the inter-cluster communication. To maximize the throughput of the network, and make the network latency predictable, all DSPIN routers are clocked by a mesochronous clock distribution, where all routers have the same frequency but different phases. DSPIN uses therefore two types of bi-synchronous FIFOs: The FIFOs between two neighbor routers solve the skew between clocks that have the same frequency, whilst the FIFOs between a router and a synchronous local subsystem interface clock domains where both frequencies and phases can be different.
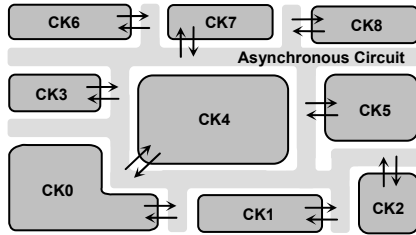

Fig. 3. ASPIN

In ASPIN, the global interconnect (network) has a fully asynchronous architecture. This type of NoC respects the GALS paradigm by providing Synchronous ⇔ Asynchronous interfaces (black arrows in Fig. 3) at each interface between the network and a synchronous subsystem. The two efficient

Synchronous ⇔ Asynchronous converters used in ASPIN have been presented in [19].

2.3. Packet Routing

DSPIN and ASPIN are both packet switching networks. Packets are divided into flits. A flit contains a 32 bits data word, and is the smallest flow control unit handled by the routers. The first flit of a packet is the header of packet including the destination cluster address. This cluster address is defined in absolute coordinates X and Y.

When a router receives the header of a packet, the destination field is analyzed and the flit is forwarded to the corresponding output port. Round-Robin is used in order to avoid starvation, when there are simultaneous requests for the same outgoing port. As DSPIN and ASPIN use wormhole routing, the rest of the packet is also forwarded to the same port until the end of packet marker.

DSPIN and ASPIN use the deadlock free X-First algorithm to route the packets over the network. With this algorithm, the packets are first routed on the X direction and then on the Y direction. The X-First algorithm is deterministic, and guarantees the in-order delivery property of the network.

2.4. Long Wire Issue

In deep submicron processes, the largest part of the delays is related to the wires. As place and route tools have difficulties to cope with long wires, in multi-million gates SoCs, the timing closure can become a nightmare [20]. Both DSPIN and ASPIN architectures attempt to solve this problem by partitioning the SoC into isolated clusters, (or subsystems). This allows performing physical synthesis and timing closure analysis for each cluster independently, without any time constraints between different clusters.
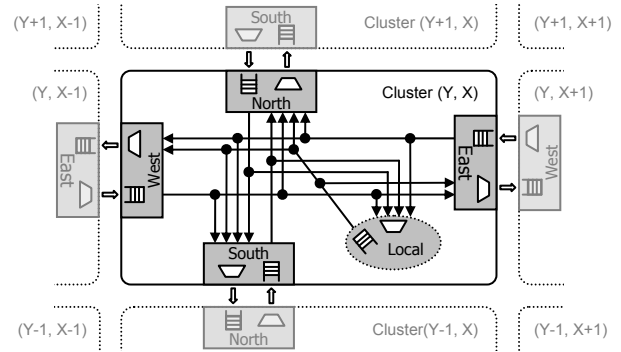

Fig. 4. Router Architecture

As shown in Fig. 4, the DSPIN and ASPIN routers are not designed as a centralized macro-cell. They are split in 5 separated modules (North, South, East, West and Local) that are physically distributed on the clusters borders. This feature allows us to classify the network wires in two classes:

- Inter-Cluster Wires: connecting modules of two adjacent clusters (white arrows). For example, the connections between East module of cluster (Y, X) and West module of cluster (Y, X+1). As those components can be made very close from each other, inter-cluster wires are short wires.

- Intra-Cluster Wires: connecting modules of the same cluster (black arrows). Those wires are the longest wires. But the wire length is bounded by the physical area of a given synchronous domain.

Since intra-cluster wires can have various lengths, depending on the routing, the differences between their delays are not predictable. Respecting delay insensitivity, in the asynchronous ASPIN implementation, the long wires are double railed and the communication uses a Four-Phase protocol.

3. Silicon Area

The actual silicon area after physical synthesis is the first important parameter. Both the DSPIN and ASPIN routers have been physically implemented.

Synthesizable VHDL models have been designed for all DSPIN components. As illustrated in Table 1, the 32-bit DSPIN router has been synthesized using *Synopsys* and the ST-Microelectronics GPLVT standard cell library. It takes 40200 μm^2 for this 90 nm process.

Regarding ASPIN, we developed a generic ASPIN generator, using *Stratus* hardware description language of the *Coriolis* platform [21]. This tool generates both a gate-level net-list and the physical layout. The total silicon area of the 32-bit ASPIN router, (using the ALLIANCE portable standard cell library [22]) is 36199 μm^2, for the same fabrication process.

Table 1. Silicon Area

	DSPIN	ASPIN
Router	40200 μm^2	36199 μm^2
Long Wire Buffers	4276 μm^2	7815 μm^2
Total	**44476 μm^2**	**44014 μm^2**

The ASPIN router area is about 10% smaller than the DSPIN area, but another factor must be accounted: the area of the long wire buffers. As discussed earlier, the Intra-Cluster wires in DSPIN and ASPIN architectures are the long wires. In some case, these long wires need to be bufferized. As ASPIN uses double railed wires, the area of the long wire buffers is about two times larger for ASPIN than for DSPIN.

4. Communication Throughput

The communication throughput is the maximum number of flits transmitted by second (a flit contains a 32 bits data word). This parameter depends on the routers micro-architecture, and on the long wire effects. As the router is physically distributed, the length of the intra-cluster long wires is a key factor, and we need a model for these intra-cluster wires. The length of these wires depends on the cluster size. In 90 nm fabrication process, 2×2 mm^2 is a rough surface estimation for a large cluster. The Fig. 5 shows a simple RC model for an intra-cluster long wire connecting one input module to four output modules.

To evaluate Communication Throughput, Packet Latency, as well as the Power Consumption, we used this Long Wire model, and extracted the SPICE model of all DSPIN and ASPIN components. The target fabrication process is the ST-Microelectronics 90 nm GPLVT transistors. *Eldo* simulations have been performed for typical conditions.

The first row of Table 2 presents the Maximum Throughput for the DSPIN and ASPIN routers. In case of DSPIN (synchronous approach), this indicates the maximum clock frequency that can be used to clock the router. In case of ASPIN (asynchronous approach), The Maximum Throughput is equal to the inverse of the time needed to pass a flit through the slowest storage stage (pipeline stage) of the router.
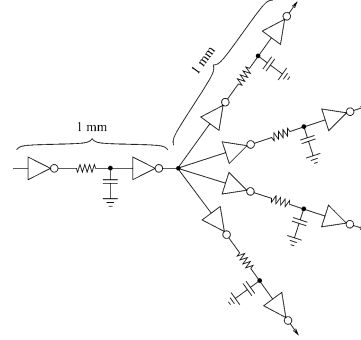

Fig. 5. Long Wire RC Model

The first row in table 2 doesn't take into account the long wire effects. The second row presents the effect of the long wires delays, using the 2 mm wires model. These long wires delays are about four times larger in ASPIN, due to the delay insensitive Four Phase protocol. The Applicable Throughputs, mentioned in the third row of Table 2, are the final evaluations. As said before, a 4 mm^2 cluster is a large cluster, so these throughputs are a worst case evaluation which can be applied to all clusters regardless of theirs size.

Table 2. Communication Throughput

	DSPIN	ASPIN
Maximum Throughput	787 MFlits/S	1131 MFlits/S
Long Wire Effect	135 ps	515 ps
Applicable Throughput	711 MFlits/S	714 MFlits/S

As a summary, the number of flits passing per second through an ASPIN cluster may be between 700 and 1100 Mega Flits depending on the cluster size. Whilst for DSPIN router the 700 Mega Flits are independent on the cluster size.

5. Packet Latency

The minimal Packet Latency is the end-to-end delay between the time a packet header enters into the first router and the time it exits the last router, assuming no contention in the network.

The path through the network can be decomposed in three parts: First router, Intermediate routers and Last router that have different latencies. Table 3 shows the latencies of ASPIN and DSPIN routers.

Table 3. Packet Latency

	DSPIN	ASPIN
First Router	3~4 T[*]	1.06 ns
Intermediate Router	2.5 T	1.53 ns
Last Router	4.5~5.5 T	1.76 ns + 1~2 T
Long Wire Effect	0 ns	0.39 ns

[*] T is the clock cycle time (2 ns for 500 MHz clock frequency)

As DSPIN is a synchronous circuit, the latency depends on the clock cycle time. The exact value depends on the clock skew relation between the network clock, and the subsystems clocks.

The latency of the First DSPIN router is between 3 and 4 clock cycles. For the Intermediate routers, a mesochronous clock distribution is used and the latency is predictable as 2.5 clock cycles. According to the synchronous circuit principles, the long wires have no effect on DSPIN Packet Latency.

The ASPIN Packet Latencies are given in nanosecond. As explained in [19], for the final router, an Asynchronous to Synchronous converter, located in the Network Interface Controller, has a synchronization latency between one and two clock cycles. Packet Latency in ASPIN directly depends on the cluster size and long wire delays. Four-Phase protocol with 2 mm wires causes an extra latency of about 390 ps per cluster.

Assuming 500 MHz as system clock frequency (clock frequency estimation for fast and large MP-SoC subsystems in 90 nm technology), equations (1) and (2) denote the Packet Latencies for 4 mm^2 clusters where N is the number of routers in the packet transmission path.

DSPIN Packet Latency = (5.00 × (N-2) + 17.0) ns   (1)
ASPIN Packet Latency = (1.92 × (N-2) + 6.60) ns   (2)

The synchronization delay at each clock boundary crossing explains why the DSPIN Latency is much higher than the ASPIN Latency.

In Shared Memory Multi-processor System on Chip (MP-SoC), the packet latency is critical for system performance. According to the above equations, the asynchronous approach in a GALS system can really improve the system performance.

6. Power Consumption

Power consumption of the communication structure in deep submicron fabrication processes is a major concern.

Although most research has focused on average power consumption or total energy consumption [23], we believe that instantaneous power consumption (or energy consumption) during one short period of time is also important for NoC characterization. In calculating the NoC power consumption, two terms must be taken into account: dissipated energy per transmitted flit and idle power consumption.

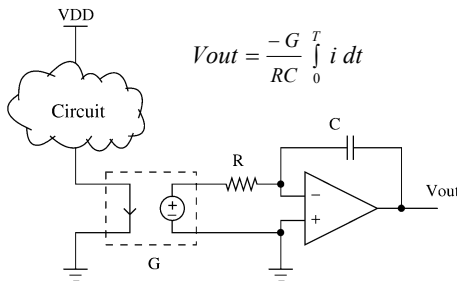

$$Vout = \frac{-G}{RC} \int_0^T i\, dt$$

Fig. 6. Current Integrator

To measure electrical energy consumed by the circuit in a defined period of time, we used a Current Integrator model in electrical simulations. The schematic of the proposed Integrator is shown in Fig. 6. The output voltage (Vout) is equal to the definite integral of the instantaneous current (i) traversing the circuit, from the beginning of the simulation.

As a first step, we have measured, for each router, the idle power consumption. An idle router means there is no packet to route. Table 4 present the results. The DSPIN power consumption is 2060 μWatt at 500 MHz, using clock gating [24]. With 640 μWatt the ASPIN power consumption is about three times lower. It is well known that the clock power dissipation in synchronous designs is not negligible, even with clock gating.

Table 4. Power Consumption

	DSPIN	ASPIN
Idle Router	2060 μWatt	640 μWatt

In a second step, the energy consumptions of two activated DSPIN and ASPIN routers have been compared. The energy consumptions have been measured for the transmission of a single five flits packet. Separated measurements have been done for the First, Intermediate and Last routers.

We have executed the measurements with four different hypotheses, depending on two parameters. The first parameter is the packet content: All flits in the packet can have a constant value, or all bits values change between two successive flits. The second parameter is the long wires capacitance: Depending on the cluster size, the corresponding power consumption is taken into account or not. Table 5 summarizes the energy consumption results for a clock frequency of 500 MHz. In this Table, N is the number of routers.

In the asynchronous Double-Rail Four-Phase protocol, one of the two rails of each bit goes to logic One and return to Zero, whether the bit content is zero or one. Consequently, ASPIN energy consumption is nearly independent on the packet content.

In small clusters, where the effect of long wires is insignificant, DSPIN and ASPIN consume approximately the same amount of energy to transfer one packet. When the long wire effect is taken into account, the energy required by DSPIN to transfer packet with constant content remains almost at the previous value, but if the packet has an alternate content, energy consumption increases. As expected, the long wire effect on ASPIN energy consumption is much more dissipative.

In a typical shared memory multi processor system using a Best Effort Micro-Network, the average activity of the routers is rather low: Most of the time, the routers are idle. According to a factor 3 between ASPIN and DSPIN for idle power consumption, the ASPIN router consume less power than DSPIN, even if the energy required for packet transmission is larger in ASPIN than in DSPIN.

Table 5. Energy Consumption during one Packet Transmission (pJ)

	Without Long Wire Effect				With Long Wire Effect			
	With Constant Content		With Alternate Content		With Constant Content		With Alternate Content	
	DSPIN	ASPIN	DSPIN	ASPIN	DSPIN	ASPIN	DSPIN	ASPIN
First Router	37	27	43	34	50	124	83	131
Intermediate Router	36	33	42	41	45	129	81	137
Last Router	36	48	42	62	45	147	81	161
Transmission Path	36×(N-2)+73	33×(N-2)+75	42×(N-2)+85	41×(N-2)+96	45×(N-2)+95	129×(N-2)+271	81×(N-2)+164	137×(N-2)+292

7. Saturation Threshold

The saturation threshold is the last important parameter for NoC characterization. The main motivation supporting the NoC paradigm is the fact that classical interconnects such as shared busses do not scale when the number of components to interconnect increases. When too many processors generate traffic, any interconnect will saturate, when the load offered by each processor reaches a point called saturation threshold. In NoC, this threshold is in principle roughly independent on the number of communicating components.

The offered load is defined, for each subsystem generating traffic, as the percentage of the maximal bandwidth:

$$\text{Offered Load} = \frac{L}{L+G}$$

Where L is the average packet length (number of flits with one flit transmitted per cycle), and G is the average number of cycles between two packets.

Before saturation, the average packet latency remains approximately constant. At the saturation threshold, it raises exponentially to an infinite value. The saturation threshold of a network depends on four elements: number of clusters, average packet length, destination packet distribution and the total storage distributed in the network.
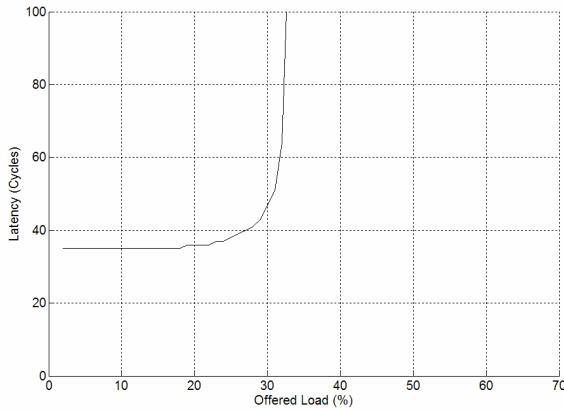

Fig. 7. DSPIN Saturation Threshold

To evaluate the saturation threshold of DSPIN and ASPIN, we have focused on a mesh topology containing 5×5 clusters. Each cluster contains one Traffic Generator/Analyzer (TGA) that plays the role of a processor and one Traffic Reverser (TR) that plays the role of a target. The traffic has a uniform random distribution: each TGA sends randomly the packets to all TR (except the TR situated in the same cluster). Each TR returns the received packets to the sender TGA. The length of packets is 9 flits and flit storage capacitance of routers is 8 per FIFO, for both DSPIN and ASPIN. To prevent deadlock, two separated networks are used for requests (from TGA to TR) and for responses (from TR to TGA). In order to take into account the network contention and have a meaningful latency measurement, the packets have a time stamp and are posted in an infinite FIFO instantiated in each TGA. The average packet latency is measured as the average number of subsystem clock cycles for a Round Trip.

For DSPIN, all components (TGA, TR and DSPIN Router) have been modeled in SystemC language as cycle accurate

simulation models. The Fig. 7 depicts the DSPIN average packet latency (in cycles) versus the offered load (in percent), obtained by cycle accurate simulations. The saturation threshold value is about 32%.

For ASPIN, the ASPIN generator provides a structural VHDL net-list of standard cells. The ALLIANCE standard cell library has been completed to include the specific asynchronous cells used in the ASPIN router. The cell behavioral models are written as *transport delay models*. As an example the VHDL behavioral model of the asynchronous standard cell MUTEX is given as below:

```
ENTITY mutex IS
PORT(
  r0      : IN STD_LOGIC;
  r1      : IN STD_LOGIC;
  g0      : OUT STD_LOGIC;
  g1      : OUT STD_LOGIC;
);
END mutex;

ARCHITECTURE RTL OF mutex IS
  SIGNAL x0          : STD_LOGIC;
  SIGNAL x1          : STD_LOGIC;
BEGIN
  g0 <= x0;
  g1 <= x1;
  x0 <= transport (not x1 and r0) or (not r1 and r0) after 10 ps;
  x1 <= transport (    x1 and r1) or (not r0 and r1) after 10 ps;
END RTL;
```

ModelSim has been utilized to perform a co-simulation including the ASPIN VHDL model and the cycle accurate TGA and TR SystemC models. The saturation threshold in ASPIN depends on the ratio between the synchronous subsystem clock frequency, and the asynchronous network throughput. In Fig. 8, the average packet latencies are plotted versus the network offered load for six different ratios of 0.5, 1, 1.5, 2, 3 and 10. A ratio of 0.5 means that the subsystems are two times faster than the network. A ratio of 2 means that the network is two times faster than the subsystems. The ratios larger than 10 practically produce the same curve as a ratio of 10.

In the previous section we said that the ASPIN throughput varies between about 700 and 1100 MFlits/S depending to the clusters size, although an estimation of maximum clock frequency for fast MP-SoC subsystems in 90 nm technology could be about 500 MHz. So, the actual minimum ratio of ASPIN throughput to system clock frequency is about 1.5.
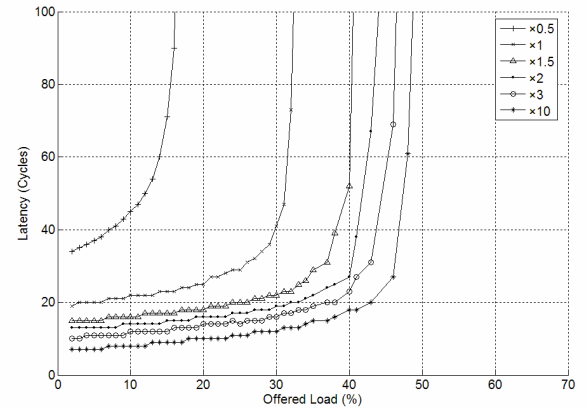

Fig. 8. ASPIN Saturation Thresholds

According to the curves of Fig. 8, the ASPIN saturation threshold is comprised between 40% and 48%, depending on the clusters size and subsystems clock frequencies. This indicates

that asynchronous approach is better for removing the global interconnect bandwidth bottleneck.

8. Conclusion

A systematic comparison between performance parameters of two different implementations of the same micro-network architecture has been presented. This NoC architecture has been designed to be used in GALS, shared memory MP-SoC. The DSPIN implementation is multi-synchronous, and the ASPIN implementation is fully asynchronous. Both architectures have been physically implemented. System level performances have been evaluated by cycle precise simulations on a 5×5 network. Physical characteristics have been evaluated by post layout SPICE simulation for ST-Microelectronic 90 nm GPLVT CMOS fabrication process. In the evaluations the long wires (intra-cluster wires) effects have been taken into account in evaluating the bandwidth, the latency and the power consumption.

- Both networks are scalable, but the asynchronous approach shows a better saturation threshold than the synchronous one.
- Regarding the silicon area, both implementations have similar foot-prints, if long wire buffers are taken into account.
- In systems containing large clusters, the energy dissipated to transmit a packet is higher in the asynchronous approach than in the synchronous approach, but the *idle* power consumption is 3 times lower. Consequently, the average power consumption is expected to be smaller in the asynchronous approach for typical shared memory MP-SoCs.
- The maximal bandwidths are similar: 700 MFlits/S for the synchronous approach, against 700 to 1100 MFlits/S (depending on the cluster size) for the asynchronous approach.
- The packet latency is clearly the strong point for the asynchronous approach, as the latency is about 2.5 times smaller for ASPIN than for DSPIN.

As a general conclusion, silicon area, power consumption and bandwidth have approximately similar values, but the asynchronous implementation accepts larger non saturating offered load than the synchronous one and its average packet latency is about 2.5 times smaller, which is very important in shared memory MP-SoC architectures.

Moreover, in large multi-clusters architectures, the risk of metastability introduced by the multiple bi-synchronous FIFOs used in the multi-synchronous approach can become a critical issue. This risk is much lower in the asynchronous approach, as the metastability is entirely confined in the Synchronous ⇔ Asynchronous converters.

Finally, we believe that the results obtained for ASPIN with asynchronous Four-Phase Double-Rail protocol can be improved by using another delay insensitive communication protocol such as m-of-n data encoding [25].

References

[1] L. Benini, G. De Micheli, "*Networks on chip: a new SoC paradigm*", IEEE Computer, vol. 35, no. 1, Jan. 2002

[2] D. M. Chapiro, "*Globally-Asynchronous Locally-Synchronous systems*", PhD thesis, Stanford University, 1984

[3] S. G. Pestana, E. Rijpkema, A. Rădulescu, K. Goossens, O. P. Gangwal, "*Cost-Performance Trade-Offs in Networks on Chip: A Simulation-Based Approach*", DATE 2004

[4] P. Wielage, K. Goossens, "*Networks on Silicon: Blessing or Nightmare?*", DSD 2002

[5] J. Sparsø, "*Future Networks-on-Chip; will they be Synchronous or Asynchronous?*" (Invited talk), SSoCC 2004

[6] P. Guerrier, A. Greiner. "*A generic architecture for on chip packet-switched interconnections*", DATE 2000

[7] A. Adriahantenaina, A. Greiner, "*Micro-network for SoC: Implementation of a 32-port SPIN network*", DATE 2003

[8] W. J. Dally, B. Towles, "*Route packets, not wires: on-chip interconnection networks*", DAC 2001

[9] J. Dielissen, A. Rădulescu, K. Goossens, E. Rijpkema, "*Concepts and Implementation of the Philips Network-on-Chip*", IP-SOC 2003

[10] M. Dall'Osso, G. Biccari, L. Giovannini, D. Bertozzi, L. Benini, "*xpipes: a Latency Insensitive Parameterized Network-on-chip Architecture For Multi-Processor SoCs*", ICCD 2003

[11] M. Millberg, E. Nilsson, R. Thid, A. Jantsch, "*Guaranteed Bandwidth Using Looped Containers in Temporally Disjoint Networks within the Nostrum Network on Chip*", DATE 2004

[12] J. Bainbridge, S. Furber, "*Chain: A Delay-Insensitive Chip Area Interconnect*", IEEE Micro, vol. 22, no. 5, September/October 2002

[13] T. Bjerregaard, J. Sparsø, "*A router architecture for connection-oriented service guarantees in the MANGO clockless Network-on-Chip*", DATE 2005

[14] D. (R.) Rostislav, V. Vishnyakov, E. Friedman, R. Ginosar, "*An Asynchronous Router for Multiple Service Levels Networks on Chip*", ASYNC 2005

[15] E. Beigne, F. Clermidy, P. Vivet, A. Clouard, M. Renaudin, "*An Asynchronous NOC Architecture Providing Low Latency Service and Its Multi-Level Design Framework*", ASYNC 2005

[16] T. Felicijan, S. B. Furber, "*An Asynchronous On-Chip Network Router with Quality-of-Service (QoS) Support*", SOCC 2004

[17] I. Miro Panades, A. Greiner, A. Sheibanyrad, "*A Low Cost Network-on-Chip with Guaranteed Service Well Suited to the GALS Approach*", Nano-Net 2006

[18] I. Miro Panades, "*Buffer memory control device (Dispositif de commande d'une mémoire tampon)*", Patent pending

[19] A. Sheibanyrad, A. Greiner, "*Two Efficient Synchronous⇔Asynchronous Converters Well-Suited for Network on Chip in GALS Architectures*", PATMOS 2006

[20] R. Ho, K. W. Mai, M. A. Horowitz, "*The future of wires*", IEEE, vol. 89, no. 4, April 2001

[21] http://www-asim.lip6.fr/recherche/coriolis/

[22] http://www-asim.lip6.fr/recherche/alliance/

[23] T. T. Ye, G. De Micheli, L. Benini, "*Analysis of power consumption on switch fabrics in network routers*", DAC 2002

[24] L. Benini, P. Siegel, G. De Micheli, "*Saving Power by Synthesizing Gated Clocks for Sequential Circuits*", IEEE Design & Test, vol. 11, no. 4, October 1994

[25] W. J. Bainbridge, W. B. Toms, D. A. Edwards, S. B. Furber, "*Delay-Insensitive, Point-to-Point Interconnect Using M-of-N Codes*", ASYNC 2003