Fault-Tolerant Techniques to Manage Yield and Power Constraints in Network-on-Chip Interconnections

Anelise Kologeski, Caroline Concatto, Fernanda Kastensmidt, Luigi Carro

Fault-Tolerant Techniques to Manage Yield and Power Constraints in Network-on-Chip Interconnections

Anelise Kologeski, Caroline Concatto, Fernanda Lima Kastensmidt, Luigi Carro

PGMICRO - PPGC - Instituto de Informática
Universidade Federal do Rio Grande do SUL (UFRGS)
Av. Bento Gonçalves, 9500 – Prédio 43412, sala 213
CEP: 91501-970 – Porto Alegre – RS - Brazil.
{alkologeski, cconcato, fglima, carro}@inf.ufrgs.br

Abstract. The use of fault-tolerant mechanism is essential to ensure the correct functionality of integrated circuits after manufacturing due to the massive number of faults that may occur during the process. In this work, we propose a set of fault-tolerant techniques to cope with faulty wires in Network-on-Chip (NoC). The most appropriate technique is chosen by taking into account the number of faulty wires and their location in the NoC. The goal is to combine different techniques to reduce overheads in area, delay and power. The use of testing and diagnosis can minimize costs associated with embedded fault-tolerant mechanisms once the architecture adapts itself to work in different faulty scenarios. The proposed fault-tolerant strategy uses a lightweight adaptive routing combined with data splitting, which is able to send the data in one clock cycle. The power penalty has a low correlation with the number of faulty interconnections. Results for MPEG4 and VOPD applications running on the NoC with different faulty case-study scenarios show that the proposed techniques can tolerate many faulty interconnections with a low area, performance and power overheads.

Keywords: adaptive routing; data splitting, fault tolerance, interconnections, multiple faults, NoC.

1 Introduction

The use of embedded fault-tolerant strategies in System-on-Chip (SoC) architectures becomes crucial to improve yield and reliability, due to the huge amount of interconnections subject to the defects that comes from dimensions shrinking and aggressive transistor density. A Network-on-Chip (NoC) offers better scalability and performance than a traditional bus, and therefore it is alternative communication architecture inside of a complex System-on-Chip.

Nevertheless, according to [1], one expects up 15% of the wires faulty in recent technologies, which confirms it is necessary to consider the fault probability at the design time to ensure high yield and reliability in the devices. The use of fault-tolerant

structures grows in NoC designs, due to the fact that it is almost impossible to manufacture integrated circuits without any defect in nanometer technologies [1]. Consequently, the use of fault-tolerant methods is crucial to allow that circuits with some amount of defects still reach the market. Therefore, fault-tolerant mechanisms in NoCs are mandatory to ensure the correct functionality, the yield and the lifetime of a chip.

The problem is that the use of several embedded fault-tolerant techniques to cope with multiple faults in links can significantly increase the overheads in area, power, energy and performance. This is because most of the techniques are applied in the critical path. In addition, they can be pre-placed even on those interconnections with no defects. To offer a flexible strategy, our proposed method combines testing and diagnosis to allows fault-tolerant techniques to be activated only in the faulty interconnections. In this way, we minimize the costs associated with the embedded fault-tolerant techniques.

The strategy presented in this work is named ATARDS -*Adaptive Technique based on Adaptive Routing and Data Splitting*, and the strategy combines a light-weight adaptive routing (LAR) and data splitting (DS) to ensure NoC connectivity in presence of massive defects in the interconnections. The combination between two strategies allows to obtain better results when compared with traditional solution widely known in the literature. The present technique tolerates multiple faults scenarios in the interconnections (or links). Consequently, it sustains yield by keeping the connectivity in the network. If the fault-tolerant resources are configured previously due to testing and diagnosis phases, the performance and power overheads can be minimized. It is possible because the fault-tolerant techniques are enabled to operate only in faulty interconnections.

ATARDS avoids the need of additional wires in the link, and minimizes additional hardware in the critical path. The impact in performance and power is not seen in all parts of the architecture, since only faulty regions use the fault-tolerant mechanism. The experimental results with different faulty case-study NoC scenarios show the advantage of combining testing, diagnosis and ATARDS to reach better trade-offs with a high connectivity, reduced power overhead and large fault coverage.

This paper is organized as follows. Section 2 presents the fault and test models used in this work. Related work is discussed in Section 3, and we also demonstrate that our proposed strategy can fill some significant hole in the literature by cope with faulty interconnections using a merge of techniques. In Section 4, ATARDS strategy is presented with details. Results from synthesis, performance, energy and connectivity are reported in Section 5. Finally, the conclusions and ideas for future work are discussed in Section 6.

2 Fault and Test Models

The fault model provides and abstraction between the particular fault source and its manifestation in different layers of the architecture. We are mainly interested in

high-level fault models in NoC. They can be in the cores (core fault model) or in the interconnections (inter-core fault model) [7]. We address permanent faults due to manufacturing. They can be modeled as shorts and open circuits. In our case, shorts will be adopted to consider a specific pattern to be addressed. A short fault occurs when a wire connects with another one. The shorts may happen among wires at the same metal layer, at the top or bottom metal layers. There are three types of short faults: OR-short, AND-short, and strong driver [19]. In a NoC, the short faults, in the wires can happen among different interconnections, from router to router or from router to core. As the amount of wires grows, the number of faults will increase exponentially [19]. Therefore, strategies that increase the number of wires are more prone to faults.

The proposed fault-tolerant techniques tolerate the inter-core faults. The inter-core fault model has been defined by faults happening among any links of the network, and it has been further classified as interlink and intralink [22]. Intralink faults happen when aggressor and victim wire are into the same link. So, they may happen isolated in links between two routers and/or in links between a core and a router. Each intralink fault is not associated with other links. Interlink faults appear when aggressor and victim are in different links. Thus, each interlink fault occurs between two different interconnections that are intersecting. Multiple defects can be any combination of intralink and interlink faults, and both can be treated by ATARDS implementation, which characterize single and multiple faulty interconnections.

Besides of permanent faults due to problems with manufacturing process, the faults also can be classified as intermittent and transient, in according to the duration. Intermittent faults occur again and again considering a certain period of time, as a periodical influence by noise or crosstalk. The transient faults usually are result from alpha particles, heavy ions and radiation, and they reach quickly the circuit affecting only memory elements changing the information by one or few clock cycles. The architecture proposed by ATARDS only copes with faulty situations located in each interconnection. As a consequence, then just permanent and intermittent faults can be tolerated by the approach if they are previously detected and diagnosed.

The capability of detecting faults in interconnections such as short circuit among channels is mandatory for yield improvement. According to [19], for full-custom layout implementations, faults between wires of distinct links are less likely, but can still be observed. So, it is mandatory to extend the fault model to include interaction faults that affect different interconnections of a NoC, like explained before with interlink faults situation.

Detection and diagnoses can be developed during manufacturing test, and off-line tests also can run during the life time of the circuit. The test, proposed in[19] detects shorts between pairs of wires (including data and control wires within a single channel or between channels) for a mesh NoC with XY routing. One has a cost-effective test, which uses a 2 x 2 NoC to deal with a fault model, which expands for a larger mesh NoC. The proposed testing approach uses Walking-One Sequence as a method to detect faults in the NoC. Furthermore, in [19] can be extended to other interaction faults in the interconnections, such as crosstalk, by adapting the test sequence. The testing approach in [10] is very similar once it uses test vectors to allow testing and

diagnosis of fault interconnections. The test uses the results to configure the registers in each channel with the information about faulty links.

In [20], the authors present a method to detect defects in SoC interconnections using I_{DDT} test (analyzing the variation of the dynamic current), boundary scan and tests of delay. A *built-in self-test* (BIST) methodology for testing the inter router links of a NoChas been proposed in [21] considering the *Maximum Aggressor Fault* (MAF) model.

In this work, we consider that the test and diagnosis is done by test vectors like the ones proposed in [19] and [10]. Analyzing the results from test, it is possible to configure the registers to inform each router of the faulty channel, and to configure the control of multiplexers used for the data splitting strategy, as will be described in detail after the related work.

3 Related Work

Related techniques to mitigate faults in the link usually based on one of the following techniques: Hamming code, parity check, retransmission, redundancy, data splitting, adaptive routing or remapping [2-10, 22]. Some of them do not need detection and diagnosis offline, because they are always detecting and correcting possible faults at run-time. Normally the authors assume a single fault scenario, which means only one faulty wire in the interconnection, in accordance to MAF model [21] or considering only one transient fault. Solutions based on error detection and correction codes (EDAC) imply extra wires for parity/check bits and extra hardware placed in the critical path, for encoding and decoding blocks in each NoC link. EDAC impacts latency and power consumption. Moreover, they can deal only with one fault per link and not multiple intralink faults. The model of a single fault per link is not valid any longer, since multiple manufacturing defects are more common to be observed and in locations close to each other, as clusters of defects in nanometer technologies [11]. Then, one requires the use of a solution to tolerate massive faults, and as a consequence, the trend is to combine different techniques to cope with, achieving high reliability with an efficient solution.

Authors in [2] propose a technique that uses Hamming Code (HC) to protect all NoC links against crosstalk, permanent and transient faults. They consider single-error correction and double-error detection (SEC/DEC) [2]. One decodes the incoming data before being stored in the FIFO, and encoded when it leaves the router. Reported results show an area overhead of 39% and a delay penalty of 32% in frequency for 180nm technology, and there is no protection for multiple faults in the link. In [3], the authors propose to combine different methods to achieve fault tolerance to crosstalk and permanent faults in NoC links. The technique uses data splitting, Hamming Code at each half of the data, and retransmission to correct crosstalk faults in the links. On top of that, triple modular redundancy (TMR) is used to protect the handshake links. The two main disadvantages of this method are the high area and power overhead, which result from the combined use of the HC and TMR, leading to a final area four times larger than the no-protected router. Besides, there is performance deg-

radation in the network, due to the HC encoding and decoding, plus the time redundancy required for the data split technique. The latency in [3] is also increased around four times.

The technique proposed in [4] uses parity check, data splitting and retransmission of data to protect. The technique is similar to [3], but the authors propose the use of parity check to discover a faulty interconnection instead of HC. Extra bits for the parity check have been used in each half of the link to detect faulty wires, reducing the costs in relation to [3]. In the presence of faults, the erroneous half of the data is doubled and retransmitted. Due to the required retransmission in faulty cases, the performance penalty in [4] occurs only in the presence of a fault. The main disadvantage of this method is the use of extra wires and the area overhead compared to a router protected with HC only. The big problem observed in these techniques presented in [2-4] is to deal only with multiple faulty links (each one of single-fault), but not with multiple faults per link.

In [5], for a 64-core NoC with 32 bits of channel-width, the overhead in wires is about 137%, because each link protected by the Hamming code was completely duplicated (overhead of an interconnection plus wires to HC). The total area overhead is 22%, but the voltage scaling strategy has been used to reduce the power, saving 6.6% in power consumption when compared with non-protected NoC. However, only triple-error correction and quadruple-error detection are possible, considering that there is a duplicated interconnection in each link.

In [6], redundancy has been applied in some specific components inside of a 2-channel router, which means that there are two interconnections in each channel of the router to provide reliability in the links. The area overhead is between 12.5% and 15.5%, due to the number of buffers used. Results for a 64-core NoC (for link-size that occupies 5.45% of the total area) show that when there are 20 faulty wires the connectivity is around 90%, while for 100 faulty wires the connectivity becomes low, around 30%. Furthermore, the redundancy also degrades the latency.

The work in [7] proposes the use of partially faulty links when the traffic in the network is high. The main idea is to make a uniform distribution of traffic in the links. The links capacity can be split in groups of 25%, 50%, 75%, and 100% of wires, according to the faults in the link. The proposed technique has a power consumption overhead among 5% and 8% and an area overhead of 15% to 21%. However, [7] considers that all faults concentrates within the same group of wires (affecting exactly 1/4, 1/2 or 3/4 of the link), although faults can be distributed among the link. In this way, each data has a pre-defined position to be transmitted. For instance, the first bit of the data can be placed in group1-bit1, or group2-bit1, or group3-bit1 or group4-bit1. If there is a fault in each group, some bit of data always will be affected by the faulty wire in the group, and then the link must be avoided (but avoiding a link, the traffic can be damaged).

The works proposed in [23] and [24] combine mapping and adaptive routing to increase reliability in NoCs. Both works present a mapping strategy that concurrently takes into account the application core graph, the fault probability in the links and the routing. Their goal is to obtain the Pareto set of mapping configurations with customized routing functions that minimize the average latency and maximize the reliability

of the application. Both proposals use the same routing algorithm (APSRA), and do not cover faults between cores and routers, just between cores. The difference between [23] and [24] is the mapping algorithm. Moreover, the proposed technique can solve the problems caused by faulty links between core and router, while in [23-24] it is not possible, reducing their efficiency to 65% in the NoC with 12 routers and cores.

The works presented in [8] and [9] use adaptive routing to avoid faulty links and faulty routers, which implies in a relative low latency overhead. However, they use virtual channels and memory tables to avoid deadlock in the network, which are normally synonymous of area overhead and excessive power consumption. Besides that, they cannot cope with faulty wires between router and cores, because there is no a redundant path to re-route the data.

In [10], the authors propose a lightweight partially adaptive (LAR) routing strategy to cope with multiple defects in each link and multiple faulty links based on minimal change in the XY path. LAR provides minimal changes in the XY path of 2-D Torus NoC, and it can cope with faults that affect up to 100% of wires in a single link, once that the faulty link can be completed avoided by using a different path to forward the packets. Consequently, virtual channels and tables are not used, and the technique in [10] has just 1% of area overhead. However, LAR cannot cope with faulty wires between a router and core, because there is no redundant path to reach its target. Besides that, LAR cannot access a router when both inputs in vertical (South and North) or horizontal (East and West) are faulty, as well as cannot leave a router with both faulty outputs, because these situations also do not allow an alternative fault-free path. Results in [10] have shown that by using only adaptive routing, 34% of faulty links would still be non-protected in a single-fault scenario. This percentage can be even higher when considering multiple faulty links and specific LAR limitations. The advantage of LAR appears in the lowest overhead in area and performance compared to the others techniques based on parity check or Hamming code, making the penalty in time and power almost imperceptible. However, LAR itself is unable to tolerate a large number of multiple faulty cases, because for many combinations of multiple faulty wires in multiple links there is no available alternative fault-free path to be used. Consequently, the combination of LAR with another fault-tolerant technique able to use faulty links in some critical cases can be used to achieve a good compromise in reliability, area, performance and power overhead.

Therefore, it is evident that we still need efficient solutions to solve the problem of multiple faults in NoC interconnections, with minimum overheads and large fault coverage. For this reason, our initial idea was to combine [10] with data splitting (DS) and re-mapping of tasks to achieve good trade-offs, as can be seen in [12]. Initially [12] has a double impact in latency for each communication through the faulty interconnection, because two clock cycles are necessary to send each data with DS strategy. However, to minimizing this impact, [12] considers re-mapping of tasks, although it could not be applied in all situations of faults, keeping sometimes the time penalty still high. In the next section, significant upgrades have been done in the proposal developed in [12]. Memory elements sensitive to the level of clock were inserted, which enables to use the data splitting in only one clock cycle. Then, the re-mapping of tasks could be removed by adding memory elements, simplifying the strategy. As

the approach uses the information about the fault diagnosis together with the best fault-tolerant configuration, the proposal obtains good trade-offs in relation to traditional Hamming approach, as will be presented later.

4 The Adaptive Technique Based on Adaptive Routing and Data Splitting: ATARDS

ATARDS copes with multiple defects, interlink and intralink. ATARDS tolerates permanent faults, as shorts and open circuits, or intermittent defects such as crosstalk. ATARDS is an improvement of [12], because it does not use re-mapping of tasks and transmits a flit in two halves considering just one clock cycle. ATARDS uses latches structure to store the data and sends each half of information in different clock levels. With the new approach, the re-mapping of tasks is not necessary because a faulty interconnection does not introduce delay in the communication time (considering clock cycles). The latency in clock cycles is the same for the proposed technique and original NoC without any fault tolerance technique. The difference is in the maximum frequency for each proposal. The router frequency is limited by the hardware overhead introduced in each approach. ATARDS has lower maximum frequency when compared to the original (non-protected) router, since there is more hardware in the critical path. However, the latency (in cycles) is the same between a NoC with ATARDS or original router, but the communication time (in seconds) is different, it depends on the maximum frequency. Even with a reduction in maximum frequency, on ATARDS compared to non-protected router, it is possible to reduce the delay impact, once no extra cycle has been inserted for cases with faulty interconnections. ATARDS also does not add any extra wires in the links, as most of related work in literature does [2-6].

ATARDS has been implemented in 12-core SoCIN NoC [16] with 2D-torus topology without virtual channels. The router architecture has been implemented in VHDL, and each router can be connected to four neighboring routers with two unidirectional channel links. Each router has a local port with a processor element connected. The architecture uses packet switching and deadlock-free XY-routing. Each input channel port has a buffer with 4 slots. All routers are capable of using the lightweight adaptive routing (LAR) and data splitting (DS), however only the ones with faulty interconnections uses one of these techniques in order to minimize the overheads according to the fault case. By using test and diagnosis [10, 19-21], each router is configured with the information relative to the faulty interconnections (registers in each channel receive the information about faulty interconnections and multiplexers from DS technique receive the information about the specific faulty wires). In presence of defects, LAR technique is always the first choice, because it has minimal impact in communication time and power.

For LAR technique, the routing algorithm checks the test information before forwarding a packet. Each router is configured with the manufacturing test information about faulty-links. An additional 10-bit register is added in each router with the test results to inform if one or more of its channels are faulty. When the contemplated

output channel is indicated as faulty, an alternative path replaces the original one in the header, and the packet is re-routed through the fault-free path. Each router knows the NoC size and its own position, so it can calculate the new number of steps needed for the packet in the new path. In the 2D-torus topology of size *m* x *n*, a packet has two possible routes in the same dimension: it may go *k* steps to one way (positive) or *m* - *k* (or *n* - *k*) steps to the other way (negative). Though, a packet travels no more than *m* - *1* or *n* - *1* steps from source to destination when *m* or *n* is odd, or only *m* or *n* steps when they are even. As a consequence, the router dynamically changes the target address in the header in a packet when the original address intends to use a faulty link. LAR has a small impact in latency, less than 1% for the simulated cases. This little impact can be explained because on the average, the opposite path is not much larger than the original path, and for all considered cases the alternative path was not heavily congested.

However, LAR cannot cope with fault cases when there is no redundant path. For these blocking positions, the faulty channel cannot be discarded, because the connectivity needs to be sustaining, and another strategy must be used. When fault affects both input and output channels in the same direction, the router becomes inaccessible, as presented for router R6 and R11 in Figure 1. In addition, when the fault affects the channel that connects cores and routers, there is no alternative path too, as shown in Figure 1 for MED CPU and IDCT cores. The combination of LAR with another fault-tolerant technique can enable to use faulty links with a good compromise in reliability, area, performance and power overhead. So, for simplicity, one aggregates data splitting in LAR approach. Figure 1 shows the 12-core 2D-torus NoC with the MPEG4 application mapped into the system.

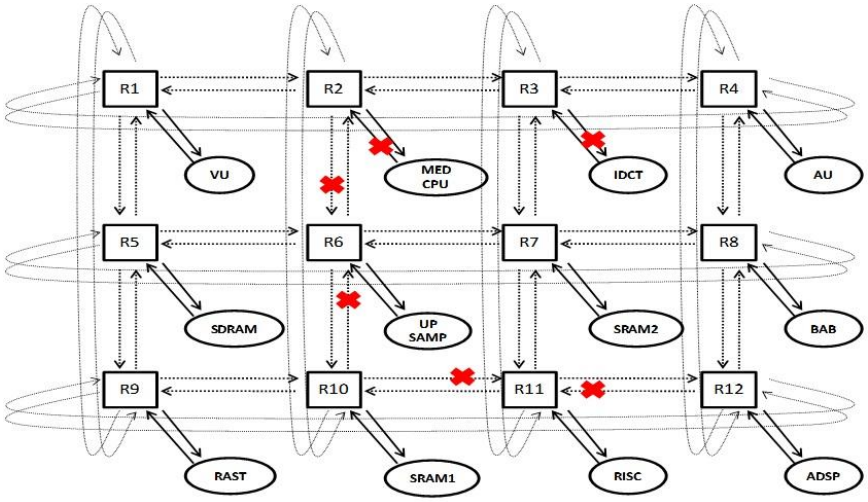

Fig.1. MPEG4 benchmark mapped in a 12-core 2-D Torus NoC. The limitation of the LAR technique is shown by the indicated faulty links that cannot be protected by LAR mechanism.

Data splitting technique can use the partial link by selecting just the fault-free wires for each transmission. DS uses multiplexers to select the fault-free wires in each input and output channel of the router, including the local channel. The data is sent in two parts by using 50% of the interconnection in each moment. The control of each multiplexer is configured off-line based on the test results. An example of configuration can be found in Figure 2 (a) for an 8-bit link, where four faulty wires are considered. Two latches (L1 and L2) have been used to store each half of the data, because the DS technique can send the data in one cycle of the clock, at clock high and low levels. When clock is high, the first data half is transmitted through the link and it is stored in L1. At the second moment, at clock low, the second half of the data is transmitted and stored in L2. In the next cycle, both data are stored in L1 and L2 so the data can be placed together in the input buffer, as can be seen in the waveforms in Figure 2 (b). When DS solution is not necessary, the multiplexers can be bypassed and turned off [14-15].

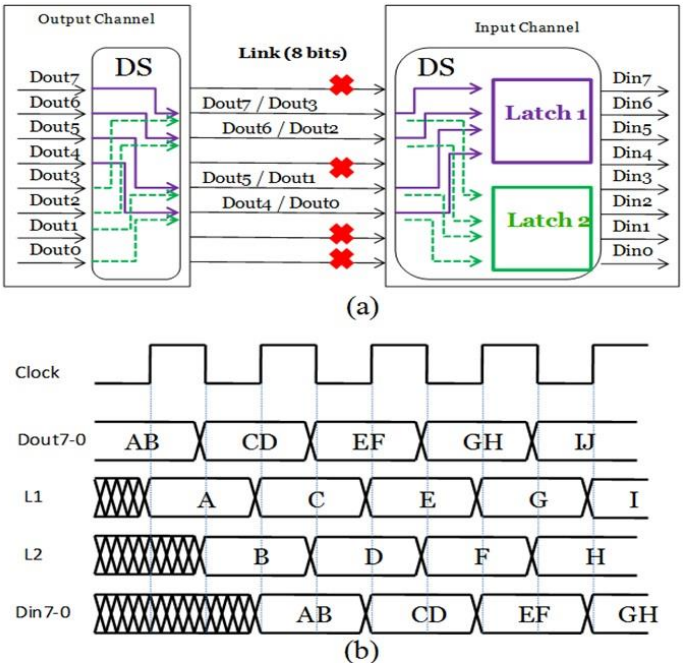

Fig.2. DS technique implementation in the ATARDS approach: (a) an example of configuration using DS; (b) waveforms for a hypothetical communication showing the L1 and L2 latches processing the data in one clock cycle.

In Figure 3, a flowchart has been used to explain the order of application of the LAR and DS strategies. Firstly the approach applies test vectors in the NoC to test and diagnose faults in the wires of the interconnection. So, after that, one tries to use LAR strategy. LAR being a possible solution to isolate the faulty links has minimum power overhead, since LAR usually introduces 1% in power overhead (see the synthesis

results). LAR, needs to avoid deadlock situations: LAR can ensure a deadlock-free communication when there is only one faulty link in each interconnection's group in row and column of the NoC. A row of interconnections is, for example, all horizontal interconnections placed among Router 1, Router 2, Router 3 and Router 4 (Figure 1). A column of interconnections is a group of vertical interconnections, for example, placed among Router 1, Router 5 and Router 9 (Figure 1). When there are at least two different faulty links in a row or in a column of interconnections, it is necessary to use DS at least once. When LAR is not an option, one applies DS. DS solution will be used only when "up to X faulty wires?" is affirmative. For our approach, the X value corresponds to 50% of the wires into an interconnection. In cases when the faulty channel has more than 50% of faulty wires, the approach isolates the faulty interconnection. The flowchart needs to be repeated for each interconnection in the NoC.

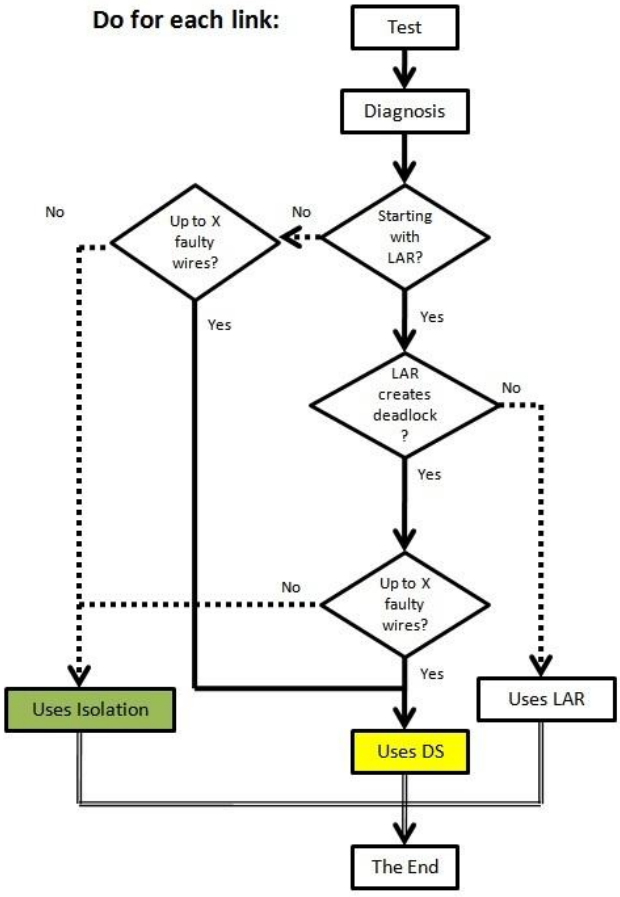

Fig.3. Flowchart to decided what technique will be used by the ATARDS approach. The X value is 50% of the wires in the interconnection.

4.1 Fault Coverage

To compare the fault coverage, one chooses to present the following approaches for each NoC's router: non-protected, LAR, ATARDS and Hamming code (HC). To measure the fault coverage one considers two cases: acceptable number of faulty wires in each interconnection and acceptable number of faulty interconnections for each solution. Figure 4 shows a situation with only one faulty interconnection, and this interconnection can have one or more faulty wires. As LAR provides a new path when an entire link is faulty it has the best solution in that case. HC is the worst case, once it protects against only one fault per link. On the other hand, ATARDS can cope with only 50% of faulty wires and the non-tolerated strategy cannot be able to accept faults without any protection.

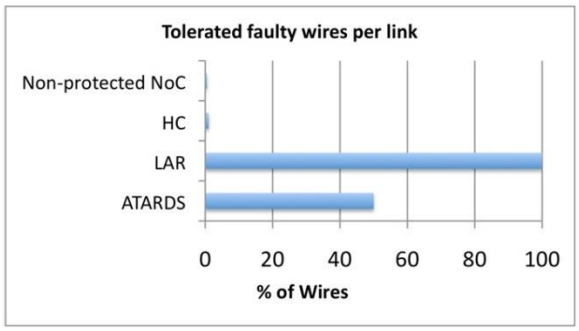

Fig. 4. Faulty tolerable number wires in each link for each strategy to sustaining connectivity.

Figure 5 shows connectivity results for a scenario with multiple faulty interconnections and just one faulty wire per link. For this scenario, LAR is the weaker strategy because it accepts only one faulty interconnection in each row and each column of interconnections without causing deadlock. Thus, LAR tolerates seven faulty links of the 12-core NoC as shown in Figure 1. For multiple faulty links with single-fault, just DS and HC can protect the entire set of interconnections in the NoC with a successful rate, while DS can still consider multiple faulty wires within an interconnection.

4.2 Connectivity

ATARDS can sustain 100% of connectivity in the NoC with a large range of multiple fault combinations, once it combines techniques that can be better utilized in accordance with the type and fault location. ATARDS can completely protect the NoC when there are up to 50% of faulty wires. Multiplexers have been used to avoid the faulty wires, shifting the information into an interconnection and using the levels high and low to send the information in the same clock cycle. One expects to use LAR when there are more than 50% of faulty wires per link.

Figure 6 shows the percentage of NoC connectivity according to the faulty wires percentage in a 12-core 2-D Torus NoC with 8-bit link. We are considering the best

fault distribution for each strategy. We also take into account the number of links used by the application in the NoC. The scenario regards the defect's location and the application.

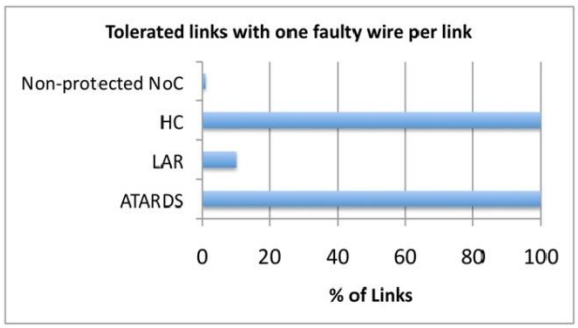

Fig. 5. Tolerated number of faulty interconnections with only one faulty wire per link in 12-core NoC.

Figure 1 presents MPEG4 mapping into a NoC, for that specific mapping 24 links are not used. Wherefore, faults in these links do not affect the connectivity, and therefore we consider that faults first happen in these interconnections, for all strategies. ATARDS can sustain 100% connectivity with up to 50% of faults distributed in each interconnection because of the DS capability. For the particular case of MPEG4, it can sustain 100% connectivity even with 60% of faults (considering the best case of fault distribution).

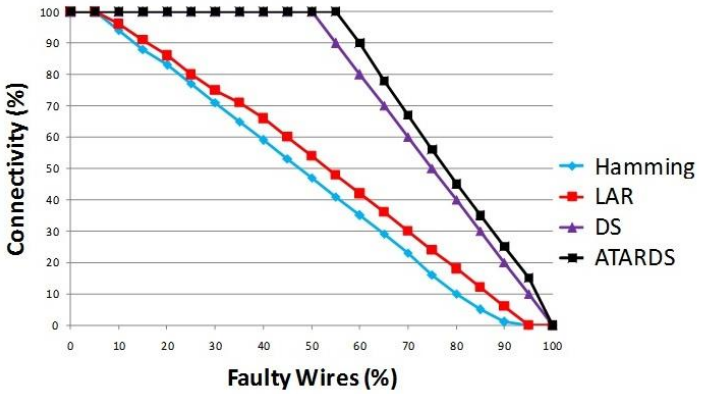

Fig. 6. NoC connectivity percentage in a generic 12-core NoC with MPEG4 benchmark.

ATARDS can have better results because LAR and DS have been combined to improve the fault tolerance. Data splitting (DS) only presents good results when 50% of the wires are faulty-free in each link. LAR, by itself, can consider up seven faulty links completely faulty in a 12-core NoC to avoid deadlock situations in according to

its limitations: only one fault in each row and column of the NoC can be considered, as explained in Figure 1. Hamming code has the best case of protection when there is just one faulty wire in each interconnection, that means that the efficiency is reduced as soon as possible if faults happen in more than one wire within the interconnection.

Figure 7 shows connectivity results where the number of faulty wires have been considered between 0 and 100, and the percentage calculated based on the total number of wires in the network, considering 8-bit link. The faults can be distributed in any wire of the NoC, and the worst and the best scenario compared for each approach, considering a generic case for a 4 x 3 and 8 x 8 NoC, as applied in [6]. For our approach, the best scenario happens when the faults occur in 7 specific links for a 4 x 3 NoC or 16 in an 8 x 8 NoC (ATARDS using LAR). When DS becomes necessary, the best case is when the faulty wires are at most 50% of faulty wires in each interconnection.

The best scenario for [6] occurs when the faults are completely distributed among the interconnections, because redundancy and duplication are used by the authors. The worst case for [6] is not clearly specified by the authors, but we compare with our worst case in an 8 x 8 NoC scenario (when the faults happen in more than 50% of the interconnections). For instance, when an interconnection with 8 wires has more than 4 faulty wires. Considering an 8 x 8 NoC with 100 faulty wires, ATARDS presents 6% less of connectivity, while [6] presents almost 70% of loss. It happens because 100 faulty wires are easily tolerated by our strategy when there are many wires and interconnections considered. When the total number of interconnections is lower, like in a 4x3 NoC case, our proposed technique shows up 30% less of connectivity, because there are few interconnections in the network and 11.57% of the total wires are faulty.

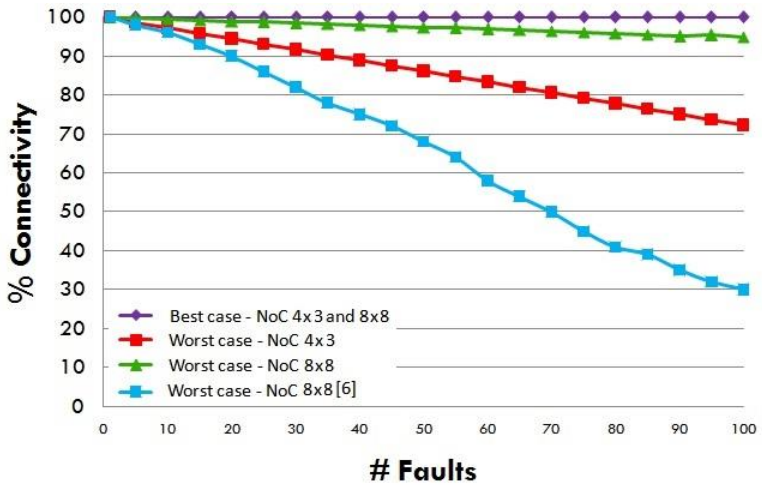

Fig.7. Comparison of connectivity in two generic NoCs considering ATARDS and the solution proposed in [6] with 8-bit link.

5 Experimental Results

ATARDS has been compared to LAR and HC technique in terms of area, performance and power. Table 1 shows the synthesis results for each router developed based on RASoC [18], with 8-bit and 32-bit links, using Synopsys Power Compiler tool with 65 nm CMOS standard cell library. Besides the data bits in the link, the SoCiN network has 2 extra wires per link to set the packet control and other 2 extra wires to do the handshake and verify the buffer availability in each channel.

ATARDS has the largest area overhead because of its configurability properties. HC presents the largest performance overhead, due to the encoding and decoding blocks with long chains of XORs. HC also incurs the largest overhead in the number of wires, once each link needs to send extra codification, using 4 and 6 extra wires for 8 and 32-bit link, respectively. Both maximum frequency and normalized frequency at 300 MHz have been considered to calculate the power results. For ATARDS, there are different types of power results, in according to the number of routers using the approach: the number of routers using DS active depends of the amount of faults present in the wires and its location. Therefore, some routers can turn off the DS solution when it is not necessary in the network. Then, when ATARDS turn off the DS solution, it is running like a LAR router, and this situation is useful in the absence of faulty links to improve the power and energy results.

	Router	**Area (μm^2)**	**Critical Path Delay (ns)**	**Router Power (µW)@ Max. Freq.**	**Router Power (µW)@ 300MHz**	**Total # of Wires**
8 Bits	Non-protected	5360.3	1.11	334.06	111.3	864
	LAR	5260.3	1.11	338.19	112.7	864
	ATARDS (DS on)	6978.7	1.71	498.49	255.9	864
	ATARDS (DS off)	6978.7	1.71	216.64	112.7	864
	HC	5948.8	2.04	295.21	180.6	1152
32 Bits	Non-protected	13850.3	1.26	811.04	306.8	2592
	LAR	14071.3	1.26	819.12	308.4	2592
	ATARDS (DS on)	19910.4	2.09	1392.5	873.6	2592
	ATARDS (DS off)	19910.4	2.09	488.95	308.4	2592

Table 1. Synthesis results for 65 nm technology. The total number of wires was considered for the NoC with 12-core 2-D Torus.

Some power and energy results also are available for the SoCIN NoC with 12-core 2-D Torus [16]. A 4 x 3 NoC is often used by MPEG4 and VOPD benchmarks. As the

behavior of these two applications is very similar, the results are in a very close range, and could be aggregated in the same value of power overhead. Figure 8 shows the best case scenario at power consumption and fault coverage for each approach (HC, ATARDS, DS and LAR). HC copes with up to 72 faulty channels, but a single-fault needs to be considered in each channel, showing its limitations. LAR can tolerate up to 7 completely faulty links, in a specific configuration. For multiple faulty wires and faulty links scenario HC and DS are not a solution, on the other hand, ATARDS can cope with better than just DS.

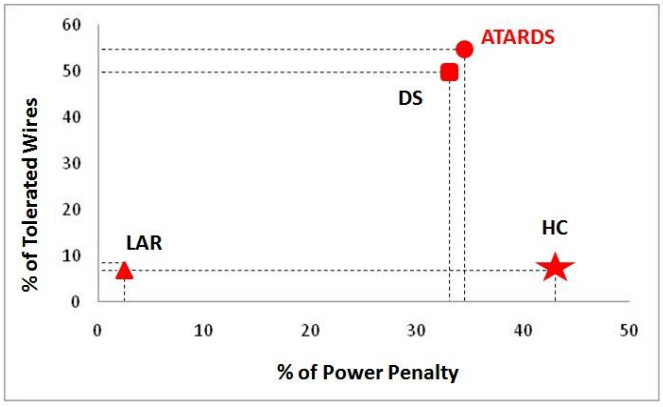

Fig. 8. The number of faulty wires tolerated by each fault-tolerant technique (considering the best fault distribution scenario) and the power penalty results, on the average, for MPEG4 and VOPD benchmarks mapped into a12-core 2-D Torus NoC. The values are very similar between 8 and 32-bit links. The results are normalized according to the power of the original non-protected NoC.

To measure the energy and power consumption of wires and routers in the NoC we considered a generic packet injection rate (1 flit/node/cycle) and a NoC size with 12-core. The power consumption in each wire has been calculated using simulations at Spice level based on the distributed π-model for the wires [17]. We assumed values between 1 mm and 1.5 mm of wire length for each link in 8 bits. The total power is the sum of the power in the router and in the wires. Figure 9 depicts the energy overheard having the non-protected router as base, for a case considering the average between VOPD and MPEG4. The energy overhead was obtained multiplying the total power by the communication time (in seconds) at maximum frequency. HC has the higher overhead in energy because there are 4 extra wires in an 8-bit link design. LAR solution has a low impact in energy, because it is similar to non-protected router. For ATARDS there is a variable impact in energy. When there are few faulty links in the network, some ATARDS routers can avoid the DS block to improve the energy, bypassing it. For instance, when there are only 1 faulty link requiring two routers with DS on, minimizing the energy results. So, all the other routers will run with LAR on,

and their DS block turned off. In the HC case, there is no variation with or without faults, and there is almost 110% of energy overhead.

Figure 10 depicts power overhead at 300 MHz. For the experiment in Figure 10, all proposed scenarios are running at the same frequency. The injection rate reduced to 25% of switching activity. ATARDS has 22% of power overhead in relation to the non-protected router, while the Hamming code has 37% of power overhead. The power results can be easily converted to energy results if you consider an equal execution time for all proposed scenarios. Latency and throughput results are not taken into account, because the frequency for the considered techniques is 300 MHz, consequently the traffic is the same for all situations.

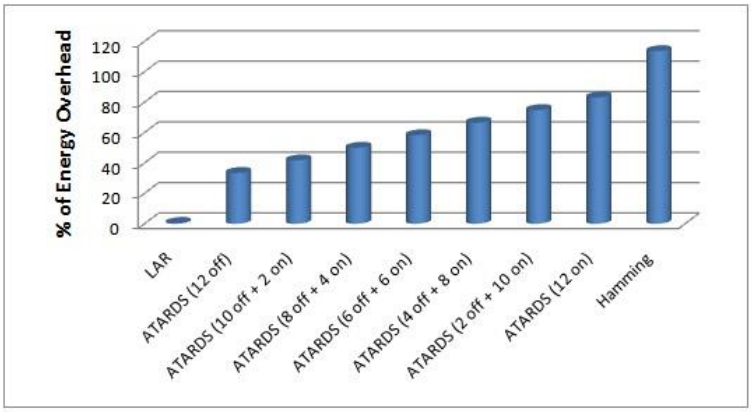

Fig. 9. The percentage overhead in energy for each configuration approach with 8-bit link design.

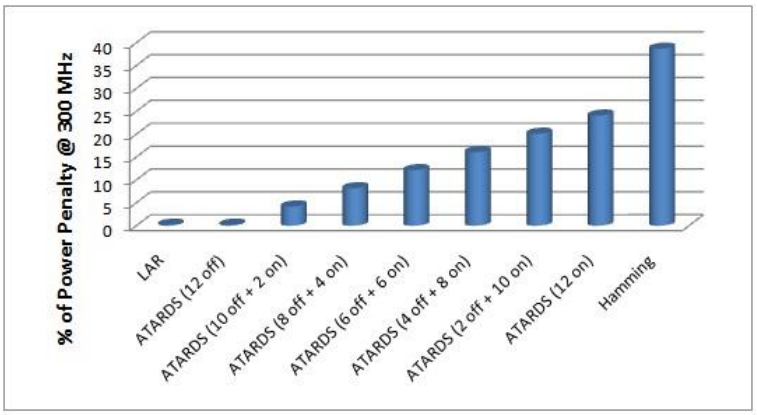

Fig. 10. Power penalty @300MHz for all strategies with 8-bit link design.

6 Conclusion

An adaptive strategy for fault tolerance in NoC interconnections has been presented. The strategy named ATARDS is able to improve the yield in the presence of many faulty wires and many faulty interconnections. The technique merges LAR, DS and memory elements (latches) to decrease latency and sustain reliability. When a data uses data splitting, it is sent using only one clock cycle and no time penalty is incurred to transmit the packet through the NoC.

The DS block is used only in situations that LAR cannot cope with it. In fault-free situations, DS can be turned off to save power consumption and energy, and ATARDS runs like a LAR router. Besides that, there is no need for extra wires in ATARDS, and because that we can obtain good energy results. Moreover, ATARDS has a variable impact in power dissipation, depending on the faults location, while the HC has an excessive and constant impact. The energy can be saved when compared to well-known strategies such as Hamming code.

For MPEG4, the strategy can sustain high connectivity even when there are 60% of faulty wires, considering the 8-bit link design with the best distribution of faults. Meanwhile, HC can protect the NoC only if no more than 8.3% of the wires are faulty, considering a single-fault scenario per link.

7 References

1. Dehon, A.; Naeimi, H.;, "Seven strategies for tolerating highly defective fabrication," *IEEE Design & Test of Computers*, vol.22, no.4, pp. 306- 315, 2005.
2. Frantz, A. P.; Kastensmidt, F. L.; Carro, L.; Cota, E.;, "Dependable Network-on-Chip Router Able to Simultaneously Tolerate Soft Errors and Crosstalk," *Proceedings of 2006 International Test Conference (ITC)*, vol. 1, pp. 1 – 9, 2006.
3. Lehtonen, T.; Liljeberg, P.; Plosila, J.;, "Online Reconfigurable Self-Timed Links for Fault Tolerant NoCs," *VLSI Design*, vol. 2007, Article ID 94676, pp. 1-13, 2007.
4. Braga, M.; Cota, E.; Kastensmidt, F.L.; Lubaszewski, M.; , "Efficiently using data splitting and retransmission to tolerate faults in networks-on-chip interconnects," *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, vol., no., pp.4101-4104, 2010.
5. Ganguly, A.; Pande, P.P.; Belzer, B.;, "Crosstalk-Aware Channel Coding Schemes for Energy Efficient and Reliable NOC Interconnects," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol.17, no.11, pp.1626-1639, Nov. 2009.
6. Kakoee, M.R.; Bertacco, V.; Benini, L.; , "ReliNoC: A reliable network for priority-based on-chip communication," *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011*, pp.1-6, 14-18 March 2011.
7. Palesi, M.; Kumar, S.; Catania, V.; , "Leveraging Partially Faulty Links Usage for Enhancing Yield and Performance in Networks-on-Chip," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol.29, no.3, pp.426-440, March 2010.
8. Koibuchi, M.; Matsutani, H.; Amano, H.; Mark Pinkston, T.; "A Lightweight Fault-Tolerant Mechanism for Network-on-Chip". *2nd ACM/ IEEE International Symposium on Networks-on-Chip*, pp. 13-22, 2008.

9. Tornero, R.; Sterrantino, V.; Palesi, M.; Ordua, J.M.; “A multi-objective strategy for concurrent mapping and routing in networks on chip,” *IEEE International Symposium on Parallel & Distributed Processing*, pp.1-8, 2009.
10. Concatto, C.; Almeida, P.; Kastensmidt, F.; Cota, E.; Lubaszewski, M.; Herve, M.;, "Improving yield of torus NoCs through fault-diagnosis-and-repair of interconnect faults," *On-Line Testing Symposium, 2009. IOLTS 2009. 15th IEEE International*, pp.61-66, 24-26 June 2009.
11. Agrawal, V. D.;, "Testing for Faults, Looking for Defects," *Test Workshop (LATW), 2011 12th Latin American*, Keynote Talk, March 2011.
12. Kologeski, A.; Concatto, C.; Carro, L.; Kastensmidt, F.L.; , "Adaptive approach to tolerate multiple faulty links in Network-on-Chip," *Test Workshop (LATW), 2011 12th Latin American*, pp.1-6, 27-30 March 2011.
13. Shih-yu Y.; Papachristou, C.A.; , "A method for detecting interconnect DSM defects in systems on chip," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol.25, no.1, pp. 197- 204, Jan. 2006.
14. Changbo L.; Lei H.; “Distributed sleep transistor network for power reduction,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 9 , pp. 937-946, Sep. 2004.
15. Shi, K.; Howard, D.; “Sleep Transistor Design and Implementation - Simple Concepts Yet Challenges To Be Optimum,” *International Symposium on VLSI Design, Automation and Test*, pp. 1 – 4, 2006.
16. Zeferino, C.A.; Susin, A.A.; , "SoCIN: a parametric and scalable network-on-chip," *Integrated Circuits and Systems Design, 2003. SBCCI 2003. Proceedings. 16th Symposium on*, pp. 169- 174, 8-11 Sept. 2003.
17. Sakurai, T.;, "Approximation of wiring delay in MOSFET LSI," *Solid-State Circuits, IEEE Journal of*, vol.18, no.4, pp. 418- 426, Aug 1983.
18. Zeferino, C.A.; Kreutz, M.E.; Susin, A.A.; “RASoC: a router soft-core for networks-on-chip,” Design, Automation and Test in Europe Conference and Exhibition, vol. 3, pp. 198 – 203, 2004.
19. Cota, E.; Kastensmidt, F.L.; Cassel, M.; Herve, M.; Almeida, P.; Meirelles, P.; Amory, A.; Lubaszewski, M.;, "A High-Fault-Coverage Approach for the Test of Data, Control and Handshake Interconnects in Mesh Networks-on-Chip," *Computers, IEEE Transactions on*, vol.57, no.9, pp.1202-1215, Sept. 2008.
20. Shih-Yu Y.; Papachristou, C.A.; Taib-Azar, M.;, "Improving bus test via I_{DDT} and boundary scan," *Design Automation Conference, 2001. Proceedings*, pp. 307- 312, 2001.
21. Grecu, C.; Pande, P.; Ivanov, A.; Saleh, R.; , "BIST for network-on-chip interconnect infrastructures," *VLSI Test Symposium, 2006. Proceedings. 24th IEEE*, pp.6 pp.-35, April 30 2006-May 4 2006.
22. Kologeski, A.; Concatto, C.; Carro, L.; Kastensmidt, F.L.;, "Improving Reliability in NoCs by Application-Specific Mapping Combined with Adaptive Fault-Tolerant Method in the Links," *European Test Symposium (ETS), 2011 16th IEEE*, pp.123-128, 23-27 May 2011.
23. Tornero, R.; Sterrantino V.; Palesi M.; Orduna J.;, "A multi-objective strategy for concurrent mapping and routing in networks on chip," *IEEE International Symposium on Parallel & Distributed Processing*, pp.1-8, 2009.
24. Choudhury A.; Palermo G.; Silvano C.; Zaccaria V.;. "Yield Enhancement by Robust Application-specific Mapping on Network-on-Chips," In *NoCArc'09 - Second International Workshop on Network on-Chip Architectures*, pp. 37-42, 2009.