

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/236236373>

# NoC generation of an optimal memory distribution for multimedia systems

Article in *Proceedings of SPIE - The International Society for Optical Engineering* · May 2009

DOI: 10.1117/12.821523

CITATIONS

0

READS

47

6 authors, including:



**Félix B. Tobajas**

Universidad de Las Palmas de Gran Canaria

81 PUBLICATIONS 212 CITATIONS

[SEE PROFILE](#)



**Valentín de Armas Sosa**

Universidad de Las Palmas de Gran Canaria

63 PUBLICATIONS 151 CITATIONS

[SEE PROFILE](#)



**Eduardo de la Torre**

Universidad Politécnica de Madrid

131 PUBLICATIONS 928 CITATIONS

[SEE PROFILE](#)



**Roberto Sarmiento**

Universidad de Las Palmas de Gran Canaria

202 PUBLICATIONS 1,075 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



NAuTILES(Novel Autonomous Intelligent Localisable Endoscopy System) Link: <https://nautilesorg.wordpress.com> [View project](#)



CERBERO - Cross-layer modEL-based fRamework for multi-oBjective dESign of Reconfigurable systems in unceRtain hybRid enviroNments [View project](#)

# NoC generation of an optimal memory distribution for multimedia systems

Raúl Regidor<sup>a</sup>, Félix Tobajas<sup>a</sup>, Valentín de Armas<sup>a</sup>, Eduardo de la Torre<sup>b</sup>, Teresa Riesgo<sup>b</sup>, Roberto Sarmiento<sup>a</sup>

<sup>a</sup>Instituto Universitario de Microelectrónica Aplicada (IUMA), Universidad de Las Palmas de Gran Canaria, Campus Universitario de Tafira, 35017 Las Palmas de Gran Canaria;

<sup>b</sup>Centro de Electrónica Industrial (CEI), Universidad Politécnica de Madrid, 28006 Madrid

## ABSTRACT

In this paper a topological analysis of different IP distributions focusing on optimal memory placements in regular 2D-Meshes has been performed. As case study, a real MPEG-4 decoder implementation with three memories was chosen. In order to study the influence of memories in the topology of the network, Arteris NoCexplorer tool was used. The results inferred from the experiments show how the performance of a multimedia system can be improved if memories are properly located within a NoC. Furthermore, the present work serves to validate the use of Arteris NoCexplorer for simulating and modelling complex NoC based designs. In addition, a methodology for determining the best IP distribution in terms of latency and throughput is presented and its feasibility is demonstrated.

**Keywords:** Networks-on-Chip, NoC, multimedia, shared memories

## 1. INTRODUCTION

In nowadays telecommunications era, traditionally disjointed fields, like computation, communication and multimedia, are converging into new applications demanded by the society. Videoconference and Video-on-Demand (VoD), together with a higher speed access to the Internet, are the main reasons why new video compression standards are being developed in order to create suitable applications for mobile phones and digital television.

In order to transmit images or video sequences, data have to be previously compressed. During video processing, lots of resources are required, especially memory devices, because of the huge amount of space needed. According to the International Technology Roadmap for Semiconductors (ITRS), the continuous decrement of transistor's size will increase the complexity of future System-on-Chips (SoCs) [1]. Consequently, the number of Intellectual Properties (IPs) contained in a single chip will increase, as have been done in the last years. However, what makes this increment different is that future integrated circuits will contain more memory than logic.

In order to overcome future interconnection problems, Networks-on-Chip (NoCs) have been proposed as an emerging solution to support reusability demands as well. In NoCs, IPs are interconnected through a network formed by routers reducing therefore the number of global wires and allowing new elements to be easily added to the structure. A NoC can be mainly described by its topology and the strategies used for routing, flow control, switching, arbitration, and buffering. The network topology, which is the arrangement of nodes and channels into a graph, is one of the most important aspects of a NoC. Although several tools have already been developed [2] and some theoretical models have been proposed [3] [4] for determining the best topology of a NoC, no global solution has been found yet [5].

In the light of the mentioned circumstances, a topological analysis of different IP distributions focusing on optimal memory placements in regular 2D-Meshes has been performed. As case study, a real MPEG-4 decoder implementation with three memories was chosen [6]. In order to study the influence of memories in the topology of the network, Arteris NoCexplorer tool was used [7]. In the following section, three main problems related to memory accesses in NoC platforms are described. Then, the NoCexplorer workflow and the multimedia system are explained. In section 5, the proposed methodology for determining the optimal topology is outlined. Finally, simulation results obtained with NoCexplorer are presented and discussed.

## **2. MEMORY RELATED ISSUES OF NOC BASED MULTIMEDIA DESIGNS**

After a thorough study of multimedia applications and their requirements, three main problems related to memory accesses in NoC platforms have been identified and will be hereafter described.

### **2.1 Computation and communication abstraction**

In order to reach enough flexibility at the time of reusing an IP, it is mandatory that computation keeps independent from communication. In the field of video processing, memory access and reusability go hand in hand. For this reason, a proposal of a high level memory access without considering the location of the memory space is introduced in this paper. The goal is to perform memory accesses without specifying addresses and the amount of data to be read or written. Instead of that, the IP should directly request the data. All these operations should be performed introducing an abstraction level, which means to ask only for data instead of specifying memory IPs and addresses.

In order to reach this abstraction level, it is mandatory to index the information contained in each memory. For this purpose, centralized or distributed solutions could be implemented. The centralized approach consists of storing the table of contents of all memories in registers allocated in the General Purpose Processor (GPP). When an IP needs to perform a memory operation, it will have to ask the processor for the allocation of the data first and keep this information until the operation has finished.

In the distributed solution, the information could be stored in the Network Interface (NI) of each IP. In order to minimize the required number of registers, only the information needed by the IP will be saved. This latter approach reduces latency and traffic in the network.

### **2.2 Topology synthesis**

Although Network-on-Chip is a novel concept for on chip communication introduced in this century, several tools have already been developed to automatically synthesize NoCs. In order to find the best topology, a core graph of the desired SoC is usually required, in which communication data among IPs are described. Using this core graph together with design constraints, a set of topologies that fulfil the desired objective function are proposed.

However, the first step in order to obtain the best distribution of the SoC should be to determine whether the number of IPs is optimal. In Networks-on-Chip, the number of IPs plays a very important role while selecting the topology, as well as defining the characteristics of the routers. In addition, power, area and latency are also affected.

Several methodologies have traditionally been developed in order to improve the performance of a circuit by reallocating memory, but astonishingly none of them have been used in NoC based designs yet. Wuytack minimizes the required memory bandwidth using storage bandwidth optimization (SBO) [8], a technique that analyzes memory dependencies among IPs and proposes an optimized memory configuration. The required memory is divided into regions depending on the time interval they are accessed. When two or more regions are accessed at the same time, they cannot be allocated together in the same memory. These dependences are depicted in a conflict graph that determines which regions of memory come into conflict.

### **2.3 Parallelization in video MPSoCs**

Current integrated circuits related to video processing are Multi-Processor Systems-on-Chip (MPSoC), in which each IP is responsible for one or several tasks. In distributed memory systems, each processor has its own memory space and no other processor has access to it. Multi-processing systems with distributed memory usually have lower latency because memories are placed closer to the corresponding processor.

In a shared-memory system, all processors have access to the memory at any time. Because of that, latency is higher than in distributed memory based architectures, and, therefore, a cache memory is also integrated close to each processor in order to allocate most frequently accessed data. In addition, the shared-memory architecture suffers from two main problems: cache coherence and memory consistency.

Due to the traffic nature of video processing applications, most of the previously explained problems can be solved just by avoiding them. As said before, traffic in video applications is predictable and, therefore, memory accesses are known as well. From the design phase, it is well known how many memory blocks are required and which information must be stored. Consequently, a shared memory system is not necessary. On the other hand, distributed memory architecture may not be an optimal solution because memory blocks can be underutilized during certain phases of the pipeline.

### 3. NOCEXPLORER WORKFLOW

NoCexplorer [7] is a high level modelling and simulation tool for Systems-on-Chip (SoC) interconnected by Arteris NoC IP units, which analyzes the throughput of the circuit and determines whether the selected topology, together with the distribution of the IPs in the network, fulfils the requirements of the system. In order to validate a topology, NoCexplorer estimates the latency of the NoC by taking into account different factors like congestions in the queues, priorities or the capacity of the links.

Fig. 1 shows the NoCexplorer workflow proposed by Arteris. The whole functionality of the system under analysis is described in an input script file, in which the connections among IPs are specified and the traffic is modelled. With this information, NoCexplorer builds a Network-on-Chip in order to interconnect the IPs of the described system using Arteris NoC IP elements. The interconnected circuit is then simulated and analyzed. If the results are not satisfactory, the initial script file must be modified. Once both communication and computational requirements are matched, the final interconnection infrastructure can be designed using Arteris NoCcompiler tool.

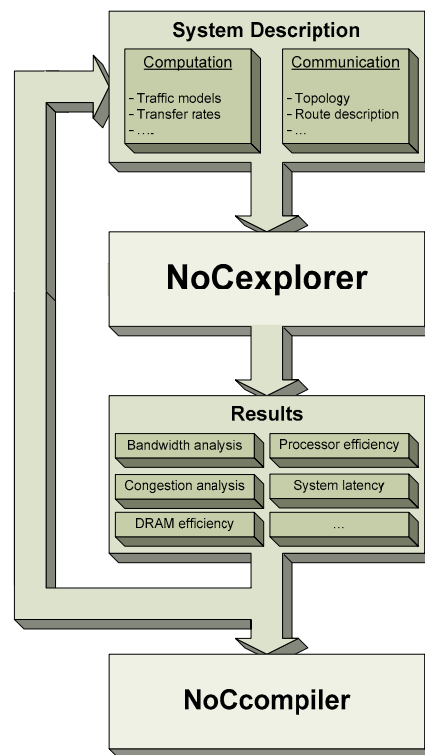


Fig. 1. NoCexplorer workflow.

The terminology used by Arteris includes several concepts that must be previously defined in order to understand the functionality of the tool. IPs that want to transmit data are called 'initiators', while the destination of the information is called 'target'. The behaviour of the system is modelled in 'processes'. For each initiator-target pair, a process is defined in which the traffic rate and the number of bytes per transaction among other parameters are specified. In order to describe the route that transaction must follow from initiator to target, a specific path containing the links for each connection must be described. The routing strategy used by Arteris NoCexplorer follows a XY-algorithm.

As can be seen, the NoCexplorer workflow is an iterative process that must be repeated until the modelled NoC fulfils the functionality of the system. However, this process can be easily automated because both the input script file and the results reports are text files. Therefore, a set of simulations that cover a wide design space can be run by automatically generating input file scripts and processing the results.

#### 4. SYSTEM DESCRIPTION

As case study, a real MPEG-4 decoder implementation [6], which has already been chosen by other research groups [9] [10] [11] [12], was used. The multimedia system is formed by twelve IPs, three of which are memories, as can be seen in Fig. 2.

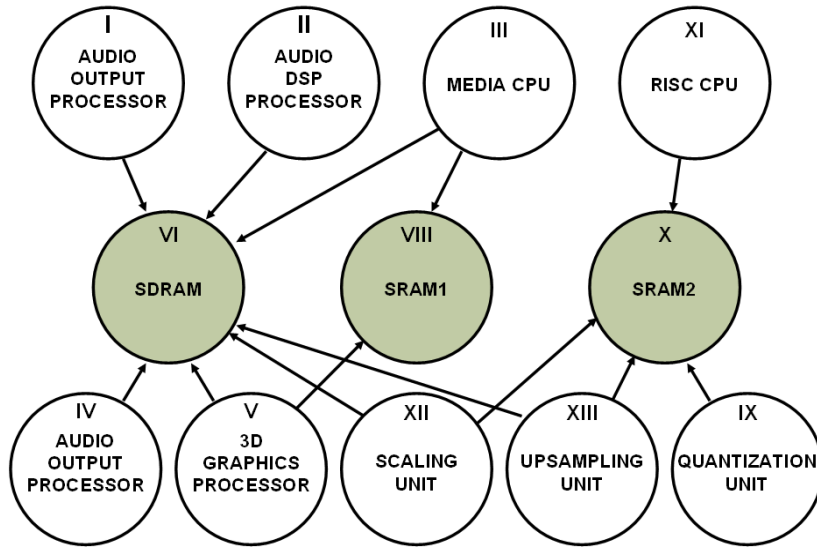


Fig. 2. System description.

The bandwidth requirements vary depending on which IP is trying to access to the memories, as represented in Table 1. Of special interest is the Upsampling Unit (XIII) because it is the IP with the highest communications requirements to the SDRAM (VI) and to the SRAM2 (X) as well. On the other side, the Audio Output Processor (I) and the Audio DSP Processor (II), which are just connected to one memory, mainly the SDRAM (VI), have the lowest traffic rates. As far as the memories are concerned, the SDRAM (VI) and the SRAM2 (X) have both very high communications demands, while the SRAM1 (VIII) is just connected to two IPs and the traffic rates are quite low compared to the other memories.

Table 1. Connections Among IPs and Bandwidth Requirements (MBpS) of the MPEG-4 Decoder Implementation.

Target Initiator	VI	VIII	X	TOTAL
I	0.5	-	-	0.5
II	0.5	-	-	0.5
III	50.0	40.0	-	90.0
IV	190.0	-	-	190.0
V	600.0	40.0	-	640.0
IX	-	-	500.0	500.0
XI	-	-	250.0	250.0
XII	32.0	-	174.0	206.0
XIII	910.0	-	670.0	1580.0
TOTAL	1783.0	80.0	1594.0	3457.0

## 5. METHODOLOGY DESCRIPTION

The objective of this section is to describe the methodology used to evaluate how the location of the memories influences in the throughput of the system.

First of all, the whole system was modelled in NoCexplorer using a random distribution of the IPs, with the aim of finding a starting topology that fulfils the communication requirements of the system.

Once all elements were correctly characterized, the impact of the SDRAM (VI) was evaluated separately by building a 2x4 NoC formed by the SDRAM (VI) and the IPs that are connected to it without modifying the initial characteristics, that is, links, clocks, queues and any other element of the system had the same properties as the starting configuration. In order to determine the influence of the memory in a 2x4 topology, all possible IP distributions were simulated. Because the experiment was focused on the influence of the position of the memory, simulations were divided in two groups. In the first group, the SDRAM (VI) was fixed in a corner of the NoC (00) and all the combinations were simulated with this restriction. Due to the symmetric nature of the topology, it is not necessary to simulate in all corners. In the second group, the SDRAM (VI) was placed in the node 01, as shown in Fig. 3. Because of the same reason as before, results can be extrapolated to nodes 02, 11 and 12. In a 2x4 NoC with one fixed IP, the number of possible IP distributions is 7! and therefore the total number of simulations was 5040 for each group.

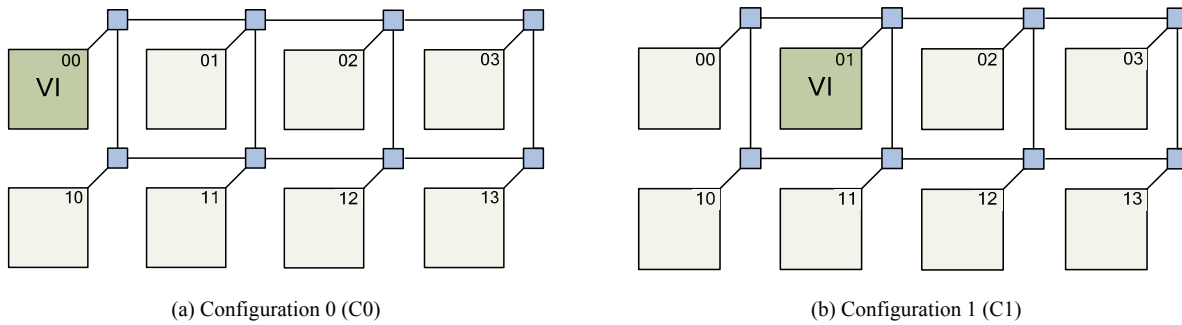


Fig. 3. Configurations for the 2x4 NoC.

Next, the influence of the two main memories was analyzed. For this purpose, the SDRAM (VI) and the SRAM2 (X) together with seven IPs were interconnected in a 3x3 NoC. Although there are nine IPs that communicate with the two memories, the Audio Output Processor (I) and the Audio DSP Processor (II) were discarded in these tests due to their low transfer rates. In this case, eight groups of simulations were executed according to the possible positions of the two memories in the topology, as represented in Fig. 4.

In order to evaluate and compare all possible IP distributions, different parameters were measured. First of all, each IP distribution was classified into valid or invalid NoC according to the reached efficiency rate. If the bandwidth requirements of the processes are not satisfied, the NoC is classified as invalid.

For all valid NoCs, the charge of each link is then measured in order to evaluate the performance. In NoCexplorer, the charge of the links is measured in terms of time spent by the link in transmitting data or waiting until the corresponding resources are assigned. When data spend more time waiting than being transmitted, the NoC has not been well sized. In order to evaluate the performance of the system, the total amount of transmission time (*total\_xfer*) was measured. In addition, the total amount of wait time for all links (*total\_busy*) was also measured.

$$total\_xfer = \sum_{i=1}^{used\_links} xfer\_time_i \quad (1)$$

$$total\_busy = \sum_{i=1}^{used\_links} busy\_time_i \quad (2)$$

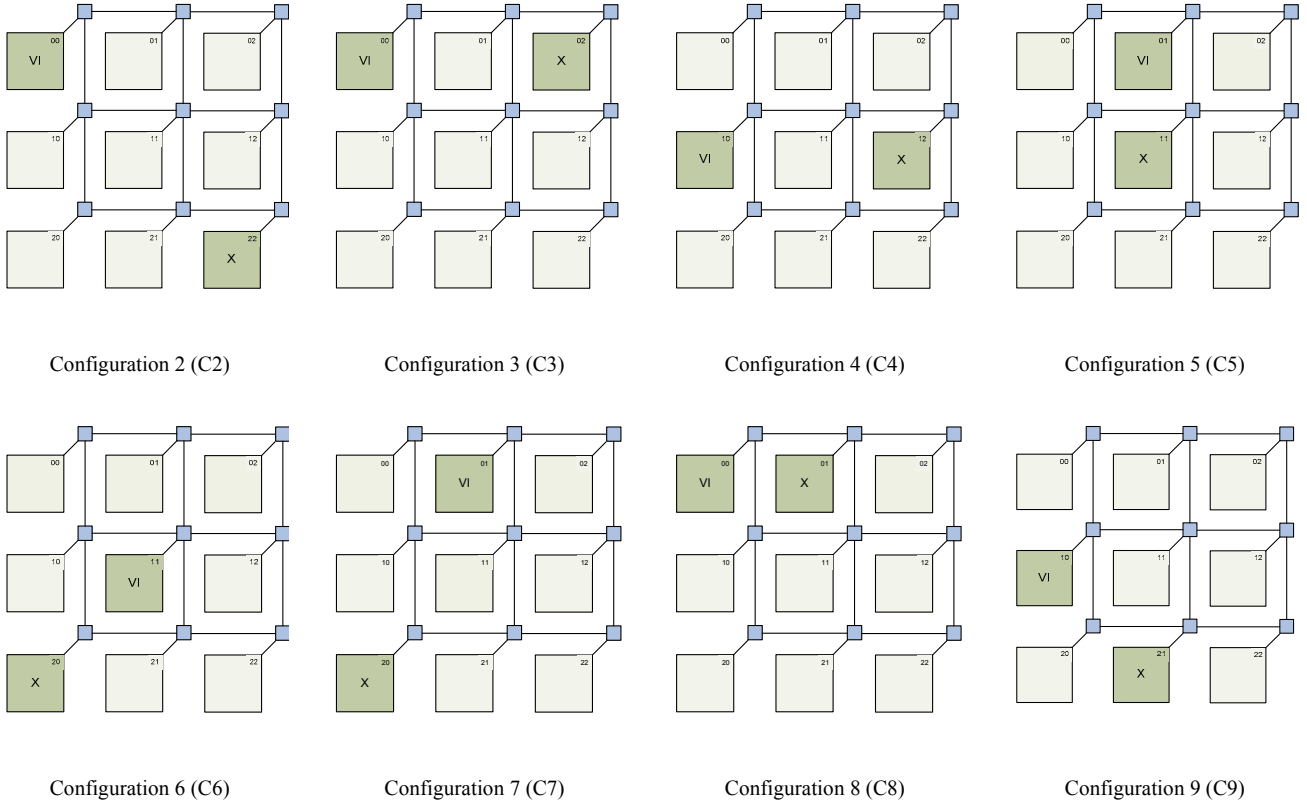


Fig. 4. Configurations for the 3x3 NoC.

In order to evaluate the functionality of a system, latency is the parameter that better represents the performance of a SoC. In this sense, NoCexplorer is a very powerful tool because of the huge amount of information that gives referring to latency. As far as queues are concerned, NoCexplorer measures latency from the IP socket interfaces, and therefore includes any delay the data incurs if it must wait within the initiator queues before it traverses the NoC. Referring to IPs, NoCexplorer shows the average and maximum latency, which are measured as the amount of time spent by packets in the network between the start of a transmission at the origin (initiator) to the start of the corresponding response packet reception at the destination (target). Because latency is measured for all initiators but they have different bandwidth requirements, a high average latency in a low demanding process does not have the same influence as in a high demanding process. For this reason, average latency and traffic rate of each initiator were combined in a new throughput measure as shown in (3). In addition, the maximum latency value reached by a process (*max\_lat*) was also considered as an indicator of the highest clock frequency that could be set for the critical path.

$$throughput = \frac{\sum_{i=1}^{initiators} (avg\_latency \cdot traffic\_rate)}{initiators} \quad (3)$$

In order to estimate the impact of the IP distribution in the power consumption of the system, the standard deviation of the transmission time (*std\_xfer*) and the wait time (*std\_busy*) were calculated according to (5) and (7). With this measure, the distribution of the power consumption in the system can be evaluated, because two NoCs can have similar values of *mean\_xfer* but different values of *std\_xfer*, which means that the power consumption of the NoC with less *std\_xfer* is distributed in a more homogeneous way.

$$mean\_xfer = \frac{total\_xfer}{used\_links} \quad (4)$$

$$std\_xfer = \sqrt{\frac{\sum_{i=1}^{used\_links} (xfer\_time_i - mean\_xfer)^2}{used\_links}} \quad (5)$$

$$mean\_busy = \frac{total\_busy}{used\_links} \quad (6)$$

$$std\_busy = \sqrt{\frac{\sum_{i=1}^{used\_links} (busy\_time_i - mean\_busy)^2}{used\_links}} \quad (7)$$

## 6. RESULTS

In this section, the results of the experiments are presented. First, the starting configuration is described and finally the analysis of the 2x4 NoC with one memory and the 3x3 NoC with two memories are discussed.

### 6.1 Starting configuration

According to the NoCexplorer workflow explained before, the first step consists of describing the functionality of the whole system in an input script file, which must also contain the communication infrastructure. In this sense, a random distribution of the IPs was chosen. In order to characterize and configure all elements in the NoC, the starting IP distribution of Fig. 5 was used. In this configuration, clock frequencies were set to 76 MHz and socket widths were set to 32 bytes, which allows traffic rates of 2.4 GBps. Although the highest bandwidth requirements come from the SDRAM (VI) and its traffic rate is less than 1.8 GBps, the network was oversized in order to take into account the overhead introduced by the NoC protocol.

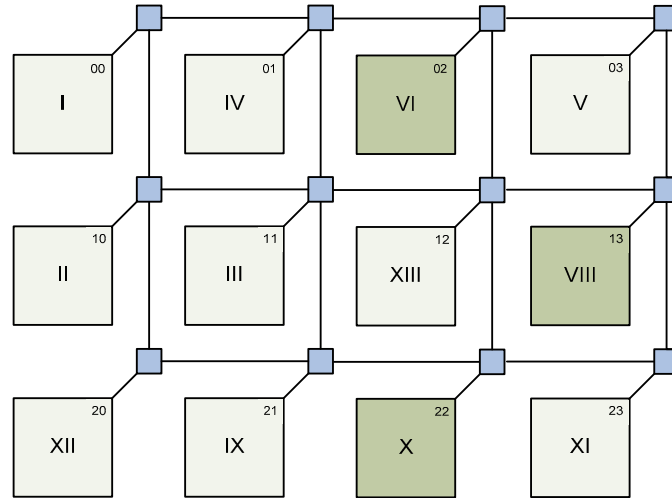


Fig. 5. Starting configuration.

Traffic was modelled by defining thirteen processes that correspond to the thirteen connections of the system. For simplicity it was supposed that the traffic rate remains constant all the time and the amount of data sent and received by an IP is the same. Consequently, each process was divided in a store process and a load process with the half part of the corresponding traffic rate respectively, and the length of all transactions was set to 128 bytes. In order to store the transactions generated by each process, thirteen queues were defined. Due to the diversity of the traffic rates, the dimension of each queue was selected according to the bandwidth requirements.



## 6.2 2x4 NoC analysis with one memory

In this analysis, all possible IP distributions were simulated for configurations C0 and C1 using the same elements as in the starting configuration. Because the bandwidth requirements of the 2x4 NoC are considerably low compared to the whole system formed by the twelve IPs, all required traffic rates were reached and therefore all IP distributions were valid. Quantitative data are summarized in Table 2.

Table 2. Connections Among IPs and Bandwidth Requirements (MBpS) of the MPEG-4 Decoder Implementation.

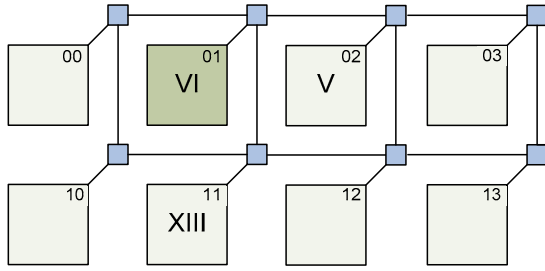
		C0	C1
used links	Min	26	24
	Max	28	28
max_lat ( $\mu$ s)	Min	0.59	0.59
	Max	1.00	1.10
throughput	Min	18.10	18.11
	Max	18.80	18.79
total_xfer	Min	348.90	335.00
	Max	601.10	494.30
total_busy	Min	116.70	110.30
	Max	212.80	181.80
std_xfer	Min	15.30	15.26
	Max	20.67	19.25
std_busy	Min	6.67	6.55
	Max	9.47	9.12

Globally speaking, the IP distributions of configuration C1 present a better performance than configuration C0. The first parameter to point out is the number of used links. In C1 there are IP distributions that used 24 links while in C0 the minimum number of used links is 26. The reason for this difference lies in the location of the SDRAM memory (VI). In C0 the memory is placed in a corner and therefore there are only two nodes (01 and 10) that have direct access to it. Consequently, the path of the packets arriving from remaining nodes must obligatory include either node 01 or node 10. As a result, the total amount of wait time (*total\_busy*) is higher in C0 than in C1. Due to the lower number of links used in C1, the total amount of transmission time (*total\_xfer*) in C1 is less than in C0. Despite all of this, the differences of maximum latency and throughput between C0 and C1 are not noteworthy.

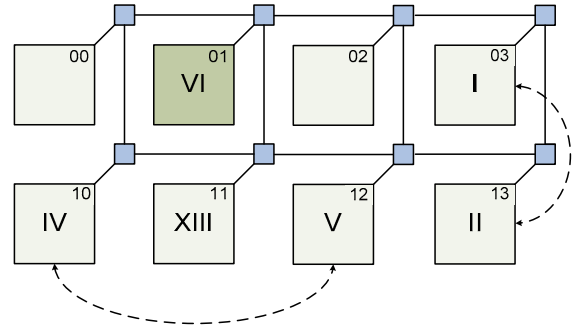
Among all the IP distributions generated in C1, the topologies that achieve better results in terms of less *total\_xfer*, less *total\_busy*, less *std\_xfer* and less *std\_busy* are represented in Fig. 6. In order to obtain the lowest amount of transmission time in the network, the two IPs with the highest traffic rate should be placed close to the memory. In particular, the optimal positions are the node 02 for the 3D Graphics Processor (V) and the node 11 for the Upsampling Unit (XIII). Nevertheless, the lowest variation of charge among the links is obtained when the 3D Graphics Processor (V) is placed in the opposite corner of the network. As far as designing a system with the lowest *total\_busy* rate, the distribution of the IPs should follow the pattern shown in Fig. 6 (b), in which the IPs with the lowest rate should be located opposite to the SDRAM memory (VI).

## 6.3 3x3 NoC analysis with two memories

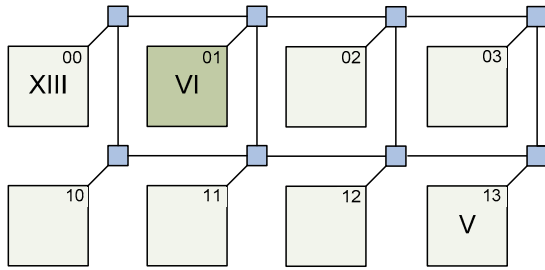
The results for the analysis of the topologies with two memories are summarized in Table 3. Contrary to the previous study, a considerable amount of IP distributions does not fulfil the bandwidth requirements, although the characteristics and properties of the elements were the same as in the starting configuration.



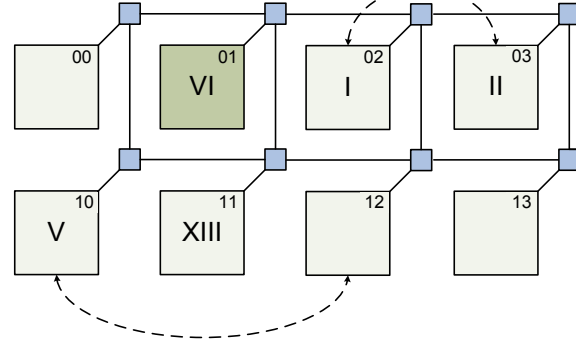
(a) *total\_xfer*



(b) *total\_busy*



(c) *std\_xfer*



(d) *std\_busy*

Fig. 6. Best IP distribution patterns for obtaining (a) less *total\_xfer*, (b) less *total\_busy*, (c) less *std\_xfer* and (d) less *std\_busy*.

Table 3. Results of the 3x3 NoC Analysis with Two Memories

		C2	C3	C4	C5	C6	C7	C8	C9
used links	Min	35	35	35	35	35	35	35	35
	Max	40	40	40	42	42	41	42	40
max_lat ( $\mu$ s)	Min	0.36	0.35	0.34	0.41	0.34	0.34	0.34	0.36
	Max	0.86	0.80	0.70	12.60	28.30	0.78	0.82	0.91
throughput	Min	39.87	39.88	39.67	40.12	39.66	39.87	41.81	39.66
	Max	54.76	56.39	53.47	57.54	111.57	54.26	61.12	55.06
total_xfer	Min	734.90	670.90	640.70	697.70	656.00	684.00	731.90	655.30
	Max	897.50	1090.60	928.20	891.20	919.20	961.60	1056.60	983.60
total_busy	Min	301.40	277.30	280.40	307.00	277.30	284.60	296.70	278.30
	Max	512.20	501.40	503.20	491.30	537.40	523.00	597.50	509.20
std_xfer	Min	15.90	15.95	16.09	16.45	14.60	15.50	14.73	15.32
	Max	19.07	20.16	19.23	20.02	18.87	19.52	21.82	19.75
std_busy	Min	8.32	8.07	8.39	9.06	8.14	8.43	8.93	8.13
	Max	13.58	13.72	14.36	15.96	14.11	13.34	15.57	13.84

After having analyzed all possible combinations of all configurations, there was no IP distribution that uses less than 35 links for transmitting packets and in some cases all available links in the network were used as in configurations C5, C6 and C8. As far as maximum latency is concerned, configurations C4, C6, C7 and C8 have the less value, while the best IP distribution of C5 reaches the highest minimum value for the peak latency. Referring to throughput and *total\_xfer*, C4, C6 and C7 achieve the best results in both categories. On the other side are C2, C5 and C8 with higher values for *total\_busy* as well.

According to the results of Table 3, the eight configurations under study can be classified into three groups. On the one hand, configurations C4, C6 and C9 show the best global values (G1). On the other hand, C2, C5 and C8 can be grouped together because of the bad results in almost all categories (G2). In the last group, C3 and C7 do not show whether good not bad results and therefore are left far behind (G3).

The reasons for the differences among the groups are extremely related to the location of the memories. The configurations belonging to the group G1 have the highest number of direct connections between initiators and targets. So, nodes 00, 11 and 20 in C4 can access with just one hop to the SDRAM (VI), which is located in node 10. In addition, nodes 02, 11, 22 have also direct access to the other memory (X) located in node 12. The same is found in configurations C6 and C9.

Nevertheless, the group formed by the configurations with worst results does not show the same number of direct connections. In this sense, C2 has four direct connections, but C5 has five and C8 just three. From all three configurations, the most amazing aspect to enhance is the bad performance of C5, although the number of initiators with direct access is quite high. However, a detailed analysis of the configuration reveals that the reasons for the bad results lie in the location of the memories together with the routing strategy. Although there are five IPs with direct access to the targets, the SRAM2 (X) is located next to the SDRAM (VI). As a result, all packets arriving from nodes of the last row (20, 21 and 22) must be transmitted across the switch of the SRAM2 whether they are addressed to the SDRAM (VI) or the SRAM2 (X). For this reason, the *total\_busy* value together with the *std\_busy* rate and the minimum peak latency as well are the worst results of all configurations. In the case of the configuration C8, the bad performance is caused due to the low number of direct connections, while in C2 the target IPs are located far away and therefore packets must follow a long path until they reach their destination.

## 7. CONCLUSION AND ON GOING WORK

Memory accesses have a huge influence in the performance of NoC based multimedia designs. In this paper, a deep study of all IP distributions in regular 2D-Mesh NoCs with different number of memories has been presented. The results inferred from the experiments show how the performance of a multimedia system can be improved if memories are properly located within a NoC. Furthermore, the present work has served to validate the use of Arteris NoCexplorer for simulating and modelling complex NoC based designs. In addition, a methodology for determining the best IP distribution in terms of latency and throughput has been presented and its feasibility has been demonstrated.

The immediately goals consist of widen the range of the study to other regular topologies like 2D-Torus or 2D-Folded Torus and irregular topologies as well. Moreover, other routing algorithms are going to be used, including adaptive ones. It is also mandatory to create a figure of merit formed by the combination of all measured parameters in order to determine which topology better adapts to the requirements of different situations.

## ACKNOWLEDGMENTS

This work was partially supported by the Spanish Ministry of Science and Innovation under project DR. SIMON (Dynamic Reconfigurability for Scalability in Multimedia Oriented Networks) TEC2008-06846-C02-02/TEC.

## REFERENCES

- [1] International Technology Roadmap for Semiconductors, "System Drivers", *International Technology Roadmap for Semiconductors*, 2005 Edition. [Online]. Available: <http://www.itrs.net/Links/2005ITRS/>

- [2] S. Murali and G. De Micheli, "SUNMAP: A Tool for Automatic Topology Selection and Generation for NoCs", *41<sup>st</sup> Design Automation Conference (DAC'04)*, San Diego, CA, USA, 7-11 June 2004, pp. 914-919.
- [3] D. Kim, K. Kim, J.-Y. Kim, S.-J. Lee and H.-J. Yoo, "Solutions for Real Chip Implementation Issues of NoC and Their Application to Memory-Centric NoC", *1<sup>st</sup> International Symposium on Networks-on-Chip*, Princeton, New Jersey, USA, 7-9 May 2007, pp. 30-39.
- [4] R. Moraveji, H. Sarbazi-Azad and M. Abbaspour, "Optimal Placement of Frequently Accessed IPs in Mesh NoCs", in *Advances in Computer Systems Architecture*, Springer, 2007, pp. 126-138.
- [5] U. Y. Ogras, J. Hu and R. Marculescu, "Key Research Problems in NoC Design: A Holistic Perspective", *3<sup>rd</sup> IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES + ISSS'05)*, Jersey City, New Jersey, USA, 18-21 September 2005, pp. 69-74.
- [6] E. B. van der Tol and E. G. T. Jaspers, "Mapping of MPEG-4 decoding on a flexible architecture platform", in *Media Processors 2002*, pp. 1-13.
- [7] Arteris, The Network-on-Chip Company. <http://www.arteris.com>
- [8] S. Wuytack, F. Catthoor, G. De Jong, and H. J. De Man, "Minimizing the Required Memory Bandwidth in VLSI System Realizations", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 7, n° 4, pp. 433-441, December 1999.
- [9] S. Murali and G. De Micheli, "Bandwidth-Constrained Mapping of Cores onto NoC Architectures", *Design, Automation and Test in Europe Conference and Exhibition 2004*, Paris, France, 16-20 February 2004, vol. 2, pp. 896-901.
- [10] D. Bertozzi, et al., "NoC Synthesis Flow for Customized Domain Specific Multiprocessor Systems-on-Chip", *IEEE Transactions on Parallel and Distributed Systems*, vol. 16, n° 2, pp. 113-129, February 2005.
- [11] M. Kim, D. Kim and G. E. Sobelman, "MPEG-4 Performance Analysis for a CDMA Network-on-Chip", *2005 International Conference on Communications, Circuits and Systems*, Hkust, Hong Kong, China, 27-30 May 2005, vol. 1, pp. 493-496.
- [12] M. B. Stensgaard and J. Sparsø, "ReNoC: A Network-on-Chip Architecture with Reconfigurable Topology", *2<sup>nd</sup> IEEE International Symposium on Networks-on-Chip*, Newcastle, United Kingdom, 7-11 April 2008, pp. 55-64.