Réseau sur puce sans buffer pour les applications de codage vidéo : étude de cas H.264/AVC

Buffer-less Network-on-Chip for video encoding applications: case study H.264/AVC

Hichem Mayache*[1 3], Salah Toumi[1], El Bey Bourennane[2] & Atef Benhaoues [1]

[1] *LERICA Laboratory, University of Badji-Mokhtar, Po Box 12, Annaba, 23000, Algeria.*
[2] *LE2I Laboratory, University of Bourgogne, Dijon – France.*
[3] *LABGET Laboratory, University of Larbi Tebessi, Tebessa, 12000, Algeria.*

Soumis le : 14/12/2017 *Révisé le : 09 /06/2018* *Accepté le : 11/06/2018*

ملخص:

برزت الشبكات على الرقاقة كحل للربط على رقاقة جديد قابل للتطوير، مرن وقابل لإعادة الاستخدام لتطبيقات النظام على رقاقة. مستوحاة من شبكات الكمبيوتر، فهي توفر حلا وسطا جيدا بين التكلفة والأداء. في هذا العمل، نقدم شبكة على رقاقة دون مخزن مؤقت لتطبيقات تشفير الفيديو. يستفيد العمل المقترح من المرونة وقابلية التوسع من أجل اقتراح حل اتصال أفضل لمشفرات الفيديو. الفكرة الرئيسية هي استخدام شبكة على رقاقة دون مخزن مؤقت لربط مكونات مشفر الفيديو H.264/AVC. يتم وصف الهندسة التركيبية المقترحة باستخدام لغة وصف الأجهزة على منصة FPGA فيرتكس-7- XC7V2000T. بعد ذلك، تتم مقارنة النتائج مع أعمال مماثلة باستخدام حلول الربط البيني الأخرى ويظهر التنفيذ أن الهندسة التركيبية المقترحة تقدم أقصى تردد تشغيل أعلى بتكلفة مكونات منطقية أقل.

الكلمات المفتاحية : *شبكة على رقاقة - النظام على رقاقة - تطبيق تشفير الفيديو - مشفر الفيديو - AVC / H.264 - لغة وصف الأجهزة.*

Résumé :

Les réseaux sur puce ont émergé comme une solution d'interconnexion évolutive, flexible et réutilisable pour les applications des systèmes sur puce. Inspirés des réseaux informatiques, ils offrent un bon compromis entre le coût et la performance. Dans ce travail, nous présentons un réseau sur puce sans buffer pour les applications de codage vidéo. L'idée principale est d'utiliser un réseau sur puce sans buffer pour connecter les IPs composant l'encodeur vidéo H.264/AVC. L'architecture proposée est décrite en utilisant le langage de description matérielle sur une plateforme FPGA Virtex-7 XC7V2000T. Les résultats sont comparés à des travaux similaires utilisant d'autres solutions d'interconnexion et l'implémentation montre que l'architecture proposée offre une fréquence de fonctionnement maximale plus élevée avec une consommation en ressources logiques plus faible.

Mots clés : *Réseaux sur puce – systèmes sur puce – application de codage vidéo – encodeur H.264/AVC.*

Abstract:

Networks on chip have emerged as new scalable, flexible and reusable On-Chip Interconnect solution for System-on-Chip applications. Inspired from computer networks, they offer a good compromise between cost and performance. In this work, we present a Buffer-less Network-on-Chip for video encoding applications. The main idea is to use a buffer-less NoC to interconnect the IPs composing the H.264/AVC video encoder. The proposed architecture is described using Hardware Description Language onto Virtex-7 XC7V2000T FPGA platform. Thereafter, results are compared to similar works using other interconnect solutions and the implementation shows that the proposed architecture offers a higher maximum frequency operating with a lower logic cost.

Keywords: *NoC, SoC, Video Encoder Application, H.264/AVC encoder.*

[*] *corresponding author :h.mayache@live.fr,*

1. Introduction

The design process of digital systems has evolved drastically with the emergence of new technologies such as FPGA (Field Programmable Logic Array) to take a prominent place in the conception of embedded systems. These technologies offer hardware architectures as versatile as software, which is a good compromise between utilization flexibility and processing power. Such a compromise is even more suitable for embedded systems intended for different applications in various areas. Within this background, FPGA technology will make it possible to design and upgrade real-time computing systems, while optimizing hardware resources. Alongside, technological progress continues to increase the integration density of transistors in integrated circuits (ICs) owing to the growing complexity of applications and functionalities. This requires the development of high performance embedded systems that can satisfy recent computing requirements [1]. Furthermore, the desire to reduce the time-to-market puts more pressure on the designers to achieve new competitive products. Manifestly, shared-bus-based architectures have been considered for a long as the most attractive solution because of their low logic cost and implementation simplicity [2].However, this solution failed to provide high performances with a large number of connected IPs(Intellectual Properties). The bandwidth of a bus is shared by all the connected IPs that makes it insufficient. Moreover, to connect a large number of IP cores, length of the bus should be important which implies significant power consumption. In addition, for such an architecture it is possible to make only one reading (or writing) operation at a time. All these limitations have led designers to provide alternatives. Out of this, Network-on-Chip (NoC) has emerged as a promising solution to overcome some of the bus-based architectures limitations [3] [4]. It solves integration complexity problems by providing scalable, reusable, robust and low power consumption on-chip communication solution with a larger bandwidth [5].

Similarly, the Network on Chip is composed of network links and routers (R). These latter are connected to processing elements (PEs) via a network interface (NIC). Each router has five ports for the four directions (east, west, south, and north) and the local PE. In traditional NoCs, each router is composed of Buffer, crossbar and arbiter, which leads to significant area, power and latency overheads. Obviously, the need of a high performance On-Chip communication system resides in the complexity of signal processing applications such as video compression standards: MPEG 2, MPEG 4, H.264 and many others [6] [7]. These standards have to satisfy requirements of real-time and very high definition systems, which make them more and more sophisticated. H.264/Advanced Video Coding is a video compression standard motion-compensation-based block-oriented used for video content: recording, compression, and distribution. Up to no, several H.264/AVC hardware implementation solutions have been proposed even they are not always well adapted for optimizing the functioning [8].Some works [8] [9] [10] [11] [12] [13] [14] focused on different goals such as: power consumption, maximum frequency operating, data pipelining, parallelism etc.… by proposing intra-coding and inter-coding blocks. These implementations often use a cascade interconnection, which is a real disadvantage for data parallelism, as some modules must be duplicated. Some authors have exploited similarity points between redundant blocks to reduce the FPGA cost by connecting the H.264/AVC cores through a network on chip [15].Although such a solution turned out faster with equivalent cost, there remains an optimization margin in terms of logical resources. Moreover, the use of buffer-based network on chip to interconnect the H.264/AVC modules is not as effective as it might sound, because video coding systems require an on-chip interconnection that guarantees Quality of Service (QoS) and provide communication system well suited for video applications.

In this paper, we propose a new solution based on NoC architecture for video applications offering bandwidth guarantees such as point-to-point connection, but also the flexibility and the scalability of a network on chip. The main originality of our work is to adapt a network on chip through a circuit switching mechanism in order to reduce latency. In addition, this approach aims to ensure high operating frequency, optimize FPGA area utilization and reduce power consumption by removing the router buffers.

The remainder of the paper is structured as follows. Section 2 introduces briefly the fundamentals of networks on chip. Insection 3, we will detail the principles of a video encoder and more precisely the case of the H.264/AVC standard. Subsequently we will examine the different hardware implementation approaches of the H.264/AVC. Section 4 presents the proposed buffer-less NoC adapted for video applications while detailing the contribution of this approach compared to other existing solutions. Synthesis results of the implementation on Xilinx Virtex-7 FPGA prototyping platform are exposed and discussed in section 5. A comparison and an analysis between the proposed and previous works are also performed in this section. Finally, section 6 reviews the proposed work, gives the conclusion and future works.

2. Network on Chip

During the last decades, the integrated circuits development has been base on the principle of the reuse of previously designed intellectual property (IP) cores [16] [17]. This concept was intended to design higher performances systems-on-chip, with shorter time to market. Bus-based architecture has been considered for a long as the most adapted interconnection solution for SoCs [18].Unfortunately, this solution loses its usefulness with the increase of the connected IPs, which makes it unable to deal the requirements of recent applications.

Performance, power consumption, latency and real-time constraints have forced designers to propose a new efficient on-chip interconnect solution. Therefore in the 2000s some authors developed a new paradigm baptized Network on Chip [3][4], inspired from computer network principles. Furthermore, it has been demonstrated by other works that for an important number of connected cores and large data flow, network-on-chip based systems achieve better performances compared to bus-based systems [19].

Network on chip is a non-centralized on-chip interconnection model made of: set of elements called nodes (routers) and wires to make the physical connection between the router sand the intellectual properties (IPs) [20]. As IP cores are different and may be memories, processor elements (PEs) or CPUs they will have different communication protocol sand the access to the network will be made through a block named network interface (NI). Network interface constitutes the boundary between calculation and communication; it enables to get the best of the reuse concept by standardizing flowing packets on the network (Fig. 1).

Since they emerged, networks on chip stand out by their wealth of configuration starting with the topology, which is the placement of routers on the network. Routers may be the same on the entire network when mapped on regular topology such as mesh 2D, mesh 3D, tree topology or may have differences in their ports if the topology used is irregular [21].

Globally, in generic architectures found in literatures, the elementary part of a NoC is the router. It is an entity having a number of input and output ports, with a crossbar that allows the route of incoming packets to reach their destination, and a specific port to connect the IP core. However, in more detail routers must perform several functionalities: routing, arbitration, switching, buffering and the flow control [21].

Originally, packets must travel on the network in a well-defined way; such functionality is known as the switching mechanism. Primarily, each packet is segmented in three main parts: the header, the payload and the tail. Header part includes destination and control information, the payload part composed of data to send to the destination IP core, and finally the tail part points the end of the packet. When performance guarantees are required for the connection, the necessary communication resources along the way are reserved from source to destination; this mechanism is called the circuit switching. On the other hand, the connection takes longer to be established. An alternative approach consists of sending the whole packet once the header flit is transmitted through storage and transmission variants: store and forward strategy, virtual cut through strategy and wormhole strategy [22] [23].

Otherwise, packets (or flits) displacement is defined by the flow control policy in a way to avoid deadlock situations or to optimize the communication by minimizing logic cost or reducing the latency. The flow control is usually distributed where decisions are taken at router level. It deals with choosing the right compromise when designing the NoC to foster one criterion among a set of performance metrics: resources management, low power consumption, high operating frequency etc.… defined as quality-of-service (QoS). Certainly, the flow control has been revolutionized by the virtual channel concept that led to overcome the deadlock. The originality of VCs can be summarized in sharing a physical channel by multiple channels with dedicated buffers separated logically [21] [23].

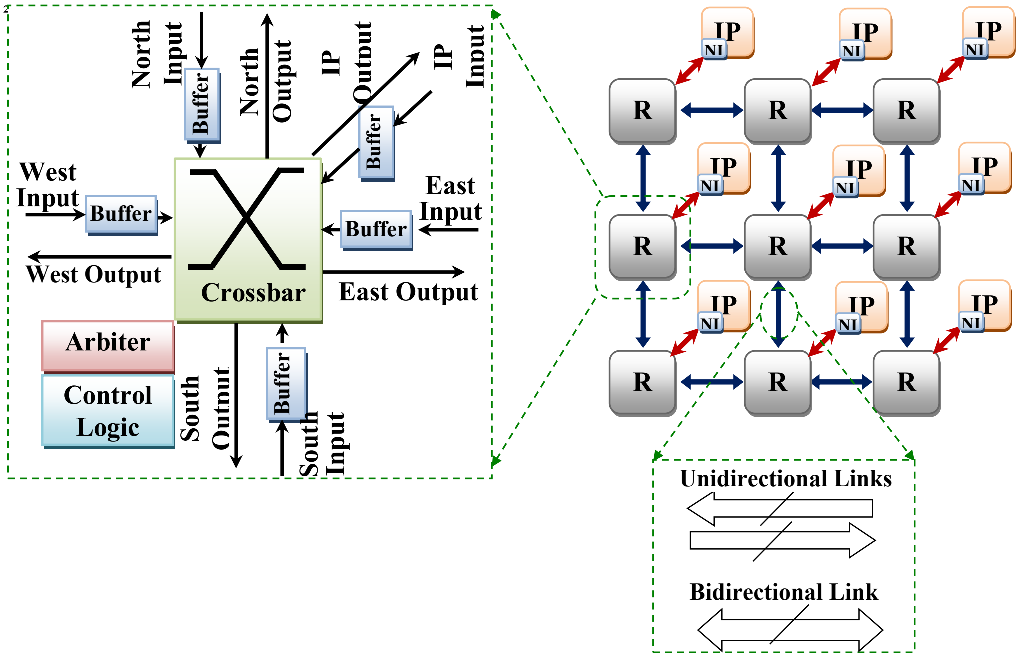

Figure 1: Architecture of Standard Mesh (3x3) Network on Chip

Packets on the network are often confronted to congestion, which imposes buffers in the router architecture to store data temporarily. Furthermore, the placement, the size and the number of buffers in the router affect the performance and the cost of the entire network. Storage policies are different amongst: a common buffer for all the ports to minimize the logic cost at the expense of operating speed, buffers for each input port which is the most usual solution or a less effective which dedicates buffer for each output port [21] [23].

The routing algorithm is the logic functionality that selects the optimal output port corresponding to the destination address located in the header for each incoming packet. Similarly, to computer networks, there is a wide variety of routing algorithms classified into adaptive or deterministic routing. XY and source routing are the most known and used in the deterministic routing class where the packets having the same destination follow identical track. On the other hand, there is an adaptive variant of XY routing and many other routing where the taken path for an identical destination is not necessarily the same [23][24].

Once the routing unit calculates the output port, multiple inputs may request the same output port concurrently; here fore, there is great need for arbitration feature to regulate the assignment of output ports. Many arbitration possibilities are available: centralized arbitration or distributed arbitration [23] [25].

In practical, the evaluation of NoC-based system performances is often known after hardware implementation as it shows how well the communication can be predictable providing service guarantees. For instance, video applications operating in real time need more performance guaranties rather than a prediction about the behavior of the on-chip interconnection. Therefore, the type of networks utilized in such applications is guaranteed service (GS NoCs). For other applications whether there is no constraints and just the accomplishment of the communication is required a best effort network (BE NoC) makes the deal [25].

3. Video coding applications

Nowadays video coding application is one of the areas requiring high performance on-chip communication system with a very high bit rate, especially that such applications may operate in:ultra-high definition television (UHDTV), digital video broadcasting including all its approaches (DVB-T, DVB-C, and DVB-HS etc.), high definition video surveillance or high definition video-gaming [26] [27]. To meet these requirements, researchers and engineers have presented solutions to implement codec's or to optimize the various processing modules in codec's [28].Compression begins by the intra-coding to remove spatial redundancy and/or the inter-coding to remove temporal redundancy, followed by a transformation and quantification to concentrate the video signal energy in defined regions, and ends with the entropy encoding [29].

Given the high frame rate in high definition video, motion estimation aims to exploit the similarity between two consecutive frames in order to reduce temporal redundancy. Previous video coding standards were based on Fixed Block-Size Motion Estimation (FBSME), which had the disadvantage of spending the same effort to estimate the motion of static or dynamic objects. Furthermore, such an approach gives poor performances in the case of two objects moving in two different directions in a block. On the other hand, H.264/AVC encoder is based on Variable Block-Size Motion Estimation (VBSME), which enables to have higher efficiency by using larger bloc size for static objects and smaller bloc size for the moving ones. H.264/AVC variable block size motion estimation is performed by: Integer Motion Estimation (IME) and Frame Motion Estimation (FME). Firstly, IME calculates the Motion Vector Predictor (MVP) in the current MB by performing the mean of MVs of the left, the top, and the right-top (or left-top) MBs. The center point of the search window is pointed by MVP. Thereafter, a motion search is carried out within the search window to find Integer Motion Vector (IMV) for each of the 41 blocks. Afterwards, FME interpolates half-pixels by six-tap filter and then quarter-pixels by two-tap filter, then carries out motion search around the refinement center and further refines IMVs into Fractional Motion Vectors (FMVs) of quarter-pixel precision. Fractional Motion Estimation educes the bit-rate while improving the video quality. However, motion estimation complexity makes it costly in computing time, hence the need to use a hardware accelerator for FME [27].After IME and FME, Motion Compensation constitutes the next stage in H.264/AVC inter-frame prediction. Motion compensation also stands out by high computational complexity. Recent designs for motion compensation try to get around that and focus on increasing performance while reducing memory traffic.

The reverse chain contains three inverse modules (the entropy decoding, the inverse transformation and quantification and the decoding module). Each standard utilize different tools compared to the previous standards to obtain improved compression performance [30].

The video coding recommendation H.264 of ITU-T (also known as MPEG-4 part 10 of ISO/IEC [31]) is one of the latest complete coding standards [32]. The H.264/AVC achieves a significant improvement in coding efficiency when compared to other coding methods. It can save as much as 25% to 45% and 50% to 70% of bitrate when compared to MPEG-4 and MPEG-2 respectively [33]. As shown in figure 2, the encoding system is composed of the forward path (encoding) and the inverse path (decoding). The forward path predicts each macro block using intra-prediction or inter-prediction; it also transforms and quantizes (TQ) the residual, then it forwards the result to the entropy encoder module. Finally, it generates the output packets in the NAL module. The inverse path involves the reconstruction of the macro block from the previously transformed data by utilizing the inverse transform and quantization (ITQ), the reconstruction module and the deblocking filter [34].

The intra-coding chain contains several elementary modules: first, the intra-prediction module calculates the residual block following two strategies (Intra4×4 and Intra16×16), these residuals are further processed by the forward transform, quantization, inverse quantization, inverse transform, and finally reconstructed as a reference for the processing of next macro blocks [8]. Based on figure 2, many modules are connected using a point-to-point link. The deblocking filter used in H.264/AVC is more complex than other filter used in previous video compression standards [35]. First, this filter is highly adaptive and applied to each edge of all the 4x4 luma and the 2x2chroma blocks in a 16x16 macro block of pixels. Second, it can update three pixels in each direction in which the filtering takes place. Third, in order to decide whether the DBF will be applied to an edge, the related pixels in the current and neighboring blocks must be read from memory and processed [36].

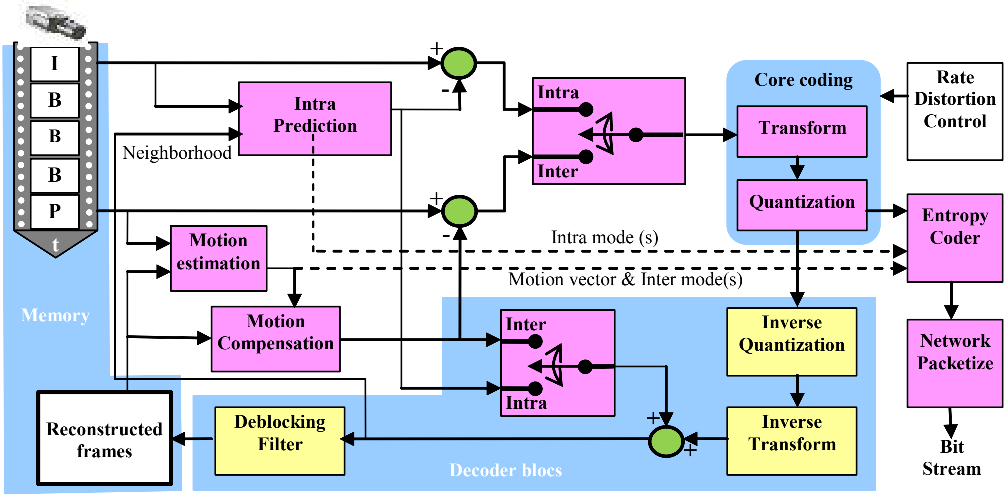

Figure 2: Basic coding architecture for H.264 encoder [31]

4. The proposed buffer-less NoC for Video applications

4.1. State of art

The evolution of video applications gave rise to new real-time systems, more complex but also with more requirements that have prompted researchers to propose solutions based on hardware implementations either partial or total of H.264/AVC codec. In recent years, several hardware implementations (hardware IPs) have been proposed for the H.264/AVC modules [8] [33] [34] [36] [37]. Different communication strategies have been used to implement these IPs on FPGA devices in order to ensure real-time processing. Generally, point-to-point solution has been for a long the most adopted interconnection approach [8][28]. The shared buses are also used for connecting IPs [29] with an embedded processor. The processor is used generally for task management.

In the H.264 encoder, we note that the elementary IPs are organized into distinguished groups. IP cores in the same group often communicate with each other but do not communicate with other groups IPs, except through a single IP. In this sense, a NoC can be used for the intra-coding chain IPs and another NoC for the deblocking filter IPs. In some recent studies, several authors have demonstrated the utility of using NoC structure in many applications [4] [38] [39].

In [40] the authors have proposed hardware architecture for the intra-coding module used in the H.264 encoder. Using a data bus (32-bits bus size), a data-Parallelism based hardware architectures are proposed for the intra-prediction, the integer transform, the quantization, the inverse integer transform, the inverse quantization, and the mode decision modules. In [8] the authors have reproduced these architectures using a 128-bits bus size in order to ensure good memory management. As mentioned in figure 3 [8], the intra-prediction module is decomposed in two modules (the Intra4×4 and the Intra16×16) using both the same chain of modules (the integer transform, the quantization, the inverse integer transform and the inverse quantization). In [8] and [40], the authors use a point-to-point interconnection with multiplexers/de-multiplexers in multi-input/output cases. Some IPs (T/Q/Q^{-1} and T^{-1}) are used several times to ensure rapid processing and to avoid the use of multiplexers/de-multiplexers. Duplicating the H264/AVC IP modules increases significantly the used FPGA resources. To overcome the shared-bus limitations, and block duplication, another approach exploiting networks on chip has been implemented [41]. The idea behind this solution is to get the best of IP reuse principle. This approach allows the exploitation of similarities between the intra-coding chain and the inter-coding chain by removing the redundant blocs in order to reduce the FPGA resources.

Thereby, this solution is not that efficient and the resources optimization is not important because the used NoC in this work is buffer-based. More precisely, the use of buffers in the router increases logical cost and decreases the operating frequency, which constitutes the major drawback of this approach.

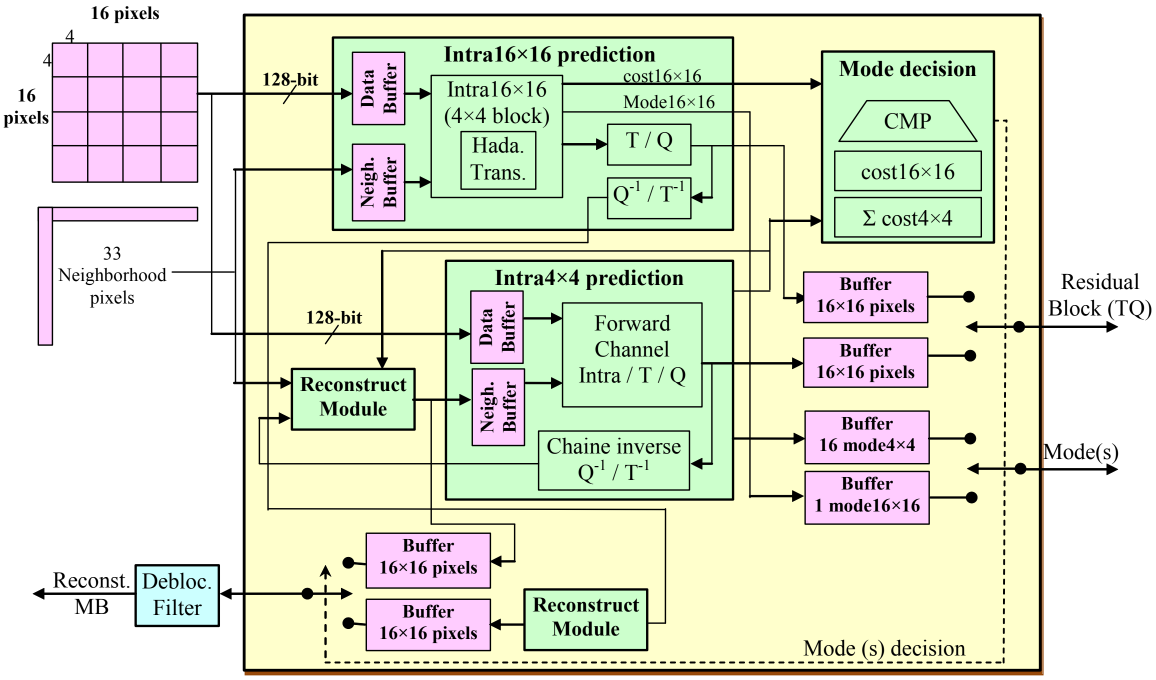

Figure 3: Hardware architecture for the Intra-coding module proposed in [8].

In this paper, we propose to use a buffer-less network-on-chip adapted for video coding applications. The main originality of the proposed work is to use a specific router that combines the strengths of a switch and a generic router. In what follows we will go into further detail on our proposed architecture.

4.2. The proposed architecture

Traditionally, networks on chip have often been designed using router architectures containing buffers in order to store data flowing on the NoC, especially when these architectures have been enhanced by the advent of virtual channel concept. This concept has allowed a better management of congestions while overcoming deadlock problem. In fact, NoC bandwidth efficiency is improved by these buffers because they minimize the rate of lost or crushed packets in the network.

Conceptually, removing buffers may seem like a bold choice as we expect a higher latency with a lower bandwidth. However, these buffers also have significant drawbacks in terms of FPGA resources cost, static and dynamic power consumption and design complexity. Accordingly, some authors have tried to reduce their impact by working on the concept of low-cost router [1] [42] [43], when other works explored how to remove totally buffers in router architectures [44] [45] [46]. Furthermore, it has been shown that a 60% buffers reduction may save 40% of power consumption in the best cases without a significative alteration of the system performance and buffer elimination leads to considerable latency reduction [44]. For these reasons, we chose to adapt this approach to video encoder (case study: H.264/AVC) in order to provide more efficient on-chip interconnection system. Figure 4 shows the proposed network on chip architecture based on buffer-less router (BLR).As presented in the figure 4, we made the choice of a 2D Mesh architecture for the NoC based interconnection solution. Such an architecture is characterized by its regularity and ease of implementation. Our proposed buffer-less router (BLR) constitutes the basic element of the network (Fig. 4). It is simply composed of crossbar and table of commutation. The role of the switching table is to assign the output port of each packet according to its label.The particularity of our proposed buffer-less routing is the use of a switching technique instead of using conventional XY routing algorithm. More precisely, the commutation mechanism relies on labels added in the header packet. This label is used by each buffer-less router (BLR) to send the packets to their destination address.

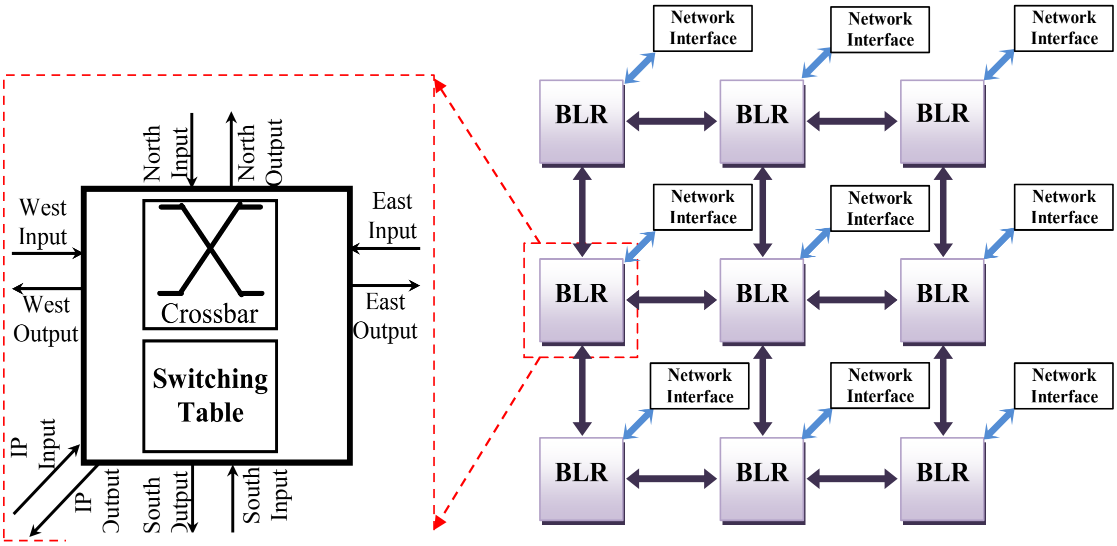

Figure 4: The Buffer-less Network on Chip proposed architecture

Given that the On-Chip interconnection may constitute a bottleneck for the system performance, our work has been focused on the On-Chip interconnect solution for video encoding application,H.264/AVC encoder in our case. The originality of our proposed is to take advantage of the strengths of Networks-on-Chip while removing the buffer drawback. Thereby, we have considered H.264/AVC as a system on which we have not acted, and we simply proposed an improvement the interconnection between its different blocks.

Our proposed architecture enables to interconnect the H.264/AVC modules through network interface as shown in figure 5. Let us take as an example the case of the entire intra-coding chain. Firstly, a microprocessor performs the initial setup of the network by loading the switching tables of each router according to the different scenarios. Secondly, the packets enter the network intended to the intra-coding bloc, through the network interface (NI) which assigns a first label. Once the treatment is performed, the processed data are returned with a new label, which will allow the access to the DCT and quantification bloc. This label change is managed by the network interface, and the connected router will check its switching table and select the corresponding output port.

Similarly, the parquets will continue to travel on the network until the end of the processing intra chain. It should be noted that the first label awarded at the beginning refers to the performed scenario. Such an approach presents a performance advantage because of the considerable gain in processing time consequently to the elimination of writing and reading times on the buffers and the switching table speed. Furthermore, deleting buffers releases much more resources and makes the solution more interesting.

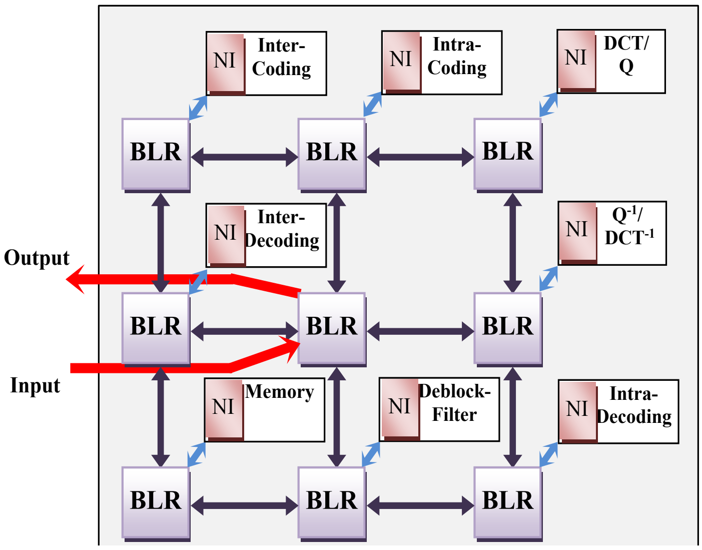

Figure 5. H264/AVC Intra Coding Chain mapping using the proposed Buffer-less NoC

5. Simulation results and discussion

The hardware implementation of H264/AVC components and the buffer-less network on chip dedicated for video application shas been described in hardware description language (VHDL). We use Xilinx ISE 14.1 Design Suite software tool to: synthesize the architectures, perform the timing analysis and visualize RTL schematics. Multi-language HDL simulation environment Model Sim 9.2 has been used for simulation and debugging. The design of the whole architecture of the H264/AVC elements interconnected by the proposed buffer-less NoC has been achieved with the embedded development kit EDK Design tool. In sum, the entire system has been implemented on Virtex-7 XC7V2000T FPGA platform.

The evaluation of our implemented buffer-less NoC for video applications is carried out after generating the synthesis report for 3x3 mesh topology, store and forward switching, handshake signal for the flow control, XY routing and round robin arbitration. The performance analysis in terms of slice LUTs, slice registers and maximum operating frequency is summarized in table 1 for different network dimensions. By focusing on these results, it seems clear that the proposed solution is easy to implement by the relatively low logic cost. Furthermore, it appears from these results that this solution is quite effective especially for high performance applications such as video encoding (or decoding) through its fairly high frequency. Thus, we can notice easily that a good balance between the low logic cost and the operating frequency is reached. In addition, it is also transparent from the comparison realized with the synthesis results of [41] (Tab. 1) that our proposed has reduced LUTs consumption by around 70% for the three NoC dimensions implemented. Likewise, our buffer-less NoC solution operates at 429.70 MHz, 427.56 MHz and 422.08 MHz for 2x2, 3x3 and 4x4 mesh NoC dimensions, which is a much higher maximum frequency with respect to the frequencies reaches by buffer-based NoC [41]. Indeed, our approach seems quite superior since it operates at a faster clock frequency this is mainly due to the suppression of buffers. The latter have an important logical cost and slow down the entire communication system frequency during writing (or reading or deleting) data.

Table 1. Implementation results on Virtex-7 XC7V2000T FPGA and comparison between the proposed buffer-less NoC with conventional NoC [41]

Performance metrics **Dimension the of mesh NoC**	**Proposed Buffer-less NoC**			**Conventional NoC [41]**		
	Number of slice registers	**Number of slice LUTs**	**Maximum Frequency**	**Number of slice registers**	**Number of slice LUTs**	**Maximum Frequency**
NoC2×2	78/2,443,200	285/1,221,600	429.70MHz	67/2,443,200	972/1,221,600	174.932MHz
NoC3×3	191/2,443,200	603/1,221,600	427.56MHz	127/2,443,200	2,154/1,221,600	172.828MHz
NoC4×4	296/2,443,200	1,127/1,221,600	422.08MHz	240/2,443,200	3,645/1,221,600	171.725MHz

In order to evaluate more objectively our buffer-less NoC we perform a comparison with previous work that has achieved a similar architecture but with a conventional network on chip (buffer-based NoC) [41]. To this end, we have ensured to implement the same intra-coding chain using the using the same IPs used by [41] then we lined up on the same metrics i.e. maximum operating frequency, logic cost and same packet size for the purpose to have common benchmarks. As in the work of [41], we decomposed the intra-prediction module into 4x4 and 16x16 intra prediction modules and in each sub-module we included: direct DCT module, direct quantization module, inverse quantization and inverse DCT modules as detailed in figure 5.Thereafter, we compared synthesis results of our implemented chain connected using the buffer-less NoC with those of the author [41]; this comparison is summarized in table 2. The examination of these implementation results of the intra-chainshows a slight optimization in FPGA resources up to 8.50 % with a significant improvement in the operating speed from 165.78 MHz to 197.61 MHz (19.20 %) which demonstrates the efficiency of our proposed.

Table 2: Comparison between synthesis results of H.264/AVC IPs connected: using a 3x3 Mesh conventional NoC on XUPV5 FPGA platform [41] and the proposed 3x3 Mesh buffer-less NoC onXC7V2000TFPGA platform

		Number of slice registers	**Number of slice LUTs**	**fully used LUT Slices %**	**Maximum Operating Frequency**
Synthesis result of H.264/AVC IPs connected by 3×3 mesh conventional NoC [41]	**3×3 mesh conventional NoC**	127	2,154	5.89%	172.83MHz
	Intra4×4 prediction	1,477	1,703	38.02%	278.43MHz
	Intra16×16 prediction	2,650	2,180	49.33%	282.03MHz
	DCT & Quantization	1,821	1,603	59.49%	594.11MHz
	Inverse Quantization & Inverse DCT	1,865	1,712	38.70%	394.06MHz
	Deblocking filter	6.702	8.890	29.20%	188.64MHz
	Implementation results ofthe IPs of the chain connected by a conventional 3x3 mesh NoC	**14,642**	**18,242**	**37.10%**	**165.78MHz**
Synthesis result of H.264/AVC IPs connected by the proposed 3×3 mesh NoC	**3×3 mesh buffer-less NoC**	191	603	57.53%	427.56MHz
	Intra4×4 prediction	1,477	1,703	44.51%	295.84MHz
	Intra16×16 prediction	2,650	2,180	54.69%	305.36MHz
	DCT & Quantization	1,821	1,603	63.21%	620.46MHz
	Inverse Quantization & Inverse DCT	1,865	1,712	42.18%	418.67MHz
	Deblocking filter	6.702	8.890	31.98%	207.94MHz
	Implementation results of the IPs of the chain connected by a buffer-less 3×3 mesh NoC	**14,706**	**16,691**	**49.01%**	**197.61MHz**

The last result table (Tab. 3) presents synthesis results of the based-Microblaze entire System implemented using the three different approaches:

- The H.264/AVC IPs are cascaded [41],
- The H.264/AVC IPs are connected by a conventional NoC [41],
- The proposed approach with H.264/AVC IPs cores connected by a buffer-less NoC.

It appears from these synthesis results (Tab. 3) that our approach is as effective as or better than the two other approaches [41] in terms of logical cost. However, it is understand able through the operating frequency that the proposed buffer-less on-chip interconnection is faster of about 27.49 % than the cascaded approach and about 19.20 % than the buffer-based NoC [41] with respect to these two other approaches. In addition, with such a reduction in FPGA resources and the abolition of buffers we expect a significant decrease of power consumption.

Table 3. Synthesis results of the based-Micro blaze System implemented for the H.264/AVC IPs using three different approaches

	Number of slice registers used	**Number of slice LUTs used**	**Maximum Frequency**
SoC for H.264/AVC IPs connected in cascade [41]	23,589	25,092	155MHz
SoC for H.264/AVC IPs using 3x3 Mesh conventional NoC [41]	21,768	23,112	165.78MHz
SoC for H.264/AVC IPs using the proposed 3x3 Mesh buffer-less NoC	**21,835**	**22,559**	**197.61MHz**

6.Conclusion

This paper has proposed a new buffer-less network on chip adapted for video applications. The main originality of our proposed solution is to take advantage of this new on-chip communication aradigm, which is Networks-on-Chip, while adapting it to the field of video applications. Video encoding (decoding) applications is one of the high-performance applications, which make the NoC solution obvious especially as it has proven itself and has emerged by its performances. The idea was to eliminate buffers from the router architecture to gain in FPGA area, but also in maximum operating frequency.

In order to prove the efficiency of our solution, we implemented the3x3 mesh buffer-less NoC as well as the H.264/AVC elementary modules on Xilinx Virtex7 platform FPGA in hardware description language VHDL. Afterward, the set has been connected to the Micro Blaze hardware platform. An initial comparison between the proposed buffer-less NoC and exiting works was carried out and it gave encouraging results. Indeed the synthesis results demonstrated the improvement, compared to buffer-based NoC solution, by reducing the logic cost by more than 65 %, while increasing also the operating speed from 175 MHz to 430 MHz (in the best case). Finally, our approach demonstrated its adaptation to video applications that makes it able to be improved before applying it for newest standards such as HEVC standard.

References

[1] C. A. Nicopoulos, D. Park, J. Kim, N. Vijaykrishnan, M. S. Yousif, C. R. Das:ViChaR: A Dynamic Virtual Channel Regulator for Network-on-Chip Routers. The 39th Annual IEEE/ACM International Symposium on Microarchitecture 2006 (MICRO'06), December 9-13, 2006, Orlando, Florida, USA.

[2] T. C. Xu, P. Liljeberg, H. Tenhunen : A Study of Through Silicon Via Impact to 3D Network-on- Chip Design, International Conference on Electronics and Information Engineering (ICEIE 2010), 1-3 August 2010, Kyoto, JAPAN.

[3] W. J. Dally and B. Towles :Route packets, not wires: On-chip interconnection networks. In Proceedings of the 38th Conference on Design Automation, June 22, 2001, Las Vegas, NV, USA, pp. 684–689.

[4] L. Benini and G. De Micheli :Networks on chips : a new SoC paradigm. Computer, vol. 35, n°1, 2002, pp.70–78.

[5] P. P. Pande, C. Grecu, M. Jones, A. Ivanov, and R. Saleh : Performance evaluation and design trade-offs for network-on-chip interconnect architectures. IEEE Transaction on Computers, vol.54, n°8, 2005,pp. 1025–1040.

[6] T. Sikora :The MPEG-4 video standard verification model. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 7, n° 1, 1997,pp. 19-31.

[7] I. Richardson :Real-time implementation of H.264 Video Coding. IEEE SOC Conference, September 17-20, 2008, Newport Beach, CA, USA.

[8] K. Messaoudi, E. B. Bourennane, S. Toumi, M. Touiza : Data-Parallelism Based Hardware Architecture for the Intra-Coding Module in the H.264 Encoder, Springer, The Arabian Journal for Science and Engineering, vol.39, n°5, 2014, pp. 3781-3797.

[9] S. Saponaraa, M. Martinab, M. Casulaa, L. Fanucci, G. Masera : Motion estimation and CABAC VLSI co-processors for real-time high-quality H.264/AVC video coding, Microprocessors and Microsystems, vol.34, n° 7–8, 2010, pp. 316–328.

[10] Y. W. Huang, B. Y. Hsieh, T. C. Chen, L. G. Chen : Analysis, Fast Algorithm, and VLSI Architecture Design for H.264/AVC Intra Frame Coder, IEEE Transactions on Circuits and Systems for Video Tech, vol. 15, n° 3, 2005, pp. 378-401.

[11] B. A. Ringnyu, A. Tangel. E. Karabulut : Implementation of different architectures of forward 4×4 integer DCT for H.264/AVC encoder, The 10thInternational Conference on Electrical and Electronics Engineering (ELECO), November 30 – December 2, 2017, Bursa, Turkey.

[12] C. W. Ku, C. C. Cheng, G. S. Yu, M. C. Tsai, T. S. Chang : A High-Definition H.264/AVC Intra-Frame Codec IP for Digital Video and Still Camera Applications, IEEE Transactions on Circuits and Systems for Video Tech., vol. 16, n° 8, 2006, pp. 917-928.

[13] B. G. Kim, J. H. Kim, C. S. Cho : A Fast Intra Skip Detection Algorithm for H.264/AVC Video Encoding, ETRI Journal vol. 28, n° 6, 2006, pp. 721-731.

[14] R. Mukherjee, A. Banerjee,I.Chakrabarti, P. K. Dutta,A. K. Ray :An efficient FPGA based implementation of forward integer transform and quantization algorithm of H.264. IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), August 13-14, 2016, Mangalore, India.

[15] I. J. Barge, C. Ababei:H.264 video decoder implemented on FPGAs using 3×3 and 2×2 networks-on-chip. The International Conference on ReConFigurable Computing and FPGAs, December 6-7, 2017, Cancun, Mexico.

[16] L. A. Barroso, K. Gharachorloo, R. McNamara, A. Nowatzyk, S. Qadeer, B. Sano, S. Smith, R. Stets, and B. Verghese : Piranha: a scalable architecture based on single-chip multiprocessing. In Proceedings of the 27thInternational Symposium on Computer Architecture, June 10-14, 2000, Vancouver, BC, Canada,pp. 282–293.

[17] K. Olukotun, B. A. Nayfeh, L. Hammond, K. Wilson, and K. Y. Chang : The case for a single-chip multiprocessor. In Proceedings of the 7th International Symposium on Architectural Support for Parallel Languages and Operating Systems, Cambridge, MA, October 01-04,1996, USA,pp. 2–11.

[18] C. M. Lien, Y. S. Chen, C. S. Shih :On-Chip Bus Architecture Optimization for Multi-core SoC Systems. In Proceedings of 5th IFIP WG 10.2 International Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, SEUS May 7-8, 2007, Santorini Island, Greece, pp 301-310.

[19] C. A. Zeferino, M. E. Kreutz, L. Carro, Altamiro A. Susin :A Study on Communication Issues for Systems-on-Chip. The 15thSymposium on Integrated Circuits and Systems Design (IEEE - SBCCI'02), September 14, 2002, Porto Alegre, Brazil.

[20] W. J. Dally and B. Towles : Route packets, not wires: On-chip interconnection networks. In Proceedings of the 38th Conference on Design Automation, June 22, 2001, Las Vegas, NV, USA, pp. 684–689.

[21] S Pasricha, N Dutt : On-chip communication architectures: system on chip interconnect, Morgan Kaufmann, vol. 1, 2008.

[22] S. Kumar, A. Jantsch, J. P. Soininen, M. Forsell, M. Millberg, J. Oberg,K. Tiensyrja and A. Hemani : A Network on Chip Architecture and Design Methodology. Proceedings of the IEEE Computer Society Annual Symposium on VLSI (IEEE - ISVLSI'02), April, 25-26 2002, Pittsburgh, PA, USA.

[23] G. De Micheli, L. Benini:Networks on Chips, Morgan Kaufmann 2006.

[24] M. Dehyadgari, M. Nickray, A. Afzali-kusha, Z. Navabi : Evaluation of Pseudo Adaptive XY Routing Using an Object Oriented Model for NOC. The 17th International Conference on Microelectronics, December 13‑15, 2005, Islamabad, Pakistan.

[25] W.J. Dally, B. Towles: Principles and Practices of Interconnection Networks. Morgan Kaufmann, 2004.

[26] ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, H.264/14496-10 AVC Reference Software Manual, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, 2009.

[27] Y. Song, A. Akoglu: An Adaptive Motion Estimation Architecture for H.264/AVC, Journal of Signal Processing Systems, vol. 73, n°2, 2013, pp. 161–179.

[28] K. Messaoudi, E. B. Bourennane, S. Toumi : Reconfigurable Hardware Intelligent Memory Controller for H.264/AVC Encoders, International Journal of Computer Science and Information Security (IJCSIS), vol.8 n° 9, 2010, pp. 8-16.

[29] K. Messaoudi, M. Touiza, E.B. Bourennane, S. Toumi,Hardware/Software Co-Design with MicroBlaze Soft-Core Processor for the Integer Transform Algorithm Used in the H.264 Encoder, International Review on Computers and Software (I.Re.Co.S), vol. 5 n°. 3, 2010, pp. 348-354.

[30] I. J. Barge, C. Ababei : A Network-on-Chip Based H.264 Video Decoder Prototype Implemented on FPGAs, IEEE The 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), April 30 - May 2, 2017, Napa, CA, USA.

[31] ITU-T and ISO/IEC JTC 1,Advanced Video Coding for generic audio-visual services, ITU T Rec. H.264 and ISO/IEC 14496-10 (AVC), May 2003 (and subsequent editions). (Last consultation: 01/11/2017)

[32] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi : Video coding with H.264/AVC: Tools, Performance, and Complexity, IEEE Circuits and Systems Magazine, vol. 4, n° 1, 2004, pp.7-28.

[33] T. C. Chen, C. Lian, and L. Chen : Hardware Architecture Design of an H.264/AVC Video Codec, Asia and South Pacific Conference on Design Automation, 24-27 January 2006, Yokohama, Japan.

[34] K. Babionitakis, G. Doumenis, G. Georgakarakos, G. Lentaris, K. Nakos, D. Reisis, I. Sifnaios, N. Vlassopoulos: A real-time H.264/AVC VLSI encoder architecture, Springer –Journal of Real-Time Image Processing, vol. 3, n° 1-2, 2008, pp. 43–59.

[35] P. List, A. Joch, J. Lainema, G. Bjøntegaard, M. Karczewicz :Adaptive Deblocking Filter, IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, n° 7, 2003,pp. 614-619.

[36] K. Messaoudi, E. Bourennane, S. Toumi and G. Ochoa : Performance Comparison of Two Hardware Implementations of the Deblocking Filter Used in H.264 by Changing the Utilized Data Width, IEEE, The 7th International Workshop on Systems, Signal Processing and their Applications, 9-11 May 2011,Tipaza-Algeria.

[37] K. Messaoudi, E. Bourennane, S. Toumi, M. Touiza, A. Yahi :A Highly Parallel Hardware Implementation of the Deblocking Filter Used in H.264/AVC CODECs, In proceedings of the International Conference on Software Engineering and New Technologies, ICSENT'12, pp. 26-38, December 15-17, 2012, Hammamet, Tunisia.

[38] G. Schelle and D. Grunwald : Onchip interconnect exploration for multicore processors utilizing FPGAs, The 2nd Workshop on Architecture Research using FPGA Platforms (WARFP-2006), February 12, 2006, Austin, Texas, USA.

[39] K. Motamedi, N. Ioannides, M. H. Rümmeli, I. Schagaev :Reconfigurable Network on Chip Architecture for Aerospace Applications, 30thIFAC Workshop on Real-Time Programming and 4th International Workshop on Real-Time Software, PP. 131-136, October 12–14, 2009,Mrągowo, Poland.

[40] http://hardh264.sourceforge.net/H264-encoder-manual.html. (Last consultation:01/11/2017)

[41] K.Messaoudi, H.Mayache, A.Benhaoues, E.Bourennane, S.Toumi :Connection of H.264/AVC hardware IPs using a specific Networks-on-Chip. Microprocessors and Microsystems, vol. 39, n° 8, 2015, pp. 609-620.

[42] J. Kim :Low-cost router microarchitecture for on-chip networks. The 42nd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO-42), December 12-16, 2009, New York, NY, USA.pp. 255-266.

[43] A.Kodi, A. Sarathy, and A. Louri :iDEAL. Inter-router dual-function energy and area-efficient links for network-on-chip (NoC) architectures, The 35thInternational Symposium on Computer Architecture (ISCA-35), June 21-25, 2008, Beijing, China.

[44] T. Moscibroda and O. Mutlu :A case for bufferless routing in on-chip networks. In Proceedings of the 36th Annual International Symposium on Computer Architecture (ISCA-36), June 20-24, 2009, Austin, USA. pp. 196-207.

[45] M. Hayenga, N. Jerger, and M. Lipasti :SCARAB: A single cycle adaptive routing and bufferless network. In proceedings ofthe 42nd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO-42), December 12-16, 2009, New York, NY, USA.pp.244-254.

[46] C. Gómez, M. E. Gómez, P. López, J. Duato :Reducing packet dropping in a bufferless NoC. In proceedings of the 14th International European Conference on Parallel Processing (Euro-Par’08), 26-29 August, 2008, Las Palmas de Gran Canaria, Spain.