THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

IEICE ICDV 2011

FPGA Implementation of a Low Latency and High Throughput Network-on-Chip Router Architecture

Nam-Khanh Dang, Thanh-Vu Le-Van, Xuan-Tu Tran*

SIS Laboratory, University of Engineering and Technology, VNU Hanoi
144 Xuan Thuy road, Hanoi 10000, Vietnam. (*) Corresponding author: tutx@vnu.edu.vn

Abstract — The Network-on-Chip (NoC) paradigm has recently been known as a promising solution for designing large complex Systems-on-Chip (SoCs), especially when the semiconductor technology turns into 3D integration era. This paper presents the design of a NoC router architecture which provides low latency, high throughput communication. The NoC router architecture is implemented on a Xilinx technology FPGA chip for prototyping purpose with an obtained latency of 8.1*ns* and a maximum throughput of 123*Mflits/s* on each communication channel. These results can be improved when the design is implemented with ASIC design flows.

Keyword: Network-on-Chip (NoC), On-chip communication, router architecture.

1. Introduction

Thanks to the rapidly evolution of the semiconductor technology, more and more Intellectual Properties (IPs) can be integrated into a complex System-on-Chip (SoC) to meet high demand of new applications. This leads to a big challenge for on-chip communication design because of many critical limitations of conventional communication models based on shared buses or point-to-point links; especially, when the semiconductor technology turns into 3-D integration.

Recently, the Network-on-Chip (NoC) paradigm has become a promising solution for on-chip communication of large complex systems thanks to its potential advantages such as: high throughput communication, high scalability and versatility, as well as good power efficiency [1], [2], [3]. In consequence, many works and/or proposals on NoC architectures and methodologies have been presented in literature last several years. However, the design methodology has not yet been completed and there are still many research issues to bring NoC paradigm to the market such as network latency and communication throughput improvement [4], [5], [6], [7]. In which, Nilsson and Oberg [6] proposed a pseudochronous mechanism to reduce the network latency while Beigné *et al.* [4], [9] implement the whole NoC architecture using quasi-delay insensitive asynchronous logic to minimize the network latency thanks to the asynchronous design advantages. However, it makes the test of asynchronous NoC architectures more complex [8] due to the lack of testing tools for asynchronous logics. Kreutz *et al.* [5] analyzed the latency of different topologies to find the best matched topology for each application in considering power consumption.

In this paper, we present the design of an efficient router architecture that can be used to build high throughput, low latency NoC architectures by optimizing the router architecture and using a double-crossbar. The 2-D mesh topology has been selected to develop our NoC architecture in order to be easily implemented with current semiconductor technology and to apply routing algorithms. However, other 2-D topologies such as 2-D torus or 2-D folded-torus can be easily developed from the proposed router architecture. The design of a complete NoC architecture is then implemented and validated on FPGA devices from Xilinx to evaluate the performance of the design. This implementation is targeted to prototyping purposes before the design will be synthesized and implemented on ASIC technology. Preliminary experimental results show that the design implemented on Xilinx Virtex-5 FPGA [11] has a very small latency of 8.1*ns* and a high throughput of 123*Mflits/s* per communication channel (two communication channels with different priority levels can be concurrently implemented). These results can be improved when the design is implemented with ASIC design flows in the next step.

The remaining part of this paper is organized as follows: Section 2 describes the design of our NoC architecture, from network topology to communication mechanism. Section 3 introduces our proposed router architecture used to develop out NoC architecture. Verification

and implementation of the design based on Xilinx FPGA technology is presented in Section 4. Finally, conclusions and perspectives will be given in Section 5.

2. Network-on-Chip design

2.1. Network topology

As computer networks, there are many topologies can be used to design NoC architectures such as ring, fat-tree, butterfly, 2-D mesh, 2-D torus, 2-D folded torus, etc. Between these topologies, 2-D mesh and 2-D torus topologies are usually preferred in developing NoC architectures because these topologies can be easily implemented with the current semiconductor technology. The 2-D torus has a smaller network diameter and higher bandwidth. However, the structure of 2-D torus network is more complex than 2-D mesh network with long network links to establish the loops between border routers. In our case, 2-D mesh topology has been selected to develop NoC architectures in order to easily implement routing algorithms. In addition, it is easier to expand network scale with 2-D mesh topology.

In our NoC architecture, each router has five 32-bit bi-directional ports which are connected to four neighboring routers (North, East, South, and West) and a nearest IP core via a Network Interface (NI) as depicted in Figure 1. The communication mechanism between routers is established by using a handshake protocol with two virtual channels providing system's quality of service (QoS).

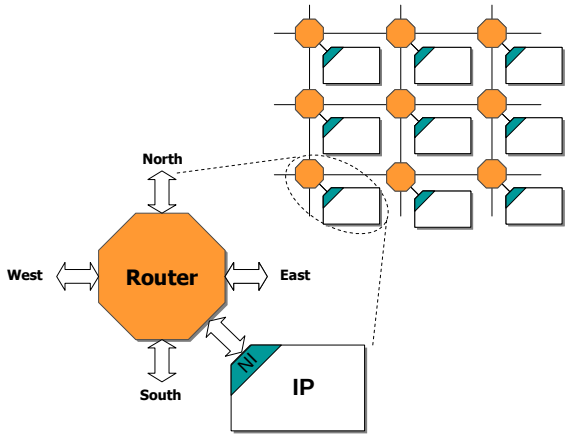

Figure 1: 2-D mesh NoC architecture.

2.2. Communication mechanism

The communication mechanism used in this NoC architecture is packet switching with Wormhole commutation mode and source routing algorithm. The message is therefore split and encapsulated into packets at the source before being transferred onto the network. Each packet is always composed by a header flit, following by one or more data flits (including body flits and a tail flit). The size of each flit is 34 bits, where 32 bits are used for data and two most significant bits are used for control purposes. The formats of these flits are described in Figure 2.

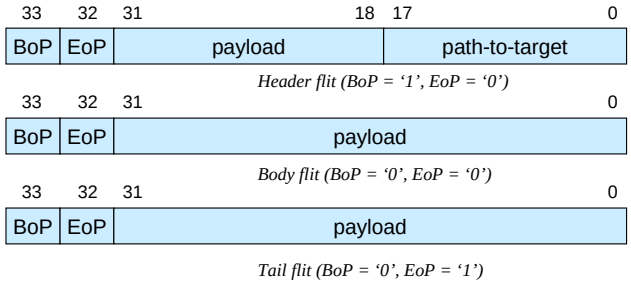

Figure 2: Flits' format.

Begin-of-Packet (BoP) and End-of-Packet (EoP) are two control bits specifying the type of a flit. Header flit has BoP = '1' and Tail flit has EoP = '1'. If both BoP and EoP equal '1', the packet has only one flit while if both values equal '0', this is a body flit. Thanks to these control bits, the network router can easily recognize the type of an incoming flit (header flit, body flit, or tail flit) and make a routing decision. To route a packet on network, routing information have to be included in the header flit (in the "path-to-target" field). This field will be shifted to the right at each router for next routing direction after two least significant bits are used due to source routing algorithms.

Figure 3 presents the communication interface between two network routers (i.e., between a network router and an IP core).

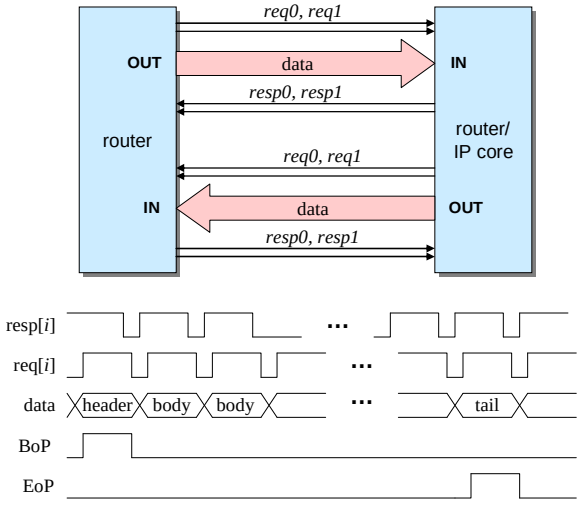

Figure 3: Router-to-router/IP core interface and handshake protocol.

The network links are 34-bit bidirectional channels, in which 32 bits are used for data and two bits are used to encode flit information as described above. In addition, “req0/req1” and “resp0/resp1” signals are used to establish a handshake protocol for communication between network elements. This protocol also guarantees that there is only one virtual channel implemented on the physical channel in one time moment.

3. Proposed router architecture

To be used in developing 2-D mesh NoC architectures, we proposed a network router architecture having five 32-bit bi-directional input/output ports (North, East, South, West, and Local) connected to four neighboring routers and the nearest IP core. The router architecture is composed of input modules, output modules, and a double-crossbar as depicted in Figure 4.

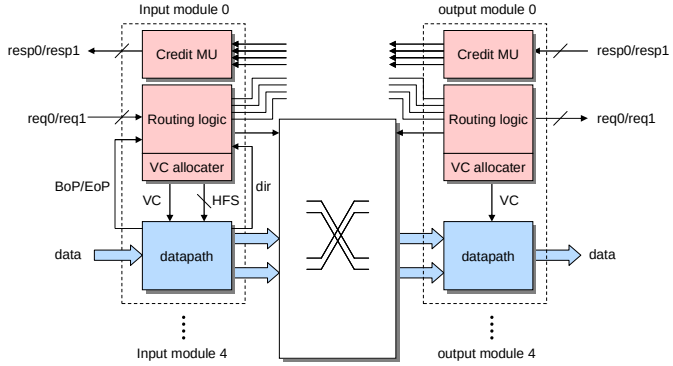

Figure 4: Router architecture.

In such architecture, each input/output module has its own datapath and control logic unit. The datapath is divided into two communication channels in order to implement two virtual channels, VC0 and VC1. In which, the VC0 has a higher priority and is used for transferring important control information or providing real-time services. The VC1 provides best-effort service and is usually used for transferring bulk data on the network.

Beside the datapath, the control logic unit is composed of VC allocater, routing logic, and credit management unit. In which, VC allocater allocates the incoming flit into its corresponding virtual channel; routing logic unit analyzes the routing information and generates necessary control signals to the crossbar to implement routing paths. A header flit shifting (HFS) signal is also sent to datapath to shift the path-to-target field for getting new routing information used at the next router. To synchronize with the output modules, routing logic unit also generate handshake signals to the output modules. The credit management unit (Credit MU) manages flit flow control. This will implement the handshake protocol between routers at flit level. The detail of the input/output modules is explained as follows.

At the input module, the datapath is composed of a Pre-processor, buffers and multiplexer/de-multiplexer as described in Figure 5.

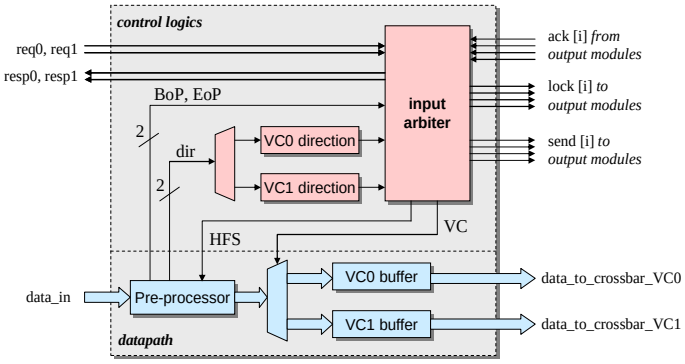

Figure 5: Micro-architecture of input module (IM).

Because Wormhole commutation mode is used, the buffers have only 1-flit depth. The coming data packets will be allocated into their corresponding virtual channel depending on the value of req0 and req1 signals. The outputs of virtual channel buffers are connected to corresponding crossbars. The use of two crossbars allows the router to establish two different routing paths on different virtual channels at the same time. This provides high throughput communication and low latency for the network architecture. Of course, the routing paths should be on different directions. Routing decision, credit management, and other control circuits are implemented in the input arbiter.

At the output module, the architecture is a little bit simpler, as described in Figure 6.

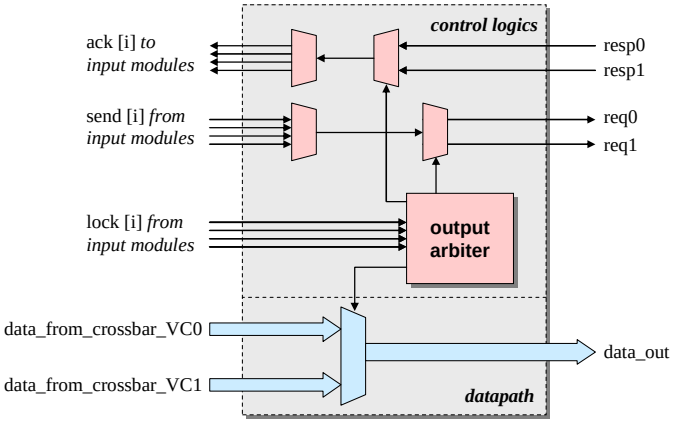

Figure 6: Micro-architecture of output module (OM).

In this architecture, routing decision, virtual channel allocation, and credit management are implemented by an output arbiter and several multiplexers and/or de-multiplexers, called control logics. The datapath is just a 34-bit multiplexer controlled by the control logics.

The crossbar of the router is a double crossbar used to connect 5 input modules to 5 output modules as presented in Figure 7. Thanks to this double-crossbar, the router can establish two concurrent communication channels on different virtual channels. The quality of service of the whole NoC architecture is then improved.

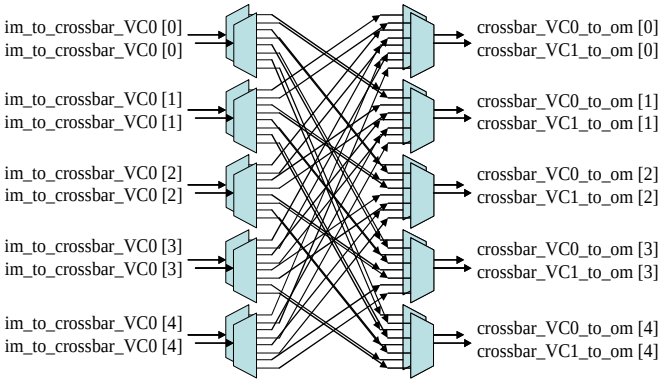

Figure 7: Double-crossbar architecture.

4. Implementation and Verification

To validate the proposed router architecture, we have modeled a 3×3 NoC architecture as described in Figure 8. This network has 9 network routers connected to 9 IP cores. For testing purpose, the IP cores are very simple; they just generate the data with defined routing information corresponding to testing scenario, transmit the data onto the network, and receive data flow from the network.

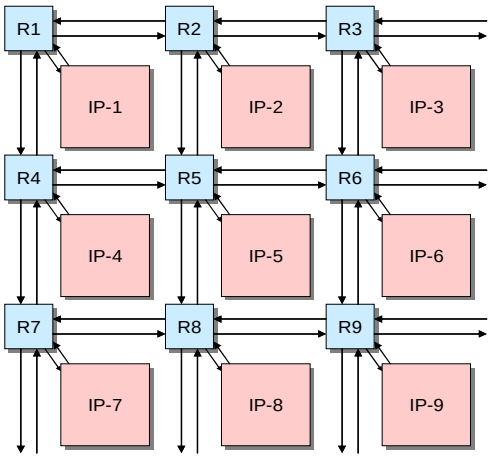

Figure 8: A 3×3 NoC architecture.

The NoC architecture is then synthesized and implemented on a Xilinx Virtex-5 FPGA chip [11] for verification purpose. The FPGA-based prototyping does not demonstrate the feasibility of the NoC architecture but also enables accurate evaluation of performance, area, and various design trade-offs [10]. The hardware overhead of the network architecture without IP cores is briefly reported in Table 1. Number of occupied slices of each router is 717, approximately 1% available resource of the targeted FPGA device.

Table 1: Hardware implementation report on Xilinx Virtex-5 FPGA chip (xc5vlx330)

Resources	Used	Available	Percentage (%)
Slice LUTs	10791	207360	5.2%
Slice LUT-Flip Flop pairs	13806	207360	6.7%
Slice Registers	3600	207360	1.7%
I/O buffers	915	1200	76%

To verify the router architecture, we have implemented the following test cases, as depicted in Figure 9, to check all routing paths of the network router.

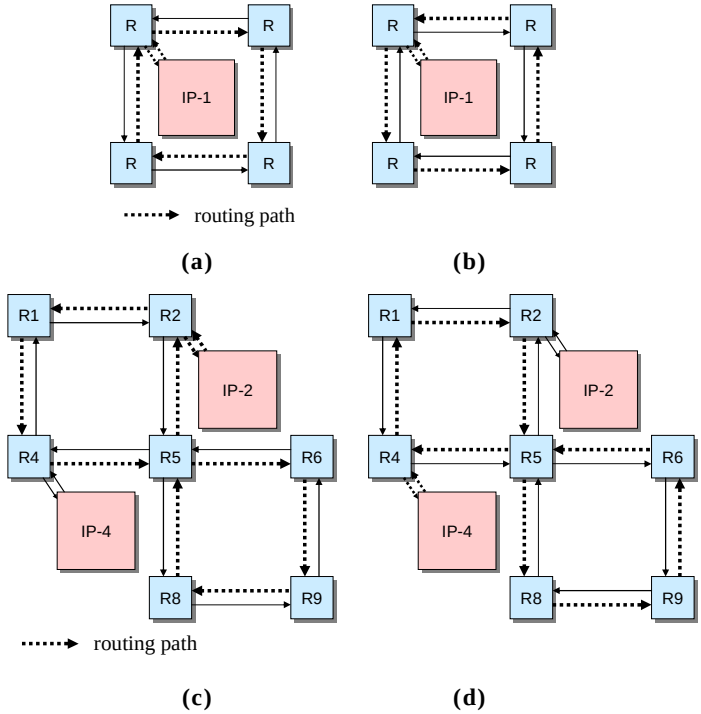

Figure 9: Verification model for 4-router group.

First, the IP core will generate data and routing information to test the loop routing path between 4 routers as presented in Figure 9(a) and Figure 9(b), for clockwise and counterclockwise, respectively. Then, to test the remaining routing paths of the router, we implement the

routing paths as presented in Figure 9(c) and Figure 9(d). In Figure 9(c), test data are generated by the IP core 2 (IP-2), then passed through R2, R1, R4, R5, R6, R9, R8, R5, R2, and finally received by the IP-2. In Figure 9(d) test data are generated by IP core 4 (IP-4), then passed through R4, R1, R2, R5, R8, R9, R6, R5, R4, and finally received by IP-4. These routing paths are opposite to each other. The verification results approved that all routing paths inside the router architecture have been passed for the test.

From the experimental results, it shows that we need only once cycle to transmit a data flit from an input port to an output port of the router. The obtained latency on the targeted FPGA device is 8.1*ns*. The maximum throughput achieved on each communication channel is 123*Mflits/s*. As presented above, two communication channels can concurrently established, the maximum throughput can be reached is 246*Mflits/s*.

5. Conclusion and Future Works

In this paper, we have presented the design and implementation of a NoC router architecture on a Xilinx FPGA Virtex-5 chip. The preliminary experimental results show that the design can be used to develop low latency and high throughput NoC architectures based on 2-D mesh/ torus topologies. The obtained latency is 8.8*ns* and the maximum throughput achieved on each communication channel is 123*Mflits/s*. Because two communication channels can be concurrently established at the same time on different directions and different virtual channels thanks to its architecture with double-crossbar, the maximum throughput of a router can be reached to 246*Mflits/s*, equivalent to 7.872Gbps. This can improve the NoC performance and guarantee the quality of service of the system. The experimental results can be improved when the design will be implemented with ASIC design flows in next step.

Acknowledgement

This work is partly supported by Vietnam National University, Hanoi (VNU) through research project No. QGDA.10.02 (VENGME).

References

[1] A. Jantsch and H. Tenhunen (Eds). Networks on Chip. Kluwer Academic Publisher, February 2003.

[2] L. Benini and G. De Micheli. Networks on Chips: a New SoC Paradigm. IEEE Computer, 35(1):70–78, January 2002.

[3] W.J. Dally and B. Towles. Route Packets, Not Wires: On-Chip Interconnection Networks. In Proceedings of the Design Automation Conference (DAC), pp. 684–689, Las Vegas, NV, USA, June 2001.

[4] Edith Beigne, et al. An Asynchronous NoC Architecture Providing Low Latency Service and its Multi-level Design Framework. In Proceedings of the 11th IEEE Int'l Symposium on Asynchronous Circuits and Systems (ASYNC), pp. 54–63, 2005.

[5] M. Kreutz, C. Marcon, L. Carro, N. Calazans, and A.A. Susin. Energy and Latency Evaluation of NoC Topologies. In Proceedings of the 2005 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 5866–5869, Vol. 6, 2005.

[6] Erland Nilsson, Johnny Oberg. Reducing Power and Latency in 2-D mesh NoCs using Globally Pseudo-chronous Locally Synchronous Clocking. In Proceedings of the 2004 International Conference on Hardware/Software Codesign and System Synthesis, pp. 176, November 2004.

[7] Radu Marculescu, Umit Ogras, Li-Shiuan Peh, Natalie Enright Jerger, and Yatin Hoskote. Outstanding Research Problems in NoC Design: System, Microarchitecture, and Circuit Perspectives. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 28(1), January 2009.

[8] Xuan-Tu Tran, Yvain Thonnart, Jean Durupt, Vincent Beroulle, Chantal Robach. Design-for-Test Approach of an Asynchronous Network-on-Chip Architecture and its Associated Test Pattern Generation and Application. IET Journal on Computers and Digital Techniques, Volume 3, Issue 5, pp. 487-500, September 2009.

[9] Edith Beigne, E.; Clermidy, F.; Lhermet, H.; Miermont, S.; Thonnart, Y.; Tran, X.T.; Valentian, A.; Varreau, D.; Vivet, P.; Popon, X.; Lebreton, H. An Asynchronous Power Aware and Adaptive NoC Based Circuit. IEEE Journal of Solid State Circuits, Volume 44, Issue 4, pp. 1167-1177, April 2009.

[10] Umit Ogras *et al*. Challenges and Promising Results in NoC Prototyping using FPGAs. IEEE Micro, pp.86-95, September – October 2007.

[11] Xilinx Virtex-5 FPGA. www.xilinx.com