Automatic Embedded Multicore Generation and Evaluation Methodology: a Case Study of a NOC Based 2400-cores on Very Large Scale Emulator

Omar Hammami, X. Li, L. Burgun, S. Delerse

Automatic Embedded Multicore Generation and Evaluation Methodology: a Case Study of a NOC Based 2400-cores on Very Large Scale Emulator

O. Hammami, X. Li
ENSTA ParisTech
32, Bvd Victor, Paris
FRANCE
{hammami,xli}@ensta.fr

L. Burgun, S. Delerse
EVE Team
2 bis, Voie La Cardon
Parc Gutenberg, Batiment B 91120 Palaiseau
FRANCE

Abstract— **Future generation embedded multicore will be based on hundreds of processors connected through Network on Chip (NOC) . Design productivity of embedded multicore is a major challenge for the semiconductor industry. In this paper, an automatic very large scale NoC design methodology based on FPGA IP is proposed to accelerate the embedded multicore design productivity using very large scale multi-FPGA platform emulation. The large scale multiprocessor is divided into pieces of FPGA IP. Local NoC is designed for each FPGA IP. Then a top level NoC connects all the local NoCs together to form the large scale multiprocessor. In this paper we extend our previous work to reach 2400-cores NoC based multiprocessor which is generated using this methodology within 3 days.**

Keywords-Very large scale MPSOC, multi-FPGA, NoC

I. INTRODUCTION

ITRS Semiconductor roadmap [1] projects that hundred of processors will be needed for future generation embedded multicore designs. Among the various challenges of embedded multicore design, productivity with constraints is paramount [2]. Very large scale multiprocessor (VLSM), which has beyond 512 PEs or even more [3], is very difficult to be implemented on to ASIC or FPGA circuit because of its complexity and the lack of EDA supported design framework. To the best of our knowledge there is no reported work in the world in 2010, besides the RAMP project, for prototyping multiprocessor larger than 512 cores on multi-FPGA platforms or on single ASIC chip. [6] To our best knowledge, there is no report on either high level modeling (e.g. SystemC) or high level performance evaluation of VLSM. As the *simulation wall* blocks the VLSM design path, FPGA platform based emulation is selected as the only solution for billion cycle applications. Automatic design work flow is mandatory at this scale to accelerate the design and debug process.

In this paper we describe our automatic design and prototyping methodology, which is based on FPGA IP reuse. The large scale multiprocessor is divided into pieces of FPGA IP, which fills one FPGA device. Local NoC is designed for each FPGA IP. Then a top level NoC connects all the local NoCs together to form the large scale multiprocessor. A 2400-core NOC based embedded multicore is generated and implemented on a large scale multi-FPGA emulation platform of 100 FPGA circuits. Comparing to RAMP Blue framework, our system is purely NoC based, which is more flexible in communication architecture based on multi-FPGA Platform.

II. RELATED WORK

Emulation and FPGA-based emulator have been recognized as efficient techniques for the performance evaluation and validation of multiprocessors. In [5] design space exploration of multi-processor on multi-FPGA platform [6] have been conducted with masters on chip, network on chip on a second chip and slaves on a third chip. Automatic design space exploration of multiprocessor on chip on a single large scale FPGA chip have been conducted [7] with automatic tuning and test of the multiprocessor. The combined effects of compiler, architecture and place and route have been addressed for a single chip in [8]. A complete network on chip emulation platform has been proposed in [9] and a multiprocessor platform in [10]. Finally large scale multiprocessor project have emerged in [11]. The state of the art of 512+ cores multiprocessor is restricted in 2010 to the University of California Berkeley project, RAMP (Research Accelerator for Multiple Processors) [4].

	RAMP Blue	**BB-672**
Max number of processor	1008	672
FPGA number and type	84 Virtex-II Pro 70	56 Virtex-4 LX200
Emulation frequency	90 MHz	10 Mhz
Processor	MB V4.0 no optional units	MB V6.0 full optional units
OS & compiler	ucLinux & GCC	Standalone & GCC
FPU	64 bits FPU / 12 cores	32 bits FPU / core
DDR memory	250 MB / processor	Not implemented
Monitoring & Debug	Control network Debug interface	Monitoring network Dynamic probes
Network topology	Cluster 3D mesh	Cluster mesh
Interconnection	Crossbar switch + Ethernet	Network on Chip
Communication	Message passing	Distributed shared memory
Language	MPI or UPC	ENSTA proper C driver
Benchmarks	Netpert (for network) NPB class-S (on 256 cores)	OCP-IP micro-benchmarks (on 672 cores)
Design flow	Not reported	Full automatic

The RAMP project has evolved through different versions in the communication paradigm and in the number of processors. To the best of our knowledge, the RAMP blue [12] and our platform [20] are the only two multiprocessors which have surpassed the milestone of 512 cores on multi-FPGA platforms. The basic idea of multi-FPGA platform emulation is the same, while the approaching framework is quite different. Although Berkeley decided to go for its own board designs we selected industry-class emulator in order to focus on the methodologies and design flows. In this paper we address a study case of 2400-core MPSOC using our automatic NoC design methodology based on L-T level topology and multi-FPGA platform emulation.It is important to note that our project is based on EVE multi-FPGA platform which is presented in section IV; there are no interconnection constraints. The emulator allows any type of communication architecture to be implemented. So comparing to RAMP framework, our system can be fully NoC based. And consequently, our communication architecture is more flexible. More NoC topologies can be used in our system design on both local and top levels of NoC architecture.

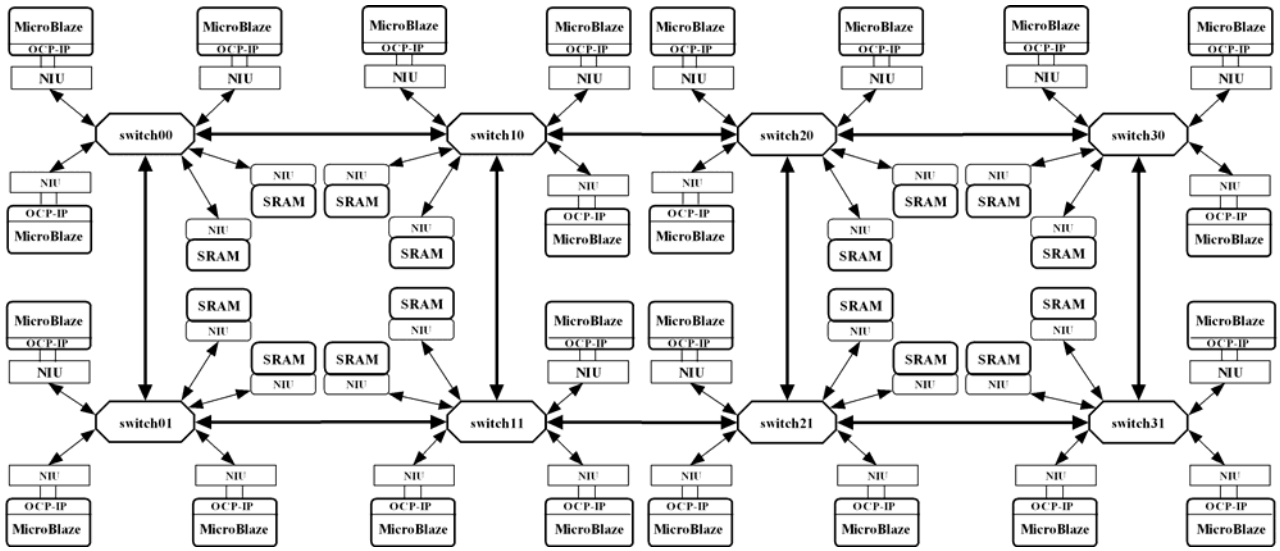

Figure 1 FPGA IP architecture with 24 MicroBlaze processors and 16 SRAMs

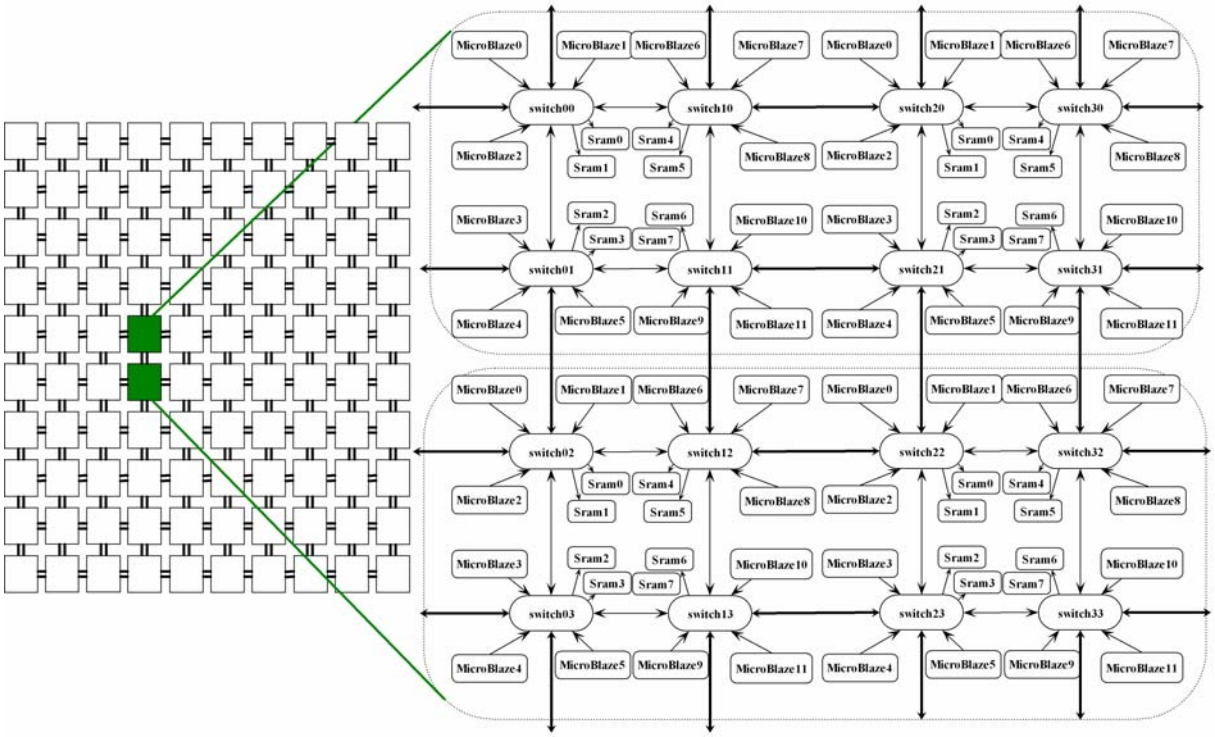

Figure 2 Extension to 2400-cores NOC based MPSOC.

III. AUTOMATIC DESIGN FRAMEWORK BASED ON FPGA IP AND MULTI-FPGA PLATFORM

A. Processor Element and NoC communication service

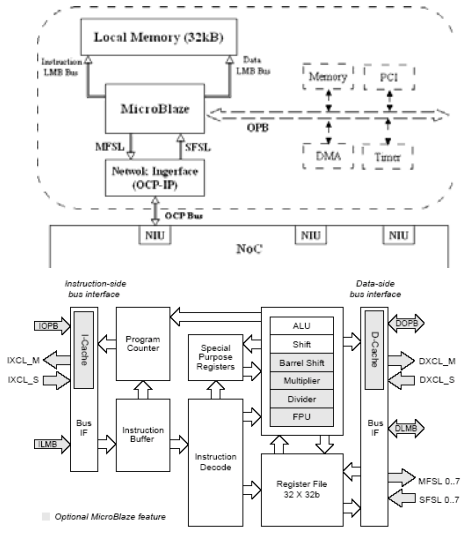

Figure 3 (a)Processor Tile (b) MicroBlaze core block diagram

MicroBlaze soft IP is used as basic processor element in SSM IP. It is a 32-bit 5-stage single issue pipelined Harvard style embedded processor architecture provided by Xilinx as part of their embedded design tool kit (ED. The MicroBlaze processor is flexible, and gives the user control of a number of features such as the cache sizes, interfaces, and execution units like: selectable barrel shifter (BS), hardware multiplier (HWM), hardware divider (HWD), and floating point unit (FPU). Our SSM design is OCP-IP compliant which implies that we can change processor IP by any other OCP-IP compliant processor IP while leaving the overall design identical. The OCP-IP protocol is used for the communication between the processors and Network on Chip. A FSL-OCP network interface is designed in order to make MicroBlaze processor compatible to OCP-IP protocol as shown in Figure (a). This interface gets data from MicroBlaze processor through FSL link and transfers data to SRAM through Network on Chip under OCP-IP protocol. It encapsulates and decapsulates data between FSL links and OCP-IP bus interface.A NoC communication service library of driver is written in C for the FSL-OCP network interface to support OCP-IP basic and extended communication modes. As FSL link works like 32 bits FIFO interface, the address and data are transferred in two phases between MicroBlaze processor and FSL using basic micro C code 'putfsl()' and 'getfsl()' in EDK library. In the service library, 5 OCP-IP compliant communication modes are supported: Write, Read, Read Exclusive, Write Non-Post and Write Conditional. Parallelized application codes use these drivers for data communication and synchronization between processors and SRAMs.

B. FPGA IP architecture

This important multiprocessor architecture requires efficient IP design and reuse. Although we use Xilinx EDA tools and IPs for the Xilinx target Virtex-5 LX-330 no multiprocessor soft IP is available which matches our need. The architecture of the FPGA IP is based on a mesh-based network on chip connecting 24 processors organized as with 3 processors and 2 SRAM on chip memories per switch. Mesh is chosen as the NoC topology for an easy multi-FPGA implementation in this case.

C. Emulation Platform: EVE Zebu Server

Large scale emulators are composed of dozens of large scale FPGA chip and are unbeatable in terms of performance accuracy tradeoffs. We target large scale emulator such as the Eve Team [17] Zebu Server emulation platform composed of up to 100 Xilinx Virtex-5 LX330 chips. ZeBu-Server is a flexible, scalable emulator, with a capacity up to 1 Billion ASIC gates. Providing high bandwidth and multi-MHz performance at any level, ZeBu-Server primarily aims at large-scale, multi-core chip and system emulation applications.

Figure 4 EVE Zebu Server multi-FPGA platform

D. Automatic Design Flow based on multi-FPGA platform

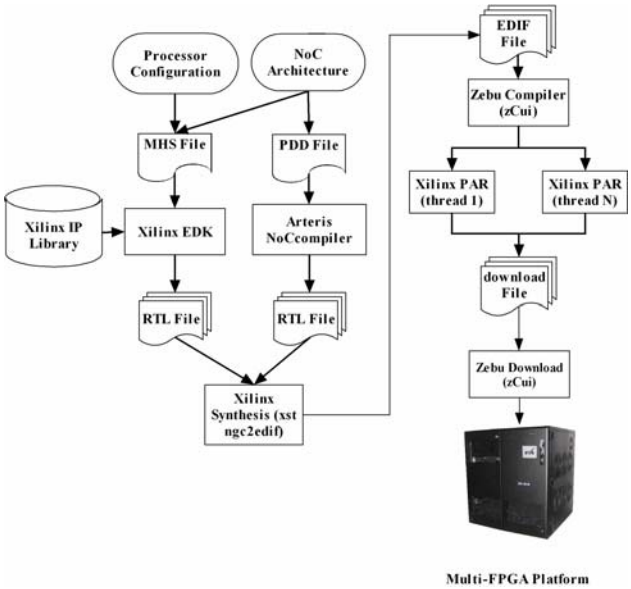

Figure 5 Automatic Embedded Multicore Design Flow

The automatic workflow of our VLSM multiprocessor design is presented in the Figure 5. Design automation tools of 3 commercial companies are combined together to generate our multi-FPGA MPSoC. The Xilinx EDK tool is used to generate our FPGA multiprocessor using Xilinx IPsOnce the RTL files

of FPGA are generated, they are reused for the multi-FPGA large scale multiprocessor synthesis, which can largely reduce system design time.erent NoC files are synthesized for each FPGA on different FPGA chips of Zebu platform. These NoC RTL files are generated by Arteris NoCcompiler tool, which allows the export of NoC using the Arteris Danube Library. Eve Zebu compiler takes the EDIF files converted by Xilinx synthesis tools for the implementation. Different FPGA IPs are analyzed and distributed onto FPGAs. Xilinx place and route tools are used to generate the download bit files of FPGA. This phase can be parallelized to reduce the turnaround time. Finally the compiled execution files and download bits files are downloaded to EVE Zebu Server multi-FPGA platform using Zebu download tools. The execution time is recorded after the execution. There are 100 Xilinx Virtex 5 LX330 FPGAs available on the Zebu Server platform we used. By duplicating the FPGA IP onto the 100 FPGAs with 2 FPGA IPs on each FPGA, a VLSM of 2400-cores is generated automatically. In the Figure 2, each rectangle represents one FPGA IP. The general architecture and interconnection between FPGA IPs are presented in the figure. The global architecture stays as 10x10 cluster mesh topology. Based on the local NoC architecture of FPGA IP, two more external input and output ports are added to each switch to connect their neighbor FPGA IPs. Our architecture is symmetrical and based on the local NoC, which makes it a good candidate for design modularity and IP based design and reuse. Table I shows the resource utilization of FPGA IP on each FPGA circuit.

Table I FPGA IP resource utilization

FPGA Virtex5 LX330 Resource	Utilization
Number of DSP48s	87 %
Number of RAMBs	78 %
Number of SLICEs	79 %
Number of LUTs	58 %

IV. Future Work

We have currently achieved the hardware design and validation of directory-based cache coherency support for our embedded multicore. This next version will be evaluated using multicore benchmarks from the embedded world EEMBC [21] and eventually to general purpose multicore benchmarks [22].

V. Conclusion

Next generation embedded multicore will be based on hundreds of processors. Network on chip design is very complex and in order to reach efficient working silicon in reasonable time large scale prototyping is needed. We proposec an automatic design methodology based on FPGA IP and multi-FPGA platform emulation. This elegant and technology scalable technique eases the implementation of 2400 cores multiprocessors on large scale commercial emulator. We plan to complete this work with directory based cache coherency support and intensive benchmarking. We believe that time has come fort EDA to move up the scale and integrate with parallel computer architecture, automatic parallelization compiler and operating system research results to efficiently tackle in a unified and integrated framework the design productivity gap.

References

[1] International Technology Roadmap for Semicoductors(ITRS)) report 2009 Edition. Available http://www.itrs.net/

[2] A.A.Jerraya and W.Wolf, “Multiprocessor Systems-on-Chips”, Morgan Kaufman Pub., 2004.

[3] John H. Kelm, Daniel R. Johnson, Matthew R. Johnson, Neal C. Crago, William Tuohy, Aqeel Mahesri, Steven S. Lumetta, Matthew I. Frank, Sanjay J. Patel, Rigel: “An Architecture and Scalable Programming Interface for a 1000-core Accelerator”, International Symposium on Computer Architecture (ISCA'09), June 2009.

[4] Wawrzynek, J.; Patterson, D.; Oskin, M.; Shin-Lien Lu; Kozyrakis, C.; Hoe, J.C.; Chiou, D.; Asanovic, K., “RAMP: Research Accelerator for Multiple Processors,” Micro, IEEE , vol.27, no.2, pp.46-57, March-April 2007.

[5] Li, X.; Hammami, O., NOCDEX: Network on Chip Design Space Exploration Through Direct Execution and Options Selection Through Principal Component Analysis, Industrial Embedded Systems, 2006. IES '06. International Symposium on 18-20 Oct. 2006 Page (s):1 – 4

[6] CHIPit Platinum Edition – ASIC Emulation and Rapid Prototyping System – v.2.0., 2004, ProDesign www.uchipit.com

[7] R. Ben Mouhoub and O. Hammami, “MOCDEX: Multiprocessor on Chip Multiobjective Design Space Exploration with Direct Execution,” EURASIP Journal on Embedded Systems, vol. 2006, Article ID 54074, 14 pages, 2006.

[8] O.Hammami, "Heterogeneous Multiprocessor on chip Compiler, Architecture, PLace and Route Design Space Exploration", in IEEE MELECON, May 5-7, 2008, Ajaccio, France

[9] N. Genko, D. Atienza, G. De Micheli, J. M. Mendias, R. Hermida, F. Catthoor, A Complete Network-On-Chip Emulation Framework, Proceedings of the conference on Design, Automation and Test in Europe Volume 1 DATE '05, March 2005

[10] Nava, M.D.; Blouet, P.; Teninge, P.; Coppola, M.; Ben-Ismail, T.; Picchiottino, S.; Wilson, R.; An open platform for developing multiprocessor SoCs, Computer, Volume 38, Issue 7, July 2005 Page(s):60 – 67

[11] E.S. Chung, E. Nurvitadhi, J.C. Hoe, B.Falsafi, K.Mai Virtualized Full-System Emulation of Multiprocessors using FPGAs, 2nd Workshop on Architectural Research Prototyping WARP-2007

[12] K. Asonvic, RAMP: Research Accelerator for Multiprocessors, 2nd Workshop on Architectural Research Prototyping, WARP-2006

[13] Arteris S.A. http://www.arteris.com

[14] NoC Solution 1.10, NoC Compiler user's Guide, o918v2rs4, Dec. 2008, Arteris www.arteris.com

[15] Danube 1.10 – Packet Transport Units Technical reference – 04277v3rs6 – March 2008, Arteris. www.arteris.com

[16] Xilinx. Embedded system tools guide. Available on: http://www.xilinx.com/ise/embedded/edk docs.htm.

[17] EVE TEAM Zebu-XXL Emulation platform. www.eve-team.com

[18] X.Li and O.Hammami, An Automatic Design Flow for Data Parallel and Pipelined Signal Processing Applications on Embedded Multiprocessor with NoC: Application to Cryptography, International Journal of Reconfigurable Computing, Volume 2009 (2009), Article ID 631490

[19] Xinyu Li, Omar Hammami, BB-762: Design and Implementation of 762 Processor Multiprocessor and OCP-IP Benchmarking, DATE 2009, university booth.

[20] O. Hammami, X.Li, L.Larzul and L.Burgun, Automatic Design Methodologies for Large Scale MPSOC and Prototyping on Multi-FPGA Platforms, IEEE International SoC Design Conference (ISOCC) 2009, invited talk

[21] EEMBC, MultiBench™ 1.0 Multicore Benchmark Software , http://www.eembc.org/benchmark/multi_sl.php

[22] C. Bienia and K. Li. PARSEC 2.0: A New Benchmark Suite for Chip-Multiprocessors In Proceedings of the 5th Annual Workshop on Modeling, Benchmarking and Simulation, June 2009.