

# Virtual Circuit Switching Based Smart NOC

P. Jashuva<sup>1</sup>, Sk. Rahil Hussain<sup>2</sup>P.G. Student, Department of ECE, DVR & HS MIC College of Technology, Kanchikacherla, AP, India<sup>1</sup>Assistant Professor, Department of ECE, DVR & HS MIC College of Technology, Kanchikacherla, AP, India<sup>2</sup>

**ABSTRACT:** NoC solutions need special kind of CPUs that consume less power, are cheaper, smaller, but still have high performance requirements. To fulfil all these demands, they are getting more and more complex as the number of transistors is rapidly growing which has led to the emerging of multiprocessor systems-on-a-chip. This is developed on Xilinx 14.2 version by using the Verilog tool kit to show the performance on the design of FPGA unit.

**KEYWORDS:** NoC, CPU, Power, Cheaper, Smaller, Performance.

## I. INTRODUCTION

This State of the Art Report covers computer architecture topics with emphasis on Network on Chip (NoC). The reader is introduced to the basic ideas behind NoC and general computer architecture concepts before presenting an in-depth analysis of three important NoC components: CPU, Interconnect and Memory architecture.

NoC stands for Network-on-Chip and is a term for putting a complete system on a single piece of silicon. NoC has become a very popular word in the computer-industry, but very few agree on a general definition of NoC [19]. There are several alternative names for putting a system on a chip, such as system on silicon, system-on-a-chip, system-LSI, system-ASIC, and system-level integration (SLI) device. Some might say a large design automatically makes it a NoC, but that would probably include every existing design today. A better approach would be to say that a NoC should include different structures such as a CPU-core, embedded memory and peripheral cores. This still is a wide definition which could imply that any modern processor with an on-chip cache should be included into the NoC-community. Therefore a more suitable definition of NoC would be a complete system on a single piece of silicon, consisting of several types of modules including at least one processing unit designated for software execution, where the system depends on no or very few external components in order to execute its task.

A repetition is a task expressing how a single sub-task is repeated; each instance of the repeated task operates with sub-arrays of the inputs and outputs of the repetition. Making the repetitions independent and therefore parallel by construction, it allows expressing data parallelism. For a given input or output, all the sub-array instances have the same shape, are composed of regularly spaced elements and are regularly placed in the array. The RSM constructions can be used to compactly model repetitive architectures. It is especially useful for architectures with a large number of identical components, like a repetition of routers, where a compact representation both simplifies the modelling stage and concretely identifies the components with identical properties. The available modelling mechanisms in RSM are directed towards two aspects. The first aspect is that RSM helps specify the shape of a repetition using a multidimensional form and allows the representation of the potential instances as a multidimensional array. This repetition can be specified for an instance or a port of a component. RSM also provides a mechanism for expressing the potentially complex topologies of the links between these arrays of ports or instances. Complex, regular and repetitive structures such as cubes and grids can be modelled easily in a compact manner via the RSM package.

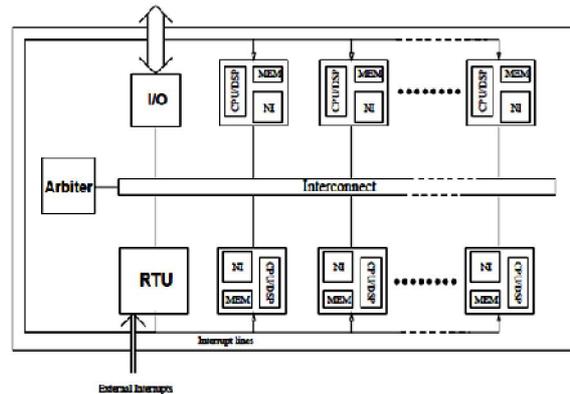


Figure 1: System architecture.

As NoC grew in quantity of IP cores, bus architectures and crossbars were exposed of their deficiencies. Shared bus architectures led to resource contention and hierarchical bus architectures and crossbar designs generated complexity. Interconnection networks offer an attractive solution to this communication predicament and are becoming persistent in NoC systems. A well-designed interconnection network makes a well-organized use of limited communication means while providing low latency, high bandwidth communication amongst different IPs with a minimum cost and low energy-dissipation. Undeniably, as system density and integration continued increasing, quite many designers discovered that it is more efficient to route packets, not wires. Utilizing an interconnection network in a NoC than a dedicated wiring permits 6 limited bandwidth to be shared so that it can be used resourcefully. In contrast, dedicated wiring is idle mostly. Using a network also administers regular, organized use of communication resources, making NoC easier to design, repair, and optimize.

## II. ARBITER

The Arbiter unit shares resources among several potential bus masters in a round-robin fashion. The implementation is done generically, so it can be configured to handle arbitrary number of nodes. It is implemented as an asynchronous unit. A round-robin token passing bus or switch arbiter guarantees fairness (no starvation) among masters and allows any unused timeslot to be allocated to a master whose round-robin turn is later but who is ready now. A reliable prediction of the worst-case wait time is another advantage of the round-robin protocol. The worst-case wait time is proportional to number of requestors minus one. The protocol of a round-robin token passing bus or switch arbiter works as follows. In each cycle, one of the masters (in round-robin order) has the highest priority (i.e., owns the token) for access to a shared resource. If the token-holding master does not need the resource in this cycle, the master with the next highest priority who sends a request can be granted the resource, and the highest priority master then passes the token to the next master in round-robin order.

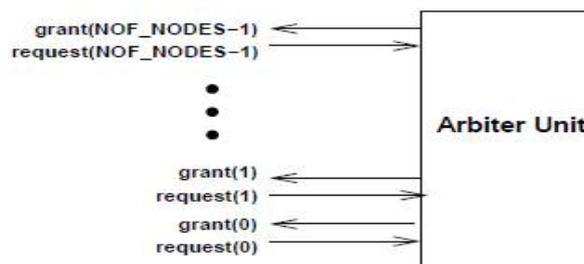


Figure 2: Arbiter interface unit.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

## Multi processor Architecture

- **Moore's law is making the multiprocessor a commodity part**
  1. 500M transistors on a chip, what to do with all of them?
  2. Not enough ILP to justify a huge uniprocessor.
  3. Really big caches that increases, diminishing miss returns.
- **Chip multiprocessors (CMPs)**
  1. Multiple full processors on a single chip
  2. Example: IBM POWER4: two 1GHz processors, 1MB L2, L3 tags
- **Multiprocessors a huge part of computer architecture**
  1. Multiprocessor equivalent of H+P is 50% bigger than H+P
  2. Another entire cores on multiprocessor architecture
    - ❖ Hopefully every other year or so (not this year)

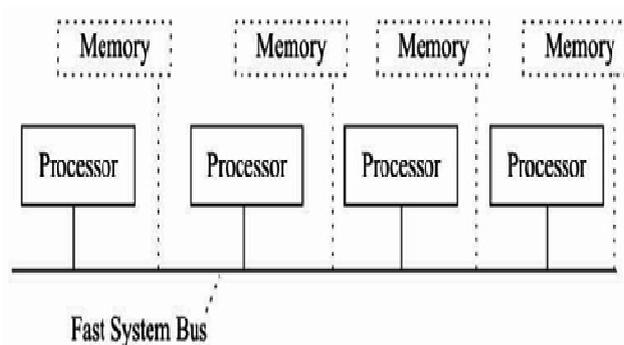


Figure 3: Network Architecture.

- Multiprocessing can be very power efficient
- Recall: dynamic voltage and frequency scaling
- Performance vs. power is NOT linear
- Example: Intel's Xscale
- 1 GHz " 200 MHz reduces energy used by 30x
- Impact of parallel execution
- What if we used 5 Xscale at 200 MHz?
- Similar performance as a 1Ghz Xscale, but 1/6th the energy
- $5 \text{ cores} * 1/30\text{th} = 1/6\text{th}$
- Assumes parallel speedup (a difficult task)
- Remember Ahmdal's law

The memory on each node is a dual-ported, zero wait-state Xilinx block RAM. Memory accesses are made transparent through a wrapper module, which makes it easier to change between different target technologies. Both internal and external communication is managed by a Network Interface (NI). The NI encapsulates the complexity of communication and address translation, acting as a simple MMU.

In distinct usage in distributed computing, the term "Network Architecture" often describes the structure and classification of distributed application architecture, as the participating nodes in a distributed application are often referred to as a "network". For example, the applications architecture of the public switched telephone network (PSTN) has been termed the Advanced Intelligent Network. There are any number of specific classifications but all lie on a continuum between the dumb network (e.g. Internet) and the intelligent computer network (e.g. the telephone network).

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

Other networks contain various elements of these two classical types to make them suitable for various types of applications. Recently the context aware network, which is a synthesis of two, has gained much interest with its ability to combine the best elements of both. A popular example of such usage of the term in distributed applications, as well as PVCs (permanent virtual circuits), is the organization of nodes in peer-to-peer (P2P) services and networks. P2P networks usually implement overlay networks running over an underlying physical or logical network. These overlay network may implement certain organizational structures of the nodes according to several distinct models, the network architecture of the system. Network architecture is a broad plan that specifies everything necessary for two application programs on different networks on an Internet to be able to work together effectively.

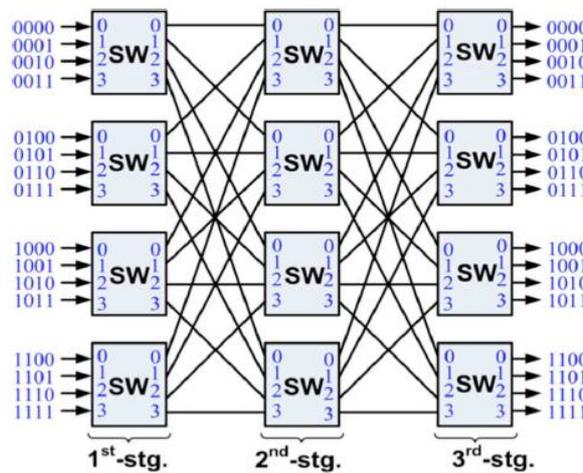


Figure 4: Network architecture with 3 stage pipeline.

Network architecture is the design of a communication network. It is a framework for the specification of a network's physical components and their functional organization and configuration, its operational principles and procedures, as well as data formats used in its operation. In telecommunication, the specification of a network architecture may also include a detailed description of products and services delivered via a communications network, as well as detailed rate and billing structures under which services are compensated. The network architecture of the Internet is predominantly expressed by its use of the Internet Protocol Suite, rather than a specific model for interconnecting networks or nodes in the network, or the usage of specific types of hardware links. The Open Systems Interconnection model (OSI model) is a product of the Open Systems Interconnection effort at the International Organization for Standardization (ISO). It is a way of sub-dividing a communications system into smaller parts called layers.

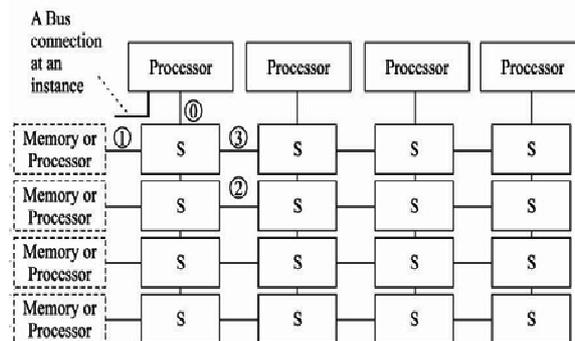


Figure 5: Modified Cross bar switch Multiprocessors network architecture for NoC.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

Each processor has switch to memory bus horizontally and processor-to-switch links vertically. A switch S having four I/O paths (0, 1, 2, 3) provides the following twelve paths (assume even paths horizontal, odd paths vertical): 0-2, 0-1, 0-3, 1-0, 1-2 1-3, 2-0, 2-1, 2-3, 3-0, 3-1, 3-2.

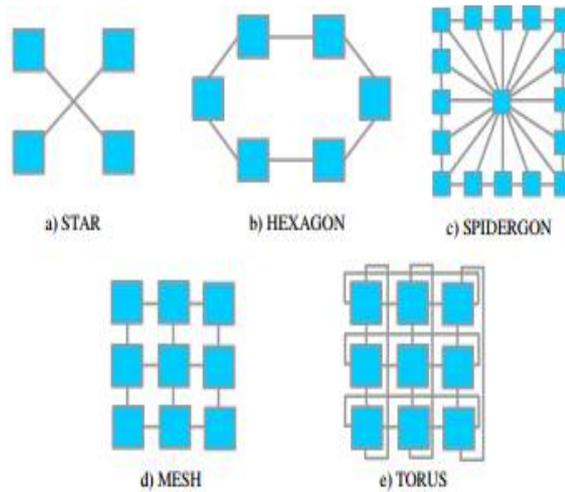
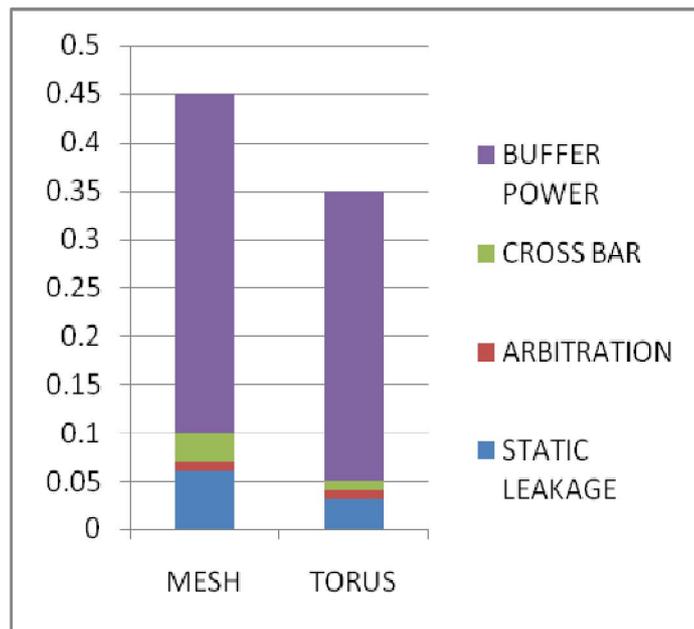


Figure 6: Examples of network architecture

### III. PERFORMANCE CHART



## IV. EXPERIMENTAL RESULTS

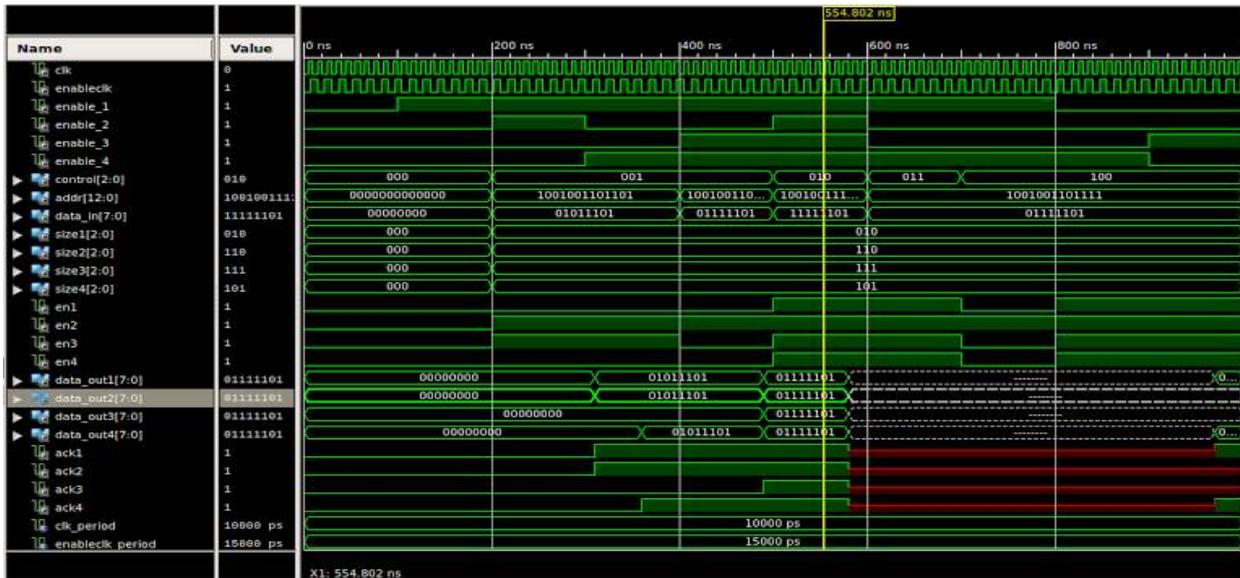
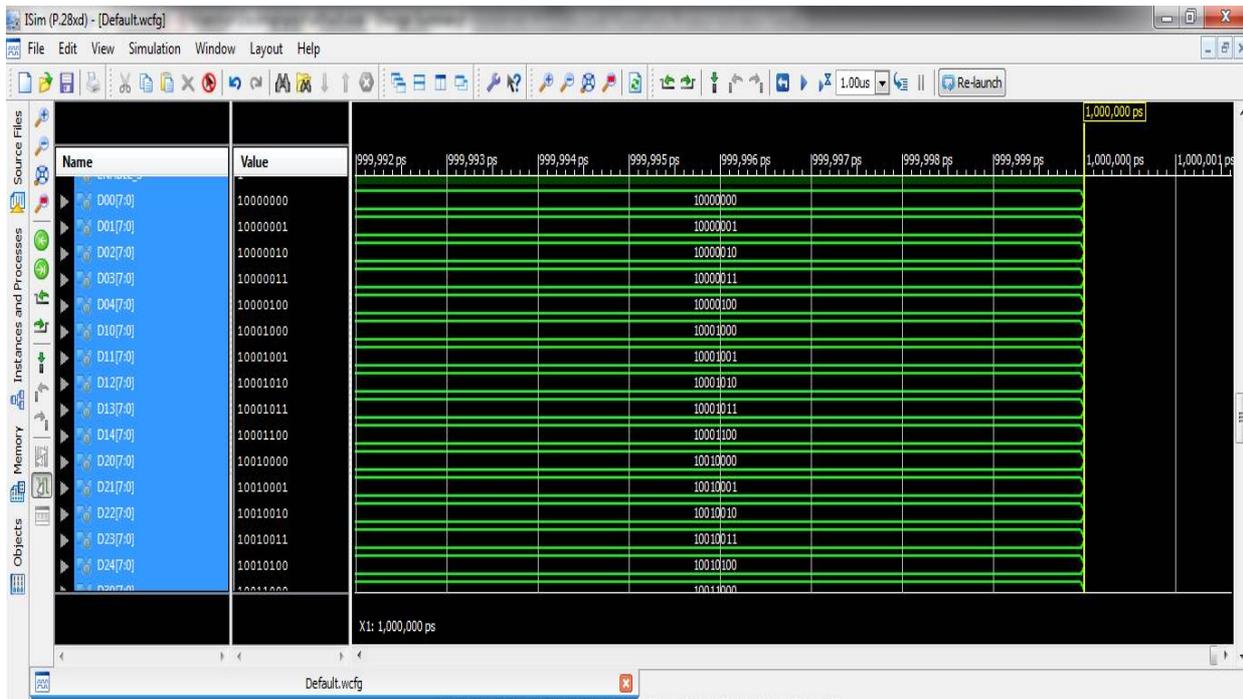
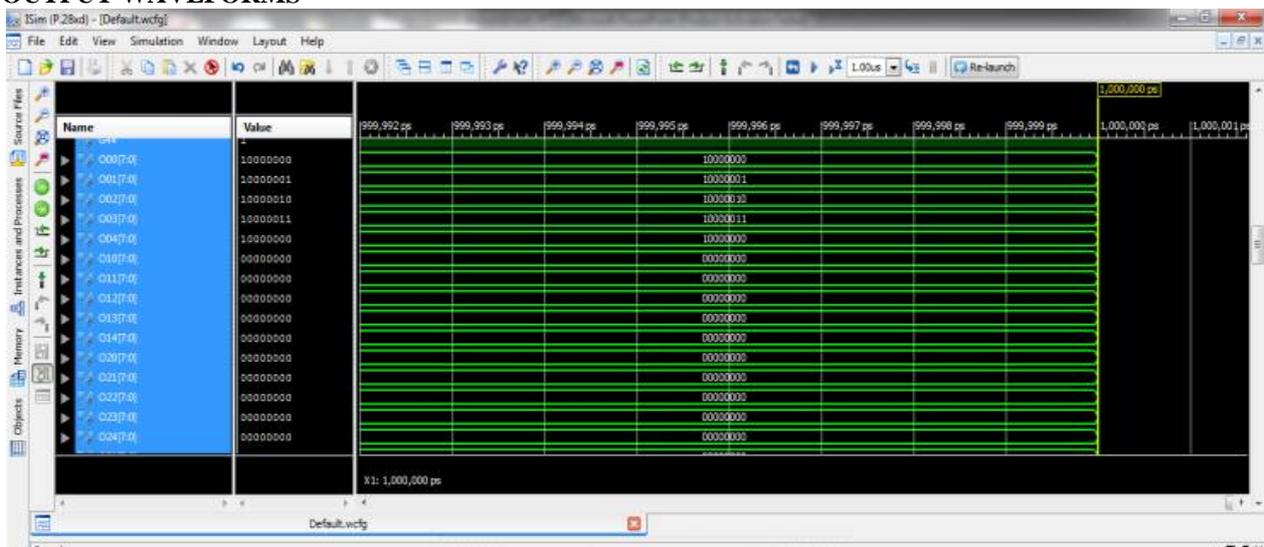


Figure 7: Simulation results for the proposed NoC multiprocessor

## INPUT WAVEFORMS



## OUTPUT WAVEFORMS



Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	7863	5472	143%
Number of Slice Flip Flops	530	10944	4%
Number of 4 input LUTs	15612	10944	142%
Number of bonded IOBs	78	240	32%
Number of GCLKs	1	32	3%

Table 1: Device Utilization

## V. CONCLUSION

This project shows that it is possible to develop (including a whole CPU core) a multiprocessor NoC that fits on a single FPGA in very short time. A platform concept gives rapid design time and keeps the designers focused on debugging the application and not the platform itself. An “all synthesizable” approach is far from feasible in most projects but it is a convenient way of getting rapid results. A system view is crucial when designing NoC. All parts interact and focusing on narrow HW or SW issues does not give a satisfying solution. The platform concept will be dominating until IP companies have solved today’s problems with reusability and interoperability.

## VI. FUTURE WORK

Design for Test is assumed to be implemented with standard tools and design flows. This can be accomplished due to our “all synthesizable” approach. Our scalability relies on a “perfect” interconnect. Today, a shared bus is used, which must be improved in the future with point-to-point/crossbar options in the generator. Software targeting is an important area and was solved by letting the user partition the application at thread level. This can be done automatically with respect to thread communication and code/data size. Inter-processors communication HW support will be integrated from existing research projects at MRTC. With automatic generation of synthesis scripts, a pushbutton design flow seems within reach. Virtual memory and dynamic memory allocation HW support will be added.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 11, November 2016

## REFERENCES

- [1] Phi-Hung Pham, Junyoung Song, Jongsun Park, and Chulwoo Kim, 'Design and Implementation of an On-Chip Permutation Network for Multiprocessor System-On-Chip', IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS, VOL. 21, NO. 1, JAN 2013.
- [2] Sven Eklundavanceraddatararkitektur, Studentlitteratur, Lund, 1994, ISBN 91-44-47671-X.
- [3] Global sources. System-on-a-chip sets new rules in the industryglobal sources MArS 10, 1999.  
<http://www.globalsources.com/MAGAZINE/EC/9905/NOCREP.HTM>
- [4] Bill Cordan, Palm chip Corporation An efficient bus architecture for system-on-chip design Custom Integrated Circuits, 1999. Proceedings of the IEEE 1999 pp: 623{626
- [5] Sibabrata Ray, Hong Jiang. A reconfigurable bus structure for multiprocessors with bandwidth reuse,Journal of Systems Architecture 45, 1999
- [6] Hammond Lance, OlukotunKunle. Considerations in the Design of Hydra: A Multiprocessor-on-a-Chip Microarchitecture, Stanford Technical Report CSL-TR-98-749, Stanford University, 1998.
- [7] Lars-Hugo HemertDigitalakretsar, Studentlitteratur, Lund, 1996, ISBN 91-44-00099-5.
- [8] John L. Hennessy, David A. Patterson Computer Architecture A Quantitative Approach, second edition, Morgan Kaufmann Inc, San Francisco California, 1996, ISBN 55860-329-8.
- [9] Vincent P. Heuring& Harry F. Jordan Computer Systems Design and Architecture, Addison Wwesley,California, 1997, ISBN 0-8053-4330-X.
- [10] Howard Sachs, Mark Birnbaum VSIA Technical Challenges Custom Integrated Circuits, 1999. Proceedings of the IEEE 1999 , 1999 , Page(s): 619 -622
- [11] Geert Roosseel, Sonics Inc. Decouple core for proper integration eeTimes Jan 3, 2000.[www.eetimes.com/story/OEG20000103S0048](http://www.eetimes.com/story/OEG20000103S0048)
- [12] Jon Turino, SynTest Technologies, Inc. emphDesign for Test and Time to Market - Friends or FoesTest Conference, 1999. Proceedings. International, 1999 , Page(s): 1098 -1101
- [13] Lewis, Je Intellectual Property (IP) Components, <http://www.ireste.fr/fdl/vcl/ip/ip.htm>
- [14] OlukotunKunle, Bergman Jules, Kun-Yung Chang and BasemNayfeh.Rationale, Design and Performance of the Hydra Multiprocessor, Stanford Technical Report CSL-TR-94-645, Stanford University,1994.
- [15] RealChip Custom communication Chips.System-on-Chips,<http://www.realchip.com/Systems-on-Chips/systems-on-chips.html>
- [16] Rincon Ann Marie, Cherichitty Cory, Monzel James. A, Sauer David, R, Trick Michael, T.IBM Microelectronics Corp.Core Design and System-on-a-Chip Integration, IEEE Design & Test of Computers. Volume: 14 4 ,Oct.-Dec. 1997, pp: 26{35
- [17] Rincon Ann Marie, Lee William. R and Slattery Michael.IBM Microelectronics Corp.The Changing Landscape of System-on-Chip Design, Custom Integrated Circuits, 1999.Proceedings of the IEEE 1999.
- [18] M. Alho, J. Nurmi: Implementation of interface router IP for Proteo network-on-chip. The 6th IEEE International Workshop on Design and Diagnostics of Electronics Circuits and Systems, Poznan, Poland, 2003.
- [19] T.A. Bartic, J.-Y. Mignolet, V. Nolle, T. Marescaux, D. Verkest, S. Vernalde, R. Lauwereins: Topology adaptive network-on-chip design and implementation. IEE Proceedings – Computers and Digital Techniques, 8 July 2005, Volume 152, Issue 4, pages: 467–472.
- [20] T. Bjerregaard, J. Sparso: A Router Architecture for Connection-Oriented Service Guarantees in the Clockless Network-on-Chip. Proceedings of the Design, Automation and Test in Europe Conference and Exhibition, 2005, Volume 2, pages: 1226–1231.
- [21] M. Dall'Osso, G. Biccari, L. Giovannini, D. Bertozzi, L. Benini: Xpipes: a Latency Insensitive Parameterized Network-on-chip Architecture For Multi-Processor SoCs. Proceedings of the 21st International Conference on Computer Design, 13–15 October 2003, pages: 536–539.

## BIOGRAPHY



Mr. SK. Rahil Hussain was born in Guntakal, Ananthapur (Dist.), A.P, India. He received Bachelor of Technology in Electronics and Communication from ANNA University, Chennai, T.N, India. He has received Master of Technology (VLSI) from K L University, Andhra Pradesh, India. He is a Secretary of International Faculty and Students Multi Research Consortium. His research interests include Analog VLSI Design, Wireless Communication and Sensor Networks, Digital VLSI Design and Low Power Memory Design and Fault Diagnosis. He is working as Assistant Professor in Department of Electronics & Communication Engineering and Coordinator for Polytechnic Program, DVR and Dr. HS MIC College of Technology, Kanchikacherla, Krishna (Dist.), A.P. India. He has Published Nine International Journals and One International Conference.



Mr. P. Jashuva was born in Andhra Pradesh, India in 1991. He received Bachelor of Technology in Electronics and Communication Engineering (ECE) from DJR College of Engineering and Technology in 2013. Presently he is pursuing Master of Technology in VLSI & Embedded Systems Engineering in DVR and Dr. HS MIC College of Technology, Kanchikacherla, Krishna (Dist.), A.P. India. His research interests include field-programmable gate array architecture design, reconfigurable systems and NoC, high performance communicating circuitry and VLSI design.