Research Journal of Engineering Sciences ISSN 2278 – 9472
Vol. **3(2),** 6-11, February **(2014)** *Res. J. Engineering Sci.*

A Novel NOC Architecture for SoC based Ultra Lightweight Crypto-Processor Using Present and Katan Algorithm

T. Blesslin Sheeba[1] and P.Rangarajan[2]
[1]Sathyabama University, Chennai, INDIA
[2]Department of EEE, RMD Engineering College, Chennai, INDIA

Available online at: www.isca.in, www.isca.me
Received 24th September 2013, revised 14th January 2014, accepted 6th February 2014

Abstract

The performance computation of the Crypto-processor implemented on SoC platform is research of interest now. The traditional buses or wires have the problem of scalability, complexity and timing, from here we target this point and present a novel Network On Chip (NOC) architecture to overcome the cons. Network On Chip (NOC) consists of storage and I/O resources interconnected by network of switches for network computation. Two ultra lightweight cryptographic algorithms are presented in this paper namely PRESENT and KATAN. It is developed using Altera Cyclone IV E. The NOC architecture consists of different topology, switching and routing techniques based efficiency requirements. Finally the computed efficiency for the processor running at 330 MHz and taking 5.047 sec for computation using 0.323 mm^2 cell area in 180 nm technology.

Keywords: Present, Katan, Network On Chip (NOC), System On Chip (SOC), Cyclone IV E.

Introduction

Several years back the cryptography is introduced as evolution of secure communication. That is data or information shared must be protected from the trespassers (i.e. from unwanted users). The cryptography is the process of encryption and decryption taken place at transmitter and receiver end respectively. Lot of algorithm has been proposed for encryption and decryption process[1]. In recent year there are several Ultra lightweight Cryptography has been proposed in advance to lightweight cryptography. In contrast to software implementation which will take more resources and computational time the hardware implementation also proposed but it has the demerits of compactness, soothe concept of crypto-processor was evolved by implementing it in SoC.

The concept of SoC is evolved as solution for implementation of more complex algorithm and design with more core processor. The rise of Network On Chip (NOC) is due to the growing complexity interconnection design and chip architecture[2,3]. The design and implementation of computational intensive cryptographic algorithm in Hardware is a tough task but the disadvantage of compatibility and flexibility is experienced. To succeed this problem the ultimate choice is a multi core platform called as SoC which will maximize the computational power and flexibility. The three conceptual thinking if we consider an SoC is design time decision, computation, storage and I/O. 64 core and 80 core have been in corporate in the single chip which is presented in the articles[4,5] respectively, However this rapid increase in the number of core in SoC implementation will lead to severe system degradation caused by bus architecture for inter connections.

If we consider the Crypto-processor implementation using SoC means the bus constraints will result message stalling and signal interference etc. Thus Network On chip (NOC) is preferred as global solution for intra SoC communication. The Network on Chip architecture is made of network adaptor, routing nodes and links. The network adaptor is to interface the core to NOC and decouple the core from communication, the routing node will route the data from source to destination according to the protocol, and finally the link will connect the different nodes. There are several parameters that will decide the computational efficiency of the processor with Network On Chip (NOC). If we design NOC architecture then we have to consider the topology, routing techniques and switching method which will contribute to the QoS of the architecture.

Ultra Lightweight Cryptography Algorithm: Present: Algorithm: The plaintext P of 64-bit length is defined by $b_{63} \ldots b_{64}$ which is initial state. The three stages addRoundkey, sBoxLayer, pLayer of Present algorithm will change the initial state in every iteration i for $0 \leq i \leq 15$. The operation of addRoundkey is that it will XOR the current state $b_{i,63} \ldots b_{i,0}$ with round key of i^{th} term as $RK_i = rk_{i,63} \ldots rk_{i,0}$ follows.

The non-linear sBoxLayer which consists of 16 copies of a 4-bit S-box is the second stage. The S (w_i) is applied to each S-box for $w_{15}, \ldots, w_0$ where w_i is defined by

$$w_i = b_{4*i+3} || b_{4*i+2} || b_{4*i+2} || b_{4*i+1} || b_{4*I}$$

The 64 most significant bits (MSB) of the current state of the key register K is the current round key RK_i. By shifting the key register $K= K_{127}k_{126}\ldots k_1k_0$ to the left by 61 bits and passing the left most 8-bit through two S-boxes of present we can generate the key for next round i+1. The 5 bit round counter is XORed

with 5-bit $k_{66}k_{65}k_{64}k_{63}$. The round key bit $RK_{i+1} = k_{127} \ldots k_{64}$ is formed by resultant 64 of MSB.

Architecture of Present: The scaling of 64-bit implementation to 16- bit implementation will lead to implementation will lead to area reduction. Further scaling will lead to degradation in throughput with small amount of area reduction due to complexity operation of permutation. The top level of datapath is shown in figure 1.

Data storage: The shift register SR1 is stored with $b_{63} \ldots b_0$ states. In one clock cycle it performs a 16-bit circular left shift. The 16, 4-bit clock is combined and consider as 64-bit shift register. For the round operation the 16 MSB are tapped out of SR1.

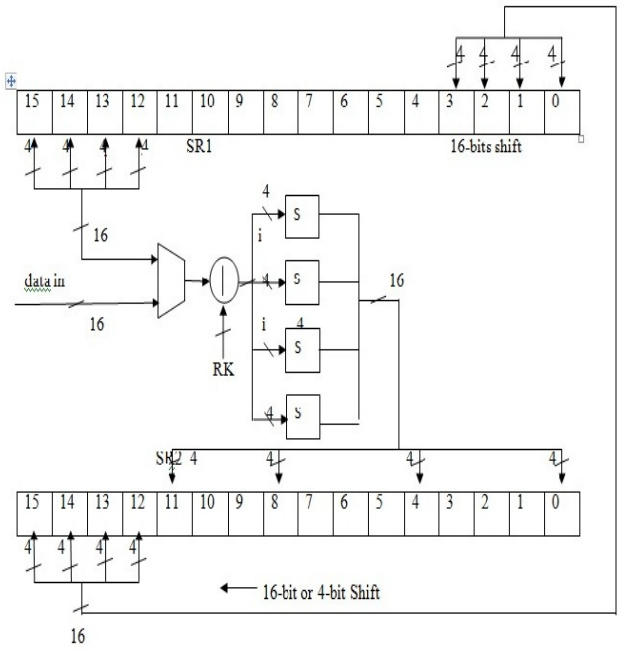

Figure-1
16-bit Datapath of Present

S-box Implementation and Permutation layer: The round key RK_i is XORed with incoming data as result of round operation initialization and stored in four S-boxes. The architecture utilizes four S-boxes for round operation and two for key scheduling. SR_2 is responsible for implementation of permutation operation which perform 4-bit shift during round operation and 16-bit shift while copy the content to SR_1. The input to the block position 12, 8, 4 and 0 of SR_2 is taken from 16-bit S-box. 16 MSB of SR_1 is used to compute the 4-bit block of 15, 11, 7 and 3 during first clock cycle of round operation. The block 14, 10, 6 and 2 are computed by shifting the SR_2 by 4-bits in the next subsequent cycle. To complete the round function the above operation is continued for another two cycle as a result 8 clock cycle for each round operation is achieved.

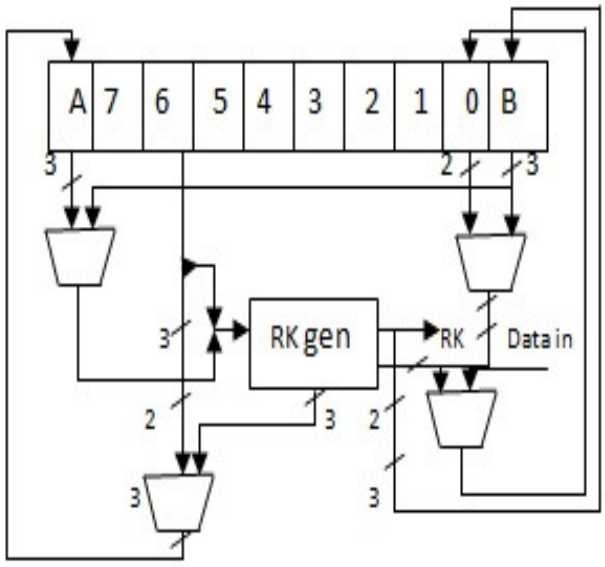

Figure-2
Key Scheduling of Present

Key Storage and Scheduling: 128-bit shift register is used to store the key which performs 16-bit circular left shift. During the process of first four clock cycle by tapping the 16 MSB from key and passing them to RKGen the key RK_1 for first round is obtained. The three extra taps are placed as shown in figure 2 to shift the 61 bits by 16-bit. The register A is used to store the lost 3bits during first round. Register A and 13MSB from the key are passed to RKGen for subsequent round key. Two S-boxes were stored in RKGen foe S-box operation, to compute the XOR with round counter a 5-bit XOR is needed and to choose the appropriate bit for round key generation multiplexers are contributed.

Katan: Katan family consists of three block ciphers with various block sizes: 32, 48 and 64 bits. All ciphers have 80-bit keys. Each of the Katan algorithms loads the data block into two internal shift registers L1 and L2. Using nonlinear functions, it performs the 254 rounds which form the registers feedback figure3. One of nonlinear functions uses specific irregular value (IR) in addition to several register bits. It depends on the round number. The requirements of katan are extremely low because of the following collection in factors: i. Katan uses shift registers, which can be implemented easily; feedback functions are very simple too though they provide required nonlinearity, ii. it processes small blocks of data – 32 to 64 bits, iii. Its internal state is small and its size is a little bit greater than the block size.

Block cipher of katan can be used as a cryptographic kernel of mounting other kinds of cryptographic primitives over it. The set of cryptographic functions over katan was recently proposed in[6]. This set includes: i. block cipher – katan algorithm itself, ii. Pseudo random number and stream cipher generator, iii. Hash function.

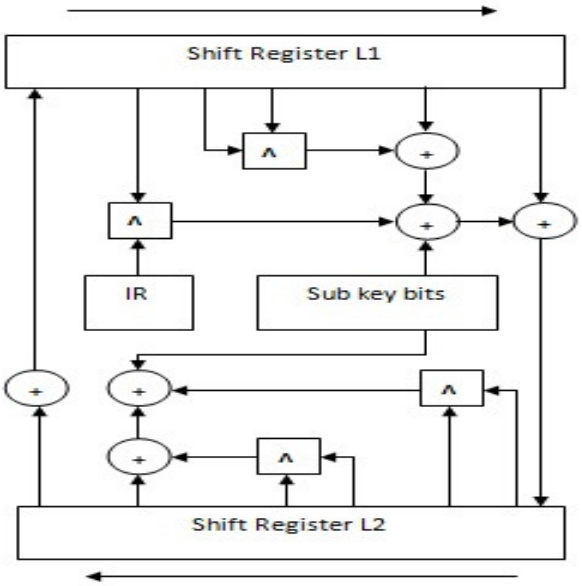

Figure-3
Round Function of Katan

To minimize expenses, the hashing add-on should be as lightweight as possible. One of hash function with a thin hashing layer over the internal block cipher is crunch algorithm, which took part in the first stage of SHA-3 contest. One of crunch versions is based on the double-pipe Merkle-Damgård construction. The double-pipe version allows reaching higher cryptographic strength comparably to the main version with practically the same overheads[7]. Using the compression function structure similar to crunch (strengthened version) and the 64-bit katan64 block cipher, we can build a lightweight compression function figure-4.

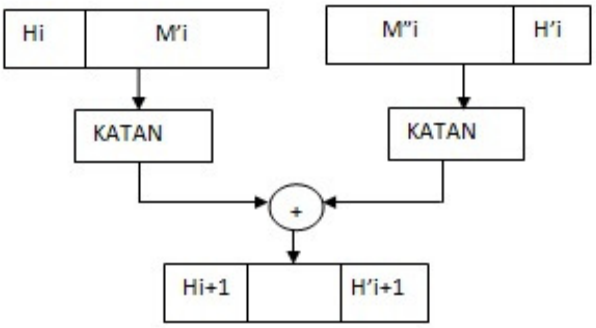

Figure- 4
KATAN 64-based Compression Function

Compression function of double-pipe crunch version encrypts every block of the message twice: concatenated with Hi and H,i values. We slightly modified the structure of the crunch compression function: as the block size of the KATAN64 cipher is relatively small, every block of the message is separated into two halves: M'i and M''i, which is processed by the block cipher in parallel. Final hash value is a result of the final transformation of HN and H'N values (last message block processing output values).

Proposed System: SoC implementation of Crypto-Processor using NOC: The figure-5 sows an overall architecture for ultra lightweight crypto-processor implemented in System On Chip (SoC) using Nios II processor with NOC for interconnection of cores or different modules. The two algorithm presented here are present and katan. The algorithm was stored in SRAM which is used by processor for different encryption and decryption of data based on the application, the control logic for selecting the cryptographic techniques is stored in processor internal memory. The DMA controller is used to access the processor during the traffic of packet occur.

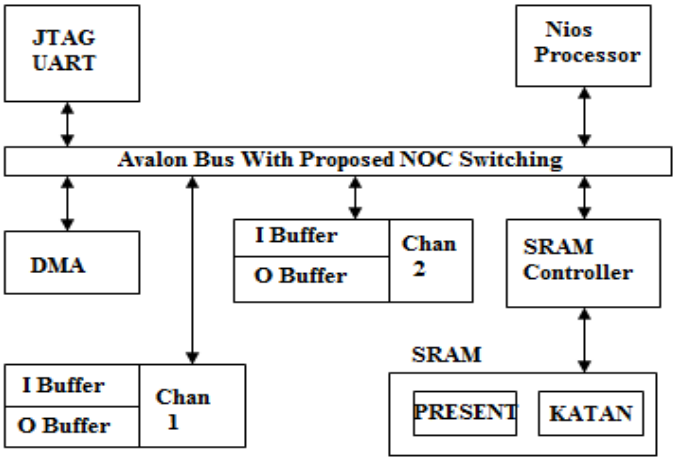

Figure-5
Crypto-Processor using NOC

The figure 6 shows the general architecture for System on chip connected with certain protocol for routing, switching and topology called as NOC.

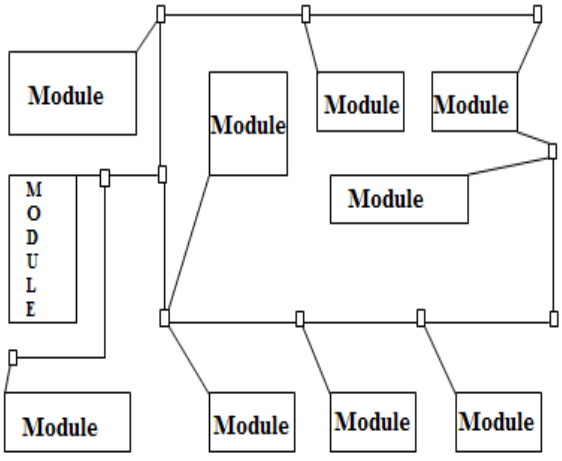

Figure-6
General NOC Architecture

It is clear from the block that even an IP core also consists of different type's module as shown. The key role of network On Chip comes here the placement of the block should be placed in such a way that routing of signal should be simple and should not cause any propagation delay and the latency be maintained. The topology must be selected in a way so that more number of nodes can be inserted without interference of the signals and node failure will not cause packet error. Finally the switching should be chosen for effective data transfer. Hence the before said parameters like Topology, Routing, Switching for proposed Network On Chip is discussed in the next section.

Proposed NOC Architecture: The below figure 7 shows cryptographic process of data with NOC interconnect.

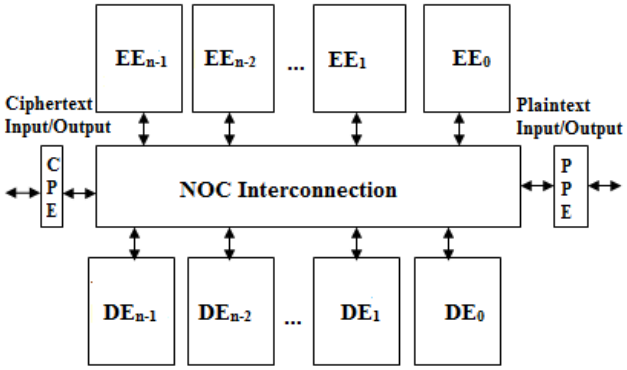

Figure-7
Proposed NOC Architecture

The present architecture is based on Packet switching for data cryptography. As shown in architecture it has EEs (Encryption element), DEs (Decryption Element), CPE (Ciphertext Processing Element), PPE (Plaintext Processing Element) are connected via Network On Chip for encryption and decryption of data sent to NOC, the number of EE and DE is not restricted for hardware implementation. From figure 7 it is clear that PPE at the right side of the architecture is responsible for two jobs, they are slicing the input plaintext into N units and header should be added to the pack for sending them to each EE then it receive the decrypted packets from DEs and remove the headers before sending it to NOC. On the other hand CPE will do the same process but with the Ciphertext.

Generally the NOC architecture is made of certain protocol for effectiveness, in this paper we are considering very two important parameters namely topology and switching. The 2D-mesh topology and wormhole switching are implemented in our novel NOC.

2D-MESH Topology: In this types of topology each of the processing element are interconnected with one another. Every node in this set up not only sends the bit but also relay data from the other Processing Element. The above figure 8 shows the Number of Processing Element (i.e. Encryption Element, Decryption Element) connected in a 2D-mesh network. The description of PEs and Des are explained in previous section. The added advantages of using the 2D-mesh topology is that data can be transmitted from different processing element simultaneously which is efficient in terms for crypto-processor. Modification of the topology in reconfigurable techniques is done easily. There is increase proficiency to find the isolation and detection of error. It is very secure since dedicated line is provided

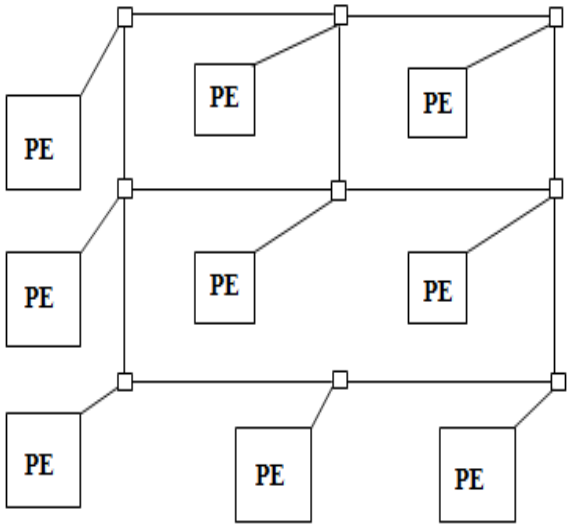

Figure-8
2D-MeshTopology

Wormhole Switching: The proposed NOC will rely on Packet Switching, so here we consider Wormhole Switching because it has the advantages like the complete paper need not to be stored in the switch while waiting for the header flits to be routed to next stage, it not only reduce the delay but also need small buffer space. Channel allocation and bandwidth are decoupled. The process of wormhole switching is shown in below figure 9.

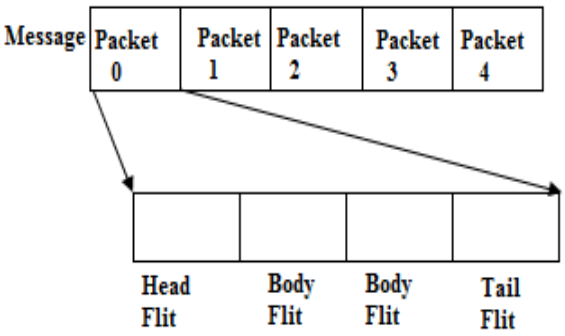

Figure-9
Process of Wormhole Switching

The main term which should be considered during Wormhole switching is FLITS (**F**low **C**ontrol Dig**its**). The large packet in the process are broken and called as flits, the flits are arranged

as Head flit, body flit followed by tail flit as shown fig. during the transmission of flit the whole body of flit which approached first is transmitted and then the next (i.e.) the interference of the flit is not allowed. It is simply for control the flow of plaintext or cipher text during encryption and decryption process. Here in crypto-processor the data (plaintext) from the EE is given to PPE and the data (ciphertext) from the DE is given to CPE by the NOC With the help of these types of flit controlled Packet switching.

Results and Discussion

The various result of the proposed NOC architecture for the crypto-processor implemented on SoC is discussed below. The table 1 shows the comparison for operation time based on the NOC architecture over the conventional techniques, and found to be more efficient as the time was reduced than the conventional one.

Table-1
Operation Time

Types	Operation Time (Sec)
Conventional Architecture	6.219
NOC Architecture	5.047

The table 2 provides the comparison chart for the proposed NOC Architecture computed over a different parameters namely technology size, cell area and data frequency. It is found that three different technologies have been taken In to account as 130 nm, 90nm and proposed NOC in 180 nm and we conclude as technology scale reduces the efficient in terms of area and frequency is increased.

Table-2
Comparison Table for Proposed NOC

NOC Prototype	[8]	[9]	Proposed NOC
Technology Size	130nm	90nm	180nm
Total cells Area	0.260 mm^2	0.082 mm^2	0.323 mm^2
Data Frequency	500 MHz	500 MHz	330 MHz

The computation of the latency is found and the comparison graph for the same is shown in figure 10. The graph shows that the latency of the packet was increased in the conventional architecture that should be less which is achieved in the proposed NOC architecture. Finally efficient computation of the topology used in the proposed architecture was done and the comparison was done between the mesh and 2D-mesh topology. The Gate count for 2D-mesh was found to be far less than the traditional topology, and dynamic power consumption is increased in the conventional topology and reduced in better manner in our proposed architecture.

Table-3
Comparison Table for Topology using Cyclone 1V E

FLITS	Mesh		2D-Mesh	
	Gate Count	Dynamic Power (mW)	Gate Count	Dynamic Power (mW)
2	32750	11.53	18368	6.99
4	46598	20.40	28654	12.29
8	75382	36.88	47038	22.32
16	134479	69.45	85362	42.03
32	250559	134.36	163330	81.27
64	490820	261.65	316173	159.33

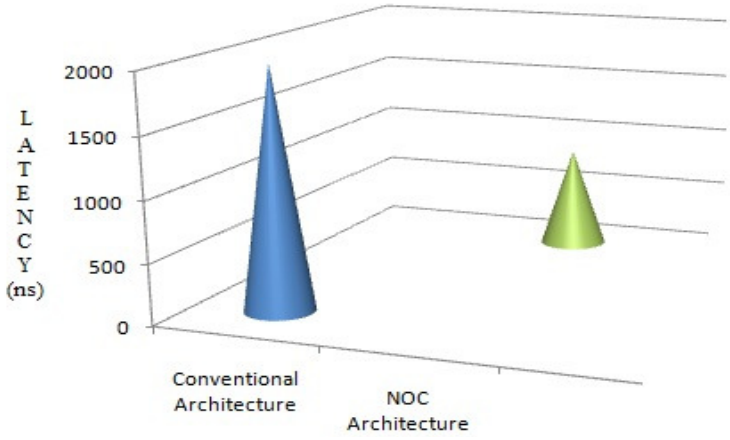

Figure-10
Comparison graph for Latency

Conclusion

We develop a Crypto-Processor implemented on System On Chip (SoC) using novel Network On Chip (NOC) which is implemented with 2D-Mesh Topology, Wormhole Switching, and effective routing techniques for Cryptographic application. Then demonstrate it performance base on the parameter like Latency, Throughput, Area, Frequency and technology. Finally we find the proposed approach is efficient in terms of cycle time, throughput and area.

References

1. Advances in Ultralightweight Cryptography for Low-cost RFID Tags: Gossamer Protocol Pedro Peris-Lopez, Julio Cesar Hernandez-Castro, Juan M. E. Tapiador, and Arturo Ribagorda **(2009)**

2. W.J. Dally and B. Towles, Route Packets, Not Wires: On-Chip Inteconnection Networks, in *the 38th annual Design Automation Conference*, 684-689 **(2001)**

3. Guerrier P. and Greiner A., A Generic Architecture for On-Chip Packet-Switched Interconnections," in *Design, Automation and Test in Europe (DATE)*, 250-256 **(2000)**

4. S. Bell et al., "TILE64 Processor: A 64-Core SoC with Mesh Interconnect," Solid-State Circuits Conference, 2008. Digest of Technical Papers. IEEE International, 88-598 **(2008)**

5. S. Vangal et al., An 80-Tile 1.28TFLOPS Network-on-Chip in 65nm CMOS, Solid-State Circuits Conference, 2007. Digest of TechnicalPapers. IEEE International, 98-589 **(2007)**

6. S. Panasenko and S. Smagin. Energy-efficient cryptography: application of KATAN. SoftCOM 2011, 19 International Conference on Software, Telecommunications and Computer Networks. Split – Hvar – Dubrovnik, September 15-17, 2011, Proceedings (SS2 – Special Session on Green Networking) **(2011)**

7. E. Volte. CRUNCH. A SHA-3 Candidate. // Available at http://www.voltee.com – 27 February 2009 **(2009)**

8. E. Rijpkema, K. Goossens, A. Radulescu, J. Dielissen, J. van Meerbergen, P. Wielage and E. Waterlander, Trade-offs in the design of a router with both guaranteed and best-effort services for networks on chip, *IEE Proc. Computers and Digital Techniques,* **150(5),** 294-302 **(2003)**

9. M. Panades, A. Greiner and A. Sheibanyrad, A Low Cost Network-on-Chip with Guaranteed Service Well Suited to the GALS Approach, *Proc. the 1st Int'l Conf. and Workshop on Nano- Networks),* 1-5, **(2006)**