Escape Path based Irregular Network-on-chip Simulation Framework

Naveen Choudhary
College of technology and Engineering
MPUAT
Udaipur, India

M. S. Gaur
Malaviya National Institute of Technology
Jaipur, India

V. Laxmi
Malaviya National Institute of Technology
Jaipur, India

ABSTRACT

Network-on-Chip(NoC) has been proposed as a solution for addressing the communication infrastructure design challenges of future high-performance nanoscale architecture of SoCs. IrNIRGAM is a discrete event, cycle accurate simulator targeted at irregular topology based Network on Chip (NoC) research. The generic, modular, and extensible framework of IrNIRGAM provides substantial support to experiment with direct network based NoC designs in terms of routing algorithms, applications on various topologies and related performance parameters such as throughput, communication latency etc. IrNIRGAM is written in SystemC and C++. Topology represents the most important characteristic of NoC architectures and essentially defines the physical interconnection of the router nodes. In IrNIRGAM, input buffered routers can have multiple virtual channels (VCs) and uses wormhole switching for flow control. The packets are split into an arbitrary number of flits (flow control units) and forwarded through the network in a pipelined fashion. A Round-Robin scheme for switch arbitration is used in the router nodes to provide fair bandwidth allocation while effectively preventing scheduling anomalies like starvation.

General Terms

Network-on-Chip, System-on-Chip, Interconnection Networks, Embedded Systems, VLSI

Keywords

Network-on-Chip; Simulation; System-on-Chip; IP Core; Interconnection Networks

1. INTRODUCTION

The greatest challenge in designing today's Systems-on-Chip (SoCs) comes from global interconnects. The global interconnects tends to increase the complexity of the the SoC due to the growing number of devices on the same chip, in addition to causing severe synchronization errors, unpredictable delays and high energy consumption. As a consequence, it has been suggested to replace these custom wires with structured Network-on-Chip [1, 2, 3]. Typical SoCs that implement the Network-on-Chip (NoC) approach consist of a number of heterogeneous devices such as CPU or DSP cores, embedded memory and application specific components that communicate using packet switching. One of the most important phase in designing a NoC is the topology or structure of the network along with setting of various design parameters (such as frequency of operation, link-width, etc). Early works [1, 2] in this research domain favored the use of standard topologies such as meshes, tori, k-ary n-cubes or fat trees under the assumption that the wires can be well structured in such topologies. However most SoCs, especially application-specific SoC, are heterogeneous, with each core having different size, functionality and communication requirements. Thus, customized irregular topologies tailor made according to the application are more suitable in such scenario.

Numerous Network-on-Chip simulations Framework have been developed in the Network-on-Chip research domain. Noxim [5] NoC simulator is one such NoC simulator, which allows to experiment with various routing algorithms on different traffic distributions, but can support only mesh topology. Moreover sharing of physical channel bandwidth with the help of virtual channels is absent. Nostrum-NNSE [5] is another such NoC simulator. BookSim [6] developed at Stanford University, supports broad range of network architectures including flattened butterfly with various routing algorithms but with limited number of traffic injection models. NIRGAM [7, 8] developed at MNIT, Jaipur, India in collaboration with University of Southampton, UK is a NoC simulator which provide extension flexibility and modularity for traffic, algorithms, applications, and functionality for existing architectures. NIRGAM is aimed primarily at the NoC research community wherein it provides researchers with convenient and efficient mechanism to experiment with NoC design in terms of routing algorithms and applications on various topologies. Users can easily plug-in their own routers and applications. The simulator is capable of dynamically loading a router and attaching any user-specified application library to any core. However in its present form NIRGAM supports regular Mesh and torus topologies with source, deterministic and adaptive routing. However the support of distributed table based routing is presently absent in NIRGAM. However generic regular topologies can become irregular for supporting over sized region or due to faults in switches or links. Moreover for most application specific NoCs, the customized irregular topology according to the communication requirement of the application under consideration are considered appropriate for optimized network performance. To address the support of irregular topologies, the NoC simulator NIRGAM[7, 8] is extended to develop IrNIRGAM. Routing in irregular topologies can be performed by using source routing or distributed routing [9]. In source routing, the source node specifies the routing path on the basis of deadlock-free routing algorithm (either using table lookup or not). The computed path is stored in the packet

header, being used at intermediate nodes to reserve channels. IrNIRGAM supports source routing but the source routing has the drawback of having packets with long length headers leading to additional consumption of network bandwidth as packets are required to contain the whole routing path in source routing.

For efficiency reasons most irregular topology NoCs have proposed the use of table based distributed routing [10]. In distributed routing, each intermediate node has to make a routing decision based on the table entries in the node characterizing the local knowledge of the networks. By repeating this process at each intermediate node, the packet should be able to reach its destination. The routing algorithm running on each switch may use only the addresses of the current and destination nodes to compute the path (deterministic routing) or may also use information collected from other nodes about traffic conditions in the network (adaptive routing). The major issue in designing routing algorithm is to avoid or recover from deadlock. Most of the proposed routing algorithms for irregular topologies use deadlock avoidance for efficiency and better resource utilization in comparison to deadlock prevention & deadlock recovery.

Prominent examples of topology agnostic table based distributed routing algorithms are up*/down* [11], Left-Right [12], L-turn [12], DOWN/UP [13], prefix-routing [14], smart-routing [15], and FX [16]. These algorithms have in common that they are based on turn prohibition, a methodology which avoids deadlock by prohibiting a subset of all turns in the network. However, in some cases, these routing functions are not able to supply any minimal path between some pairs of switches. In [17] a generic methodology based on escape path is proposed to extend such routing function for supporting greater adaptivity and increased use of minimal paths. The methodology proposed by silla et al. [17] allows messages to follow minimal paths, in most cases, reducing message latency and increasing network throughput. Moreover the methodology enforces the deadlock free route to be followed only when the minimal path is occupied by other traffic/packet. This methodology assumes that all the physical channels in the NoC can be split into two virtual channels i.e. original virtual channel and the new virtual channel. Moreover the presence of a given deadlock free routing functions based on turn prohibition [18] for the given irregular NoC is also assumed. The methodology further proposes to extend the given routing function in such a way that newly injected messages can use new channels without any restriction as long as the original channels are used exactly in the same way as in the original routing function. The modified routing function allows a packet arriving on a new channel following minimal path to be routed to any channel without any restrictions but preferably with higher priority to new channels as new channel assures minimal paths and higher adaptively (flexibility). If no new channels are available due to congestion, one of the original channels following the original routing function must be provided. However, once a packet acquires an original channel it is not allowed to do transition to a new channel anymore to avoid deadlock situation.

IrNIRGAM incorporates distributed table based routing for supporting deadlock free routing in irregular Network-on-Chip. Deadlock free deterministic as well as adaptive routing based on escape paths are supported in IrNIRGAM. Each router/node in the NoC has a table of the format <source_tileid, destination_tileid, next_tileid> to support deadlock free deterministic routing and a table of the format <source_tileid, destination_tileid, next_tileid, virtual_channel_no.> to support deadlock free adaptive routing based on escape paths. The virtual channel number basically shows whether the minimal path or deadlock free path is being followed by the packet. These tables can be filled offline according to any of the deadlock free topology agnostic routing algorithms such as up*/down* [11], Left-Right [12], L-turn [12], DOWN/UP [13], and prefix-routing [14], etc. Section 2 elaborates the escape path based topology agnostic routing. Section 3 presents the IrNIRGAM architecture along with performance metric and functionality of various modules of IrNIRGAM. Section 4 presents experimental results on IrNIRGAM and finally we conclude in Section 5.

2. TOPOLOGY AGNOSTIC NoC ROUTING

Up*/down* [11] routing algorithm is distributed in nature, and is implemented using table-look-up. The routing table in each switch must be established before data packets can be routed. To do so, a breadth-first spanning tree (BFS) on the Interconnection topology graph N is computed. Up*/down* routing is based on an assignment of direction to the operational links/channels, including the ones that do not belong to the tree. In particular, the "up" of each link is defined as end connected to 1) the router node which is closer to the root in the spanning tree or 2) the router node with lower ID, if both ends are connected to router nodes at the same tree level. Links looped back to the same router node are omitted from the configuration. The result of this assignment is that each cycle in the network has at least one link in the "up" direction and one link in the "down" direction. To eliminate deadlocks in up*/down* routing, a legal route must traverse zero or more links in the "up" direction followed by zero or more links in the "down" direction. Thus, cyclic dependencies between channels are avoided because a message cannot traverse a link along the "up" direction after having traversed one in the "down" direction. Such routing not only guarantees deadlock-freedom, but also provides adaptivity owing to existence of multiple valid paths using "up/down" rule. The route look-up tables can be constructed to support both minimal and non-minimal adaptive routing [9]. Minimal routing usually provides better performance because messages occupy fewer resources on average. However, in some cases, up*/down* routing may not be able to supply any minimal path between some pairs of switches. In [17] Silla et. al proposed general methodologies for the design of adaptive routing algorithms for networks with irregular topology. Routing algorithms designed according to these methodologies allows messages to follow minimal paths in most cases, reducing message latency and increasing network throughput. Given an interconnection network and a deadlock-free routing function defined on it, it is possible to duplicate all the physical channels in the network, taking advantage of spare switch ports or by splitting the physical channels into two virtual channels. In both cases, the graph representation of the network contains the original and the new channels. Silla et. al. [17] then propose to extend the routing function in such a way that newly injected messages can use new channels without any restriction as long as the original channels are used exactly in the same way as in the original routing function. In this paper original channels are

made to use up*/down* deadlock free routing function and new channels are allowed to follow the shortest available path to the destination. The modified routing function allows a packet arriving on a new channel following shortest path to be routed to any channel without any restrictions but preferably with higher priority to new channels as new channel assures shorter paths and higher adaptively (flexibility) . If no new channels are available due to congestion one of the original channels following up*/down* must be provided. However, once a packet acquires an original channel following up*/down* path, it is not allowed to do transition to a new channel anymore to avoid deadlock situation. Moreover to increase adaptivity and to ensure that most packets follow the minimal path, the new packet entering the network can only leave the source switch by using the new channels that provide a minimal path toward the destination.

3. IrNIRGAM ARCHITECTUR AND FUNCTIONALITY

IrNIRGAM supports various topologies, routing algorithms and outputs performance metrics in the form of latency and throughput.. Wormhole switching is used for flow control. Each packet is divided into flits (Flow Control Units). The first flit is known as the head flit and contains information about destination, last flit is known as tail flit while intermediate flits are data flits. IrNIRGAM stands for Irregular topology based NoC Interconnect Routing and Application Modeling. The modular structure of IrNIRGAM allows for easy integration of various routing algorithms and applications.

3.1 IrNIRGAM Architecture

Figure 1 shows the overall architecture of IrNIRGAM simulator. Various parameters such as topology type (Irregular/Mesh/Torus) and size, simulation and traffic generation cycle, routing algorithms and buffer size are provided to the simulation engine through configuration files. Application cores and routing algorithms can be dynamically attached to core engine at the run time, this allows for modification of traffic and routing without the need to compile the code again. The output performance metrics are also generated to plot graphs using GnuPlot. A flit in IrNIRGAM broadly contains the fields: packet type, packet header, virtual channel id, source tile and simulation data containing the information regarding packed identifier, flit identifier, flit type, flit header etc, virtual channel identifier to facilitate wormhole switching, communicating source tile identifier and simulation related data such as various timestamps, number of switches traversed by the flit etc.

IrNIRGAM core engine implements Network-on-Chip. NoC is modeled as a network of tiles and is represented by module "NoC". Each tile is represented by module "NWTile". The major components of each tile and the modules that implement them are as follows:

- Input Channel Controller (IC): Represented by module "InputChannel". Each tile consists of one IC for each neighbor, and an IC for ipcore. For example, a tile having 4 neighbors will have 5 ICs: IC-0, IC-1, IC-2, IC-3 and core IC-4. Each IC consists of one or more virtual channels (VCs). Each VC consists of a fifo buffer.
- Controller: Represented by module "Controller". Each tile consists of one Controller which implements router to service routing requests from all ICs of the tile.
- Virtual Channel Allocator(VCA): Represented by module "VCAllocator". Each tile consists of one VCA that services requests for virtual channel allocation from all ICs of the tile.
- Output Channel Controller (OC): Represented by module "OutputChannel". Each tile consists of one OC for each neighbor, and an OC for ipcore. Each OC consists of an array of registers r_in[], one for each input port.

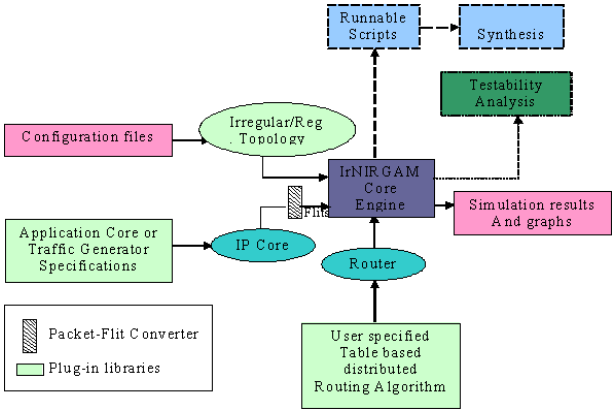


Fig 1: IrNIRGAM Simulation Framework

ipcore: Represented by module "ipcore". Each tile consists of an ipcore (IP) to which an application or traffic generator can be attached.

3.2 Topology

Topology defines the way various processing nodes are placed and interconnected to each other in a network. All the topologies are characterized by the following properties

1. Diameter: The maximum distance between two nodes in the network.
2. Bisection width: The minimum number of wires that must be cut when the network is divided into two equal set of nodes.
3. Node Degree: Number of channels connecting a node to its neighbors.

For improved performance and to satisfy resource constraints, the topology with a small diameter, small node degree and large bisection width are generally preferred. IrNIRGAM supports irregular topology with upper limit of node degree as 16 and maximum number of allowed nodes as 128. These limits allow IrNIRGAM to incorporate wide spectrum Irregular topology as well as regular topology as with the help of generic irregular topology we can design any regular topology also.

3.3 IrNIRGAM Performance

Throughput: Bandwidth of a communication network is one of it's important characteristics and is a measure of the number of

bits traveled across a network per second. Throughput is an apposite metric when the rates at which the packet sent across the network is taken under consideration. Throughput can be quantified in two different ways like throughput per output channel of a node and average throughput of the whole network. In other words throughput is a measure of the fraction of maximum load that the network is capable of handling. Moreover in IrNIRGAM, throughput (in Gbps) is also calculated for each output channel of a router separately by considering the number of flits passed through it. Using this, average throughput for each router and the entire network can also be estimated.

Latency: Latency of a packet can be defined as the delay between the injection of its header flit at the source node and the reception of its tail flit at the destination node. Instead of defining latency of every flit or packet, average latency of the entire network is of more significance. In NIRGAM, average latency (per packet and per flit) is estimated for each output channel as well as for the entire network in clock cycles.

4. EXPERIMENTAL RESULTS

4.1 Experimental Setup

For comparing the performance in IrNIRGAM while supporting irregular topologies with regular topologies, a set of application specific irregular topologies were generated. For performance comparison, the IrNIRGAM, extended version of NIRGAM supporting irregular topology with the provision of supporting distributed table based escape path as well as distributed table based deterministic routing for preventing deadlock condition was used. The IrNIRGAM is made to work on the application specific customized irregular topology. Each core in IrNIRGAM injects one flit every 2 clock cycles into the network i.e. flit interval is kept as 2 clock cycles since a core approximately take 2 clock cycles to process a flit for further transmission in IrNIRGAM. For performance comparison IrNIRGAM was run for 10000 clock cycles with applied packet injection interval to evaluate the network performance.

The communication energy for the generated NoC is estimated according to the energy model proposed in [19]. The average energy consumption by router in transmitting a bit is can be evaluated using the power simulator Orion [18] for the given technology. Moreover the dynamic bit energy consumption for inter-node links (El_{bit}) can be calculated using the following equation [21].

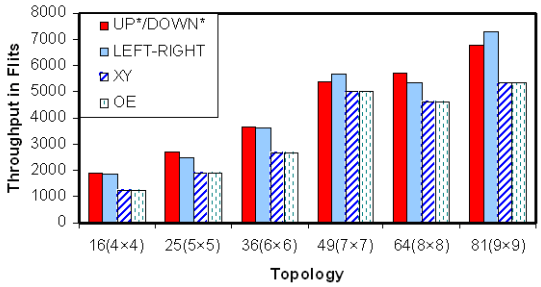
$$El_{bit} = (1/2) \times \alpha \times C_{phy} \times V_{DD}^2$$

Where α is the average probability of a 1 to 0 or 0 to 1 transition between two successive samples in the stream for a specific bit. The value of α can be taken as 0.5 assuming data stream to be purely random. C_{phy} is the physical capacitance of inter-node wire under consideration for the given technology and V_{DD} is the supply voltage.

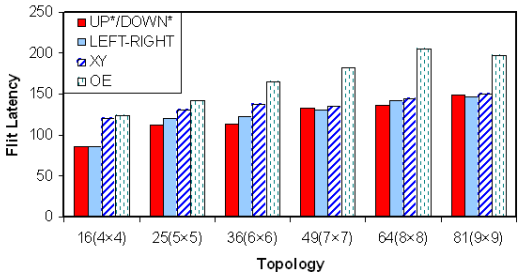
The performance of customized irregular topologies with up*/down* and Left-Right routing function with 2D-Mesh NoC with XY and OE routing were compared for the packet injection intervals according to the application requirement. The sizes of the tiles are kept same in the irregular topologies as in regular 2D-Mesh.

4.2 IrNIRGAM and Deterministic Topology Agnostic Routing

Figure 2 summarizes the comparative performance results averaged over 50 application specific irregular NoCs/topologies generated with the objective to have multiple deadlock free paths to avoid congestion and improve performance. Here for irregular topologies distributed table based deterministic routing is used with permitted node degree (nd_{max}) = 4 and permitted channel length (e_{max}) was taken as 2 times the length of the core/node for number of cores varying between 16 to 81 and 2D-mesh with similar task to core(tile) mapping and same number and size of cores with XY and OE (odd-even) routing.



(a)



(b)

Fig. 2: Performance comparasion of irregular topology (with distributed table based deterministic routing) and 2D Mesh on IrNIRGAM (a) throughput (in flits) and (b) flit latency (in clocks)

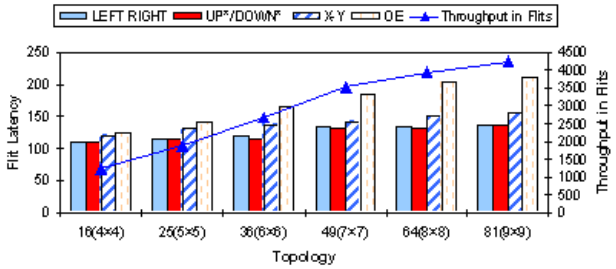
Figure 2 shows the comparative results regarding throughput and average flit latency. Application specific irregular topologies with distributed table based deterministic up*/down* routing function shows on average an increase in throughput of 32.3% and reduction in average flit latency of 11.3% and 28% in comparison to 2D-Mesh with XY and OE routing respectively. Similarly application specific irregular topologies with distributed table based deterministic Left-Right routing function shows on average an increase in throughput of 30.2% and reduction in average flit latency of 9% and 26% in comparison to 2D-Mesh with XY and OE routing respectively.

4.3 IrNIRGAM and Escape Path Based Topology Agnostic Routing

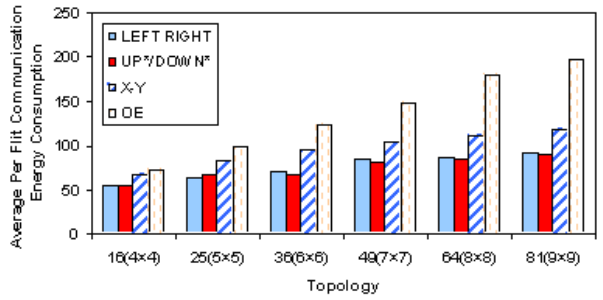
4.3.1 Experiment with random benchmarks

Figure 3 shows the performance comparison of irregular topologies with 2D-Mesh averaged over 50 various irregular

topologies with varying number of cores from 16 to 81, permitted node degree (nd_{max}) = 4 and permitted channel length (e_{max}) was taken as 2 times the length of the core/node. The irregular topologies with Left-Right routing shows reduced average flit latency in the range of 9.4 clocks to 18.4 clocks and 13.2 clocks to 69 clocks in comparison to 2D-Mesh with XY and OE routing respectively. Similarly the IrNIRGAM with up*/down* routing shows reduced average flit latency in the range of 10 clocks to 20.9 clocks and 13.8 clocks to 76 clocks in comparison to 2D-Mesh with XY and OE routing respectively. The average per flit communication energy comparison of irregular topology with 2D-Mesh shows reduction in the range of 18.5% to 25.8% and 24.6% to 53% in comparison to XY and OE routing respectively for Left-Right routing and similarly reduction in per flit communication energy in the range of 18.8% to 29.2% and 25.2% to 54.7% in comparison to XY and OE routing respectively for IrNIRGAM with up*/down* routing were observed.



(a)



(b)

Fig 3: Performance comparasion of irregular topology (with distributed table based escape path routing) and 2D-Mesh on IrNIRGAM (a) Average flit latency (in clock cycles) and (b) Average communication energy consumption per flit (in pico joules)

4.3.2 Experiment with realistic MMS benchmark

To evaluate the applicability of IrNIRGAM with realistic application, a generic Multi Media System (MMS) application was considered. MMS is an integrated video/audio system which includes an h263 video encoder, an h263 video decoder, an mp3 audio encoder and an mp3 audio decoder. The application was partioned into 40 distinct tasks and then these tasks were assigned and scheduled onto 25 selected IPs. These IPs range from DSPs, generic processors, embedded DRAMs to customized ASICs. The communication trace graphs as shown in Figure 4 for the same were obtained from the work presented by Hu et al. [19]. The performance analysis of customized Irregular topology supporting escape path based up*/down* routing in comparison to regular mesh topology with Xy and OE routing for the stream of information satisfying the communication trace graph of Figure 4 are summarized in Figure 5.

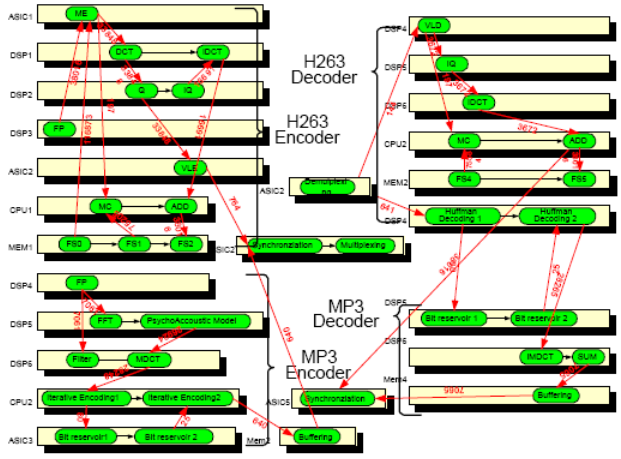
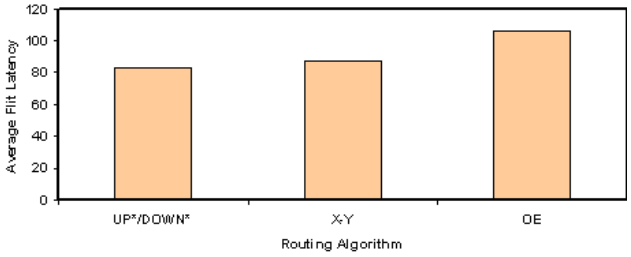
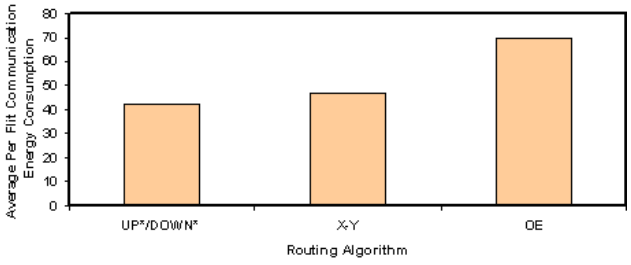


Fig 4: Communication Trace Graph for MMS

Figure 5 shows reduction in latency on average of 4.2 clocks and 22.5 clocks in support of customized irregular NoC for equivalent throughput in comparison to the 2D-Mesh with XY and OE routing respectively with smart task to core mapping as proposed in [19] for the MMS as shown in Figure 4. Similarly customized irregular NoC for equivalent throughput showed reduction in average per flit communication energy of 9% and 39% in comparison to 2D-Mesh with XY and OE routing respectively with smart task to core mapping.



(a)



(b)

Fig 5: Customized irregular NoC and 2D-Mesh performance comparison for MMS (a) Average flit latency (in clock cycles) and (b) Average communication energy consumption per flit (in pico joules)

5. CONCLUSION

In this paper an extensive simulation framework: IrNIRGAM supporting regular as well as irregular topology is presented. The topology is one of the most important aspects for any SoCs communication performance. Since most application specific SoC designs favor the use customized irregular topology tailor made according to application, The IrNIRGAM provides the generation of irregular topology or communication infrastructure according to the application requirement. In addition to the above IrNIRGAM provides a modular architecture which is easily extendible if desired. The important traffic models like CBR and Bursty are also supported by IrNIRGAM. Moreover the support of the popular routing algorithms in NoC domain like deterministic XY, OE and source routing and distributed table based deterministic and escape path based routing is also provided in the IrNIRGAM simulation framework.

Therefore IrNIRGAM facilitate researchers of the NoC domain to experiment with the various communication infrastructure/topologies, routing algorithms and communication traffic(application) to choose the best as per their need. Moreover various performance metrics are evaluated by IrNIRGAM such as latency, throughput and bit energy based energy model permit the designers to optimize their designs according to various NoC performance aspects. Further IrNIRGAM also provides run time integrated plots of the requisite performance metrics to enable the analysis of the design easier and accurate.

6. REFERENCES

[1] Dally . W. J., and Towles. B. 2001. Route packets, not wires: on-chip interconnection networks, In IEEE Proceedings of 38th Design Automation Conference (DAC), 684–689.

[2] Benini. L. and De Micheli. G. 2002. Networks on chips: A new SoC paradigm. Journal of IEEE Computer, vol. 35, no. 1, 70–78.

[3] International Technical Roadmap for Semiconductors, http://public.itrs.net/, 2004.

[4] Noxim the NoC Simulator, available at http://noxim.sourceforge.net/

[5] NNSE: The Nostrum NoC Simulation Environment, avaiable at http://www.ict.kth.se/nostrum/NNSE/

[6] Jain. L., Al-Hashimi. B. M., Gaur M. S., Laxmi. V. and Narayanan. A. 2007. In proceedings of NIRGAM: A simulator for NoC interconnect routing and application modeling. In proceedings of DATE.

[7] NIRGAM. http://www.nirgam.ecs.soton.ac.uk

[8] Duato. J., Yalamanchili. S. and Ni. L. 2003. Interconnection Networks : An Engineering Approach, Elsevier.

[9] Boden. N. J., Cohen. D., Felderman. R. E., Kulawik. A. E., Seitz. C. L., Seizovic. J. N., and Su. W. K.. 1995. Myrinet : a gigabit-per-second local area network. In IEEE MICRO, 29-36.

[10] Schroeder. M.D. et al. 1991. Autonet: A high-speed self-configuring local area network using point-to-point links. Journal of Selected Areas in Communications, vol. 9.

[11] Jouraku. A., Funahashi. A., Amano. H. and Koibuchi. M. 2001. L-turn routing: An adaptive routing in irregular networks. In proceedings of international conference on Parallel Processing, 374-383.

[12] Sun. Y. M., Yang. C. H., Chung Y. C and Hang. T. Y. 2004. An efficient deadlock-free tree-based routing algorithm for irregular wormhole-routed networks based on turn model. In proceedings of International Conference on Parallel Processing, vol. 1, 343-352.

[13] Wu. J., and Sheng. L. 1999. Deadlock-free routing in irregular networks using prefix routing. Technical Report. DIMACS.

[14] Cherkasova. V. L. and Rokicki. T. 1996. Fibre channel fabrics: evaluation and design. In proceedings of the 29th international conference on System Sciences, Vol. 1, pp. 53-62.

[15] Sancho. J. C. et al. 2000. A flexible routing scheme for networks of workstations. In proceedings of international conference on High Performance Computing.

[16] Silla. F. and Duato. J. 2000. High-performance routing in networks of workstations with irregular topology. In IEEE Transactions on Parallel and Distributed Systems, vol. 11, 699-719.

[17] Glass. C. and Ni L. 1992. The turn model for adaptive routing. In proceedings of 19th international symposium on Computer Architecture, 278– 287.

[18] Hu. J. and Marculescu. R. 2003. Energy-aware mapping for tile-based NOC architectures under performance constraints. In proceedings of ASP-DAC.

[19] Wang. H-S. et al. 2002. Orion: A power-performance simulator for interconnection network. In proceedings of international symposium on Microarchitecture.

[20] Hu. J. 2005. Design methodologies for application specific networks-on-chip. Doctoral Thesis. Carnegie Mellon University.