Benchmarking of On-Chip Interconnection Networks

Daniel Wiklund, Sumant Sathe, and Dake Liu*
Dept. of Electrical Engineering
Linköping University
S-581 83 Linköping, Sweden
{danwi,sumant,dake}@isy.liu.se

Abstract

More complex on-chip interconnection structures such as networks on chip emerge today. As the number of interconnect architectures rises there is a need to do an impartial evaluation of the performance of the interconnect structure. This is important for both the designer of the interconnect as well as for the system designer in order to achieve best performance vs. cost tradeoff. The work presented in this paper describes a method to specify, execute, and evaluate benchmarks for on-chip interconnects. The benchmarking method uses formal traffic specifications together with architecture independent constraints to form the benchmark specification. This specification is adapted to the simulation flow available for the interconnect and simulated to get the wanted results. The benchmark method is evaluated using two related examples where throughput is the main focus in the results. These examples show the applicability of the method.

Keywords: Benchmarking, network on chip, simulation.

1. Introduction

One of the key components in contemporary system design is the interconnect structure. The purpose of the interconnect is to provide the possibility to communicate between different functional blocks according to the system specification. Dependent on the system specification, this structure can be anything from a few wires to a full-fledged network, e.g. the Internet. There has been a lot of work in the design and evaluation of the general purpose style networks used between computers and between boards. In contrast, this paper will only address the interconnect structures that are used inside a single chip. Being limited to on-chip gives certain constraints on the interconnect design. E.g. silicon cost will limit the size of the interconnect components thus allowing only small buffers and relatively simple functionality. Where a component such as a router in a computer network can occupy an entire 19-inch rack mount box, the on-chip counterpart has to fit a space smaller than 1 mm^2 [1, 2].

The popularity of networks for use on-chip is increasing. Many different research projects have been started in the last few years [1, 3, 4, 5, 6]. As the number of proposed network architectures increase there is a growing need for benchmarking of these networks. This is necessary in order to assess the relative performance of the different interconnect architectures for the applications intended and to find the bottlenecks in the interconnect systems.

The only way to get benchmarking methods that is fair and usable for comparisons is to create rather formal methods for specification of the benchmarking premises. The methods can not be limited to a specific type of interconnect but must apply to any type of interconnect structure. This is necessary in order to assure that the methods in themselves do not limit the usability of the results.

Section 2 gives an introduction to networks on chip. Section 3 discusses simulation of interconnect performance. Section 4 introduces some definitions while section 5 describes the benchmarking method. Section 6 discusses a benchmarking example and the associated results. Finally, section 7 concludes the paper.

2. Networks on chip

The traditional method for communication on chip is to use either point-to-point links or time-division buses such as ARM AMBA. These structures have inherent problems with scalability and flexibility. The point-to-point interconnects suffer from severe inflexibility and the only way of create this flexibility is to add more links. A bus is rather the opposite. It is flexible in connecting several ports but at the price of significantly lower performance per port.

A way to lower the impact of these problems is to merge these two opposites into a network structure. This network-on-chip (NoC) will consist of a shared set of links and routers that will give higher scalability than the bus and

*This work is supported by the Swedish Foundation for Strategic Research (SSF) through the Stringent electronics research center at Linköping University.

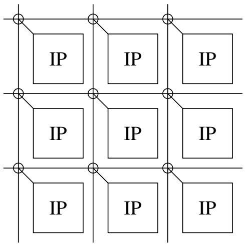

Figure 1. Typical mesh network

larger flexibility than the point-to-point links. However the freedom given in design of a network makes choosing the parameters involved a non-trivial task. There are many major factors to take into account when assessing the performance of the network. E.g. there are topology, routing algorithm, router arity, physical link design, etc.

All these options sum up to a vast design space that will yield significant difference in performance between network implementations and traffic patterns.

The topology is perhaps the most obvious design choice. The 2-d mesh has turned out to be the most popular topology because of its simplicity. A section of a 2-d mesh can be found in Fig. 1. The circles in the figure represent the network routers.

3. Simulation of interconnect performance

An useful analytical model of performance for a network-style interconnect is difficult (if not impossible) to achieve. This is due to the inherent complexity of the network design space and how the traffic patterns affect the network performance. The only practical method of assessing the interconnect performance is through simulations with real-world traffic [5, 7].

Simulations have the inherent drawback of the results not being better than the stimulus used. For a real-world application it is important to extract the appropriate traffic model. Without this model it is more or less impossible to tell whether the simulated interconnect will fulfill the performance specification. Even with an appropriate traffic model, comparison of simulation results for different implementations are problematic. All parameters in the simulations must be strictly controlled to make the comparison fair.

These drawbacks with simulation together give rise to the question of benchmarking as a method to compare and assess the relative and absolute performance of different architectures and implementations. We have identified this need and addressed it through the development of a benchmarking process for simulation of interconnect structures on-chip.

From the above discussion, the importance of benchmarking for evaluation is evident. The main contribution presented in this paper is a method to specify, perform, and evaluate benchmarks for interconnect structures.

Table 1. Comparison of DSP and NoC benchmarks

	DSP	NoC
Constraints	· Native precision · Include round/sat · Only core DSP (i.e. no accelerators)	· Transaction level · Only complete transactions (no loss) · Implementable NoC
Spec	BDTI examples: · Real block FIR · Complex block FIR · Vector dot product	· Traffic pattern · Packet size(s) · Number of ports
Target info	· Word length · Clock frequency · Purchase cost	· Word length · QoS in hardware · Hardware cost
Results	· Cycle cost · Program memory use · Data memory use	· Throughput · Latency · Buffer usage

4. Terminology and definitions

This section contains definitions of some concepts central to benchmarking. These definitions are reasonably general but it should be noted that this work only addresses the benchmarking of interconnect traffic. The problem of benchmarking computational resources has already been addressed by several instances [8].

Defnition 1 *A* benchmark *is the combination of the specification(s) that have been used and the result(s) that have been achieved in the process of benchmarking.*

Defnition 2 *A* bottleneck *is a performance limiting factor in a system.*

Defnition 3 *A* benchmarking method *is a specified method used to create (i.e. specify) and run benchmarks in order to find bottlenecks.*

Defnition 4 *A* benchmarking process *is the process of using a benchmarking method and benchmark specification in order to get benchmark results and finding bottlenecks.*

5. Benchmarking of interconnects

There are a number of differences between benchmarking of computational resources, e.g. DSP processors, and interconnection networks. The most important differences are shown in Table 1. The two upper sections relate to the benchmark specification and the two lower sections relate to the results. The idea behind each category is similar, e.g. constraints make both the DSPs and NoCs comparable in the sense of disallowing "cheating".

The constraints set the limitations of the interconnect to be benchmarked such that it has to be implementable and reliable. The specification in turn tells what kind of traffic should be simulated over the interconnect.

The results are a combination of the target information and simulation results. Target information are architectural choices such as word length (i.e. link width), if quality of service (QoS) is supported in hardware, etc. Some results given from the simulations are measures on throughput, latency, buffer usage, etc.

5.1. Benchmarking method

The principle of the benchmarking method is to create a traffic model for the interconnect simulation that reflects the benchmark specification. This traffic model is then used to excite the network model in the appropriate simulator. The simulations will give (approximations of) the performance for the given network(s) under the traffic specified.

The benchmarking flow is as follows:

1. Translate the benchmark specification into a traffic model for the simulator. It is important to note that this translation might become suboptimal. Dependent on the impact of the design flow the benchmark may also take the tool chain and methodology into account.
2. Execute the simulations with the traffic model and one or more network models. This step involves executing the traffic model as stimuli over the hardware model given in the network description.
3. Collect the results from the simulations.

5.2. Benchmark specification

The specification of the benchmark is basically the traffic pattern specification. Such a specification can be presented in many different ways. Examples are Kahn graphs with communicating processes and communicating synchronous data flow graphs.Another description style is multiple state machines and flow charts. The important point is that the model used must be able to describe the system accurately.

5.3. Interpretation of results

The most important part of the benchmarking process is to interpret the simulation results correctly. Even direct figures like a throughput measure can be misleading if taken out of context. The only appropriate way of reading a simulation result is to couple it with the traffic situation and interconnect architecture used for the specific simulation. Because of this and the general complexity of interconnect systems, each comparison must then be based on an individual interpretation to find the specific differences in each case.

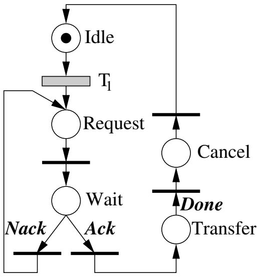

Figure 2. Petri net for source (Waiting time T_1 is random in this case)

6. Benchmarking example

This section presents two benchmarking examples where the impact of burst size (i.e. packet size) and the total data rate of the transfers are varied. The main target result of the benchmarks is throughput.

6.1. Specification

The traffic is considered in a network with 64 combined sources/sinks. The outgoing data rate for each source is varied over a range from 5% to 80% of the cycles (assuming synchronous model). Also the packet size is varied from 20 to 3000 words per packet for each of the data rates. These communicate with each other in a random fashion with uniformly distributed starting times. The two benchmarks differ in how the source/sink pair selection have been specified. The first case is totally random selection of source and sink over the entire set. The second case assumes locality where the sources/sinks are distributed evenly in a square (over a 2-d surface). The sources are then selected randomly while the sinks are selected randomly within a radius of two sinks away from the source.

These traffic models are not intended to model a specific application but is rather selected to show the applicability of the methods described in this paper.

6.2. Network

The network used for the benchmark run is a circuit switched network that has been published earlier [2, 5]. One highlight of the network specification is that the topology can be arbitrary. In this benchmarking case we have selected a 2-d mesh with sources/sinks connected to each router node.

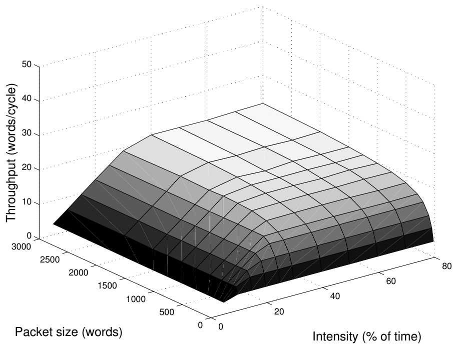

Figure 3. Throughput for random traffic

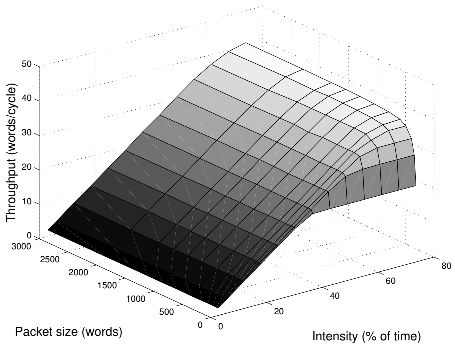

Figure 4. Throughput for traffic with locality

Another highlight is the route setup scheme where a packet traverses the same set of wires as will be used for the circuits while locking the circuit. The basic functionality of a network source is shown as a simplified Petri net in Fig. 2. When a transfer is initiated a routing request is sent through the network towards the destination. When this request has reached the destination an acknowledgment is returned showing that it is clear to send the payload. If the request for some reason cannot reach the destination, a negative acknowledgment is returned and the process has to start over.

6.3. Results

The theoretical maximum throughput at the inputs is 64 words per cycle. The simulation results for the first sink selection case can be found in Fig. 3. The graph shows a low saturation level about five words per cycle for the smallest packet size (20 words). The saturation level increases to about 21 words per cycle for larger packet sizes. The reason for the relatively low saturation limit with small packets is that the overhead will dominate the transfer.

For the second case (and a sane network allocation) the throughput will reach the levels shown in Fig. 4. Small packet still give a low saturation limit of about 22 words per cycle whereas the large packets will raise the throughput to roughly 40 words per cycle. These two benchmark runs clearly shows the impact of locality for this specific network.

7. Conclusions

This paper shows the problem of evaluating different on-chip interconnect structures in an impartial way. A benchmarking method has been developed to alleviate this problem. The benchmarking method has been specified and tested on two examples.

The terminology related to benchmarking has been defined. We also describe the benchmarking method and how to generate benchmarks that are fair, e.g. in the sense of not putting any constraints that can only be fulfilled by some architectures/implementations of the interconnect. Further the issue of benchmark specification has been discussed. The specification is used as an input to the benchmarking process and must be strictly controlled. The results of the benchmark are different communication performance measures, e.g. throughput and latency.

The benchmarking method has been tested on two examples where throughput has been the primary result variable. The interconnect considered is a circuit switched network and the benchmark clearly shows the importance of locality in such a network.

References

[1] K Goossens, J van Meerbergen, A Peeters, and P Wielage, "Networks on silicon: Combining best-effort and guaranteed services," in *Proceedings of the design automation and test conference*, Mar. 2002.

[2] S Sathe, D Wiklund, and D Liu, "Design of a switching node (router) for on-chip networks," in *Proc of the ASICON 2003 conference*, Oct. 2003.

[3] P Guerrier and A Greiner, "A generic architecture for on-chip packet-switched interconnections," in *Proc of the design and test in Europe (DATE) conference*, 2000.

[4] I Saastamoinen, D Sigüenza-Tortosa, and J Nurmi, "Interconnect IP node for future system-on-chip designs," in *IEEE int'l workshop on Electronic design, Test, and Applications*, 2002.

[5] D Wiklund and D Liu, "Socbus: Switched network on chip for hard real time systems," in *Proc of the Int'l Parallel and Distributed Processing Symposium (IPDPS)*, Apr. 2003.

[6] W J Dally and B Towles, "Route packets, not wires: On-chip interconnection networks," in *Proc of the design automation conference (DAC)*, 2001.

[7] S G Pestana, E Rijpkema, A Radulescu, K Goossens, and O P Gangwal, "Cost-performance trade-offs in networks on chip: A simulation-based approach," in *Proceedings of Design, Automation and Test in Europe Conference*, Feb. 2004.

[8] Berkeley Design Technology, Inc. (BDTI), "Evaluating DSP processor performance," *http://www.bdti.com*.