On-Chip Interconnection Network Topologies

An Emerging Critical Design

Derek Taylor **TJ Giuli** **Paul Lassa** **Tom Fountain**

Stanford University
Stanford, CA 94305
{dat,giuli,plassa,fount}@Leland.Stanford.EDU

Abstract

This paper analyzes current research in an emerging field of study: on-chip interconnection networks. On-chip interconnection networks have become an increasingly important focus as researchers determine how best to make use of the extraordinarily large number of transistors (>1 Billion) available by the year 2010. This paper focuses on recent research that has been done recently on new microprocessor designs aimed at this technology, as well as research performed in FPGA technologies that are merging with this study. In particular, this paper examines results presented by the MIT Raw Machine and the Berkeley Pleiades research projects. Finally, we examine the application of traditional network methodologies to on-chip networks.

1 Introduction

The semiconductor industry roadmap projects that VLSI technology advances will allow for 1.4 billion transistors on a single chip by 2011 [NTR98]. On-chip interconnection networks will uniquely determine the effectiveness of future system designs since communication will play a crucial role in determining the system's performance. Effectively dealing with these communication issues requires that researchers accurately project the constraints that will form the basis of design decisions in such an arena. Researchers have only recently begun to explore the vast design space and define the constraints posed by this technology.

Currently the most widely considered constraint on the problem of a billion-transistor design is the constraint of wire delay. Researchers estimate that by the year 2007, it will take a signal as much as 40 cycles to cross a chip [BLAA99]. This constraint on communication demands that researchers deal effectively with multiprocessing solutions that exploit spatial locality to reduce the communication delay in computation. This paper analyzes the constraints that researchers project will dominate the design decision for on-chip interconnection topologies.

In Section 2 of this paper, we propose the issues that researchers believe to be the driving considerations in selecting an on-chip network topology. Section 3 reviews some of the approaches researchers have taken to solve these problems. In particular, in Section 3.1, we look at solutions to the load balance problem. Section 3.2 reviews some of the approaches to the homogeneity problem. Section 3.3 examines solutions to the locality and layout problems posed by on-chip interconnection networks. Section 3.4 provides some of the solutions to the power consumption problem. In Section 4, we speculate what the future directions will be for research in this area. Finally, we draw our conclusions.

2 Topology Considerations

The first problem facing the designers of any interconnection network is the network topology. On-chip networks, unlike conventional off-chip networks, avoid the packaging constraints of a chip's pin-out limitation and backplane and board bisection bandwidth. Instead, on-chip networks are constrained by on-chip bisection bandwidth, or the number of available communication channels on the chip. Designers must design for the optimum network (in terms of latency, throughput, and path diversity) by trading channel bandwidth (area devoted to communication channels) for processing grain size (area devoted to the processing tiles). Thus, a measure of the quality of the topology is the load balance between the network and the processing unit [AMY98]. Proper apportionment of silicon area balances the load between the processing units and the network for the widest class of applications. For example, if the network is idle most of the time while the processing units are busy, more area should be devoted to the processing units to increase performance since the area used in the network is wasted for this application. Conversely, if the network is heavily loaded and the processors are regularly stalled waiting for network traffic to complete, then chip area could be used more efficiently by giving more of the processing area to the network to increase the channel bandwidth.

Associated with the problem of grain size is the issue of homogeneity of the processing tiles. Researchers are divided over the question of whether processing tiles must be homogeneous. This decision directly affects the choice of network topology as a heterogeneous network forces the network topology to be irregular [ZWGR99]. However,

nearly all researchers agree that regular topologies are easier to implement and route. Thus, a measure of quality for the topology is its regularity.

Researchers must also address the problem of selecting a topology suitable for the application pool intended for use on the system. Most researchers assume that a network that exploits spatial locality, such as a 2-D mesh or a fat tree yield the greatest opportunity for performance gains for the widest array of applications [BLAA99]. Others have analyzed the option of global interconnects through crossbar switches or multi-stage interconnection networks (MIN; eg. k-ary n-flies) [ZWGR99]. These researchers agree that networks that exploit locality are the clear winners for the majority of applications suited to these systems. Thus, another measure of quality is its ability to exploit spatial locality. While this measure is not unique to on-chip networks, it is important to note that researchers continue to emphasize this quality.

A fourth measure of quality for a network topology is its ability to map to a 2-D layout. The network topology must map to a 2-D plane if it is to be realizable using current VLSI technologies. As such, researchers generally assume 2-D meshes or 2-D fat trees [BLAA99]. However, hierarchical patterns have been proposed that overcome the scaling limitation of increasing hop counts for topologies that assume the "mesh" wiring pattern [ZWGR99, Dally91].

As the number of signals increase on a chip, designers must increasingly pay attention to the power consumption of the network. Higher clock speeds and more transistors on the chip further exacerbate this problem. In this regard, designers must tradeoff power for performance. For example, a full crossbar interconnection, while extremely inefficient in terms of silicon area use and also power consumption, offers fast performance and great flexibility [ZWGR99]. In any case, a network that consumes less power is clearly superior to another network with equivalent performance.

Designers of on-chip interconnection networks must attend to all the preceding measures of a topology's quality without neglecting the base measures of a network's quality: latency, throughput, and path diversity. Designers can easily derive *zero-load* formulas (assuming no network contention) for these properties directly from the topology selected.

In traditional network design, the cost of the network is often the overriding factor that drives design decisions. For example, although crossbar networks have traditionally been known to provide the best performance and flexibility, networks are never actually built as crossbars because of the extraordinarily high cost. For on-chip networks, cost plays a different role in the selection of a topology. The cost of an on-chip network is a function of the silicon area that the network consumes rather than the cost of its packaging (chips, boards, cables, etc.). Area translates directly into cost via the yield of the fabrication process for the chip.

This cost property of the network poses an interesting opportunity for system designers. The authors of this paper believe that for on-chip interconnection networks, fault tolerance can directly translate into *lower* cost through higher manufacturing yield. DRAM manufacturers have for a long time used redundancy to improve yield. If an on-chip system uses homogenous tiles and after production, the tester discovers that one or more of the processing elements is defective, the chip can still operate at very reasonable performance levels as long as the network itself is tolerant enough to operate the system without the defective units. Thus, a final measure of quality for an on-chip interconnection network is its fault-tolerance. While much of this fault-tolerance will be achieved through the routing strategy employed, the topology must support a network with path diversity and configurable nodes so that faulty processing elements can be "shut off" without affecting the rest of the system or consuming any power.

Table 1 summarizes the quality measures unique to on-chip interconnection networks that we have found in the literature. Please note that the final measure of quality, *fault tolerance* is not a topic that we found discussed much in the literature. Rather, this issue is one the unique contributions that this paper makes to the topic. We discuss this issue further in Section 4, Future Directions.

Network Qualities	Pros	Cons
Balanced Load	Higher efficiency	Difficult to implement
Homogeneous Elements	Regular network topologies allow for simpler routing	Elements can have limited functionality or functionality that is not always used for special functions
Good Locality	Increase performance; lower cost; lower power consumption	Not all applications benefit; non-uniform access times
2-D Layout easy	Simpler implementation	Can only use a limited number of topologies
Low power consumption	Energy efficient	More complicated to implement; requires careful circuit design
Fault Tolerance	Can lead to lower cost. More reliable performance	More complicated design

Table 1: Quality Measures of on-chip Networks

3 Proposed Solutions

While several researchers have been addressing the key problems posed by the on-chip network challenge, none have comprehensively addressed all of the issues. The following sections examine each of the major network qualities and the solutions researchers propose. Some of the studies examined in the following sections are not focused on billion-transistor technology, however the solutions and methodologies presented are applicable and provide a unique contribution to the topic. Table 2 in the Appendix summarizes the arguments presented by each of the researchers.

3.1 The Load Balance Problem

The Raw Project at MIT presents the balance problem as an offshoot inherent in any FPGA-type (reconfigurable) system [AMY98]. Moritz, et al. propose a model that measures the performance of an application given a description of the application's processing, memory, and communication requirements. They introduce a set of "balance constraints" to assist the process of optimizing the model parameters to find the best number of processors *P*, each with the correct processing power *p*, sufficient communication bandwidth *c*, and memory *m*. The "balance constraints" follow directly from our intuitive understanding that performance will improve if we transfer part of the area of any underutilized resource to the area of an over-utilized resource to increase the performance of the bottleneck resource. The authors define these variables and a set of equations providing their model system. They then show how to optimize these parameters to estimate the optimal configuration for the system [AMY98].

The weakness of the Raw Group's study is that they do not vary the network topology or consider any alternatives to their static, compiler-scheduled network. Later work on the Raw Machine [AMY98, BLAA99] indicate that the Raw Machine actually requires two networks, one static and one dynamic, since the static network insufficiently handles applications with dynamic data communication patterns. The authors of this paper believe that this secondary network will disrupt the balance that the original study was aiming to achieve, since suddenly there is a new resource, unaccounted for in the original study, that is apparently underutilized (the Raw Group believes that their dynamic network will be used rarely [BLAA99]).

Andre DeHon, at Berkeley, argues from an analysis of FPGA technology that it is not always the most area efficient strategy to achieve high utilization of the logical units [DeHon99]. He argues that while additional interconnect allows the designer to use the logical units more heavily, it often causes less efficient use of the interconnect's resources. This is because an interconnect that always allows high utilization of the logical units (such as a crossbar) consume more area. Furthermore, in many FPGA devices the interconnect resources consume as much as 80-90% of the effective area on the device. Thus, he argues that in many cases less interconnect rather than more, along with an efficient mapping of the application's communication to the available channels will extract the most area-efficient use of the silicon [DeHon99]. He also proposes heuristics for finding the most efficient mapping of the application's dependency schedule to the available network. The limitation of the paper is that the analysis applies only to a fat tree topology. The author acknowledges this limitation and agrees that a study of these same effects should be done for mesh-based topologies [DeHon99].

3.2 The Homogeneity Problem

Zhang, et al. investigates the problem of heterogeneous components connected in an interconnection network for future system-on-a-chip platforms [ZWGR99]. Their study was conducted in the context of a system with reconfigurable nodes. As not all the elements of the system to be integrated into the chip are the same, the researchers propose a heterogeneous layout pattern based on the 2-D mesh. They propose a "generalized mesh" structure with communication channels routed along the side of every module and routers at each junction. As this network architecture is irregular, the network channels must be individually optimized for connectivity requirements of each individual module [ZWGR99].

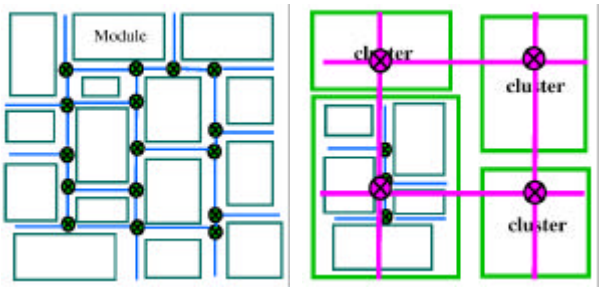

Figure 2: Generalized and Hierarchical Mesh

The greatest weakness of heterogeneous designs is their lack of generalization to an organized design methodology. However, Zhang, et al. address this issue by proposing a "hierarchical segmented interconnect network" that both addresses the problems posed by the non-regularity of the network and at the same time reduces the mesh topology's inherently high latency for distant communication. Figure 1 illustrates both the generalized mesh and the hierarchical mesh. The hierarchical segmented network, similar to earlier proposals [Dally91] simply creates longer channels that span several nodes, thus allowing a shorter number of hops between distant nodes. Zhang, et al. propose that a heterogeneous cluster of system components could be connected by a generalized mesh at a higher level of wiring to provide efficient inter-cluster communication [ZWGR99].

The Raw Machine is an example of a typical study that ignores the requirements of non-homogeneity. Essentially,

all diverse elements are abstracted away within the homogenous processing tile of the Raw Microprocessor so that the more regular network topology issues can be determined [AMY98].

While researchers do not agree on the feasibility of homogeneity, researchers generally agree on the need for regularity where possible. Where this is impossible, abstraction is generally used to hide the irregularity. As the transistor counts increase for an on-chip system, the assumption of abstracting away any regularity becomes more natural as processing elements can include more functionality. However, this raises a separate issue (not addressed in this paper) of the efficiency of silicon area use within the processing elements themselves.

3.3 The Locality and Layout Problem

Zhang, et al. further analyze the possibility of a "global interconnect network" wherein the network provides routes with identical costs for all connections between any pair of nodes [ZWGR99]. For example, a crossbar network, a MIN, or a multiple-bus network can provide this type of global connection within the system. Zhang, et al. rejects the global interconnection because it fails to take advantage of spatial locality and regularity in the connections. These global networks therefore waste wire resources due to the large area overhead. They also consume the most energy due to long buses and a large number of switches. Conversely, "local interconnect networks" such as a mesh, torus, and fat tree allow for reduced energy consumption and greater efficiency of wire usage while getting similar performance measures due to the locality property of many algorithms. These networks also have the property of mapping easily into a 2-D layout.

Some researchers do not publish the fact that they considered any alternative to the mesh topology. The Raw Machine at MIT [BLAA99] is a good example. The project specifies that the system is comprised of a simple static mesh network, along with a dynamic mesh to be used only for dynamic memory accesses. In none of the published papers that we found is the network topology analyzed in relation to alternatives. Furthermore, the very comprehensive study that was conducted to determine the primary constraints on the allocation of the resources within the system was based on a preliminary assumption of a simple mesh network [AMY98].

While the mesh topology is an ideal, simplistic starting point for the design of an on-chip interconnection topology, it should by no means be the stopping point. Early research [Dally91] demonstrates a method that can be used to provide for better scaling as the network size grows. This method was called "Express Cubes." Essentially, the idea proposes "fast channels" that skip ahead over several nodes and shorten the hop count for distant messages. The ring topology (or torus in higher dimensions) can be viewed as a special case of this topology with a single fast channel connecting the endpoints in a dimension. Although the earlier proposal by Dally does not assume on-chip interconnects, the principles applies naturally to the VLSI problem of on-chip connects. Using advanced VLSI techniques, a designer can create wider wires in high levels of metal wiring to provide "fast channels" for distant communication. Since most applications exhibit spatial locality in their communication patterns, the fewer fast channels should be able to handle the load on them for most application classes even though the expanded width of the wires limits the number of available express channels.

3.4 The Power Consumption Problem

Zhang, et al. provide a unique analysis of the problem of interconnects in terms of energy consumption. They point out that the mesh topology has an advantage in that it naturally segments the network. The mesh can accomplish significant power savings using selective activation : only the segments that are really necessary to provide a connection are activated [ZWGR99]. They then extend the power consumption analysis to relate the increased efficiency of a hierarchical network (as originally explained by Dally [Dally91]). They provide a model for the energy consumption per signal within the network. The model is essentially the sum of the energy consumed by the switches and in the wires. The energy consumed in the wires due to the distance traveled is slightly increased due to the larger driver and increased capacitance and resistance of the wire. However, this slight increase is more than offset by the fewer number of hops. Therefore, the energy consumed in hops drops by the number of nodes skipped in the hierarchy. Thus, Zhang et al. conclude that hierarchical networks are more efficient not only in terms of speed, but also power consumption.

4 Future Directions

We believe that there are four main opportunities for key research in the area of on-chip interconnection networks. First, the load balance problem has not sufficiently been analyzed. Second, the important research of "Express Channels" [Dally91] should be reanalyzed in terms of on-chip interconnection networks. Third, the power consumption issue of on-chip networks must be analyzed in greater depth. Finally, we feel that for on-chip networks, fault tolerance in the chip can be uniquely designed into the system and network such that the system is not only more reliable, but also cheaper.

The load balance analysis of the Raw Project [AMY98] does not analyze the problem in terms of a variety of topologies. The Raw project assumes only a 2-D mesh topology from the start of the analysis. The authors of this paper feel that with a topology that uses "Express Channels," much of the non-local traffic load is redistributed, decreasing the load on the local channels. These channels can then potentially be made smaller ,

allowing for more area devoted to processing tiles. Thus, a more generalized model can more accurately reflect the tradeoffs the come from the initial topology selection.

The earlier work on "Express Cubes" did not assume interconnection networks built completely on chip. While this work is applicable because of VLSI technology, research on this topic should verify the performance models of the topology in terms of wire delay constraints and VLSI packaging limitations such as the number of metal layers. Additional analysis should also be performed as to what class of applications can benefit from "Express Channels." Specifically, research should determine for what class of applications there is sufficient non-local communication so that "Express Channels" are warranted or unwarranted.

Power consumption issues for billion-transistor systems have not been sufficiently explored. In particular, we feel that architectures may become power limited rather than area limited in terms of the number of wiring channels available for communication. Furthermore, topology implementations that reduce power consumption through selective activation need to be explored.

This paper suggests that fault-tolerant on-chip interconnects can actually reduce the cost of the system. If a chip has many redundant compute nodes connected with a fault-tolerant network then errors in the fabrication process do not necessarily mean that the affected chips cannot be sold. This analysis is based on the same type of redundancy that DRAM manufacturers use to increase the yield of their processes. More study should be done that examines fault tolerant network strategies and compares the implementation cost vs. the cost gain in yield. The analysis should also measure the performance costs for a system operating under a fault. The fault model for the on-chip interconnect would account for node failures and wire breakage. Additionally, the design would require a testing process that would be able to determine exactly which nodes and links were broken. The study should determine whether adding more area to the chip for a fault-tolerant interconnect – and thus increasing both the probability of an error occurring on the chip and the cost of the chip – pays a higher yield and more revenue.

A fault-tolerant model would also affect the compiler. The compiler must assume a dynamic model of the processing nodes so that a new compilation is not required for every possible system error. Thus, a Raw-type system [AMY98] with only a static interconnection network would not provide this type of fault-tolerance.

5 Conclusions

We feel that researchers have missed a critical point in failing to address the issue of fault tolerance in their on-chip network designs. The authors of this paper believe that this issue can be one of the crucial issues driving future topology considerations because of the ability of a good model to reduce cost by increasing the yield of the fabrication process. A good starting point for research in this area is to review research in the fault tolerant area for off-chip networks and determine what principles can be cross-applied. Then we can do further research to determine the best models for future on-chip networks that will increase yield and thus lower cost.

In the future, on-chip interconnection networks will uniquely determine the effectiveness of future system designs since communication will play a crucial role in determining the system's performance. We feel that topologies should allow for effective balancing of the load across a wide class of applications. We also feel that topologies that take advantage of VLSI technologies to provide flexible mapping to 2-D layout with hierarchical patterns will be more effective than those that do not . Additionally, we feel that topologies can be designed which minimize power consumption. Topologies that adapt to these crucial issues may well play the dominant role in the technology of the next decade.

6 References

[AMY98] Agarwal, A., Moritz, C. A., & Yeung D., "Exploring Optimal Cost-Performance Designs for Raw Microprocessors" Proceedings of the International IEEE Symposium on Field-Programmable Custom Computing Machines, FCCM98, April 1998.

[BLAA99] Barua, R., Lee, W., Amarasinghe S., Agarwal a., "Maps: A Compiler-Managed Memory System for Raw Machines" To appear in the Proceedings of the IEEE Workshop on FPGAs for Custom Computing Machines '99 (FCCM '99), Napa Valley, CA, April 1999.

[Dally91] Dally, W. J., "Express Cubes: Improving the Performance of k-ary n-cube Interconnection Networks", IEEE Transactions on Computers, Sept. 1991.

[DeHon99] DeHon, A. ``Balancing Interconnect and Computation in a Reconfigurable Computing Array (or, why you don't really want 100% LUT utilization)." Proceedings of the 1999 ACM/SIGDA Seventh International Symposium on Field Programmable Gate Arrays (FPGA '99, February 21-23, 1999).

[NTR98] *The National Technology Roadmap for Semiconductors: Technology Needs,1998 Update.* Semiconductor Industry Association: Austin, Texas, 1998.

[ZWGR99] Zhang, H., Wan, M., George, V., Rabaey, J., "Interconnect Architecture Exploration for Low-Energy Reconfigurable Single-Chip DSPs", Proceedings of the WVLSI , Orlando, FL, USA, April 1999.

7 Appendix

Papers:	*Exploring Optimal Cost-Perf. Designs for Raw Microprocessors*	*Maps: A Compiler-Managed Memory System For Raw Machines*	*Express Cubes: Improving the Performance of k-ary n-cube I/C Networks*	*Balancing Interconnect and Computation in a Re-configurable Array*	*Interconnect Arch. Exploration for Low-Energy Reconfig. Single-Chip DSPs*
Description of Problem:	Balance/Grain size-Computing/Mem/ Communications	Latency limitations of a large centralized memory system.	Mesh topology great for locality, creates large network diameter (latency)	Balance of computation vs. wires	Interconnection flexibility vs. power consumption; global vs. local I/C
Constraints:	Total Real estate cost, inter-communication	Clock cycle constraint, wire delays do not scale w/ technology	Given mesh & wire bisection limits, how to reduce latency, increase throughput	Fixed wire-schedule, minimize height of sub-trees/wire lengths/distance	Single module (Maia) mapped to I/C, minimize power/fixed delay
Measure quality of Solution:	Fixed Cost, Optimal Performance among different applications	Performance with static compiler scheduling of communications	How it gets you better performance at less cost. (locality + low latency)	Area minimization; area overhead of interconnect	Minimize power consumption/wire delay
Why is problem Hard:	Comm. too slow, computing stalled. Computing too slow, comm. wasted	Different apps w/ different needs. Complexity, overhead of dynamic.	Topologies have opposing qualities of locality vs. latency; node delays dominate over wires	Very difficult for 2D or 3D topologies.	Regularity vs. flexibility and adaptability. Mesh vs. Global Interconnection.
Key Issues:	75% of chip area for processing and local comm.	Static vs. dynamic	Mesh exploits locality well, but adds latency; MIN is opposite	Optimization between computation and Interconnect.	Locality vs. Delay
Description of Solution:					
What Contributions:	Compute time equal to Comm time. Proposed tile size, ratios of Comp/Mem/Comm	Loop unrolling and control localization. Acknowledge need for Dynamic	Hybrid of advantages of both. Locality & latency; Fits existing networks.	Area utilization is not directly correlated with compact/ high computation/LUT usage.	Interconnect flexibility vs. power consumption.
Strong points:	Analytical framework to evaluate trade-offs; Reduce search space	Shows speedup for varied number of tiles across several apps.	3 different ways of implementation : basic/Multiple/ Heirarchical.	Useful I/C model scales relative area vs. locality; CAD meth. for Mapping to Struct.	Compare using energy as a metric; good topology spread; Energy & delay model.
Weak Points:	Static I/C; Only 1 topology, 2D Mesh. Only varied a few parameters	Only a single proc (MIPS R2000); Multiple tile compared with single tile Limitations of static.	No prediction for 2010 (submitted in 1989) ; added modules and components; some O/H for short hops that do not need express.	Only tree of meshes; only utilization considered; fixed wire schedule; too many other complexities.	Narrow scope; only mapped single module (Maia chip); performance based on energy/wire optimization.
What Remains TBD:	Other topologies; more real apps choices; verify accuracy of chosen analytical models.	Other topologies; Altering the processing elements ; variety of real-world apps.	Applicability to on-chip topologies and structures; no pin-limit; analysis with varied traffic patterns.	Apply to other tops; performance characterization; optimize/variable wire schedule.	Apps performance, Other architectures
Critical issues of Topic Addressed:	- Locality - Grain-size	- S/W static Compiler - Locality - Comp Vs Comm	- Limitations of the Mesh topology - Locality vs. latency	- Area - Mapping of design	- Power consumption - Wire delay model
Key issues not addressed:	- Power - Other topologies	- Limitations of static compiler	- Compare apps on MIN to same on Express Cube - Quantitative ex.	- Performance - Other topologies	- Other module types - Area / cost - Performance

Table 3: Analysis Issues for Research Papers