

# Virtual Point-to-Point Links in Packet-Switched NoCs

Mehdi Modarressi, Hamid Sarbazi-Azad, Arash Tavakkol

*Sharif University of Technology and IPM School of Computer Science, Tehran, Iran*

*modarressi@ce.sharif.edu, {azad, arasht}@ipm.ir*

## Abstract

*A method to setup virtual point-to-point links between the cores of a packet-switched network-on-chip is presented in this paper which aims at reducing the NoC power consumption and delay. The router architecture proposed in this paper provides packet-switching, as well as a number of virtual point-to-point, or VIP (VirtUal Point-to-point) for short, connections. This is achieved by designating one virtual channel at each physical channel of a router to bypass the router pipeline. The mapping and routing algorithm exploits these virtual channels and tries to virtually connect the source and destination nodes of high-volume communication flows during task-graph mapping and route selection phase of the NoC design process. The evaluation results show a significant reduction in power and latency over a traditional packet-switched NoC.*

## 1. Introduction

The advances in the semiconductor technology enable us to integrate a large number of processing elements into a single chip. The lack of scalability in bus-based systems and large area overhead and design effort of point-to-point dedicated links on one hand, and the scalability and performance of switch-based networks and packet-based communication in traditional parallel machines on the other hand, have motivated the researchers to propose the Network-on-Chip (NoC) architecture to overcome complex on-chip communication problems [1]. However, although NoC addresses some problems, e.g. scalability, the need for complex and multistage pipelined routers results in a high router-to-link energy/delay ratio and increases the delay and energy of communication.

In point-to-point connections, the packets travel on dedicated pipelined wires which directly connect their source and destination. As a result, they can yield the ideal performance and power results [2]. A network-on-chip increases these ideal values by the delay and power related to the router pipeline stages (buffer read/write, route selection, arbitration and crossbar

traversal). An analytical and experimental comparison of the power and performance metrics of dedicated point-to-point links and NoCs can be found in [2] and [3], respectively.

A number of methods have been developed to close this existing power and performance gap [2][4][5]. Express Virtual Channels, proposed in [2], is a method which allows packets to virtually bypass intermediate routers along their path. The bypass paths are made by a set of network virtual channels (VC). A packet starts traveling on an express VC either on some specific nodes (in the static express VC flow control scheme) or on every node (in dynamic express VC flow control) and bypasses some router pipeline stages as long as they are traveling on express VCs. In both schemes, express VCs are restricted to connect nodes only along a single dimension and cannot be allowed to turn.

In this work, we develop a packet-switched router architecture which can provide low-power and low-latency dedicated virtual point-to-point (VIP) links for the source and destination of high-volume communication flows by bypassing the pipeline of the intermediate routers. In this design, one VC of each physical channel is designated to bypass the router pipeline. In this virtual channel, the buffer is replaced with a latch which holds the flits arriving on the VC and is connected to the router outputs via a reconfigurable connection. By configuring these reconfigurable connections a priori, the flit stored in a latch is directed to a proper output and then to the latch in the next router until it is delivered at the destination node. The architecture allows the designer to optimize the network for the target application by trying to find as many VIP links as possible for communication flows of the application (at the mapping and route selection phase of the NoC design process) and set up them in the network upon starting the corresponding application on the NoC. As a result, the proposed NoC design holds the packet-switched NoC advantages while can provide dedicated point-to-point links for some selected communication flows.

This method differs from those methods supporting both traditional circuit-switching and packet-switching simultaneously [6] since in our method, paths are formed by single latches (as extra virtual channels) and

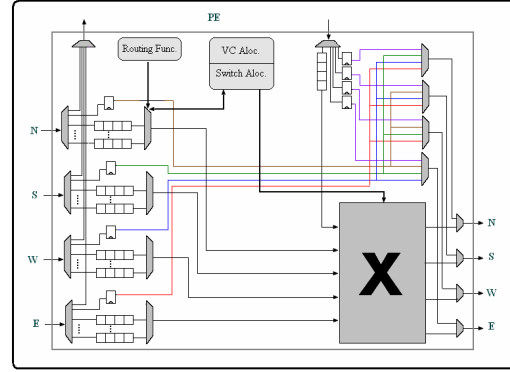
do not use the packet-switched network resources, expecting the inter-router links which are shared between them. However, VIPs do not reserve the links, but they are only prioritized over the packet-switched network. As a result, a link is devoted to a VIP only when there is a VIP flit available for sending over it; hence the VIP connections do not reduce the network throughput. In addition, by using preconfigured VIP links, the long setup time of the circuit-switching is removed. It is also different from express virtual channels [2], since the VIP connections are not limited to span along a single dimension and the router architecture gives hardware support for dynamically reconfiguring the connections to adapt to the target application communication pattern. Moreover, the VIP connections are constructed between the communicating nodes and completely connect the source and destination of a communication flow.

Since VIPs are constructed based on the target application, this paper proposes an application-specific NoC design methodology. However, unlike most of the existing application-specific NoC design methods which try to customize the NoC for the communication pattern of a single application, the VIP circuits are dynamically reconfigurable and can be configured based on the traffic pattern of the currently running application. It is a critical feature as several different applications are integrated onto today's single SoC chips and communication characteristics can be very different across the applications.

In the next sections, the proposed router architecture, the proposed NoC design methodology which exploits the architecture to optimize the NoC for a target application, and the evaluation results are presented.

## 2. The Proposed NoC Architecture

Figure 1 displays the microarchitecture of the router in our study. As shown in the figure, in one virtual channel (virtual channel 0) in each physical channel, the buffer is replaced with a latch (1-flit buffer). These virtual channels are devoted for establishing VIP links between nodes. Moreover, in order to completely bypass the router pipeline, there is a multiplexer associated with each output port through which one of the latches connects to the output port. Since a packet coming through an input port does not loop back, each latch is connected to three multiplexers. The output ports are further a multiplexed of the output of these multiplexers which carry VIP flits and the crossbar outputs which handle regular packet-switched flits.



**Figure 1. The proposed router architecture**

In the architecture, applying a simple look-ahead routing [7] (or using a flag for VIP links, since no routing is performed for the flits traveling on VIP links), a flit destined for the local PE is ejected after the de-multiplexer. Thus, the flits save the power and delay related to the router pipeline stages in the destination node. In our architecture, the VC allocator, switch allocator, and route computation units only handle the VCs other than that designated for VIP links.

A VIP link for a communication flow is established by directly connecting the latches in the routers along the path between the source and destination nodes of the communication. To this end, the multiplexers in the source and intermediate routers are configured (via the select line) in such a way that the latches in each router are connected to proper output port and then to the latch in the next router along the VIP link.

The source node of a communication trace selected for traveling on a VIP link sends corresponding packets using VC number 0 of the output port through which the flits should be sent. A latch in the processing element (PE)/router interface is allocated for keeping the flits of the communication flow. Since the VIP is already set up, the multiplexer of the designated output port selects this latch and connects it to the output. When the flits reach the next node, they are de-multiplexed based on their VC identifier and the VIP flits (identified by VC identifier 0) are stored in the VIP latch of the input port. Similarly, this latch and the other latches in the intermediate routers along the path are connected to the proper output port and construct a VIP link toward the destination. The use of multiplexers provides the flexibility to change the connectivity between the latches and output ports and allows constructing new VIP links at run-time in order to meet the demands of various applications.

By increasing the priority of the VIP latches over the crossbar, the point-to-point link flits are able to gain automatic passage through the output port without any contention. As a result, flits traveling on VIP

circuits will not find any busy channel in their way towards the destination node. If there is not a demand for very low-latency communication on a VIP link, a fair arbitration between the VIP link and packet-switched flits can be carried out. In this case, the flits still benefit from the low-power communication provided by VIP links. Prioritizing the flits traveling on VIP links over packet-switched flits may produce starvation if a VIP link always has incoming flits to forward. It can be avoided by informing the source of a VIP about the starvation in one of the intermediate routers along the path using a method like the one proposed in [2].

Since the flits bypass the router pipeline, the power consumption related to buffer read and write, routing calculation, arbitration, and crossbar traversal are removed. Instead, the flits are being latched in each node across a VIP link. Since the latches provide pipelining over VIP links and also act as a repeater for them, VIP links can offer a power and latency close to physical pipelined point-to-point links. However, the flits should travel through one 4-to-1 and one 2-to-1 multiplexer. Amongst different options [8], we use transmission gates to implement the multiplexers. The capacitance evaluation using Orion power model [9] for 70 nm technology shows that the capacitance of the multiplexer-based connections (with the size of the transmission gates in the crossbar of the Orion library) is  $7.2 \times 10^{-14}$ F. We set the length of the NoC links to 1mm which has an approximate capacitance of  $2 \times 10^{-12}$ F. Therefore, multiplexer overheads can be completely affordable due to this wire-length which is a realistic length in 70 nm chips [10].

## 2.1. The VIP construction algorithm

The VIP links are exploited to improve the power and performance metrics of the NoC when running a specific target application. In this work, we focus on reducing the energy consumption of the NoC. To this end, the high-volume communication flows are selected and a VIP link is reserved between their source and destination nodes, provided that there are sufficient free resources in the network. In general, by directing the high volume communications through the VIP links a larger number of packets take advantage of the low-power and low-latency communication provided by these links and more power saving can be achieved. Furthermore, more reduction in average message latency (as an important criterion for NoC performance) can be obtained, as well.

The problem is to physically map the cores of an application, described by a *Task Graph* (TG), onto

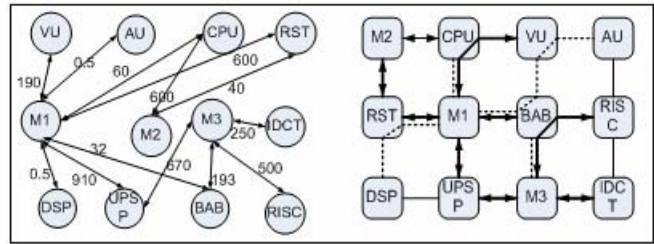


Figure 2. The MPEG4 task-graph and its mapping on a  $4 \times 3$  mesh

different tiles of a mesh network and then find as many VIP links as possible for the task-graph edges such that the power consumption of the NoC is minimized.

The core mapping is accomplished by modifying NMAP, a simple and fast heuristic power-aware core mapping and route generation method presented in [11], with respect to VIP construction. We map the task-graph cores in the order specified in the baseline NMAP where the core having maximum communication demand is placed onto one of the mesh nodes with maximum number of neighbors. Afterwards, we map the core that communicates most with the already mapped cores and find a route for its communication flows. The route selection is done in two phases. The first phase is done as soon as the source and destination nodes of an edge are mapped into the network and involves finding a VIP for the edge. To this end, all unallocated mesh nodes are analyzed for placing the newly selected core. The core is mapped onto the node which allows constructing more VIP connections between it and the already mapped cores to which it is connected in the task-graph. The edges for which a VIP could not be found should be directed through the packet-switching network. After mapping all task-graph nodes and establishing VIP links, the unmapped edges are selected in order of their communication volume and a path is found for them through the packet-switched network. The algorithm considers all shortest paths between the source and destination of the edge and selects the one with minimum overlap with VIP connections. If it is not possible to find a path with non-overlapping link, a path which overlaps with lower-traffic VIP connections is selected. The found paths should not violate the bandwidth of the network links (router input ports) to avoid congestion. For example, Figure 2 displays the task-graph and VIP links constructed in a  $4 \times 3$  mesh for an MPEG4 decoder [12]. The VIP links and packet-switched routes are shown in bold and dashed lines, respectively. After finding a route for all task-graph edges, VIP links are constructed by configuring the VIP multiplexers, while circuit switched routes are

established by appropriately setting the routing table of the network routers.

### 3. Evaluation

To evaluate the proposed NoC architecture and design methodology we select an MPEG4 decoder [12] as a case study. We perform simulations using Ximulator, a fully parameterized simulator for interconnection networks [13]. We augment it with the Orion power library [8] to calculate the power consumption of the networks. Simulations experiments are performed for a 128-bit wide system. Moreover, the process feature size and working frequency of the routers is set to 70nm and 250 MHz, respectively in the Orion library. In the simulation, packets are generated with exponential distributions and the communication rates between any two nodes are set to be proportional to the communication volume between these two nodes in the task-graph. This task-graph-based simulation approach has been introduced in [14].

Table 1 displays the power consumption of a conventional NoC and the proposed NoC architecture. In the conventional architecture, mapping and routing is performed based on the baseline NMAP algorithm on a mesh with 2 VCs per physical channel. The proposed NoC is evaluated in two cases: when the VIP latches are replaced with one virtual channel (1 VC and 1 latch per physical port), and when the latches are added to the current set of VCs as an extra VC (2 VCs and 1 latch per physical port). The latter case does not affect the packet-switched network throughput, but imposes more area overhead. The results show that by virtually connecting the cores communicating frequently in the proposed router architecture we can effectively reduce the power consumption and average message latency of the NoC. The area overhead is estimated using Orion and scaling the area analysis results in [15] to 70nm technology and shows 15% overhead for the first case (1 VC + 1 latch) and 25%, for the second case (2 VCs + 1 latch). This overhead can be compensated by the obtained power reduction and performance improvement.

**Table 1. The power consumption and latency (cycles for 32-flit packets) in the proposed and a conventional NoC**

	Conv. NoC	Proposed NoC		Proposed/Conv.	
		1 VC+ latch	2 VC+ latch	1 VC+ latch	2VC+ latch
Power (W)	1.80	0.88	0.96	0.48	0.52
Latency	78.25	33.12	33.10	0.42	0.42

### 4. Conclusion

In this paper, we presented a packet-switch router architecture that can provide low-power and low-latency dedicated virtual point-to-point (VIP) links between the source and destination of communication flows. This is achieved by means of a subset of virtual channels which bypass the pipeline of the intermediate routers along the path. Afterwards, amongst different potential applications of the VIP links, we focused on power reduction and developed an application-specific NoC design methodology which exploits the VIP links to reduce the power consumption of the NoC. Simulation results showed that compared to a conventional NoC, the proposed NoC architecture reduces the power consumption by 45%, on average.

### References

- [1]L. Benini, and G. De Micheli. "Networks on Chip: a New Paradigm for Systems on Chip Design", in *Proc. Design, Automation and Test in Europe (DATE)*, 2002, pp. 418–419.
- [2]A. Kumar, et al. "Express Virtual Channels: Towards the Ideal Interconnection Fabric", in *Proc. 34<sup>th</sup> ISCA*, 2007.
- [3]H. G. Lee, et al., "On-chip Communication Architecture Exploration: A Quantitative Evaluation of Point-to-point, Bus, and Network-on-Chip Approaches", in *ACM Trans. on Design Automation of Electronic Systems (TODAES)*, Vol.12, No.3, 2007.
- [4]W. J. Dally, "Express Cubes: Improving the Performance of K-ary N-cube Interconnection Networks", in *IEEE Trans. on Computers*, Vol. 40, No. 9, 1991.
- [5]G. Michelogiannakis, et al., "Approaching Ideal NoC Latency with Pre-Configured Routes", in *Proc NOCS'07*, 2007, pp. 153-162.
- [6]J. Duato, et al., "A High Performance Router Architecture for Interconnection Networks", in *Proc. Int. Conf. Parallel Processing*, 1996, pp. 61-68.
- [7]H. J. Kim, et al., "A Low Latency Router Supporting Adaptivity for On-Chip Interconnects", in *Proc. Design Automation Conference*, 2005, pp. 559-564.
- [8]H. Wang, "A detailed Architectural-Level Power Model for Router Buffers, Crossbars and Arbiters," Technical Report, Princeton University, 2004.
- [9]H. Wang, X. Zhu, L. Peh, and S. Malik, "Orion: A Power-Performance Simulator for Interconnection Networks", in *Proc. 35<sup>th</sup>. MICRO*, Turkey, 2002.
- [10]R. Mullins, et al., "The Design and Implementation of a Low-Latency On-chip Network", in *Proc 11th ASPDAC*, 2006.
- [11]S. Murali, and G. De Micheli, "Bandwidth-constrained Mapping of Cores onto NoC Architectures", in *Proc. Design Automation and Test in Europe (DATE)*, 2004, pp. 896-901.
- [12]K. Srinivasan, and K. Chatha, "A Low Complexity Heuristic for Design of Custom Network-on-Chip Architectures", in *Proc. Design Automation and Test in Europe (DATE)*, 2006.
- [13]Ximulator NoC Simulator: [www.ximulator.org](http://www.ximulator.org), 2007.
- [14]J. Hu, and R. Marculescu, "Application Specific Buffer Space Allocation for Networks on Chip Router Design", in *Proc. IEEE/ACM Intl. Conf. on Computer Aided Design*, 2004.
- [15]M. Kim, D. Kim, and E. Sobelman, "NoC Link Analysis under Power and Performance Constraints", in *Proc. ISCAS*, Greece, 2006.