

Silicon Photonic Interconnects: Minimizing the Controller Latency

Felipe Gohring de Magalhães
PPGCC/PUCRS
Ecole Polytechnique Montréal
felipe.magalhaes@acad.pucrs.br

Mahdi Nikdast
Colorado State University
mahdi.nikdast@colostate.edu

Yule Xiong
Ciena Corp.
yule.xiong.ele@gmail.com

Fabiano Hessel
PPGCC/PUCRS
fabiano.hessel@pucrs.br

Odile Liboiron-Ladouceur
McGill University
odile.liboiron-ladouceur@mcgill.ca

Gabriela Nicolescu
Ecole Polytechnique Montréal
gabriela.nicolescu@polymtl.ca

ABSTRACT

Silicon photonic interconnects (SPIs) have emerged as a promising solution to outperform the communication infrastructure in multi-processor systems-on-chip (MPSoCs). Routing a message from one node to another in an MPSoC integrating SPIs, several photonic components (e.g., switching elements) need to be configured to realize an optical path between sending and receiving nodes. Such configurations are performed in an electronic controller, which, if not fast, imposes high latency in SPIs, constraining the application of SPIs in MPSoCs. Realizing a full exploitation of SPIs, this paper presents a lookup-table-based centralized controller (LUCC). We indicate that LUCC has the lowest latency among the state-of-the-art controllers for SPIs while it can be applied to different SPI architectures. Employing acceleration techniques based on off-line routings, we report (simulation and prototyping) a worst-case control latency smaller than 5 ns. Moreover, LUCC is experimentally integrated with a photonic switch in the lab, where we show contention resolution in one clock cycle.

CCS CONCEPTS

• **Hardware** → **Photonic and optical interconnect; Emerging optical and photonic technologies; Metallic interconnect**; • **Networks** → **Network control algorithms**;

KEYWORDS

Silicon Photonic Interconnects; Low-Latency Controller

ACM Reference Format:

Felipe Gohring de Magalhães, Mahdi Nikdast, Yule Xiong, Fabiano Hessel, Odile Liboiron-Ladouceur, and Gabriela Nicolescu. 2018. Silicon Photonic Interconnects: Minimizing the Controller Latency. In *GLSVLSI '18: 2018 Great Lakes Symposium on VLSI, May 23–25, 2018, Chicago, IL, USA*. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3194554.3194609>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GLSVLSI '18, May 23–25, 2018, Chicago, IL, USA

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5724-1/18/05...\$15.00

<https://doi.org/10.1145/3194554.3194609>

1 INTRODUCTION

Silicon photonic interconnects (SPIs) can boost the communication in multiprocessor systems-on-chip (MPSoCs) and data centers, but their full capabilities are curbed by high latency electronic controllers [1]. Such controllers are essential to configure different photonic elements (e.g., switching elements), realizing an optical path between sending and receiving nodes in an MPSoC integrating SPIs. Different techniques have been proposed for controlling and multiplexing SPIs, namely path, time and wavelength division. Path division (PD) techniques have great scalability [8], while time division (TD) ones, deployed as centralized cores, present high system controllability [4]. Still, such techniques impose high control latency in SPIs, hence further improvement in controllers response time is required for practical deployment of SPIs in MPSoCs.

In this paper, we present the design and integration of a lookup-table-based centralized controller (LUCC) for SPIs. Comparing LUCC with the state-of-the-art controllers [5, 6, 11], it has a low latency when computing requests and performing the network configuration in SPIs, while being flexible and applicable to different network topologies. LUCC enables all connections to be configured within a single clock cycle even when the switch configuration dynamically changes due to various node requests. In particular, LUCC is designed based on using a lookup table with a fast algorithm for conflict resolution. It takes advantage of an optimization algorithm to reduce the size of the lookup table, and hence reducing the memory utilization. We indicate LUCC simulation and FPGA prototyping results with a latency smaller than 5 ns in the worst case. Furthermore, we integrate LUCC with a fabricated Mach-Zehnder Interferometer (MZI)-based photonic switch for experimental validation and demonstration, where we report a low latency in the switch performance when using LUCC.

2 LUCC: A LOW-LATENCY CONTROLLER FOR SPIS

Considering silicon photonic interconnects, the data path is through optical communication (i.e., light propagation), decreasing the network latency, where the network routing and configuration are addressed using an electronic controller. For example, in emerging 3D stacked hybrid optical-electronic networks, the electronic layer is responsible to configure the photonic elements on the photonic layer [2]. The controller receives requests from input nodes (i.e.,

processing cores) and solves conflicts to guarantee message transmissions. Also, the controller is responsible for configuring different photonic switching elements, setting up optical paths between sending and receiving nodes. Control latency is the key parameter to ensure the benefits of SPIs in terms of high throughput and low latency, hence its design must focus on reducing the execution time.

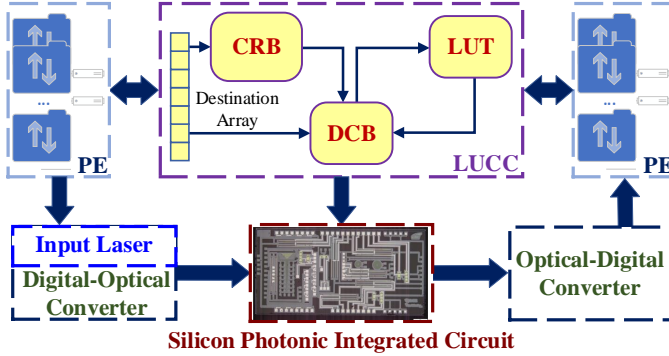


Figure 1: An abstract overview of the proposed LUCC controller architecture.

LUCC is designed to improve the control latency in SPIs. Fig. 1 illustrates an overview of LUCC architecture and how it can interact with different processing elements (PEs) and SPIs [3]. As can be seen, the main building blocks are:

- **Conflicts Resolution Block (CRB):** this block is responsible for detecting and solving destination conflicts using a given algorithm;
- **Lookup Table (LUT) - Memory:** this block is used to store static data in a form of a lookup table, available to the controller during the runtime. Using a lookup table in the memory, we can reduce the path configuration time, thus reducing control latency, and;
- **Dynamic Setup Block (DSB):** this block is responsible for on-line calculations (e.g., path attribution and reading memory addresses) employing a real-time calculation (RTC) unit.

LUCC design methodology can be divided into off-line routing analysis and on-line requests computations. Off-line routing analysis seeks to find the best routes in the network, using Shortest Path First (SPF) algorithms, such as Dijkstra algorithm [7]. The routes information is stored in fast access memories (i.e., a lookup table). During the execution, LUCC relies on three main blocks (see Fig. 1): the CRB, the memory arrays created off-line in the LUT, as well as the DSB.

CRB is a hardware block responsible for detecting conflicts in targeted PEs. A conflict is defined as any situation in which two or more source PEs are targeting the same destination PE at the same time. CRB works as follows: firstly, it analyzes all the input requests, looking for a conflict. Secondly, if a conflict is found, a Round-Robin (RR) algorithm is applied to determine which PE should have its access granted. Detecting a conflict, a matrix method is used, in which for every new request, all the source-destination pairs are mapped to a matrix \mathcal{R} of requests. Then, each column j is checked

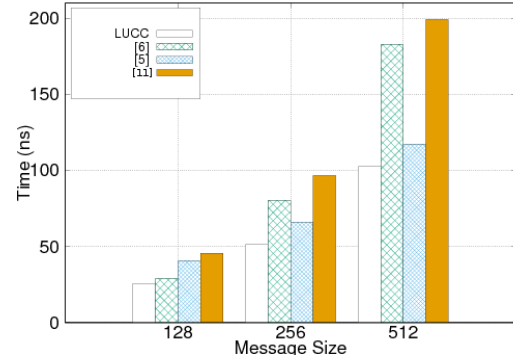


Figure 2: Comparing LUCC latency with that of the state-of-the-art controllers under different message sizes.

for any possible conflicts. The matrices are created based on the indices of the requesting input port(s) and the requested output port(s). For example, $\mathcal{R}(i, j) = 1$ denotes the input port I_i requests to access the output port O_j in the switch, such that:

$$\forall ij, I_i \text{ request } O_j \leftrightarrow \mathcal{R}(i, j) = 1.$$

As a result, the \mathcal{R} matrix for a 3×3 photonic switch, for example, where all the inputs are requesting to communicate with the output two (i.e., $\langle 0 \rightarrow 2, 1 \rightarrow 2, 2 \rightarrow 2 \rangle$) can be defined as:

$$\mathcal{R} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Considering the matrix \mathcal{R} , a conflict can be detected as:

$$\forall i'j, i \neq i', \neg \text{XOR}[\mathcal{R}(i, j)] \wedge \text{OR}[\mathcal{R}(i', j)] \implies \text{conflict}(O_j) = 1.$$

As the matrix can be accessed directly (i.e., the hardware implementation is a register), no extra processing is needed, thus accelerating the conflict detection.

The DSB is the central core of LUCC that evaluates the status signals (i.e., conflict or no conflict) from the CRB block as well as the routes accessed in the LUT. The DSB simplifies the physical design implementation, without which the CRB and the LUT would imply a prohibitive resource usage. In other words, if only the CRB and LUT blocks were considered, a large (physically unbearable) chip area would be needed, as the memory array scales with an increase in the number of paths combinations. Employing the DSB, the original LUT is compressed, and the DSB executes on-line calculations to access the correct LUT positions. We apply a reduction method to the LUT, which is generated off-line, reducing its size, enabling using the LUT when the network size scales. The reduction method is based on finding repeated (i.e., redundant) entries in the LUT, which reduces its final memory print.

3 RESULTS AND DISCUSSION

Complement and all-to-all traffic patterns are considered to assess LUCC performance under various request conditions. The complement traffic pattern is used to verify the largest paths in the network, while the all-to-all one comprises all possible communication combinations in the network (i.e., all the PEs request access

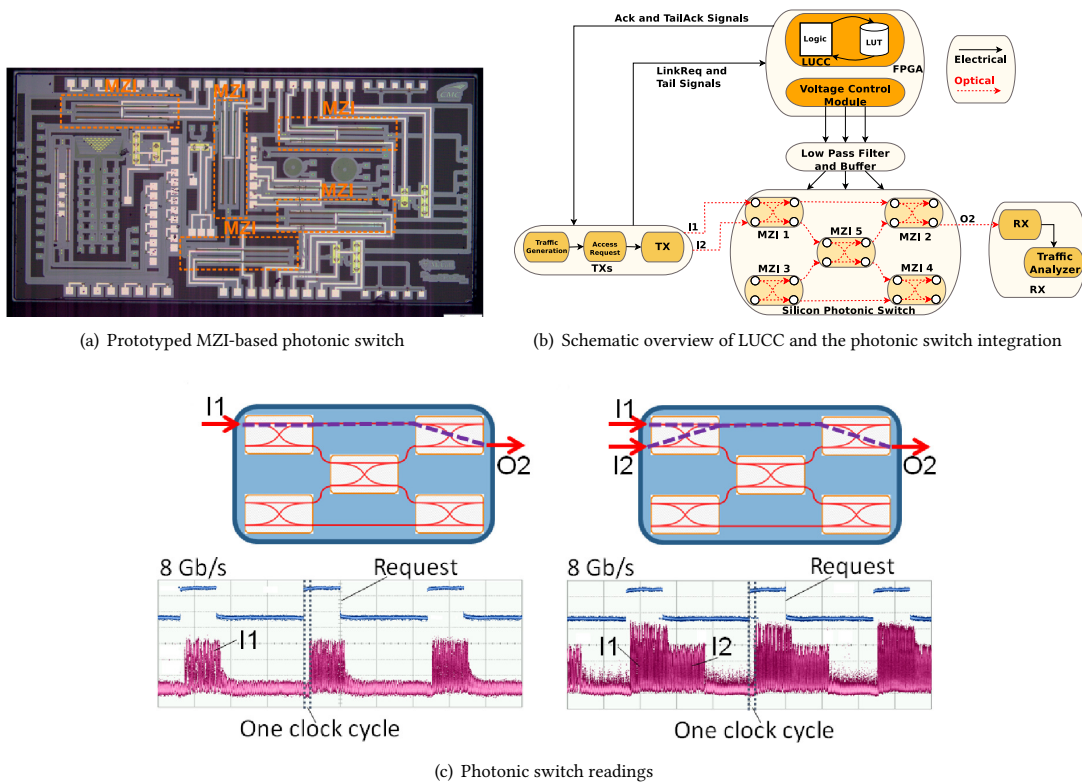


Figure 3: (a) MZI-based prototyped photonic switch; (b) overview of the FPGA-based LUCC co-designed with the MZI-based multistage photonic switch; and, (c) 4×4 MZI-based photonic switch readings.

to each other one by one). In our validations, the traffic load is not considered due to the fact that a validation based on a traffic load usually aims at evaluating topologies, while we focus on the controller evaluation. Simulations are performed in commercial simulation tools and LUCC is prototyped on Altera’s and Xilinx’s FPGAs. The obtained results for the Altera Stratix IV FPGA showed a top operation frequency of 270 MHz (≈ 3.7 ns period). As for Xilinx’s Virtex V FPGA, the top operation frequency is 295 MHz (≈ 3.4 ns period). Also, the design was synthesized using the proposed flow for the STMicroelectronics 65 nm technology process. The minimum period for this technology is ≈ 1.1 ns.

Fig. 2 compares LUCC latency with the state-of-the-art controllers in an 8×8 Beneš network, in which we indicate LUCC has the lowest latency under different message sizes. Considering a predefined photonic switch latency, the latency for each optical bit to pass through the network is rounded to 200 ps. The comparison is performed by analyzing the total time it takes for a message to be arbitrated and passes through the network: $TotalTime = CL + Nob \times TD$. Here, CL is the control latency, Nob is the number of transmitted bits, and TD is the transmission delay. Three different message sizes (128 B, 256 B, 512 B) are considered. According to Fig. 2, the solution in [6] deploys a centralized control core without any acceleration techniques, which compromises its latency and scalability. Also, the approach proposed in [11] relies on PD techniques, which can lead to a high latency as the network scales (i.e., $>8 \times 8$). Finally, the solution presented in [5] is based on an operation frequency of

5 GHz, and hence the validations are under behavioral simulations only, not considering a realistic physical implementation.

Fig. 3(a) illustrates the prototyped MZI-based photonic switch, integrated with LUCC for a realistic demonstration [9, 10]. The photonic switch is a 4×4 integrated Silicon Photonic (SiP) switch based on a Spanke-Beneš topology with five integrated 2×2 MZIs directly controlled by LUCC (see the middle block in Fig. 3(b)). Fig. 3(b) indicates the setup schematic of the FPGA-based LUCC co-designed with the MZI-based photonic switch. LUCC is prototyped in commercial FPGA from Altera. Fig. 3(c) presents the readings for the photonic switch being controlled by LUCC, in which we indicate the measured dynamic switching of a payload signal. It is generated at the FPGA running at 50 MHz for gate signal (bar and cross). Both LUCC and payload traffic are executed in the FPGA. At the receiver, we employ a photodetector connected to a digital communication analyzer (DCA) for the readings. Considering the observed waveform, it is possible to see the contention resolution in one clock cycle, as the payload is read in the switch output.

4 CONCLUSION

This work presents LUCC, a centralized controller for silicon photonic interconnects. Results, obtained through simulation, FPGA prototyping, as well as experimental integration with a fabricated photonic switch, show a fast response time when employing LUCC in SPIs. Indeed, it takes only one clock cycle delay to compute

each request in the network. Our future work comprises extending LUCC's scalability, employing LUCC for larger-radix networks. Also, for a realistic demonstration with different photonic switches, the integration of LUCC with prototyped microring resonator (MR)-based photonic switches will be performed.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support provided by ReSMiQ, CAPES and NSERC for the realization of this work.

REFERENCES

- [1] A. Biberman, B. G. Lee, N. Sherwood-Droz, M. Lipson and K. Bergman. 2010. Broadband Operation of Nanophotonic Router for Silicon Photonic Networks-on-Chip. *IEEE Photonics Technology Letters* 22, 12 (2010), 926–928. DOI : <http://dx.doi.org/10.1109/LPT.2010.2047850>
- [2] E. Fusella and A. Ciarlo. 2017. H²ONoC: A Hybrid Optical-Electronic NoC Based on Hybrid Topology. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 25, 1 (January 2017), 330–343. DOI : <http://dx.doi.org/10.1109/TVLSI.2016.2581486>
- [3] F. G. de Magalhães, R. Priti, M. Nikdast, F. Hessel, O. Liboiron-Ladouceur and G. Nicolescu. 2016. Design and Modelling of a Low-Latency Centralized Controller for Optical Integrated Networks. *IEEE Communications Letters* 20, 3 (2016), 462–465.
- [4] H. Yan, D. A. Maltz, H. Gogineni and Z. Cai. 2007. Tesseract: A 4D Network Control Plane. In *NSDI*. USENIX Association, Cambridge, MA. <https://www.usenix.org/conference/nsdi-07/tesseract-4d-network-control-plane>
- [5] J. Chan and K. Bergman. 2012. Photonic Interconnection Network Architectures Using Wavelength-selective Spatial Routing For Chip-scale Communications. *IEEE/OSA Journal of Optical Communications and Networking* 4, 3 (2012), 189–201.
- [6] J. Jian, M. Lai and L. Xiao. 2016. A Fast Hierarchical Arbitration in Optical Network-on-Chip Based on Multi-Level Priority QoS. *IEICE Transactions on Communications* E99.B, 4 (2016), 875–884. DOI : <http://dx.doi.org/10.1587/transcom.2015EBP3382>
- [7] N. Jasika, N. Alispahic, A. Elma, K. Ilvana, L. Elma and N. Nosovic. 2012. Dijkstra's Shortest Path Algorithm Serial and Parallel Execution Performance Analysis. In *International Convention MIPRO*.
- [8] P. Lotfi-Kamran, M. Modarressi and H. Sarbazi-Azad. 2016. An Efficient Hybrid-Switched Network-on-Chip for Chip Multiprocessors. *IEEE Transactions on Computers* 65, 5 (2016), 1656–1662. DOI : <http://dx.doi.org/10.1109/TC.2015.2449846>
- [9] Y. Xiong, F. G. de Magalhães, B. Radi, G. Nicolescu, F. Hessel and O. Liboiron-Ladouceur. 2016. Towards a Fast Centralized Controller for Integrated Silicon Photonic Multistage MZI-based Switches, In *Optical Fiber Communication Conference - paper W1J.2*. *Optical Fiber Communication Conference - paper W1J.2* (2016).
- [10] Y. Xiong, F. G. de Magalhães, G. Nicolescu, F. Hessel and O. Liboiron-Ladouceur. 2017. Co-design of a Low-latency Centralized Controller for Silicon Photonic Multistage MZI-based Switches, In *Optical Fiber Communication Conference - paper Th2A.37*. *Optical Fiber Communication Conference - paper Th2A.37* (2017).
- [11] Z. Li and T. Li. 2013. ESPN: A Case For Energy-star Photonic-on-Chip Network. In *IEEE International Symposium on Low Power Electronics and Design*. 377–382.