Energy efficiency, fault tolerance, and emerging on-chip interconnects for manycore architectures

Cédric Killian

UNIVERSITÉ DE RENNES 1 — Inria — UMR IRISA

HABILITATION À DIRIGER DES RECHERCHES

Energy efficiency, fault tolerance, and emerging on-chip interconnects for manycore architectures

Cédric KILLIAN

Université de Rennes 1

Spécialité Informatique

prepared in Taran Team, Inria, IRISA, Lannion

June 13, 2022

Reviewers :	Prof. Alberto Bosio	-	Ecole Centrale de Lyon
	Prof. Gabriella Nicolescu	-	Polytechnique Montreal
	Prof. Fréderic Petrot	-	Institut Polytechnique de Grenoble
Examinators :	Prof. Daniel Chillet	-	University of Rennes 1
	Research Director Gilles Sassatelli	-	CNRS, LIRMM
	Prof. Olivier Sentieys	-	University of Rennes 1

Contents

List of Acronyms **iii**

1 Introduction **1**
1 Works overview . . . 1
1.1 Context of the research . . . 1
1.2 Research focus and document organization . . . 2
2 Summary of research and teaching activities . . . 7
2.1 Curriculum Vitae . . . 7
2.2 Scientific supervisions . . . 7
2.3 Research projects . . . 10
2.4 Services . . . 11
2.5 Teaching activities . . . 13
2.6 Publications . . . 15

2 Improving energy-efficiency of nanophotonic on-chip interconnects **21**
1 Introduction . . . 21
2 Energy-Performance trade-offs in reconfigurable ONoC . . . 23
2.1 Reconfigurable ONoC for 3D Manycore architectures . . . 23
2.2 Related works . . . 23
2.3 Energy-Performance tradeoffs . . . 24
2.4 Framework for energy-performance exploration . . . 26
2.5 Results . . . 28
2.6 Conclusion and perspectives . . . 31
2.7 Research dissemination . . . 31
3 Distance Aware Approximate Nanophotonic Interconnect . . . 32
3.1 Context of the work . . . 32
3.2 Proposed data and distance aware optical link . . . 32
3.3 Data approximation scheme . . . 34
3.4 Results . . . 37
3.5 Conclusion and perspectives . . . 45
3.6 Research dissemination . . . 47

3 Fault- and Error-tolerant architectures **49**
1 Introduction . . . 49
2 Minimizing the impact of permanent errors in NoC . . . 50
2.1 Context of the work . . . 50
2.2 Related work . . . 51
2.3 Shuffling bit to minimize errors for error resilient applications . . . 52
2.4 Evaluation of the method . . . 57
2.5 Conclusion and perspectives . . . 62

2.6 Research dissemination 63
3 Timing errors in Dynamic Voltage and Frequency Scaling architectures 64
3.1 Tolerating error for energy improvement in Near Threshold Computing . . 64
3.2 Timing error detections for adaptive over-/under-clocking 70
3.3 Conclusions and perspectives 77
3.4 Research dissemination 77

4 Research perspectives **79**
1 Research perspectives 79
1.1 Fast design space exploration of multi-technologies NoC for manycore architectures 80
1.2 Cache coherence protocols for emerging on-chip interconnects 81
1.3 Fault tolerant accelerator for AI 81
1.4 Arithmetic logic circuit in 2D semiconductors for approximate computation 82

Bibliography **83**

List of Acronyms

BER Bit Error Rate . . . 3
BiSuT Bit Shuffling meThod . . . 52
BIST Built-In Self-Test . . . 53
BSL Bit Stream Length . . . 33
CNN Convolutional Neural Network . . . 57
DAG Directed Acyclic Graph . . . 24
ECC Error Correcting Code . . . 4
FET field-effect transistor . . . 82
FP Floating-Point . . . 34
HLS High Level Synthesis . . . 58
IP Intellectual Property . . . 51
LSB Least Significant Bit . . . 33
LSS Least Significant Sub-flit . . . 56
DNN Deep Neural Network . . . 81
DVFS Dynamic voltage and Frequency Scaling . . . 4
MHE Multiple Hard Error . . . 50
MMI MultiMode Interference . . . 32
MR Micro Ring Resonator . . . 23
MSB Most Significant Bit . . . 35
MSE Mean Square Error . . . 40
MSS Most Significant Sub-flit . . . 56
MWMR Multi Writers Multi Readers . . . 21
NI Network Interface . . . 56
NoC Network-on-Chips . . . 1
OOK On Off Keying . . . 23

ONI Optical Network Interface 4
ONoC Optical Network-on-Chip 2
ORA Output Response Analyzer 53
QoR Quality of Result 32
R-BiSuT Region-based BIt-Shuffling Technique 62
SEU Single Event Upset 5
SHE Single Hard Error 50
SF Sub-Flit 53
SoC System on Chip 50
SNR Signal to Noise Ratio 24
SWMR Single Writer Multiple Reader 32
TMR Triple Modular Redundancy 51
TPG Test Pattern Generator 53
TSV Through Silicon Vias 23
VC Virtual Channel 62
WDM Wavelength Division Multiplexing 23

Chapter 1

Introduction

Internships

Contents

1 Works overview 1
1.1 Context of the research 1
1.2 Research focus and document organization 2
2 Summary of research and teaching activities 7
2.1 Curriculum Vitae 7
2.2 Scientific supervisions 7
2.3 Research projects 10
2.4 Services 11
2.5 Teaching activities 13
2.6 Publications 15

1 Works overview

1.1 Context of the research

Since few years, we are witnessing the emergence of manycore architectures, namely to the implementation of massive parallelism on a single chip. Associated with the shrinking size of the transistors, announced reaching a 3nm technology in 2022 by TSMC, these manycore architectures should provide the integration of thousands of heterogeneous cores allowing huge parallel computation capabilities suitable for High Performance Computing (HPC) [Flich *et al.* 2017]. These parallelism capabilities obviously generate an enormous amount of data exchanges making the on-chip communication medium a key element of the overall system performance of the system. In the last decade, Electrical Network-on-Chips (NoC) have emerged as an efficient solution for multicore architectures, in the range of tens of cores on a-chip, to circumvent the parallelism limitations of traditional buses. Nevertheless, as the manycore era progresses, electrical NoCs suffer from scalability in terms of latency and energy due to a huge increase on the number of hops between cores [Karkar *et al.* 2016, Wolf *et al.* 2008], hence emerging technologies are called to supplement this traditional interconnect.
Recent advances in integration technologies have allowed the advent of silicon photonics [Atabaki *et al.* 2018] given rise to new on-chip interconnection media Optical-NoC (ONoC) [Le Beux *et al.* 2014, Wang *et al.* 2015]. Indeed, nanophotonic interconnects are a promising

solution to overcome bandwidth and latency issues, as optical signals propagate near speed-of-light in waveguides. However, their implementations remain challenging due to the low efficiency of the lasers, which are key devices in such interconnects.
Meanwhile, technology scaling and transistor density increase enabled voltage reduction. As a result, the intrinsic failure rate of electronics is increased [Srinivasan *et al.* 2004] while the transistor size reaches 10nm and below [Bohr 2018]. In this nanometer technology era, cores and NoCs became more sensitive to faults. This may affect their functionality, which can be crucial for application like autonomous vehicles.
In addition to the advances in on-chip interconnects and technology scaling, new computing paradigms emerged giving new opportunities to improve the energy efficiency of manycores.

For instance, approximate computing allows to rely on precision reduction of the data representations, which lowers design constraints and improves performances at the cost of Quality of Result (QoR) degradation measured with output quality metrics [Mittal 2016, Xu *et al.* 2016]. These new paradigms require to rethink the whole computing stack of computing architectures, from device to software levels.

1.2 Research focus and document organization

Building the next generation of computing architectures requires to cover different levels of abstraction, from the system level to the circuit level, and by considering emerging technologies. My research activities are cross-disciplinaries and include computer engineering, software engineering, and circuit design. I developed an expertise in on-chip interconnects based on emerging technologies such as silicon photonics as well as on fault tolerance and hardware accelerators. They are organized around two major topics: i) energy efficient on-chip interconnect networks and ii) fault tolerant architectures, and are introduced in the following sections.

1.2.1 Improving energy-efficiency of nanophotonic on-chip interconnects

Silicon photonics is an emerging technology considered as one of the key solutions for future generation of on-chip interconnects. It is based on the use of lasers and photodetectors for the conversion between electrical and optical domains, and on waveguides to carry the optical signals. It exhibits low latency, high bandwidth, and scalability capacities to answer large scale manycore architecture needs. However as a maturing technology numerous challenges have to be tackled before large adoption in future computing architectures. Besides the technological improvements, the main challenges to build an Optical Network-on-Chip (ONoC) to interconnect heterogeneous cores are i) interconnect topologies and layouts, ii) channel allocations in shared optical links, and iii) energy-efficiency.

Exploring design space and channel allocations: To address topology and allocation problems, we proposed an off-line optimization methodology allowing to explore design space, such as the number of wavelengths, the number of waveguides, and the number of laser power levels. The methodology also improves communication bandwidth allocation by providing trade-offs between execution time and communication energy costs. Our method is proposed

as a framework considering technological and architectural parameters, as well as Bit Error Rate (BER) application needs. For a given application, described as a task graph, our framework provides a set of optimized channel allocation solutions. Each solution provides a trade-off between energy and execution time. These solutions appear as a Pareto front, with two extreme bounds: a low-power solution, which tends to minimize the number of used wavelengths to reduce the crosstalk power penalty, and a high-performance solution, for which multiple wavelengths are allocated to shorten the communication time. As an example, for a 63-task application, the relative variation in the execution time and energy is 71% and 44% respectively. Solutions showing good energy- performance trade-offs are also found. Compared to baseline solutions for which laser power is fixed, our method leads to 74% energy reduction on average. The method also allows for the exploration of ONoC design parameters such as the number of wavelengths, waveguides, and laser power levels.

This work is detailed in Chapter 2 Section 2, and related work disseminations are: [IC1] [IC4], [IJ3] [IC7], [IC9], [IC12], [IC14], [NC5], [NC7], [IT4], [IT6], [IT8], [IT9]. IJ, IC, NC, and IT stand for International Journals, International Conferences, National Conferences and Invited Talks, respectively. **My list of publications is located in my Curriculum Vitae in Chapter 1 Section 2.6**.

Distance aware approximate nanophotonic interconnect: Regarding improvements on energy efficiency, we proposed to benefit from the approximate computing paradigm to reduce the power laser constraints, which are the most energy consuming component in an ONoC. Approximate computing considers error resilient applications and trades energy consumption with output quality degradation. We propose a distance aware approximate nanophotonic interconnect to improve power consumption of on-chip communications. Our proposal classifies the communications by the distance between the source and destination and its data type for approximate communication. For this latter, we propose to combine the use of approximation and truncation of data. Approximation allows to lower the laser power level, hence to save energy, by targeting a given quality of transmission, accordingly with the application output error tolerance; while truncation is the most efficient way to save power as bits are not sent, hence some lasers are kept off, but at the cost of more errors at the application level. Regarding distances of communication, only two cases are considered: short and long-distance ranges. Based on this classification, laser power levels are adapted according to the application's error tolerance. The proposed solution is a straightforward method to enhance the power efficiency of ONoC, as it only requires three power levels at each source. Our contribution is validated through simulation with SNIPER manycore simulator, which carried out Approx-bench benchmark applications running on shared-memory architectures. The result of evaluation for the proposal shows drastic laser power reduction of 53% for Streamcluster application at the cost of less than 8% of errors. Finally, we show that our solution is scalable and outperform state of the art solution as follows: 10% reduction in the total energy consumption, 35x reduction in the laser driver size and 10x reduction in the laser controller size.

This work is detailed in Chapter 2 Section 3, and related work disseminations are: [IJ1], [IC5], [IT1], [IT2], [IT3].

Other contributions: Other works on emerging on-chip interconnects have been done. However, for the sake of conciseness, they are not discussed but summarized in what follows.

- The use of Error Correcting Code (ECC) has been explored to enhance the energy efficiency of silicon photonics links [IC8]. We show that using simple Hamming coder and decoder permits to reduce the laser power by nearly 50% with a negligible hardware overhead. We also proposed an Optical Network Interface (ONI) allowing to turn on or off the use of the ECC. Indeed, using ECC reduces the effective bandwidth, hence the ONI can select to save power or to reduce transmission time regarding the application needs.

- Phase Change Material (PCM) has been explored in [IC1] to configure optical paths between writers and readers. It allows to bypass unused readers, thus reducing losses and calibration requirements needed by micro rings. We evaluate the efficiency of the proposed PCM-based interconnects using system level simulations carried out with a manycore simulator. Configurability property of PCM allows to partition the architecture for parallel execution of applications which leads to reduction in execution time compared with sequential execution of applications on all 16 clusters. Results show that up to 52% static power reduction of the interconnect and 42% execution time reduction can be reached.

- Wireless Network-on-Chip (WiNoC) is a promising solution to overcome multi-hop latency and high power consumption of modern many/multi core System-on-Chip (SoC). It is intrinsically broadcast efficient due to the use of antenna to communicate. In [IC6], we proposed a low-power, high-speed, multi-carrier reconfigurable transceiver based on Frequency Division Multiplexing (FDM) to ensure data transfer in future Wireless NoCs. The proposed transceiver supports a medium access control method to sustain unicast, broadcast and multicast communication patterns, providing dynamic data exchange among wireless nodes. Designed using a 28-nm FDSOI technology, the transceiver only consumes 2.37 mW and 4.82 mW in unicast/broadcast and multicast modes, respectively, with an area footprint of 0.0138 mm.

1.2.2 Fault- and Error-tolerant architectures

Approaching the limit of CMOS scaling makes devices becomes increasingly unlikely to be fully functional due to various sources of faults [Srinivasan *et al.* 2004], especially in harsh environment such as in space [Sec 2016]. This sensitivity is further increased with Dynamic voltage and Frequency Scaling (DVFS) that became the prominent way to reduce the energy consumption in digital systems. Indeed, effect of scaling coupled with variability issues make highly pipelined systems more vulnerable to timing errors [Stott *et al.* 2013]. This call to provide fault tolerant techniques to enhance robustness or to limit fault impacts on error resilient applications, e.g. neural networks or approximate computing [Torres-Huitzil & Girau 2017]. Besides this technology scaling, approximate computing bring the possibility to mitigate fault instead of correcting them, offering new opportunities to make circuits more reliable. The main challenges addressed in my research works are i) fault tolerant on-chip interconnect and ii) timing error detection for Dynamic voltage and Frequency Scaling (DVFS) architectures.

Fault tolerant NoC: Despite emerging technologies are explored to provide new on-chip interconnects, electrical NoCs are still good candidates for medium sized architectures. As backbone of multi- and many-cores, fault tolerance is a critical issue for NoCs where faults may lead to system failure. Existing fault tolerant approaches cannot efficiently deal with several permanent faults. To address these limitations, we proposed a mitigation technique to reduce the fault impact on data errors. This is achieved by a bit reorganization (called bit shuffling) within flits crossing a faulty router. This method is suitable for error-tolerant applications as it ensures the protection of Most Significant Bits (MSBs), by transferring the impact of the permanent faults to the Least Significant Bits (LSBs), keeping the MSBs fault-free. We evaluate the efficiency of the technique both at system level with several error resilient applications and at hardware level with synthesis on 28nm FDSOI technology.

This work is detailed in Chapter 3 Section 2, and related work disseminations are: [IJ2], [IC2], [IC3], [NC1], [NC2].

These works are in continuity with those of my PhD thesis entitled "Reliable reconfigurable Network-on-Chips". During my thesis, I proposed techniques to protect data packets from Single Event Upset (SEU), commonly called bit-flips, due to a particle strike, and to determine the location of the fault and whether a fault is temporary or permanent. The technique relies on the use of a Hamming ECC with decoders located in each input and output port of routers. Hence, a data packet is controlled when crossing a bus or a router allowing to disconnect only the faulty hardware path instead of a whole router in case of a permanent fault. For this latter, the use of handshaking and packet-retransmissions between routers, along with an error detection event table, allow to determine a permanent fault when consecutive same errors are detected and reach a threshold. I also proposed during my thesis a technique to detect and locate faulty routing logic for adaptive routing algorithm. It is based on controlling the routing decision of the previous crossed router. This control is done with a specific header that includes the addresses of the two previous routers crossed, and with specific diagonal flag connections to control is a routing bypass is normal or an error. The works published during my thesis are: [IJ4], [IJ5], [IJ6], [IC15], [IC16], [IC17], [IC18], [IC19], [IC20], [IC22], [IC23], [IC24].

Timing errors in DVFS architectures: Voltage scaling is used as a prominent technique to improve energy efficiency in digital systems since scaling down supply voltage effects in quadratic reduction in energy consumption of the system. Reducing supply voltage induces timing errors in the system that are corrected through additional error detection and correction circuits. This limit the adoption of Near Threshold Computing, where any process or temperature variation will lead to timing error, or to aggressive overclocking to improve the computing performance of an architecture. We addressed this problem by exploring two approaches. The first is to use voltage over-scaling for approximate operators targeting applications that can tolerate errors. We characterized the basic arithmetic operators using different operating triads (combination of supply voltage, body-biasing scheme and clock frequency) to evaluate the energy efficiency achievable. Moreover, this method allowed us to construct models for approximate operators. Error-resilient applications can be mapped with the generated approximate operator models

to achieve optimum trade-off between energy efficiency and error margin. On the other hand, we proposed a technique to determine the amount of timing errors or the slack available to automatically over- or under-clock the system. Based on the dynamic speculation technique, best possible operating triad is chosen at runtime based on the user definable error tolerance margin of the application. In our experiments in 28nm FDSOI, we achieved maximum energy efficiency of 89% for basic operators like 8-bit and 16-bit adders at the cost of 20% Bit Error Rate (ratio of faulty bits over total bits) by operating them in near-threshold regime.

This work is detailed in Chapter 3 Section 3, and related publications are: [IC10], [IC11].

1.2.3 Document organization

This document is built on three main chapters. Chapter 1, introduces the general context and presents a synthesis of my research works, along with my Curriculum Vitae. Chapter 2 is dedicated to my research activities on improving the energy efficiency of on-chip interconnects based on silicon photonics. Chapter 3 presents my works regarding fault- and error-tolerant architectures. Finally, Chapter 4 concludes this manuscript with perspectives.

2 Summary of research and teaching activities

2.1 Curriculum Vitae

2.1.1 Academic qualifications

2012 **PhD, in Electronic System**
University of Lorraine, LICM Laboratory, France
"Reliable reconfigurable Network-on-Chips"
Supervised by Dr. Tanougast and Prof. Monteiro

2009 **Master, in Radio-communication and Embedded Systems**
University of Metz, France
Valedictorian

2007 **Bachelor of Sciences in Electronic, Electrical Engineering, Automatic**
University of Nancy, France

2006 **Advanced technician's certificate in electronic systems** (*'BTS in french'*)
Condorcet High School, Schoeneck, France
Academy valedictorian, congratulations of the jury

2.1.2 Academic positions held

Since 2013 **Tenured associate professor**
University of Rennes 1, IUT Lannion
Taran project-team (formerly Cairn), IRISA, Inria

2018-2019 **Research scientist**
Inria
Competitive program

2012-2013 **Research Fellow**
LCOMS Laboratory, University of Lorraine

BsC Teacher
Computer science department, University of Lorraine

2009-2012 **PhD student**
LICM Laboratory, University of Lorraine

2.2 Scientific supervisions

2.2.1 Current PhD students

- **Ibrahim Krayem** - started October 2020
  - Title: Analytical modeling of Network-on-Chip based emerging technologies for fast manycore explorations
  - Co-supervised at 70% with Prof. D. Chillet

- **Jaechul Lee** - started December 2018

  - Title: Approximate computing techniques to improve energy efficiency in Optical Networks-on-Chips
  - Co-supervised at 50% with Prof. D. Chillet

2.2.2 Former PhD students

- **Romain Mercier** - Started October 2018 - December 2021
  - Title: Fault-tolerant networks-on-chip for deep learning algorithms
  - Co-supervised at 40% with Prof. D. Chillet and Dr. A. Kritikakou
  - Current position: postdoctoral researcher, Inria, Lannion, France

- **Van-Dung Pham** - Started December 2014, defended December 2018
  - Title: Design space exploration in the context of 3D integration of multiprocessors interconnected by Optical Network-on-Chip
  - Co-supervised at 30% with Prof. O. Sentieys, Prof. D. Chillet and Dr. S. Le Beux
  - Current position: research engineer at Xlim, Limoges, France

- **Jiating Luo** - Started November 2014, defended July 2018 (incl. maternity leave)
  - Title: Energy-performance trade-offs in Optical Network-on-Chips
  - Co-supervised at 50% with Prof. D. Chillet
  - Current position: research engineer at Huawei, China

- **Rengarajan Ragavan** - Started November 2013, defended September 2017
  - Title: Error Handling and Energy Estimation Framework For Error Resilient Near-Threshold Computing
  - Co-supervised at 70% with Prof. O. Sentieys
  - Current position: research engineer at Qualcomm, India

2.2.3 Post Doctoral Researcher supervisions

- **Abhijit Das** - December 2021 to November 2023
  - Subject: Cache coherent protocol based on emerging technologies
  - Co-supervised at 70% with Prof. O. Sentieys

- **Romain Mercier** - January 2022 to January 2023

  - Subject: Improving fault tolerance in ONoCs
  - Co-supervised at 70% with Prof. D. Chillet

- **Yash Aggrawal** - January to December 2021
  - Subject: Exploration of graphene interconnects for network-on-chips
  - Supervised at 100%
  - Currently assistant professor in Dhirubhai Ambani institute of information and communication technology, India

- **Joel Ortiz** - June 2020 to December 2021
  - Subject: Efficient routing algorithm for multicast in a hybrid electric-wireless NoC
  - Co-supervised at 80% with Prof. O. Sentieys
  - Currently research engineer at Tekalis, Lannion, France

- **Ashraf El Antably** - May 2016 to September 2017
  - Subject: Large scale simulation of multiprocessor system on chip with nanophotonic interconnects
  - Co-supervised at 35% with Prof. O. Sentieys, and Prof. D. Chillet
  - Currently research engineer in cybersecurity at Ausy, Toulouse, France

- **Martha Sepulveda** - November 2014 to October 2015
  - Subject: Simulation platform for a multiprocessor system on chip with nanophotonic interconnects
  - Co-supervised at 35% with Prof. O. Sentieys, Prof. D. Chillet and Dr. Le Beux
  - Currently associate professor at TUM, Munich, Germany

2.2.4 Internship supervision

- **Vincent Templier** - June to August 2019
  - Subject: Coding of Python framework for exploration of manycore architectures based ONoC
  - Supervised at 100%

- **Kushagra Aggarwal** - August to December 2018

  - Subject: Multi-levels driver for on-chip laser
  - Co-supervised at 60% with O. Sentieys

- **Jiating Luo** - May to September 2014
  - Subject: Design of a wavelength allocation controller for optical NoC.
  - Co-supervised at 50% with D. Chillet

Type	Name	Year										
		2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023
PhD Students	I. Krayem								*	*	*	*
	J. Lee						*	*	*	*	*	
	R. Mercier						*	*	*	*		
	V. D. Pham		*	*	*	*	*					
	J. Luo		*	*	*	*	*					
	R. Ragavan	*	*	*	*	*						
Postdoctoral Researchers	R. Mercier										*	*
	A. Das									*	*	*
	Y. Aggrawal									*		
	A. El Antably				*	*						
	M. Sepulveda		*	*								
Internships	V. Templier							*				
	K. Aggarwal						*					
	J. Luo		*									

Table 1.1: Overview of supervisions

2.3 Research projects

2.3.1 Project coordinator (PC)

- **French National Research Agency (ANR) young researcher grant "SHNoC"**, 2018-2023
  - Funding: 235k€(including 3 years of PhD student and 2 years PostDoc)
  - Objectives: To propose i) methods to efficiently associate electrical, optical and wireless NoCs, ii) algorithms to guarantee the quality of service (energy and performance e.g. latency, bandwidth) by routing data packets on the appropriate NoC, and iii) to propose fault tolerance techniques for optical and wireless interfaces

- **Scientific challenge grant** University of Rennes 1, 2019
  - Funding: 4k€
  - Objectives: Coding of a python framework for architectural exploration of ONoC

- **Scientific challenge** University of Rennes 1, 2016
  - Funding: 8k€
  - Objectives: hardware design of a 10Gbps electrical-to-optical conversion interface for laser-on-chip communications

2.3.2 Principal Investigator (PI) in National research projects

- **ANR PRC "Rakes"** 2019-2023 - As PI for my laboratory
  - Total funding: 690k€, amount received: 204k€
  - Consortium: TIMA (PC), IRISA/Inria, Lab-STICC
  - Objectives: Accelerate parallel programming with broadcast communications based on a hybrid wireless/wireless network on chip
- **ANR PRCE "Opticall2"** 2018-2022 - As PI for my laboratory
  - Total funding: 735k€, amount received: 192k€
  - Consortium: INL (PC), IRISA/Inria, C2N, CEA-LETI, Kalray
  - Objectives: Design optical communication links that allow communication in many-cores at wavelengths of about 1.3μm effective for broadcast
- **Labex CominLabs 3D Optical Manycore** 2014-2018 - As co-PI for my laboratory
  - Total funding: 607k€, amount received: 150k€
  - Consortium: FOTON (PC), Inria, INL
  - Objectives: study and proposal of interconnection solution in photonics on silicon for the computing cores in a 3D manycore architecture

2.4 Services

2.4.1 Reviewing activities

International journals

- ACM Transactions on Embedded Computing Systems (**ACM TECS**) 2021
- IEEE Trans. on Computer-Aided Design of Int. Circuits and Syst. (**IEEE TCAD**) 2020
- ACM Journal on Emerging Tech. in Computing Systems (**ACM JETC**) 2017, 2020
- Nano Communication Networks (**NanoCom**) 2020
- IEEE Transactions on Very Large-Scale Integration (**IEEE TVLSI**) 2016, 2017, 2019
- IEEE Transactions on Multi-Scale Computing Systems (**IEEE TMSCS**) 2017
- IEEE Embedded System Letter (**IEEE ESL**) 2014

- Electronics and Telecommunications Research Institute (**ETRI**) 2016
- Springer Journal of Design Automation for Embedded Systems (**DAES**) 2018
- SAGE International Journal of Distributed Sensor Networks (**IJDSN**) 2018

International conferences

- International Symposium on Networks-on-Chip (**NOCS**) 2020, 2021
- Design, Automation & Test in Europe Conference & Exhibition (**DATE**) '17, '18, '19
- Conference on Field Programmable Logic and Applications (**FPL**) 2016, 2018
- Conference on Design & Architectures for Signal & Image Processing (**DASIP**) 2016
- International Symposium on Computer Architecture (**ISCA**) 2016
- International Conference on New Circuits and Systems (**NEWCAS**) 2016, 2017
- ACM Symposium on Integrated Circuits and System Design (**SBCCI**) 2016

PhD Thesis

- **Rashmit Patel** - September 2021
  - Dhirubhai Ambani institute of information and communication technology, India
  - Title: Design and Simulation of Single Electron Transistor Based High-Performance Computing System at Room Temperature

Annual PhD reviewer - Defense to continue the PhD

- 4 PhD thesis in Foton, Lannion, France (2020-2023)
- 1 PhD thesis in CEA-LIST, Saclay, France (2019-2022)

2.4.2 International conference program committees

- **International Symposium on Networks-on-Chips (NOCS) 2021**, Virtual conference, Technical Program Committee (TPC)
- **International Symposium on Applied Reconfigurable Computing (ARC) 2021**, Virtual conference, local arrangement
- **NOCS 2020**, Virtual conference, TPC
- **International Conference on Embedded Systems in Telecommunications and Instrumentation (ICESTI) 2019**, Annaba, Algeria, TPC
- **International Conference on Design of Circuits and Integrated Systems (DCIS) 2018**, Lyon, France, TPC
- **ICESTI 2016**, Annaba, Algerie, TPC

2.4.3 Other services

- Organization of the thematic day "Emerging Interconnect Technologies in ManyCore architectures" of the GDR SoC2. Paris, 27 November 2017, Website

- Elected member of the Research Commission of the IUT of Lannion, 2013-2025

2.5 Teaching activities

2.5.1 Taught courses

My teaching activities at IUT of Lannion since 2013 are summarized in Table 1.2. The teaching activities are based on class teaching, composed of lectures (L), tutorials (T), laboratory sessions (S), and on para-pedagogical activities. This latter includes project/internship monitoring (P/I), and administrative responsibilities related to teaching (A). I am responsible of the second year of the DUT degree of Physical Measurement since 2015, except for the 2019-2020 period as I was on Inria mobility.

Academic year	**Teaching hours**				**Para-pedagogical hours**		**Total number of equivalent hours**
	L	T	S	Total	P/I	A	
2013-2014	10	94	121	225	6	0	231
2014-2015[1]	9	28	108	145	0	0	145
2015-2016	21	53	136	210	10	39	259
2016-2017	21	53	172	246	10	39	295
2017-2018	21	36	163	232	4	39	275
2018-2019	21	36	38	95	0	39	134
2019-2020[2]	25	36	35	96	0	0	96
2020-2021[2]	27	48	141	216	12	39	264
2021-2022	36	46	152	234	10	39	282

[1] 46H of teaching reduction for first year in the University has been applied on my second's year.

[2] 96H of teaching reduction for Inria mobility.

Table 1.2: Summary of the teaching activities.

Table 1.3 provides details on taught courses. I am currently responsible for 4 training modules for which I am in charge of the creation of the contents and evaluations, the organization and the articulation of the sessions, as well as the training of temporary employees such as PhD students or engineers from the laboratory.

2.5.2 Responsibilities

I am involved at different levels of responsibility in the IUT of Lannion. Both administrative and teaching responsibilities are summarized in Table 1.4.

Courses	Period	Level[1]	Type	Equivalent hours
Industrial IT [2]	2021 - ...	B3	L - T - S	15 - 10 - 24
Data acquisition	2020 - ...	B3	S	8
Computer architecture [2]	2019 - ...	B3	L	6
Embedded electronics	2014 - ...	B2	T - S	12 - 23
Instrument control [2]	2013 - ...	B2	L - T - S	9 - 6 - 16
Micro-controller [2]	2016 - ..	B1	L - T - S	12 - 12 - 21
Automation	2015 - 2018	B2	S	9
Signal processing	2013 - 2018	B2	S	36

[1] Bn=nth year of Bachelor level

[2] Courses for which I am fully responsible.

Table 1.3: Summary of taught courses.

Type	Title	**Year** (x = period of responsibility)								
		2013	2014	2015	2016	2017	2108	2019	2020	2021
Administration	Responsible of the second year of the DUT MP			x	x	x	x		x	x
	Elected member of the Institute Council of the IUT					x	x	x	x	x
	Elected member of the Department Council of the DUT MP				x	x	x		x	x
	IUT Lannion's representative of the digital electronics group for the design of the national program for DUT MP								x	x
Courses	Industrial IT									x
	Computer architecture							x	x	x
	Instrument control	x	x	x	x	x	x	x	x	x
	Micro-controller				x	x	x	x	x	x

Table 1.4: Summary of administrative and course responsibilities at IUT of Lannion.

2.6 Publications

2.6.1 Scientific production

Table 1.5 summarizes my scientific publications by year and type since the beginning of my career. Publications on 2010 to 2013 period was done during my PhD thesis. Table 1.6 classify my main publications by domains.

Type	**Year**												**Total**
	2010	2011	2012	2013	2014	2015	2016	2017	2108	2019	2020	2021	
Int. journal	-	-	2	1	-	-	-	-	1	-	-	2	6
Int. conf.	2	4	1	2	-	3	1	3	1	2	2	1	22
Nat. conf.	1	1	-	-	-	-	4	1	-	-	-	2	9
Inv. Talk	-	-	-	-	-	-	1	-	3	2	2	1	9

Table 1.5: Number of publication per year and type.

Ref.	**Domains**					
	Energy efficiency	Fault tolerance	Approximate computing	Emerging technologies	Hardware designs	Optimization methods
[IJ1] 2021	x		x	x	x	x
[IJ2] 2021		x	x		x	x
[IJ3] 2018	x			x		x
[IJ4] 2013		x			x	
[IJ5] 2012		x			x	
[IJ6] 2012		x			x	
[IC1] 2022	x			x		x
[IC2] 2021		x	x		x	x
[IC3] 2020		x	x		x	x
[IC5] 2019	x		x	x		x
[IC6] 2019	x			x	x	
[IC8] 2019	x	x		x	x	
[IC9] 2017	x			x		x
[IC10] 2017	x		x		x	
[IC11] 2016	x	x	x		x	
[IC23] 2010		x			x	
Total	**9**	**9**	**7**	**7**	**12**	**8**

Table 1.6: Classification of main publications by domains

The following sections present in detail the publications. Underlined names are PhD students or Postdoctoral researchers that I supervised.

2.6.2 Referred Journal Publications

[IJ1] J. Lee, **C. Killian**, S. Le Beux, and D. Chillet, "Distance Aware Approximate Nanophotonic Interconnect," ACM Transactions on Design Automation of Electronic Systems (**ACM TODAES**), accepted in August 2021, to appear.

[IJ2] R. Mercier, **C. Killian**, A. Kritikakou, Y. Helen and D. Chillet, "BiSuT: A NoC-Based Bit-Shuffling Technique for Multiple Permanent Faults Mitigation," IEEE Transaction on Computing-Aided Design of Integrated Circuits and Systems (**IEEE TCAD**), July 2021.

[IJ3] J. Luo, **C. Killian**, S. Le Beux, D. Chillet, O. Sentieys, and I. O'Connor, "Offline Optimization of Wavelength Allocation and Laser Power in Nanophotonic Interconnects," ACM Journal on Emerging Technologies in Computing Systems (**ACM JETC**). Vol. 14, no. 2, July 2018.

[IJ4] **C. Killian**, C. Tanougast, F. Monteiro, A. Dandache, "Hybrid Fault Detection for Adaptive NoC", IEEE Embedded Systems Letters (**IEEE ESL**), 4 pages, 2013.

[IJ5] **C. Killian**, C. Tanougast, F. Monteiro, A. Dandache, "Smart Reliable Network-on-Chip", Journal of Electrical and Computer Engineering, Transactions on Very Large Scale Integration Systems (**IEEE TVLSI**), IEEE, 14 pages, 2012.

[IJ6] **C. Killian**, C. Tanougast, F. Monteiro, A. Dandache, "A New Efficient and Reliable Dynamically Reconfigurable Network-on-Chip", Journal of Electrical and Computer Engineering, special issue Design and Automation for Integrated Circuits and Systems (**Hindawi JECE**), Hindawi, 16 pages, 2012.

2.6.3 Referred International Conference Publications

[IC1] P. Zolfaghari, J. Ortiz, C. Killian, S. Le Beux, "Non-Volatile Phase Change Material based Reconfigurable Nanophotonic Interconnect", Design, Automation & Test in Europe Conference & Exhibition (**DATE'22**), Antwerp, Belgium, March 2022.

[IC2] R. Mercier, **C. Killian**, A. Kritikakou, Y. Helen and D. Chillet, " A Region-Based Bit-Shuffling Approach Trading Hardware Cost and Fault Mitigation Efficiency," 2021 IEEE 34th International International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (**DFT'21**), Athens, Greece, 2021.

[IC3] R. Mercier, **C. Killian**, A. Kritikakou, Y. Helen and D. Chillet, "Multiple Permanent Faults Mitigation Through Bit-Shuffling for Network-on-Chip Architecture," 2020 IEEE 38th International Conference on Computer Design (**ICCD'20**), Hartford, CT, USA, 2020, pp. 205-212.

[IC4] J. Ortiz, **C. Killian**, H. Ammar, D. Chillet, "Min/max time limits and energy penalty of communication scheduling in ring-based ONoC", 13th International Workshop on Network on Chip Architectures (**NoCArc'20**), Oct 2020, On-line, France.

[IC5] J. Lee, **C. Killian**, S. Le Beux, and D. Chillet. 2019, "Approximate Nanophotonic Interconnects," International Symposium on Networks-on-Chips (**NOCS '19**), New York, USA. **28% acceptance rate**.

[IC6] J. Ortiz, O. Sentieys, C. Roland, **C. Killian**, "Multi-Carrier Spread-Spectrum Transceiver for WiNoC," International Symposium on Networks-on-Chips (**NOCS '19**), New York, USA. **28% acceptance rate**.

[IC7] J. Luo, V-D. Pham, **C. Killian**, D. Chillet, I. O'Connor, O. Sentieys, S. Le Beux, "Run-Time management of energy-performance trade-off in Optical Network-on-Chip," XXXIII Conference on Design of Circuits and Integrated Systems (**DCIS'18**), Lyon, France, November 2018.

[IC8] **C. Killian**, D. Chillet, S. Le Beux, V-D Pham, O. Sentieys, and I. O'Connor, "Energy and Performance Trade-off in Nanophotonic Interconnects using Coding Techniques," Design Automation Conference 2017 (**DAC '17**), Austin, U.S.A. **22% acceptance rate**.

[IC9] J. Luo, A. Elantaly, V. D. Pham, **C. Killian**, D. Chillet, S. Le Beux, O. Sentieys, I. O'Connor, "Performance and Energy Aware Wavelength Allocation on Ring-Based WDM 3D Optical NoC," Design, Automation & Test in Europe Conference & Exhibition (**DATE'17**), Lausanne, Switzerland, March 2017. **23% acceptance rate**.

[IC10] R. Ragavan, B. Barrois, **C. Killian**, O. Sentieys, "Pushing the Limits of Voltage Over-Scaling for Error-Resilient Applications," Design, Automation & Test in Europe Conference & Exhibition (**DATE'17**), Lausanne, Switzerland, March 2017. **23% acceptance rate**.

[IC11] R. Ragavan, **C. Killian**, S. Sentieys, "Adaptive Overclocking and Error Correction based on Dynamic Speculation Window," IEEE Computer Society Annual Symposium on VLSI (**ISVLSI'16**), pp. 1-6, Pittsburgh, U.S.A., July 2016.

[IC12] J. Luo, **C. Killian**, S. Le Beux, D. Chillet, H. Li, I. O'Connor, O. Sentieys, « Channel allocation protocol for reconfigurable Optical Network-on-Chip », in "SiPhotonics: Exploiting Silicon Photonics for energy-efficient high-performance computing (**SiPhotonics'15**), Amsterdam, Netherlands, January 2015.

[IC13] R. Ragavan, **C. Killian**, O. Sentieys. Low complexity on-chip distributed DC-DC converter for low power WSN nodes, IEEE International NEWCAS Conference (**NEWCAS'15**), Grenoble, France, June 2015.

[IC14] M. J. Sepulveda, S. Le Beux, J. Luo, **C. Killian**, D. Chillet, I. O'Connor, O. Sentieys. Communication Aware Design Method for Optical Network-on-Chip, in International Symposium on Embedded Multicore/Many-core Systems-on-Chip (**MCSoC'15**), Turin, Italy, Politecnico di Torino, Turin, Italy, September 2015.

[IC15] C. Tanougast, **C. Killian**, "Optimization of a Reliable Network on Chip dedicated to partial reconfiguration", in the 2nd International Conference on Control, Decision and Information Technologies (**CoDIT'14**), Metz, France, November 2014.

[IC16] M. Heil, C. Tanougast, **C. Killian**, A. Dandache, « Self-Organized Reliability Suitable for Wireless Networked MPSoC », 25th IEEE International Conference on Microelectronics (**ICM'13**), Beirut, Lebannon, December 2013.

[IC17] M. Boutalbi, C. Tanougast, M. Frihi, **C. Killian**, S. Toumi, A. Chaddad, A. Dandache, « Reliable Router for accurate Online Error Detection in Dynamic Network on Chip », 25th

IEEE International Conference on Microelectronics (**ICM'13**), pp. 1-4, Beirut, Lebannon, December 15-18, 2013.

[IC18] **C. Killian**, M. Boutalbi, M. Frihi, C. Tanougast, A. Dandache, S. Toumi, « A Dependable Low area switch suitable for FPGA based Dynamic Network on chip », International Conference on Embedded Systems in Telecommunications and Instrumentation (**ICESTI'12**), best paper, Annaba, Algeria, November 5-7, 2012.

[IC19] **C. Killian**, C. Tanougast, F. Monteiro, A. Dandache, « Strategic Placement of Reliable Routers for the Optimization of Dependable Dynamic NoC », 18th IEEE International Conference on Electronics, Circuits and Systems (**ICECS'11**), IEEE Circuits and Systems Society, pp 707-710, Beirut, Lebanon, December 2011.

[IC20] **C. Killian**, C. Tanougast, F. Monteiro, A. Dandache, « Loopback Output Router for Reliable Network on Chip », 17th IEEE International On-Line Testing Symposium (**IOLTS'11**), IEEE Computer Society, pp. 208-209, Athens, Greece, July 2011.

[IC21] **C. Killian**, M. Frihi, C. Tanougast, S. Toumi and A. Dandache, « A Dependable and Dynamic Network on chip suitable for FPGA-based Reconfigurable Systems », 6th International Workshop on Reconfigurable Communication-centric Systems-on-Chip (**ReCoSoC'11**), IEEE Circuits and Systems Society, pp. 1-6, Montpellier, France, June 2011.

[IC22] **C. Killian**, C. Tanougast, F. Monteiro, A. Dandache, « A Suitable NoC Switch for On-line Routing Fault Detection », 9th IEEE International NEWCAS Conference (**NEWCAS'11**), IEEE Circuits and Systems Society, pp. 145-148, Bordeaux, France, June 26 - 29, 2011.

[IC23] **C. Killian**, C. Tanougast, F. Monteiro and A. Dandache, « Online Routing Fault Detection for Reconfigurable NoC », 20th International Conference on Field Programmable Logic and Applications (**FPL'10**), IEEE Circuits and Systems Society, pp. 183-186, Milano, Italy, Aug. 2010.

[IC24] **C. Killian**, C. Tanougast, S. Jovanovic, F. Monteiro, C. Diou and A. Dandache, « Modeling and behavioral Co-simulation C-VHDL of Network on Chip on FPGA for Education », Reconfigurable Communication-centric SoCs (**ReCoSoC'10**), Springer, pp. 135-139, Karlsruhe, Germany, May 1 2010.

2.6.4 Invited talks

[IT1] **C. Killian** (speaker), "Tolerating errors in on-chip nanophotonic interconnects for improved energy efficiency", congrès Optique 2021 (**Optique'21**), SFO (Société Français d'Optique), Dijon, juillet 2021.

[IT2] **C. Killian** (speaker), "Tolerating Errors in Nanophotonic Interconnects for a Better Energy Efficiency", in the 6th International Workshop on Optical/Photonic Interconnects for Computing Systems (**OPTICS'20**), co-located with IEEE/ACM Design Automation and Test in Europe (DATE'20), Grenoble, France, mars 2020.

[IT3] **C. Killian** (speaker), "Approximate NoCs: communicating with errors", École d'hiver Francophone sur les Technologies de Conception des Systèmes embarqués Hétérogènes (**FETCH'20**), Montréal, Canada, 12-14 février 2020.

[IT4] **C. Killian** (speaker), "ONoCs: from offline optimization to run time adaptability", in the 5th International Workshop on Optical/Photonic Interconnects for Computing Systems (**OPTICS'19**), co-located with IEEE/ACM Design Automation and Test in Europe (DATE'19), Florence, Italy, mars 2019.

[IT5] **C. Killian** (speaker), "SHNoC: Reliable Multi-Technologies Network-on-Chip", École d'hiver Francophone sur les Technologies de Conception des Systèmes embarqués Hétérogènes (**FETCH'19**), Louvain-la-Neuve, Belgique, 24-26 janvier 2019.

[IT6] **C. Killian** (speaker), "Digital architectures to enhance Optical NoCs efficiency", journée thématique du **GDR SoC2** de la photonique sur silicium pour les architectures de calcul, Lyon, France, novembre 2018.

[IT7] **C. Killian** (speaker), J. Luo, S. Le Beux, D. Chillet, O. Sentieys and I. O'Connor. "Offline optimization of wavelength allocation and laser to deal with Energy-Performance tradeoffs in nanophotonic interconnects", in the 4th International Workshop on Optical/Photonic Interconnects for Computing Systems (**OPTICS'18**), co-located with IEEE/ACM Design Automation and Test in Europe (DATE'18), Dresden, Germany, mars 2018.

[IT8] **C. Killian** (speaker), "Energy-performance tradeoffs in optical Network-on-Chips", École d'hiver Francophone sur les Technologies de Conception des Systèmes embarqués Hétérogènes (**FETCH'18**), Saint-Malo, 24-26 janvier 2018.

[IT9] O. Sentieys (speaker), J. Sepulveda, S. Le Beux, J. Luo, **C. Killian**, D. Chillet, I. O'Connor, H. Li, « Design Space Exploration of Optical Interfaces for Silicon Photonic Interconnects », in the 2th International Workshop on Optical/Photonic Interconnects for Computing Systems (**OPTICS'16**), co-located with IEEE/ACM Design Automation and Test in Europe (DATE'16), Mar 2016, Dresden, Germany, 2016.

2.6.5 Communications, article published in national conference proceedings and workshop

[NC1] R. Mercier, **C. Killian**, A. Kritikakou, Y. Helen and D. Chillet, " BiSu: A Low Cost Bit-Shuffling Technique for Permanent Fault Mitigation in NoC Architecture," **Compas'21**, Lyon, France, 6 – 9 juillet, 2021.

[NC2] R. Mercier, **C. Killian**, A. Kritikakou, Y. Helen and D. Chillet, " BiSu: A NoC-Based Bit-Shuffling Technique For Multiple Permanent Faults Mitigation," **GDR-SoC**^{2} 2021, Rennes, France, 8 – 10 juin, 2021.

[NC3] V-D. Pham, D. Chillet, **C. Killian**, O. Sentieys, S. Le Beux, I. O'Connor, « Interface Electrique/Optique pour un ONoC », **GRETSI'17**-XXVIème colloque, JuanlesPins, France. pp.1-4, Sep 2017.

[NC4] V-D. Pham, **C. Killian**, D. Chillet, S. Le Beux, O. Sentieys, I. O'Connor, « Gestion de la consommation d'un réseau optique intégré dans un MPSoC », **Compas'16**, Lorient, France, 5 – 8 juillet, 2016.

[NC5] J. Luo, D. Chillet, **C. Killian**, S. Le Beux, I. O'Connor, O. Sentieys, "Wavelength spacing optimization to reduce crosstalk in WDM 3D ONoC," **Compas'16**, Lorient, France, 5 – 8 juillet, 2016.

[NC6] V-D. Pham, D. Chillet, **C. Killian**, S. Le Beux, O. Sentieys, I. O'Connor, « Gestion de la consommation ONoC intégré dans un MPSoC », **GDR SoC-SiP**, Nantes, France, 8 – 10 juin, 2016.

[NC7] J. Luo, D. Chillet, **C. Killian**, S. Le Beux, I. O'Connor, O. Sentieys, "Crosstalk noise aware wavelength allocation in WDM 3D ONoC," **GDR SoC-SiP**, Nantes, France, 8 – 10 juin, 2016.

Chapter 2

Improving energy-efficiency of nanophotonic on-chip interconnects

Contents

1 Introduction . . . 21
2 Energy-Performance trade-offs in reconfigurable ONoC . . . 23
2.1 Reconfigurable ONoC for 3D Manycore architectures . . . 23
2.2 Related works . . . 23
2.3 Energy-Performance tradeoffs . . . 24
2.4 Framework for energy-performance exploration . . . 26
2.5 Results . . . 28
2.6 Conclusion and perspectives . . . 31
2.7 Research dissemination . . . 31
3 Distance Aware Approximate Nanophotonic Interconnect . . . 32
3.1 Context of the work . . . 32
3.2 Proposed data and distance aware optical link . . . 32
3.3 Data approximation scheme . . . 34
3.4 Results . . . 37
3.5 Conclusion and perspectives . . . 45
3.6 Research dissemination . . . 47

1 Introduction

Silicon photonic is a promising emerging technology to replace copper interconnects for on-chip communications. Progress in the last decade have given rise to several demonstrators validating this pledge and the interest from the industry [Gunn 2006, Vlasov 2012, Boeuf *et al.* 2013, De Dobbelaere *et al.* 2017, Atabaki *et al.* 2018]. Although integration concerns are not completely solved for a complete monolithic integration of an ONoC [Stojanović *et al.* 2018], improvement in energy efficiency would facilitate the large adoption of this interconnect. Besides technological improvements, the efficiency of photonics links and ONoCs can be improved thanks to optimization techniques and new computing paradigms.

This section addresses the two following contributions:

- In section 2 we propose an off-line methodology allowing to simultaneously explore i) communication bandwidth allocation and ii) laser power levels in Multi Writers Multi Readers (MWMR) ONoCs to provide a trade-off between energy and execution time.

- In section 3 we propose a distance aware approximate nanophotonic interconnect to improve power consumption of on-chip communication at the cost of a controlled application result degradation.

2 Energy-Performance trade-offs in reconfigurable ONoC

2.1 Reconfigurable ONoC for 3D Manycore architectures

This section introduces the basic principles of photonics on silicon by detailing the reconfigurable ONoC architecture we consider for our work. It has to be noticed that our method is compatible with other architectures. We consider the 3D architecture illustrated in Figure 2.1-a: bottom layer implements processing units and memories, while top layer implements the optical interconnect. The connection between the stacked layers can be done with Through Silicon Vias (TSV) or micro-bumps [Thonnart *et al.* 2020, Vivet *et al.* 2021]. ONIs are located on the optical layer and integrate components to perform electrical-optical and optical-electrical conversions. One or several waveguides are also located on this layer. Many ONoC topologies are available in the literature [Werner *et al.* 2017]. This work focuses on ring topology, such as Chameleon ONoC [Le Beux *et al.* 2014], which exhibits low power losses, as no waveguides crossing are required, and provides huge reconfiguration capacities. In ring ONoCs, clock- and counter-clock- wise waveguides are generally considered, as illustrated in Fig. 2.1.

Fig. 2.1-b highlights the key components required for a ring ONoC. This figure depicts the communication between *Core* 1 and *Core* 3, illustrated with a blue-dotted arrow both in Fig. 2.1-a and -b. Each core is associated with a transmitter (Tx) and a receiver (Rx). To communicate, a core serializes its data to correspond to the number of laser used (1 in this example) as this latter performed serial transmissions. Serialized bits are injected to a Modulator Driver which provides the current to control the on-chip laser with respect to an On Off Keying (OOK) modulation. Micro Ring Resonators (MRs) are key components for reconfigurable ONoC. As a functional point of view, it can be tuned to be ON or OFF: ON MR allows to inject/eject optical signal to/from a waveguide to another. OFF MR do not interferes with passing signals. Hence, after the optical signal is generated with the Laser, it is injected to the waveguide with an ON MR. MRs into *Core* 2 are OFF, hence the signal propagates toward the destination Rx located in *Core* 3. The signal is then dropped by the ON MR from the Rx of *Core* 3. A photodetector is used to demodulate the optical signal and a TransImpedace Amplifier is used before de-serialization to obtain the initial data.

These architectures allow to create optical channels by activating a Tx and a Rx and is called MWMR as any core can communicate to any other by using at least only one waveguide that interconnect all the cores. Several channels can be activated at the same time. However Lasers and MRs are designed to work a given wavelength frequency, hence, two channels cannot use the same waveguide segment, as the signals will mix preventing demodulation. To avoid this limitation, and also to improve the throughput of the communications, Wavelength Division Multiplexing (WDM) allows the use of several laser in parallel. Hence, multiple channels can be created but on different wavelengths, while the number of used wavelength, i.e. the throughput, can be adapt to each communication needs. Nonetheless, it increases the channel allocation complexity, and also create inter-channel cross-talk, hence calling for optimization methods.

2.2 Related works

The high-bandwidth available in ONoC is directly related to WDM, which allows propagating multiple signals simultaneously on a same waveguide. In WDM interconnect, Wavelength

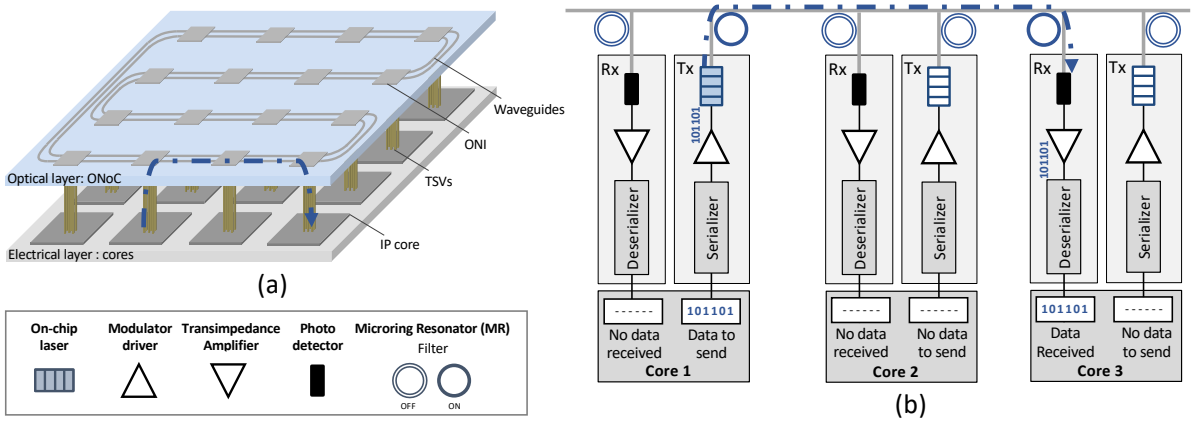


Figure 2.1: (a) Considered 3D architecture and (b) reconfigurable optical link.

Allocation plays an important role since it allows linking application level communications to resource allocation at the hardware level. In [Zang & Jue 2000], dynamic allocation from static allocation is distinguished: in the former case, the communication channels are set-up on demand at run-time while, in the latter case, the communication requests are known in advance, which allows for wavelengths to be allocated using off-line methods. For this purpose, heuristic algorithms have been proposed [Thomas & Bala 1999] and include Random Wavelength Assignment, First-Fit, Most-Used and Least-Used algorithms. Most of these approaches aim to reduce the interconnect contention in order to reduce communication latency.

Different from their off-chip counterparts, on-chip interconnects suffer from very specific challenges related to Wavelength Allocation. For instance, photonics devices such a MRs lead to undesirable mode coupling between adjacent wavelengths in photodetectors, which generates crosstalk noise and impacts ONoC performance [Chittamuru & Pasricha 2015]. Many studies have been carried out to estimate worst-case and average crosstalk noise in various ONoC topologies [Nikdast *et al.* 2015]. Furthermore, ONoC-specific Wavelength Allocation methodologies have been investigated in [Wang *et al.* 2015] [Le Beux *et al.* 2011]. These approaches rely on passive MR: no arbitration is needed to reserve optical paths before data transmission. However, these approaches suffer from a lack of scalability, are application specific and do not consider crosstalk noise in ONoC. A mapping tool has been proposed to improve the Signal to Noise Ratio (SNR) by reducing the number of communications sharing a waveguide [Fusella & Cilardo 2016]. A crosstalk mitigation technique has been developed to to increase channel spacing between adjacent wavelengths in Dense WDM (DWDM) [Chittamuru & Pasricha 2015]. Compared to these works, our methodology aims at configuring the laser output power at different levels according to the communication requirements. This strategy leads to a reduction of overall power needs and can be use together with the aforementioned approaches.

2.3 Energy-Performance tradeoffs

The key concepts investigated in this section are illustrated in Figure 2.2 (the legend is presented in Fig.2.2.g). Applications are represented as a Directed Acyclic Graph (DAG) as depicted in

Fig.2.2.a. In this example, we assume that tasks t_0, t_1, t_2 and t_3 are respectively mapped onto processors p_0, p_1, p_4, and p_8, respectively located in clusters c_0, c_0, c_1 and c_2 (see Fig.2.2.b).

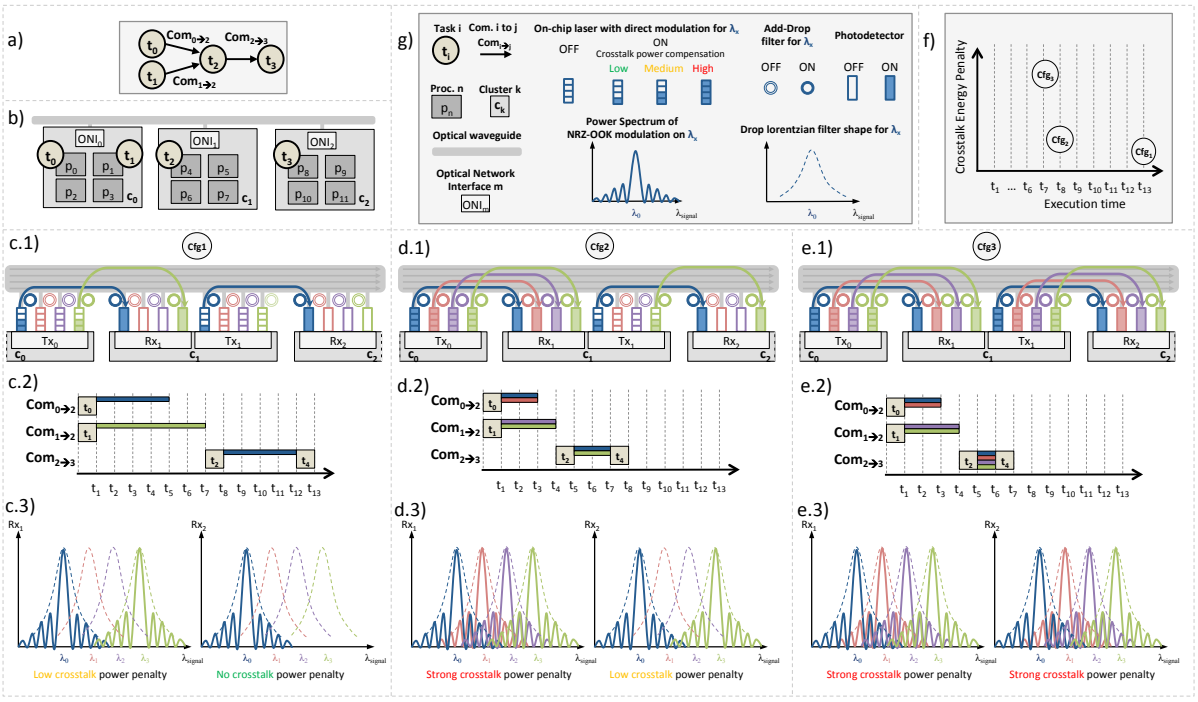


Figure 2.2: Global concept of communication scheduling for Optical Network on Chip.

Communications $Com_{0\to2}$, $Com_{1\to2}$ and $Com_{2\to3}$ are implemented using an ONoC which can be configured according to execution performance and energy requirements. Each cluster is associated with an Optical Network Interface (ONI) containing a Transmitter (Tx) and a Receiver (Rx). The Txs integrate lasers to modulate data, and a MR for each laser to inject the optical signal into the waveguide (if the MR is 'ON'). The Rxs integrate MR to eject the associated optical signal from the waveguide when the MR is 'ON'. Finally, photodetectors allow converting an optical signal into an electrical one. From the simple application given in Fig.2.2-a and if we consider a waveguide supporting 4 wavelengths, several communication configurations are possible and the following illustrates three of them:

- **Low power configuration** $(Cfg1)$: in this configuration, illustrated in Fig.2.2-c.1, a single wavelength is allocated to each communication. Wavelengths λ_0 (blue) and λ_3 (green) in Tx_0 are allocated for $Com_{0\to2}$ and $Com_{1\to2}$, respectively, and λ_0 (blue) is also used in Tx_1 for $Com_{2\to3}$. We assume a direct OOK modulation, i.e. no modulator is needed. The signals propagate along the waveguide until they reach their destination: signals from Tx_0 at wavelength λ_0 and λ_3 are ejected in cluster c_1, while λ_0 signal from Tx_1 is ejected at c_2. Since λ_0 is used between c_0 to c_1 to support $Com_{0\to2}$, it is reused to implement $Com_{2\to3}$ between clusters c_1 and c_2, thus maximizing the wavelength occupation in the waveguide. The Fig.2.2-c.2 illustrates the task and communication schedule. Processor p_4 in cluster c_1 starts executing t_2 once all the data have been received from p_0 and p_1

and sends the processed data to p_8 in c_2. The execution of the DAG in this configuration takes 13 clock cycles. In this example, we assume that the targeted BER is reached for all communications. The Fig.2.2-c.3 depicts the power spectrum of signals received at Rx_1 and Rx_2. The NRZ-OOK modulated signals are represented as cardinal sines, while the filter MRs have Lorentzian filter shapes. For the communications between c_0 and c_1, the two used wavelengths are the most distant, hence the crosstalk between λ_0 and λ_3 is limited, but the associated lasers must increase a bit the emitting power to compensate the signal degradation, named crosstalk power penalty [Bahadori *et al.* 2016b]. Regarding the communication between c_1 and c_2, as only one wavelength is received at Rx_2, no crosstalk penalty has to be compensated. It is worth mention that the power of laser also needs to compensate for propagation losses (not represented here).

- **Intermediate configuration** $\textcircled{Cfg2}$: for this configuration more bandwidth is allocated for each communication. $Com_{0\to2}$, $Com_{1\to2}$ and $Com_{2\to3}$ are respectively supported by the wavelength pairs (λ_0, λ_1), (λ_2, λ_3) and (λ_0, λ_3). This allocation leads to a reduction of execution time in 8 clock cycles (see Fig.2.2-d.2). However, in Rx_1, as all wavelengths are used, the crosstalk penalty is at maximum, hence the source lasers have to compensate a strong crosstalk power penalty. Regarding the Rx_2, as the two used wavelengths are the most distant, the crosstalk power penalty is low.

- **Fastest configuration** $\textcircled{Cfg3}$: for this configuration, the bandwidth utilization is at maximum. Compared to $\textcircled{Cfg2}$, $Com_{2\to3}$ is supported by the 4 available wavelengths. This allocation leads to the fastest execution of the DAG in 7 clock cycles but provides the hugest crosstalk (see Fig.2.2-e.2 and Fig.2.2-e.3) which needs to be compensated in each Tx.

The Fig.2.2.f plots the execution time versus the crosstalk energy penalty for each configuration. The previously presented configurations belong to a Pareto front. The configurations $\textcircled{Cfg1}$ and $\textcircled{Cfg3}$ are respectively the two bounds: the slowest and the fastest execution configurations, which are respectively the lowest and highest energy consuming. Based on the complexity of the application and the architecture (e.g. number of ONIs, waveguides, lasers, etc.) the number of solutions on the Pareto front can be difficult to determine, as well as the front itself. We propose in the following a framework to automatically extract the Pareto front of the communication scheduling.

2.4 Framework for energy-performance exploration

Figure 2.3 illustrates the design flow generating ONoC configurations according to user specifications. The flow takes as input an application mapped onto a 3D architecture. The application is modeled as a DAG characterized by task execution times, amount of data transmitted between tasks, and minimum BER to be reached. The architecture includes an ONoC implemented on top of processing cores and characterized by a topology, a number of wavelengths and waveguides. The ONoC allows for cores to communicate with each other using optical signals, which is achieved using E/O and O/E conversion. The interfaces are crossed by waveguides propagating the optical signals using WDM in both clockwise (C) and counter-clockwise (CC) directions,

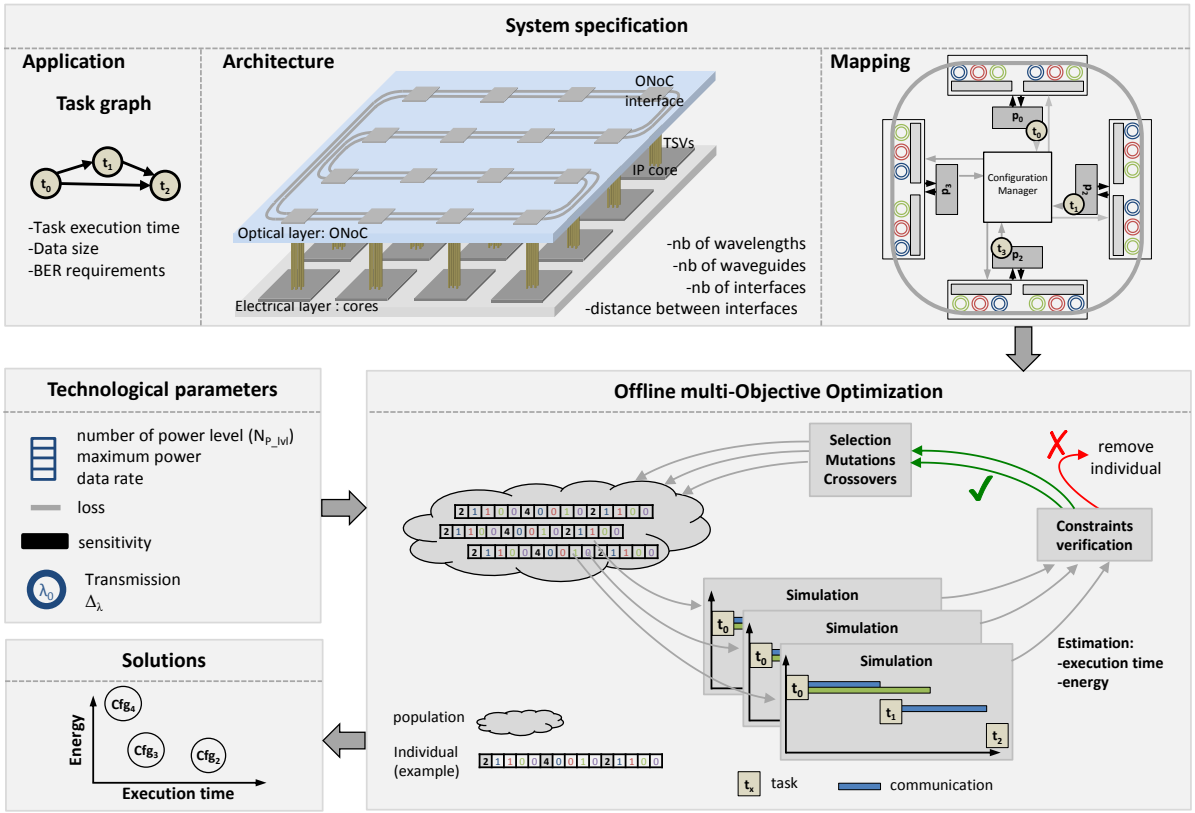


Figure 2.3: Multi-objective design space exploration.

which reduces the maximum communication distance. The mapping of the tasks on the cores gives the communications in the ONoC. The design flow relies on technological parameters since they impact the performance of optical communications. Parameters are, for instance, photodetectors sensitivity, waveguide losses and MR transmission. Regarding the laser, we consider the data-rate, the efficiency, the maximum output power, and the number of power levels available into account.

The aim of the flow is to optimize both power consumption and application execution time. For this purpose, we explore the number of wavelengths allocated to each communication, i.e. the bandwidth and the power level for each laser source. A multi-objective optimization method is implemented using a genetic algorithm, which allows extracting solutions on a Pareto front. In our genetic algorithm, the ONoC configurations (i.e. the individuals) are represented by chromosomes and the genes encode both wavelength allocations and laser power levels. Classical crossover and mutation operators are applied to evolve the initial population. Constraints validate the configurations by ensuring that i) BER requirement is reached and ii) no wavelength is used more than once on a same waveguide segment. For each valid individual, a simulation is carried out to estimate application execution time and total ONoC energy consumption (i.e. the algorithm fitness functions). The resulting ONoC configurations are thus made available to the designer for final selection. Furthermore, since all the configurations are non-dominated on at least one metric, multiple configurations could be embedded in the system in order to be loaded at run-time according to the execution context (e.g. high performance and low power).

2.5 Results

We investigate the scalability of the approach using ONoC architectures with 16, 32 and 64 ONIs, that interconnect 64, 128, and 256 cores respectively. Each ONI is connected to a cluster of four electrically connected cores. We assume two waveguides, eight wavelengths per waveguide, and a number of laser power levels $NP_{lvl} = 5$. For the ONoC with 64 cores the possible electrical laser power values are [2, 4, 6, 8, 10] mW, and for 128 cores the possible electrical laser power values are [3, 6, 9, 12, 15] mW, while for 256 cores the laser power values are [4, 13, 22, 31, 40] mW. The cores are assumed to run at 1Ghz and the laser transmit data at 10Gbps. Regarding the BER estimation, we assume the model detailed from [Xiao *et al.* 2007] and the technological parameters defined in Table 2.1.

Table 2.1: Technological parameters.

Parameter	Value	Ref
Waveguide propagation loss	-0.274 dB/cm	[Dong *et al.* 2010]
Photodetector sensitivity	-20 dBm	[Kennedy & Kodi 2017]
Laser efficiency	15%	[Kennedy & Kodi 2017]
$\Delta\lambda$	0.4 nm	[Bahadori *et al.* 2016a]
FSR	8 nm	[Bahadori *et al.* 2016a]
$-3dB$ MR bandwidth	0.26 nm	[Bahadori *et al.* 2016a]

Regarding the applications, we use a random task graph generator that provides applications including from 52 to 107 tasks and from 80 to 158 communications. The task execution time

values are randomly selected between [100, 1000] clock cycles and the communication volumes are randomly selected within [100, 1000] bytes range. The targeted BER is 10^{-9} and each task is randomly mapped on a dedicated core. As we assume shared memory within a same cluster, no latency is assumed for intra-cluster communications.

Number of Cores	**Graph ID**	**Number of Tasks**	**Number of Comm.**	**Energy (nj)**			**Execution Time (kcc)**		
				Low Power	High Perf.	Var.	Low Power	High Perf.	Var.
64	**TG 1**	55	80	141	187	1.33	41.2	23.8	1.73
	TG 2	52	78	118	194	1.64	37.9	23.4	1.62
	TG 3	57	82	102	120	1.18	39.9	21.7	1.84
	TG 4	60	92	141	195	1.38	37.3	23	1.62
	TG 5	63	93	128	193	1.51	41.2	21.8	1.89
	TG 6	62	92	147	236	1.61	43.2	24.7	1.75
	TG 7	56	87	103	148	1.44	31.7	20	1.57
	TG 8	63	91	112	156	1.39	37.5	23.4	1.60
	Average			**124**	**179**	**1.44**	**38.7**	**22.7**	**1.71**
128	TG 9	107	158	253	389	1.54	62.19	37.5	1.66
	TG 10	94	139	213	314	1.47	64.38	33.6	1.92
256	TG 9	107	158	748.35	1051	1.40	64.38	38.1	1.69
	TG 10	94	139	654.4	1153.4	1.76	61.01	34.58	1.76

Table 2.2: Energy and execution time in kilo clock cycles (kcc) for Low Power and High Performance wavelength allocations strategies.

Tables 2.2 summarizes the characteristics of the task graphs and the optimization results. Since our approach leads to a Pareto front, we only show the solutions with i) the lowest energy consumption (denoted Low Power in the table) and i) the lowest execution time (High Perf.). The solutions offer, on average, 44% energy variation and 71% execution time variation trade-offs for 64 cores architectures. As the results show, the execution time gain increases for larger architectures, while the energy gain decreases. Indeed, for these results, we use random task placement which leads to long communications between tasks, and hence high Laser power to support these communications. This simulation context is the worst as possible, and we can imagine that a smart task placement should reduce these long communications and gives the opportunity to select more often the low Laser power levels. Nonetheless, these results demonstrate the efficiency of flexible wavelengths allocation and laser output power tuning to adapt the nanophotonic interconnects according to application requirements.

Figure 2.4(a) and 2.4(b) give the distribution of laser power levels and allocated bandwidth for the low-power solutions. On average, the 2mW power level is selected in 60% of the cases and is followed by 4mW (15% on average). Higher power levels are selected to compensate for the losses experienced by long range communications. Indeed, as shown in Figure 2.4(b), all the communications are allocated on a single wavelength for a minimum crosstalk effect. Figure 2.4(c) illustrates results for high-performance solutions. On average, 45% and 15% of the lasers are configured to 2mW and 4mW respectively. The lasers configured with a higher output power emit signals experiencing significant losses (i.e. long distance communications) or crosstalk. Figure 2.4(d) shows the number of wavelengths allocated distribution for each task

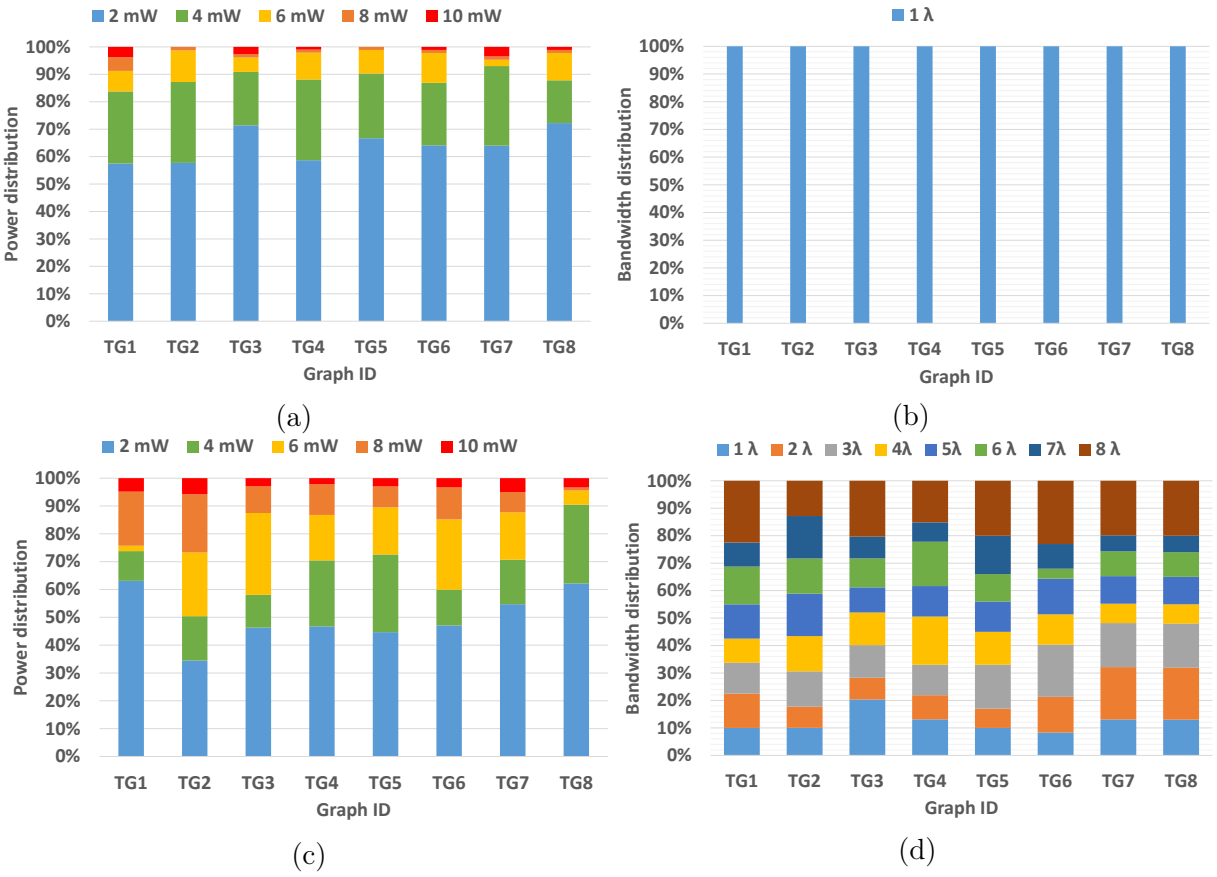


Figure 2.4: Distribution of laser power levels and allocated bandwidth for (a-b) low-power solutions and (c-d) high-performance solutions.

graph for the same solutions. Taking TG8 as an example, 13% of the communications are carried out using a single wavelength, while 20% rely on all the wavelengths. Hence, the resulting ONoC configurations do not simply allocate all the wavelengths for each communication, even though maximum execution performance is reached. Indeed, our algorithm computes the maximum wavelengths to be allocated by considering simultaneous communications and computation time that may not be reduced.

These results demonstrate that the proposed approach to combine laser output power and wavelength allocation reaches the maximum execution performance while saving energy by using only the required optical resources. Furthermore, complex applications lead to solutions with very different energy and execution time figures. This confirms the opportunity induced by our off-line approach to adapt the ONoC configuration at run-time according to the execution context and QoS requirements.

2.6 Conclusion and perspectives

Reconfigurable ONoCs are promising solution for large scale manycore architectures. However, the design space is complex to explore, and the channel allocation clearly impacts the performance of the system as demonstrated in this work. Indeed, we shown there is a trade-off between the low power and the high performance communication mapping solutions provided by our framework. It is worth mentioning that we even do not consider solutions that are not on the Pareto front of solutions and that would lead to un-optimized solutions. The proposed framework also helps to explore the design space as it allows to explore both architectural and technological parameters [IJ3]. Mapping of task will also impact the system performance, as minimizing the communication conflicts may either improve the execution time by allowing more wavelengths per communications, or less energy consumption due to less parallel communications. This topic has not been explored in our work, but techniques are available that are compatible and may be integrated to our framework [Fusella & Cilardo 2016, Wang & Cheng 2019].

We would like to emphasize that the resolution method (i.e. genetic algorithm) is not the contribution of this work but a mean to reach it. We also proposed a mathematical formulation of this energy-performance trade-offs in [IC4] along with the use the language OPL (Optimization Programming Language) and Cplex solver. It results the extraction of the two bounds of the Pareto front. This method allows to reduce the exploration space of channel allocation in order to speed up the results, or to guarantee to obtain the optimal solutions.

Regarding run-time allocations, heuristics a recent work propose online mechanism to allocate the minimum of wavelengths to satisfy the bandwidth for each communication with an online policy [Narayan *et al.* 2021]. Finally, the significant improvements currently made by the machine learning community to embedded algorithm on chip also leads to new directions for online exploration [Musumeci *et al.* 2019].

2.7 Research dissemination

The works presented in this section have been disseminated in [IC1], [IJ3], [IC4], [IC7], [IC9], [IC12], [IC14], [NC5], [NC7], [IT4], [IT6], [IT8], [IT9].

3 Distance Aware Approximate Nanophotonic Interconnect

3.1 Context of the work

As already explained, ONoC, which bases on Nanophotonic interconnects, is a promising solution to overcome bandwidth and latency issues, as optical signals propagate near speed-of-light in waveguides. However, their implementations remain challenging due to the low efficiency of the lasers, which are key devices in such interconnects. Indeed, the laser power consumption is mainly driven by the high signal power required to transmit data at low BER for accurate communications [Alexoudi *et al.* 2019].

Energy efficiency of ONoC is addressed at several aspects. At circuit level, [Hamedani *et al.* 2014] proposes to reduce the power loss driven from components such as reducing the number of MR and waveguide crossings. At data level, the use of error correcting codes has been explored to allow transmission power reduction, leading to transmission errors that are compensated by data redundancies [IC8]. At communications level, communication scheduling can improve the energy efficiency thanks to less parallel communication leading to crosstalk and energy penalty [Zhou & Kodi 2013, Postman *et al.* 2013], [IC4].

Beside these on-chip interconnect evolution, approximate computing is an emerging solution to improve energy efficiency and execution speed of embedded computing systems [Xu *et al.* 2016, Mittal 2016]. It relies on accuracy reduction of the data representations, which allows lowering design constraints and improves performances at the cost of Quality of Result (QoR) degradation measured with output errors [Sampson *et al.* 2015]. It has been deployed in numerous manners on interconnects [IJ1][Sunny *et al.* 2020], operators and memory levels [Mahajan *et al.* 2015, Sampson *et al.* 2011] as well. With various methods of approximate computing implementation, it offers storage efficiency, computation speed and power consumption improvements, while keeping a tolerable degradation at the application side.

In this work, we propose a distance aware approximate nanophotonic interconnect to improve power consumption of on-chip communication.

3.2 Proposed data and distance aware optical link

Our proposal aims to manage, for a given application, power consumption regarding the QoR. For this purpose, the proposed nanophotonic interconnect allows adapting the laser power level according to the communication distance and the transmitted data type. Without lack of generality, Fig. 2.5 illustrates our approach on a Single Writer Multiple Reader (SWMR) link. The writer (source) is connected to several readers (destinations represented by Core in the figure) using waveguides (for sake of clarity, only one waveguide is represented in the figure). Each waveguide allows transmitting $N\lambda$ signals using WDM ($\lambda_0..\lambda_{N\lambda-1}$). The signals are emitted by on-chip lasers and are combined into waveguide using a MultiMode Interference (MMI) couplers [Mandorlo *et al.* 2012]. SWMR only slightly differs from MWMR introduced in the previous section 2.1 Reconfigurable ONoC for 3D Manycore architectures. It simplify channel allocation management as each transmitter possesses a dedicated waveguide to communicate to any connected receiver. It is more efficient for small data packet transmissions such as in cache coherency, which is the case study considered for the results of this work, however it requires more waveguides and receivers.

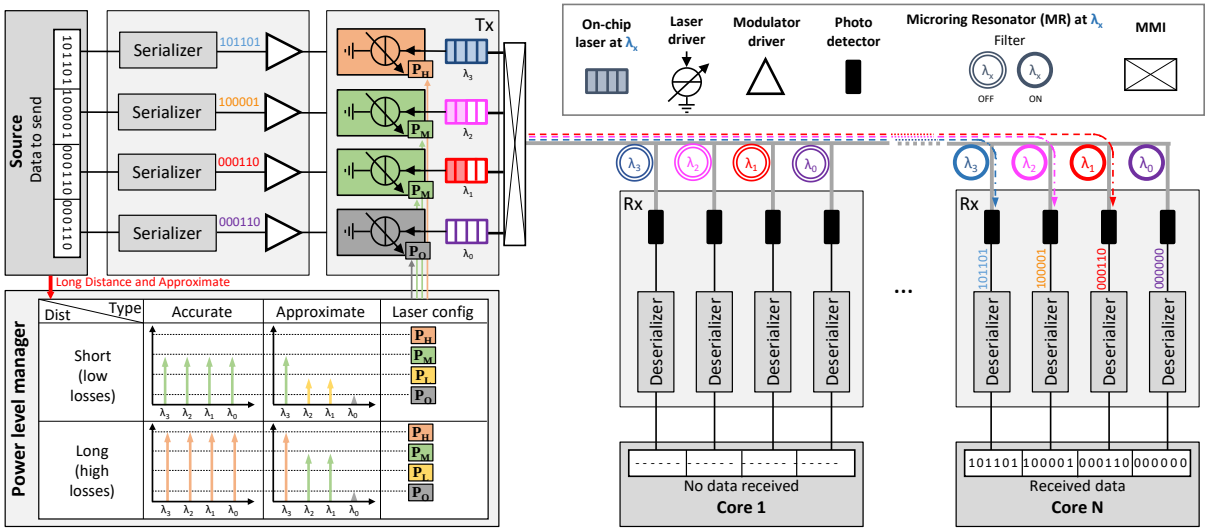


Figure 2.5: Proposed data and distance aware optical Single Writer Multi Reader (SWMR) link.

When a communication is initiated, the writer activates the lasers (optical signals are emitted after few ns typically) and the data modulate the optical signals using OOK modulation. Although we assume that data transmissions involve the use of all the optical signals, the transmissions of data are serialized according to the data bit width (N_{data}) and the number of wavelengths ($N\lambda$). For instance, $N_{data} = 32$ and $N\lambda = 8$ would lead to a Bit Stream Length (BSL) of 4, i.e. 4 bits are sent on each wavelength and 4 clock cycles are needed to transmit each data. The design of the serializer has been investigated in [IC8] [Wang *et al.* 2016] and is not detailed here. The modulated signals propagate along the waveguide until reaching the destination. At this stage, the signals are ejected from the waveguide using MRs in ON state. They are redirected to photodetectors from which opto-electronic conversions are carried out. The serial streams are then de-serialized back into the original data format. SWMR must be associated with a mechanism to activate the correct receiver on the SWMR link, which is not detailed in this present work but has been studied in [Pan *et al.* 2009] which proposes a secondary waveguide dedicated to active the receivers.

In case no communication occurs, the lasers are turned OFF and hence do not consume energy [Neel *et al.* 2015]. In case data transmission is triggered, lasers are turned ON with power levels that depend on i) the required communication quality and ii) the communication distance. To allow a reduction in consumption while limiting the driver complexity, we propose to use four laser power levels: high power (P_H), medium power (P_M) low power (P_L) and off (P_{Off}), with $P_H > P_M > P_L > P_{Off} = 0$. Fig. 2.5 presents an example assuming that the λ_3 is sent with high power, λ_2 and λ_1 are sent with medium power, while the power of λ_0 is equal to P_{Off}, meaning this wavelength is not used, which corresponds to 4 Least Significant Bit (LSB) bits truncation. The laser power values are calculated based on technological parameters (e.g. waveguide propagation loss and MR losses), system level parameters (e.g. number of readers) and the targeted BERs.

We assume that the accurate data (i.e. data that cannot be approximated) and approxi-

mated data require, on the receiver side, BER_{NotAx} and BER_{Ax} respectively. Hence, assuming $\text{BER}_{NotAx} < \text{BER}_{Ax}$, approximated data will be transmitted using lower laser power compared to accurate data. As previously explained, the actual laser power depends on the propagation loss and hence the communication distance. For this purpose, we define two ranges of communication distance, *Short* and *Long*, depending on the position of the destination on the SWMR link. This leads to the four scenarios detailed in the following:

- **Accurate & Long** is used for the transmission of sensitive data (e.g. instructions and integer data) that critically impact the execution of applications. Therefore, the transmission quality is protected by targeting low BER for all the bits (e.g. $\text{BER}_{NotAx} = 10^{-12}$). To compensate the losses induced by the long-range transmission, the highest power level is used (i.e. P_H) for all the lasers.

- **Accurate & Short** is also used to transmit sensitive data but at a shorter distance. Due to the lower losses, low BER is obtained for all the bits by configuring the lasers power level to P_M.

- **Approximate & Long** allows the long range transmission of data that can be approximated. Depending on the approximation scheme, the lasers are configured with different power levels. As detailed in Section 3.3, most significant bits are transmitted with low BER, intermediate bits are transmitted with high BER and least significant bits are truncated. To achieve this, the lasers that emit signals transmitting most significant bits and approximated bits are configured to P_H and P_M respectively. Truncated bits are not transmitted and the corresponding lasers are thus set to P_{Off}.

- **Approximate & Short** allows for short distance transmission of data that can be approximated. As for the previous scenario, the configuration of the lasers depends on the bit's significance. However, the lower losses allow reducing the power of the emitted signals, which lead to use the P_M and P_L power levels for most significant and intermediate bits respectively.

By adapting the laser powers according to both communication distance and level of approximation, our interconnect aims at minimizing the energy to transmit data bits. Since only three power levels are needed for all destinations with approximation levels, the design complexity of the laser driver remains comparable to solutions which require individual computation of power levels for each destination (*Distance Proportional*).

3.3 Data approximation scheme

The proposed communication method involves both approximation and truncation on data. In order to control different approximate level, we propose configurable bit areas according to the bit significance. Hence, several configurations are available to send data that can be approximated, and that can be tuned to achieve a given quality of result for an application.

Fig 2.6 illustrates how the technique can be used for the IEEE 754-2008 single-precision Floating-Point (FP) format and for long distance communication. Bits are grouped into 3 different areas according to their significance on the floating-point data representation:

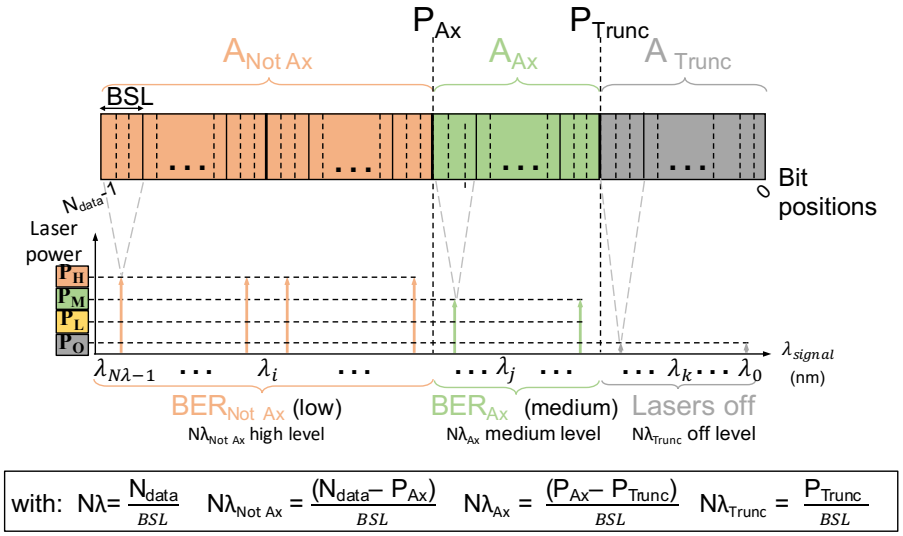


Figure 2.6: Example of data approximation scheme.

1. $A_{NotAx} = [N_{data} - 1\ ;\ P_{Ax}]$ corresponds to Most Significant Bits (MSBs), composed with the sign and the exponent of the FP number, which are always protected from approximation thanks to a transmission power allowing to reach BER_{NotAx};

2. $A_{Ax} = [P_{Ax} - 1\ ;\ P_{Trunc}]$ are intermediate bits, the highest part of the mantissa, which may tolerate approximation and are transmitted in order to reach BER_{Ax};

3. $A_{Trunc} = [P_{Trunc} - 1\ ;\ 0]$ are the LSBs, lowest par of mantissa, and are not transmitted.

These 3 groups provide flexibility to adapt the robustness of the transmitted bits with respect to the bit significance. Table 2.3 summarizes the parameters used to identify and configure the different groups of bits of a data with the proposed method. While the flexibility can be controlled through the BER to reach and by selecting the bits areas to approximate or truncate (defined by A_{Ax} and A_{Trunc} on Fig. 2.6), in this study we propose to fix these parameters based on the analysis of a set of applications. This analysis is presented in the next section.

Fig. 2.7 illustrates six possible communication scenarios to save energy for the transmission of 32 bits data. For this scenario, 8 wavelengths (λ_7 to λ_0) are used to simultaneously transmit 8 groups of 4 serialized bits. Each scenario is noted as xNA/yA/zT with: x Not Approximated, y Approximated, z Truncated bits. Regarding long distance communication (Fig. 2.7-a), we first illustrate a scenario where only approximation is used, assuming $A_{NotAx} = 12$ and $A_{Ax} = 20$. This scenario is noted as $12NA/20A/0T$. This leads to lasers at λ_7 to λ_5 and λ_4 to λ_0 configured with P_H and P_M respectively. Second scenario, $12NA/0A/20T$, involves only truncation ([Sunny *et al.* 2020]) which refers to $A_{NotAx} = 12$ and $A_{Trunc} = 20$ and it leads to laser configuration P_H for λ_7 to λ_5 and P_{Off} for lasers at λ_4 to λ_0. Third scenario, $12NA/8A/12T$, is a combination of approximation and truncation with $A_{NotAx} = 12$, $A_{Ax} = 8$, and $A_{Trunc} = 12$ which leads to P_H, P_M and P_{Off} for lasers at λ_7 to λ_5, λ_4 to λ_3 and λ_2 to λ_0 respectively. Moreover, when all of these 3 scenarios are applied in short-range communication, the laser power

Parameters	Definition
N_{data}	Data size
P_{Ax}	Position of the higher approximated bit
P_{Trunc}	Position of the higher truncated bit
A_{NotAx}	Area of accurate bits
A_{Ax}	Area of approximated bits
A_{Trunc}	Area of truncated bits
$N\lambda$	Number of wavelengths (nb of lasers)
BSL	Bit Stream Length (BSL = $N_{data}/N\lambda$)
BER_{NotAx}	BER for non approximated bits
BER_{Ax}	BER for approximated bits

Table 2.3: Summary of parameters used for the proposed method.

levels are reduced accordingly ($P_H \rightarrow P_M$ and $P_M \rightarrow P_L$) (as presented in the three scenarios of Fig. 2.7-b).

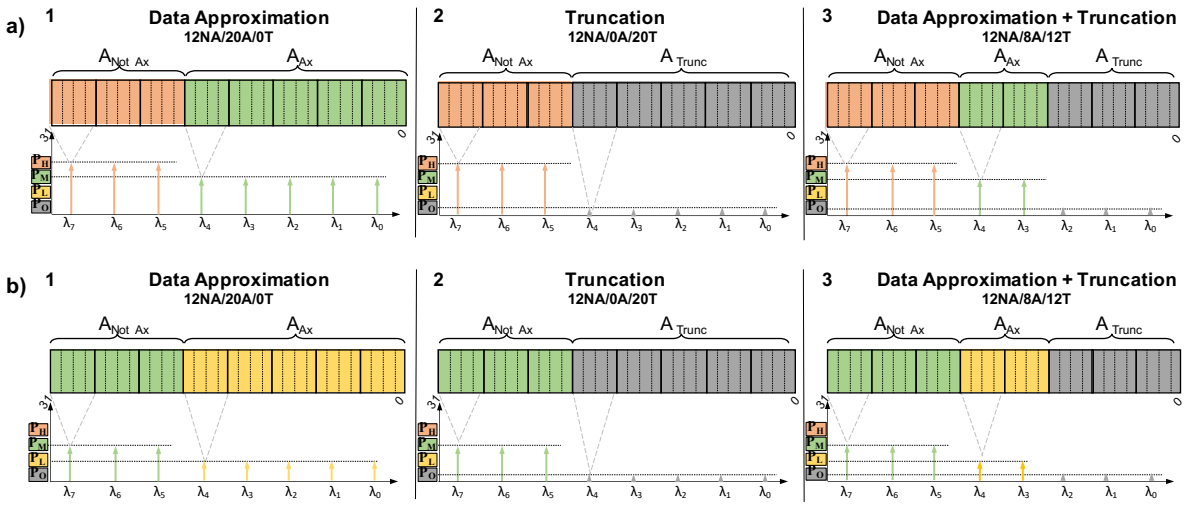


Figure 2.7: Laser power level configurations to transmit approximated floating-point data assuming a) Long distance and b) Short distance communications. $N_{data} = 32$, $N\lambda = 8$ and BSL $= 4$. xNA/yA/zT with x Not Approximated, y Approximated, z Truncated bits

Compared to the scenario where all bits are always transmitted at full power, regardless of the communication distance, our approach allows saving power, as detailed in Equation 2.1.

$$PowerGain = \frac{P}{N\lambda \times P_H} \tag{2.1a}$$

(2.1b)

$$\text{with } P = \frac{1}{BSL}\Big[(N_{data} - P_{Ax}) \times P_1 + (P_{Ax} - P_{Trunc}) \times P_2 + P_{Trunc} \times P_{Off}\Big]$$

$$\text{and with } (P_1, P_2) = \begin{cases} (P_H, P_M) \text{ for long distance,} \\ (P_M, P_L) \text{ for short distance.} \end{cases}$$

To calculate power gain, power consumption from proposals P over Baseline scenario $\lambda \times P_H$ has to be computed. For baseline laser power is configured as P_H for all the wavelengths to guarantee accurate communication for all the distances, equation 2.1a. Moreover, in equation 2.1b, to obtain power consumption from proposal, power consumption on three bit areas (defined in section 3.3) have to be computed. First, MSBs of FP $[N_{data} - P_{Ax}]$ is configured with P_H or P_M level according to the Short and Long distance. Secondly, approximate bit area $[P_{Ax} - P_{Trunc}]$ is configured with P_M or P_L to target approximate communication. Last, for the lasers that are in charge of Truncation bit area $[P_{Trunc} - 0]$ are set in off state P_{Off}. The summation of all the power consumption from those bit areas defined the total power consumption for the proposal. The actual power saving depends of bit area configuration, targeted BER and distance configuration.

The parameter values used in equation 2.1 are computed from equation 2.2 that considers the required power at the receiver and the propagation losses as follows:

$$P_s = P_d + (Hop_{s,d} - 1) \times Loss_{mr} \times N\lambda + Hop_{s,d} \times Loss_h + Loss_d + Loss_{Xtalk} \quad (2.2)$$

Where P_s is the optical power injected from source core (in dBm) and P_d is the optical power received at the destinations core (dBm). P_d depends on the targeted BER at the destination photodetector [Radi *et al.* 2021]. The BER strongly depends on the attenuation experienced by the propagating optical signals, which depend on : i) the hop count $Hop_{s,d}$, ii) the number of wavelengths (or lasers) $N\lambda$, iii) the MR through loss ($Loss_{mr}$, in dB), iv) the waveguide propagation loss per Hop ($Loss_h$, in dB/hop), v) the MR drop loss ($Loss_d$, in dB), and vi) the crosstalk loss ($Loss_{Xtalk}$, in dB). For this latter we consider the crosstalk model from [Bahadori *et al.* 2016b].

3.4 Results

3.4.1 Experimental setup

Considered 3D architecture: for experimental setup, we consider an architecture based on two layers: one electrical layer to support the computation part, and one optical layer for the optical communications (as presented on Fig. 2.8-a). The chip size considered for this section is equal to $1600mm^2$, hence the distance between two consecutive ONIs is therefore equal to $10\,mm$. As summarized in Table 2.4, the electrical layer is composed of 16 clusters of 2 by 2 cores. Ws consider a distributed share memory architecture. Each cluster integrates a last level cache (L3) shared with the other clusters. Each core has its own private L1 data (L1d), instructions (L1i) caches and L2 cache. Furthermore, a MESI protocol ensures coherency between the distributed caches. All these memory organization parameters are listed in Table 2.4. Every clusters are

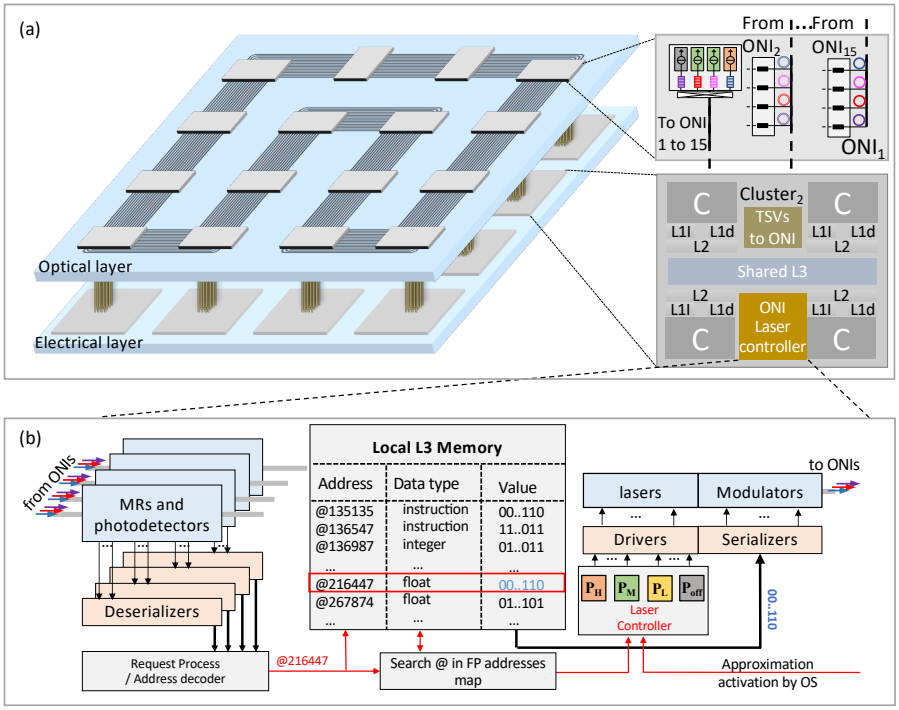


Figure 2.8: a) Hardware Architecture (16 clusters of 4 cores) b) SWMR channel management for approximate communication

connected with ONI through TSVs. The ONI blocks are located in optical layer and have 1 transmitting and 15 receiving waveguides which are featured with SWMR and WDM (as depicted in Fig. 2.5). Each waveguide transmits 8 optical signals on different wavelengths, hence each having a 4 bits stream data (i.e. BSL $= 4$ and $N\lambda = 8$).

Approximation under Cache Coherence Traffic: cache coherency involves traffic between cache memory and last level of cache that is mainly initiated by cache misses or write-back accesses [Solihin 2015]. However, the traffic initiated by cache coherence protocols also involves addresses of the data to be transmitted and is thus data type agnostic. We assume that approximation are enabled by the OS based on user requirements. This is achieved by setting or resetting flag "Approximation activation by OS" at run-time (figure 2.8). In case approximations are disabled, all communications are carried without errors using P_H and P_M for long-range and short-range communications respectively. Otherwise, the lasers are configured as follows. First, we assume that i) accurate data and approximate data are located in different address spaces of the L3 memory and ii) ranges of addresses associated to a given data type are stored in local tables. Then, from the packet destination ID, the ONI evaluates whether the communication distance is short or long. Both information (accurate/approximate and short/long) allow the ONI manager to configure the laser power levels from the laser configuration table. The details of communication between OS and ONIs are studied in [IJ3]. By considering all these parameters, laser controller configures laser powers at the different levels (P_H, P_M, P_L and, P_{Off}).

Architecture		
	# clusters	$16\,(4*4)$
	# cores per cluster	4
	ONI distance	$10\,mm$
	L1 I / D caches	$64\,KB$ each private
	L2 cache	$512\,KB$ private
	L3 cache	$4\,M$ shared (in a cluster)
	Cache protocol	MESI
Technological parameters		
$Loss_{mr}$	MR through loss	$0.02\,dB$ [Xia *et al.* 2007]
$Loss_{d}$	MR drop loss	$0.7\,dB$ [Xia *et al.* 2007]
Q	MR Quality Factor	$20,000$ [Bahadori *et al.* 2016c]
$Loss_{h}$	Waveguide loss per hop	$0.25\,dB/cm$ [Dong *et al.* 2010]
P_{d}	Optical power	$-8\,dBm$ at 10^{-12} BER
	received by photodetector	$-12\,dBm$ at 10^{-3} BER
FSR	Full Scale Range	$8nm$ [Bahadori *et al.* 2016c]
$Laser_{e}ff$	Laser efficiency	0.33 [Chen *et al.* 2021]
B_{R}	Bit-rate	10Gb/s
ONI parameters		
BER_{NotAx}	BER without approximation	10^{-12}
BER_{Ax}	BER with approximation	10^{-3}
P_{H}	High power	$707\,\mu W$
P_{M}	Medium power	$281\,\mu W$
P_{L}	Low power	$112\,\mu W$
$N\lambda$	Number of wavelengths	8
Short distance	Nb of hops from	1 to 5 hops
Long distance	source to destination	6 to 15 hops

Table 2.4: Summary of considered parameters for evaluation

Simulation platform: for evaluation, we consider error tolerant applications from Approxbench [Sampson *et al.* 2015]. Table 2.5 presents the four benchmark applications considered for the results: Blackscholes, Fluidanimate, Canneal and Streamcluster performing financial algorithms, animation, routing and online clustering, respectively. Moreover, Approxbench provides the output error metric for each application, which allows to estimate the QoR.

This output error is the difference on error metric between the application with and without approximate communications. As each application performed has specific goal, each one has a particular error metric which is defined in Table 2.5. For instance, Blackscholes is financial pricing application that predicts price of the market based on previous market values and trend. Thus, the output error metric is the Mean Square Error (MSE), and is computed on mean error of output values as follow in Equation 2.3:

$$MSE = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(Accurate_i - Approximate_i)^2} \tag{2.3}$$

Based on this equation, approximated output values $Approximate_i$ are compared with original ones $Accurate_i$ to determine output error.

These applications are simulated in multi-core simulator Sniper [Carlson *et al.* 2014]. From the simulations, we extracted the output results in order to compute the output errors and also the communication traces linked to the traffic between clusters. From the communication traces, which embed information of packet type, source, destination and sent time, we analyze communication distribution and type of data.

Applications	Descriptions	Error metrics
Blackscholes	Financial pricing	Mean error
Canneal	VLSI routing	Routing Cost
Fluidanimate	n-body simulation	Distance in elements
Streamcluster	Online clustering	Cluster center distance

Table 2.5: Used Approximate Benchmarks and associated error metrics [Sampson *et al.* 2015]

3.4.2 Power Vs Quality of Results Trade-offs

To fully evaluate the different techniques, we have to consider both power consumption and application output error. In [IJ1], we explored several applications from the Parsec benchmark suite [Bienia *et al.* 2008]. Here, for sake of conciseness, we only highlight results for the StreamCluster application. The Fig. 2.9 highlights the design space with all the possible techniques to handle approximate communications in ONoC (including the following different BER_{Ax}: 10^{-2}, 10^{-3}, 10^{-5} and 10^{-7}). The bottom of the figure shows zooms of valid solutions from the top figure. All the optical power consumption's have been normalized to the *Baseline* represented by the red cross. This baseline only uses P_H power level, hence no distance management and no approximation are considered. For more clarity, we grouped the results by different communication distance managements: *No Distance aware*, *Short/Long* and *Distance proportional*, in red, blue, and green, respectively.

In each distance management group, the solutions that do not consider approximation or truncation are represented with a cross. Moreover, in each group, some points are located on Pareto front lines which indicate the most effective results in terms of power consumption and output error. The solutions on Pareto front lines are represented with triangles and the configurations are highlighted, e.g. 4A/20T 10^{-3} stands for 4 approximated bits, 20 truncated bits, and $BER_{ax} = 10^{-3}$. For more clarity in the Fig. 2.9, the number of non-approximated bits is not shown, but can be easily deduced as the total number of bits of FP data are 32bits. We also represented the solutions that provide more than 10% of output error, there are located on the right gray part.

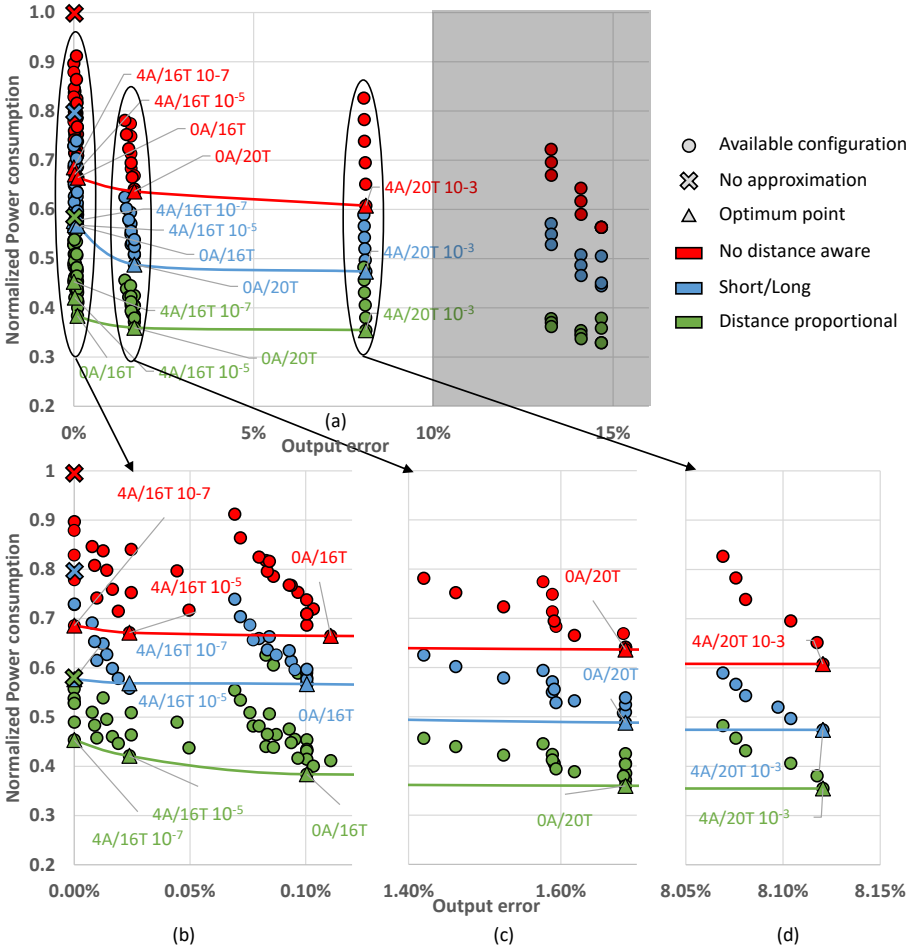


Figure 2.9: Power consumption and output error trade-off for StreamCluster application: (a) design space, (b), (c), and (d) are zooms.

First of all, we can see that there is a huge exploration space with high differences in terms of power versus QoR. The way to manage the distance of communication provides a shift from 12% for the same configuration in terms of approximating or truncating data. Unsurprisingly, the distance proportional solutions are the less power consuming, but at the cost of high complexity. It is interesting to note that, for distance proportional solutions, some configurations

can consume more that the best solution with only two distances considered. The same trend can be seen for two distances management that has solutions worse than the Pareto front of the *No Distance aware*.

Regarding the best solutions in terms of trade-offs between power consumption and output errors, we see the same trends on the different color groups. The first main cloud of solution, zoomed in Fig 2.9-b, are dominated with three points on the Pareto fronts in each group. All these solutions provide less than 0.15% of output errors. The solutions $16NA/0/16T$ with a maximum of 16 truncated bits provide a very small degradation of the output error, but allow to save a huge amount of power compared to the baseline, and for any group of distance managements. Indeed, more than 20% of power reduction is shown.

The second main trend of points, zoomed in Fig 2.9-c, are dominated by the solutions $12NA/0A/20T$. These solutions do not exceed 1.7% output errors while pushing forward the power consumption reduction. The last trend of solutions, that do not exceed the tolerable output error of 10%, is zoomed in Fig 2.9-d. These solutions are dominated by the $8NA/4A/20T$ with $BER_{ax} = 10^{-3}$. These solutions provide less than 8.15% of output error while providing the more power consumption reduction.

To conclude this exploration, first we can state that the design space is huge, and designing an ONI that is be able to configure all of any configuration is unrealistic. Secondly, we highlighted that the different solutions can be grouped in four trends, almost no error (Fig 2.9-b), few errors (Fig 2.9-c), reasonable errors (Fig 2.9-d), and unacceptable errors. All trends are dominated by one or few optimal points, hence limiting the number of interesting solutions to embed in the ONIs. As already mentioned, we target to limit the complexity of the ONI and its management by only considering one low power communication scheme. Thus, the most relevant solution to provide the highest power savings is $8NA/4A/20T$ with $BER_{ax} = 10^{-3}$ which provide a reasonable output error. Finally, regarding the distance of communication, the proposed solution of only considering two group of distance (i.e. Short/Long) provides efficient result compared to no distance management technique. Compared to the optimal solution induced by fully distance proportional links, our solution consumes only 12% additional power. However, it is important to notice that our ONI design is realistic as it requires only four laser powers, compared to [Sunny *et al.* 2020] for which the complexity linearly increases with both number of cores and number of BER levels.

3.4.3 Comparison with fine grain optical power tuning

As discussed in the previous section, fine grain tuning of the optical signal power, as proposed in [Sunny *et al.* 2020], allows reducing the laser power consumption. However, such approach leads to significant challenges related to the design of both analog and digital electronic circuits to control the lasers. In the following we compare our realistic solution with Lorax methodology [Sunny *et al.* 2020].

Scalability comparison: in [Sunny *et al.* 2020], the authors consider a given approximation level for each application. This implies that each driver is capable of targeting, for each destination, any BER from 10^{-12} to 10^{-2}. Hence, for a N-cluster architecture, each driver is capable of emitting a signal at $11 \times (N-1)$ different power levels (e.g. 165 levels per laser for 16 clus-

ters). Since no hardware evaluation is carried out in [Sunny *et al.* 2020], we study their driver complexity assuming only two BER values of 10^{-12} and 10^{-3} for each destination, while still assuming that the optimal BER can be used. Results are provided in Fig. 2.10 for a number of clusters ranging from 2×2 to 16×16. As seen on the figure, the number of power levels for our architecture is constant (4 levels corresponding to P_{Off}, P_H, P_M and P_L) while it linearly grows for *Lorax* despite the considered optimistic assumption. For instance, *Lorax* leads to 255 power levels for 128 clusters (1 level for P_{Off} and two BER levels for each of the 127 destinations), which significantly increases the design complexity of the laser driver, as discussed in the following.

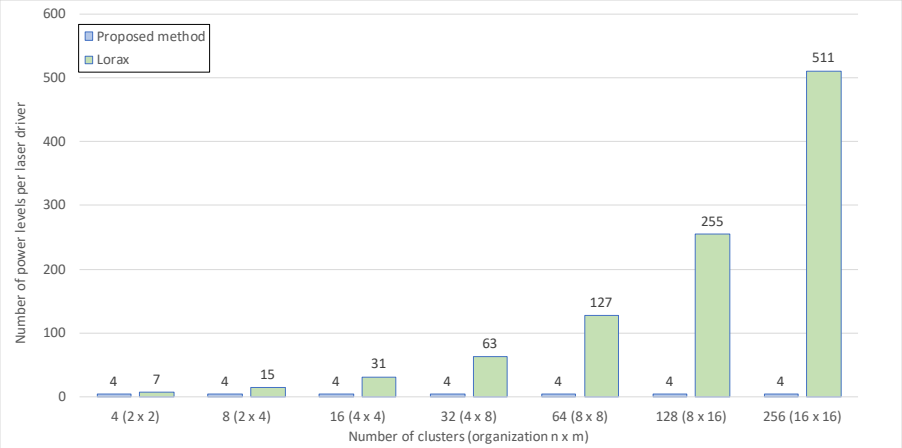
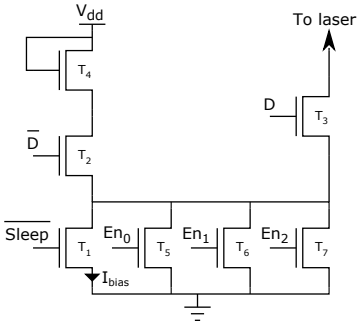


Figure 2.10: Number of power levels per laser driver according to the network size for Lorax and the proposed method.

Laser Driver Design Complexity: we now study the complexity of the CMOS drivers assuming an OOK modulation for both Lorax and our method. CMOS drivers are key components to deliver the right current to the on-chip lasers according to the selected optical power level. We designed multi-level laser drivers for our approach and *Lorax* assuming 16 clusters, as represented in Fig. 2.11. The drivers operate as follow: transistor T_1 drive a I_{bias} current to maintain the laser near the threshold area and transistors drove by En_i signals define the current in the laser emission area. In case communication is required or is predicted, Signal $/Sleep$ is set to 1, allowing to start driving current and to prepare laser for emission. This allows reducing latency penalty occurring to switch on laser from a fully off state [Hosseinabadi & Ansari 2014, Chen & Joshi 2013]. In case no optical communication occurs or is predicted, Signal $/Sleep$ is set to 0, thus leading to energy saving. Temporal penalty and optimization of driver sleep time [Lan *et al.* 2017] are out of the scope of this work but would lead to the same energy reduction for both *Lorax* and our method. Input signal D, which corresponds to the data to transmit, controls the laser emission through transistors T_2 and T_3. The number of transistors increases with the number of laser output powers and the En_i are activated according to selected output power. As detailed in Tables of Fig. 2.11-c) and d), the more transistors driving current, the higher the laser output power. As a result, the driver for *Lorax* requires 34 transistors to generate the 31 laser power levels (15 destinations and 2 levels

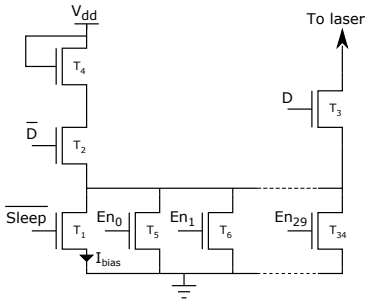
per destination, and P_{Off}). Our approach requires 7 transistors to generated the 4 power levels (i.e. P_H, P_M, P_L and P_{Off}), hence leading to 4.8× reduction in transistor count.



(a)

En_2	En_1	En_0	Current level	Power level
0	0	0	I_{bias}	P_{off}
0	0	1	I_L	P_L
0	1	1	I_M	P_M
1	1	1	I_H	P_H

(c)



(b)

En_{29}	En_{28}	...	En_1	En_0	Current level	Power level
0	0	...	0	0	I_{bias}	P_{off}
0	0	...	0	1	I_1	P_{ax} Hop 1
0	0	...	1	1	I_2	P_{notax} Hop 1
...	...	...	...	...	...	...
0	1	...	1	1	I_{29}	P_{ax} Hop 15
1	1	...	1	1	I_{30}	P_{notax} Hop 15

(d)

Figure 2.11: Laser drivers assuming 16 clusters for a) proposed approach (4 power levels corresponding to P_{off}, P_H, P_M and P_L) and b) *Lorax* (31 power levels corresponding to P_{off} and 15 destinations with 2 levels per destination). The combinations of inputs signals En_i and the corresponding power level are given in tables c) and d).

Table 2.6 summarizes driver estimations on leakage current and footprint for our proposed method and *Lorax*. When targeting a $65nm$ technology node from STM and estimation methods from [Agrawal *et al.* 2014, Dave *et al.* 2013, Agrawal *et al.* 2015], the leakage current for our method is $15nA$ against $125nA$ for *Lorax* (8× reduction). Regarding the footprint estimations, *Lorax* driver requires $8.89mm^2$, while our method only requires $0.25mm^2$ (35× reduction).

	Number of Power Level	**Transistor count**	**Leakage (nA)**	**Footprint (mm^2)**
Proposed method	4	7	15.65	0.253
Lorax [Sunny *et al.* 2020]	34	32	125.12	8.89
Lorax increase (×)	**7.75**	**4.85**	**7.99**	**35.14**

Table 2.6: Driver area estimations for a 16 cluster architecture

Input bits En_i of the drivers are defined using a digital controller. The controller complexity directly depends on number of configurable laser power levels: while the 4-level driver used for our method requires a 2-input to 3-output controller, Lorax would require a 5-input to 30-output controller. In order to evaluate the area and power overhead, we designed the controllers at the RTL level using VHDL. The controllers were synthesized for a $65nm$ ST Microelectronics technology using Synopsis Design Compiler. We targeted a 1GHz frequency corresponding to core speed assumed in the previous experiment. Prime Time was used for power estimation. As seen Table 2.7, our method leads to controller with an area of $42\mu m^2$, thus leading to a 10 ×

reduction compared to Lorax ($445\mu m^2$). Similar gain is observed for the power consumption: the method we propose leads a controller consuming a total of $28.58\mu W$ (leakage $3.25\mu W$ and dynamic power $25.33\mu W$) compared to $290\mu W$ for *Lorax*.

	Power			**Area**
	Dynamic Power (μW)	Cell Leakage Power (μW)	Total Power (μW)	Area (μm^2)
Proposed Method	25.33	3.25	28.58	42.6
Lorax [Sunny *et al.* 2020]	259.8	30.9	209.7	445.1
Gain	**10.2** ×	**9.5** ×	**10.1** ×	**10.4** ×

Table 2.7: Driver controller synthesis results on 65nm STM technology node

Communication power analysis for a comprehensive comparison, we evaluate the total power consumption per laser for *Lorax* and our approach, assuming 16 clusters, BER of 10^{-3} for approximate communications and 33% laser efficiency [Chen *et al.* 2021]. Fig. 2.12 presents the power breakdown (controller and laser) according to the communication distance. Our approach leads to power consumption of $368\mu W$ and $882\mu W$ for short-range communications (hop count 1 to 5) and long-range (hop count 6 to 16) communications respectively. Regarding the power consumption for *Lorax*, it quadratically increases and leads to average power of i) $573\mu W$ for hop count 1 to 5 and ii) $873\mu W$ for hop count 6 to 16. Hence, our method allows reducing power by 35% for short range communications while leading to only 2.5% power overhead for long range communications. Interestingly, our approach is more energy efficient for the longest communication range (hop count 12 to 15 in the figure). The significant gains obtained for short range communications are possible due to the lower static power of the driver controller. The slight power overhead for long-range communication is acceptable considering the 10× reduction in the area footprint: while the drivers for *Lorax* would occupy a total area of $8.89mm^2$ for the studied 16 clusters architecture, our approach would take an area of $0.25mm^2$.

3.5 Conclusion and perspectives

We proposed a method to design scalable approximate nanophotonic interconnects. Our approach aims at improving the interconnect energy efficiency by adapting the transmission robustness to the ap plication requirements. To save energy we proposed to use for error-tolerant data communications with bits truncation and approximation, by lowering power signals. We classified communications according to the transmission losses, thus leading to short-range and long range communications. To minimize the number of laser power levels, we also proposed a method allowing to use a same power for both accurate short-range and approximate long-range communications. The method requires only minor modification of the interconnects interface and is scalable due to the reduced number of power levels. We have evaluated the impact of various combinations of truncation and approximation on the output error of benchmark applications, namely Blackscholes, Canneal, Fluidanminate and Streamcluster. Results show that applications such Streamcluster are the best candidates to use approximation in the nanophotonic interconnect since they are error-resilient and they involve numerous floating-point data.

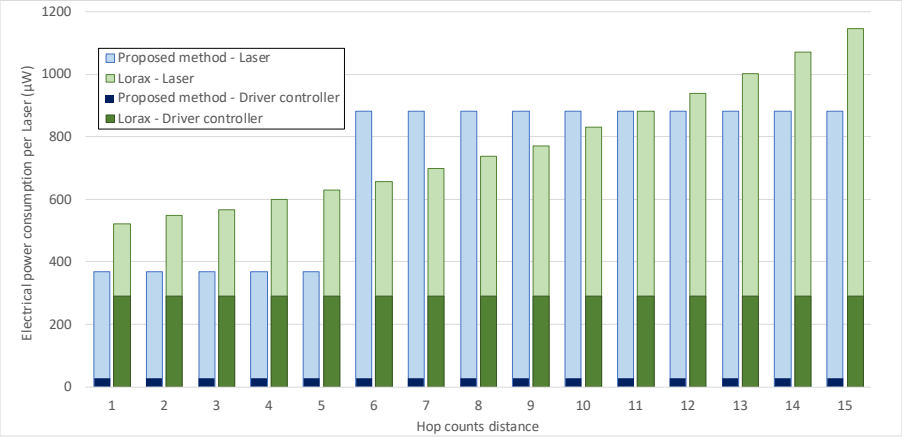


Figure 2.12: Total power consumption per hop count for Lorax and the proposed method. Results are given for a 16 clusters architecture assuming 10^{-3} BER for approximate communications.

Our method leads to 53% power reduction for an acceptable 8% output errors. Compared to state of the art method involving fine grain laser power tuning, we obtain a 35% power reduction for short-range communication. We also demonstrate 35× and 10× reductions in the laser driver and laser controller sizes respectively. Finally, we shown the opportunities to extend this work to off-chip lasers and to task mapping optimization, with the aim to take benefits from the significant energy saving available for short-range communications.

This study also demonstrates the poor scalability of methods involving fine grain tuning of laser powers. It also validates the effectiveness of our approach to maintain the driver complexity low while reaching high level of energy efficiency, especially for short-range and very long-range communications. This opens new research perspectives related to i) short-term deployment of approximate nanophotonic interconnects with mature optical devices and ii) smart mapping of tasks to maximize the utilization of most energy efficient links.

i) Off-chip laser
   Our work focuses on on-chip lasers, which are disruptive but promising technologies for large scale integration of light sources on a chip. However, as off-chip laser is more realistic approach at current development of optical chip laser [Thonnart *et al.* 2020] we will investigate the deployment of our approach to off-chip laser-based interconnects. A key challenge with off-chip lasers would be to adapt the power level according to the communication needs, as off-chip lasers cannot be dedicated to each IP and are generally shared among them. Usually, the power is configured to satisfy any transmission requirements, hence leading to worst case scenario and significant waste of energy. Approaches involving on-chip optical amplifiers such as [Thakkar *et al.* 2016], or method adapting the power regarding the number of active paths in a tree-based optical NoC [Zhou & Kodi 2013], could be used to adapt the output power according to the requirement but would significantly increase the design complexity. Furthermore, approaches involving off-chip lasers necessary require dedicated modulators, while direct modulation could be achieved using on-chip lasers. In this work, our goal is to evaluate

the potential of a prospective architecture to maximize the energy efficiency while keeping the design realistic. To apply the proposed method in the context of off-chip lasers, a straightforward solution would be to consider spatial division multiplexing by using multiple SWMR links per core. Each SWMR would then have the ability to transmit signals at a given power level (P_H, P_M and P_L) to a specific set of receivers (e.g. SWMR with P_L would reach only receivers part of short-range communications). This would lead to three sets of off-chip laser(s) dedicated to i) approximate / short range communications only (SWMR with P_L), ii) accurate / long range communications (SWMR with P_H) and iii) both approximate / long range and accurate / short range communications (SWMR with P_M). Switching ON or OFF the lasers would be achieved at software level, depending on the executed application. Such approach would ideally complement existing solutions that enable to use only a set of wavelengths to communicate [Narayan *et al.* 2021] and that could be easily extended to transmitted truncated data. Obviously, such solutions would lead to significant overhead in the design of the photonics chips (number of lasers and layout) as well as on the electronics, which will have to be evaluated.

ii) Task mapping optimization

Optimizing the task mapping in manycore architectures has been thoroughly studied in the context of electrical NoCs [Wu *et al.* 2017, Wang *et al.* 2020b, Yang *et al.* 2019], hybrid electrical-wireless NoCs [Choi *et al.* 2018] and optical NoCs [Wang & Cheng 2019]. Since the purpose of existing approaches is mainly to minimize the communication costs, they could be adapted to efficiently use the laser power levels available in our architectures. For instance, it would be possible to prioritize very long-range communications in case P_H is used, which can be obtained by optimizing at system level metrics such pJ/bit/mm. Task mapping will also allow to evaluate the efficiency of the approach when extended to larger set of communication ranges and when applied to application specific domain. For this purpose, Convolutional Neural Networks (CNNs) are particularly interesting candidates as communication volumes and data accuracy requirement strongly change between the CNN layers.

3.6 Research dissemination

The works presented in this section have been disseminated in [IJ1], [IC5], [IT1], [IT2], [IT3].

Chapter 3

Fault- and Error-tolerant architectures

Contents

1 Introduction 49
2 Minimizing the impact of permanent errors in NoC 50
2.1 Context of the work 50
2.2 Related work 51
2.3 Shuffling bit to minimize errors for error resilient applications 52
2.4 Evaluation of the method 57
2.5 Conclusion and perspectives 62
2.6 Research dissemination 63
3 Timing errors in Dynamic Voltage and Frequency Scaling architectures 64
3.1 Tolerating error for energy improvement in Near Threshold Computing . . 64
3.2 Timing error detections for adaptive over-/under-clocking 70
3.3 Conclusions and perspectives 77
3.4 Research dissemination 77

1 Introduction

Approaching the limit of CMOS scaling makes devices become increasingly unlikely to be fully functional due to various sources of faults [Srinivasan *et al.* 2004], especially in harsh environment such as in space [Sec 2016]. This sensitivity is further increased with DVFS that became the prominent way to reduce the energy consumption in digital systems. Indeed, effect of scaling coupled with variability issues make highly pipelined systems more vulnerable to timing errors [Stott *et al.* 2013]. This call to provide fault tolerant techniques to enhance robustness or to limit fault impacts on error resilient applications, e.g. neural networks or approximate computing [Torres-Huitzil & Girau 2017].

This section addresses the two following contributions:

- In section 2 we explore the fault tolerance of electrical Network-on-Chip by using mitigation techniques to reduce permanent errors instead of correcting them.
- In section 3 we propose voltage over-scaling based approximate operators for applications that can tolerate errors, and we presente a Dynamic Speculation Window in double-sampling for timing error detection and correction in FPGAs

2 Minimizing the impact of permanent errors in NoC

2.1 Context of the work

Since several decades, technology improvements and transistor shrinking enabled high transistor density per chip according to the Moore's Law [Schaller 1997], reaching today billions of transistors per chip. The integration of many resources on a single chip gave birth to System on Chip (SoC). While frequencies and chip density met the power limit, performance increase has been reached by adding more cores on the chip. Nowadays, SoC includes a large number of cores, memories and hardware accelerators. However, the increase in number of cores induced more and more data transfers. As a result, conventional communication means, such as buses and point-to-point links, cannot ensure efficient communication. To address this gap, NoC appeared as a scalable solution to manage communications between a large number of cores [Xu *et al.* 2005], such as in ACAP Xilinx devices [Swarbrick *et al.* 2019] and Kalray MPPA-256 Bostan manycore processor [Dinechin & Graillat 2017].

Meanwhile, technology scaling and transistor density increase enabled voltage reduction. As a result, the intrinsic failure rate of electronics is increased [Srinivasan *et al.* 2004] while the transistor size reaches 10 nm and below [Bohr 2018]. In this technology era, NoCs became more sensitive to permanent faults, which affect their functionality. Manufacturing defects are the cause of more and more permanent hardware failures due to the engraving thinness affecting the device reliability [Dubrova 2013]. During system lifetime, aging defects [Kundu & Chattopadhyay 2014], like electromigration, Negative-Bias Temperature Instability (NBTI), Hot Carrier Injection (HCI) and Time-Dependent Dielectric Breakdown (TDDB), become additional permanent failure sources on devices. Furthermore, radiations present in high altitude and space are an additional source of permanent faults due to high-energy alpha particles and neutrons destructive hits [Sec 2016]. Manufacturing and aging defects are well know to induce permanent hard errors, e.g., radiations can induce permanent damages due to particle strikes [Mut 2016] [Sexton 2003], such as Single-Event latch-up (SEL), Single-Event Burnout (SEB), Single Event Snap Back (SESB), Single Event Hard Errors (SEHEs), Single Event Gate Ruptures (SEGRs) or Single Event Dielectric Ruptures (SEDRs). In addition, Total Ionizing Dose (TID) effect increases the sensibility of the transistors to these external effects [Duzellier *et al.* 2002]. The accumulation of these events leads to multiple, adjacent or not, permanent faults in a same data word [Rusu 2010]. In particularly, electromigration leads to bridging between adjacent wires [Yu & Ampadu 2011] causing adjacent permanent errors. These fault types are generally modeled by the stuck-at fault model in the literature [Dally & Towles 2004]. Next, we will use the term Single Hard Errors (SHEs) for faults that impact only one bit and Multiple Hard Errors (MHEs) for faults impacting several adjacent bits.

To deal with permanent faults, fault tolerant techniques are commonly applied on the NoC [Radetzki *et al.* 2013, Werner *et al.* 2016, Khalil *et al.* 2019]. These techniques are generally split into three categories: i) detection, ii) diagnosis and iii) correction/mitigation of faults. This work focuses on the third category. Fault correction/mitigation techniques are usually based on i) mitigation through routing algorithms [Fu *et al.* 2014], ii) hardware reconfiguration through spare resources or default backup path [Werner *et al.* 2016], iii) correction through circuit replication [Dubrova 2013] and iv) information redundancy [Liva *et al.* 2016].

Although the aforementioned approaches are efficient for single permanent fault, they are less adequate for multiple permanent faults. They introduce high costs, in terms of latency, area and power consumption, while their mitigation capabilities are limited, as discussed in Section 2.2.

2.2 Related work

Initially, we present the fault tolerant techniques for NoCs mitigation/correction, which can be grouped into four sub-categories, described in the following paragraphs. Then, we present similar approaches applied in other domains than dealing with faults in NoCs.

Routing algorithms avoid faulty paths or faulty regions in NoCs [Fu *et al.* 2014]. For instance, only the remaining healthy resources of NoCs are used during transmission [Chen *et al.* 2019]. The most fault tolerant are adaptive routing algorithms which are generally table-based [Chen *et al.* 2019] or logic-based [Jain *et al.* 2019], including rules to avoid congestion and deadlock during packet transmissions. Other hand, semi-adaptive routing algorithms are generally based on deterministic deadlock-free routing algorithms [Khichar *et al.* 2017], reducing hardware costs and performances, i.e. handled fault number. Therefore, the hardware cost drastically increases with the size of the NoCs. Using routing algorithms can be efficient, as long as the number of faults is limited. Otherwise, the latency may become higher than the acceptable limit, and thus, some Intellectual Propertys (IPs) become unreachable. Hence, this solution is less suitable for large NoCs and multiple faults.

Reconfiguration replaces a faulty element of the NoC by using spare resources at different levels [Chang *et al.* 2019, Mohammed *et al.* 2019]. As spare resources can be used only once, these techniques have large overhead in terms of area and power consumption, while they can tolerate few faults. Other reconfiguration approaches use default-backup paths to avoid data corruptions and packet re-transmissions [Ebrahimi *et al.* 2013]. Although default-backup paths have low area and power consumption, the latency drastically increases under multiple faults, due to the routing complexity. In the worst case, several IPs become unreachable. Last, NoCs can be reconfigured in degraded mode, using only the remaining healthy resources [Chen *et al.* 2019]. Although this method handles permanent faults with lower hardware costs, the latency, throughput and the network congestion increase. Moreover, some IPs become unreachable due to the removal of faulty resources.

Circuit replication, called N-Modular Redundancy, replicates N times, fully or partially, the architecture and votes the replicated outputs. The most popular approach is Triple Modular Redundancy (TMR) [Dubrova 2013], where a module is replicated three times. Several works focus on reducing hardware costs. In [Mukherjee & Dhar 2019], the last level gates of the triplicated modules are designed with a triple transistor logic, which acts as a voter circuit, and thus, replaces the voter. In [Balasubramanian & Naayagi 2017], redundant logic is added in the isolated combinational circuits, avoiding the replication of the entire modules. In these techniques, multiple faults are masked if they occur in the same module. However, if more than one module is affected, the voter cannot correct the output. The area and power consumption overhead remains significantly high, e.g., more than three times for TMR.

Information redundancy inserts additional bits inside messages using ECCs. The most commonly used coding scheme for NoC is the extended Hamming code, which can detect two faulty bits and correct only one. Despite the increase of the bus size of the complete NoC, Hamming

code is efficient for correcting single faults [Liva *et al.* 2016]. The number of correctable bits can be increased by encoding the message on two dimensions [Satya Sai Lakshmi *et al.* 2020] or on several interleaved ECCs [Yu & Ampadu 2008]. However, using ECCs to correct more than one faulty bit dramatically increases the area and the power overheads [Sánchez-Macián *et al.* 2014] due to the necessary increasing of the flit size, leading to almost duplicating or triplicating the size of the flits [Satya Sai Lakshmi *et al.* 2020, Yu & Ampadu 2008]. As a result, the application of multiple ECCs are limited against multiple faults.

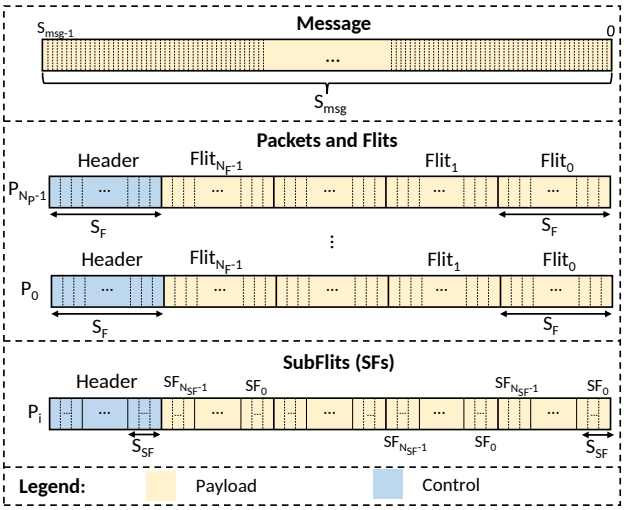
Last, but not least, recently a fifth categories emerges; the use of approximate computing for fault mitigation, already proposed in energy-efficient and high-performance communications [Reza & Ampadu 2019]. Such approximated approaches have been proposed in other research fields and apply a similar idea of transferring the impact of faults towards LSBs. In telecommunication, bit-shuffling methods, called *interleaving*, manage adjacent errors by spreading the errors across several packets. However, they are limited to serial transmissions [Shi *et al.* 2004]. As in NoCs data are forwarded through buses, permanent faults impact every flit that crosses a faulty bus or router. Since faults always appear in the same positions in each flit, the application of interleaving methods is less appropriate. In memory, bit-shuffling methods, called *scrambling*, are used to prevent memories from faults and increase their lifetime [Han *et al.* 2017]. Some works have been proposed in the NoC-based fault tolerant field, which are based on approximate communication. In [Najafi *et al.* 2019], an approach statically changes the assignment of lines in data-path bus, by placing the MSBs on the borders of the bus, to attenuate the electromagnetic influences between neighbored lines. Contrary to our method, the assignment of lines cannot be modified during execution, and thus, external effects and manufacturing defects cannot be addressed by this technique. In [Wang *et al.* 2020a], a multi-plane NoC is proposed to increase performance using a second buffer-less network, where approximate packets can be dropped during the message forwarding, in order to avoid congestion, while the main network ensures a 100% accuracy in packet transmissions. This method needs two parallel networks to operate, which increases drastically the hardware costs compared to the proposed method.

Contrary to the aforementioned approaches, our work efficiently addresses the mitigation of multiple permanent faults for data transferred through the NoC, based on a low overhead hardware mechanism.

2.3 Shuffling bit to minimize errors for error resilient applications

The Bit Shuffling meThod (BiSuT) approach mitigates multiple permanent faults, which can occur in NoC, and especially on the data-path part of the interconnect. We consider faults that can be located in i) the interconnections between routers, or ii) the buffers and the crossbar within each router. Due to nano-scale technologies and power scaling, devices and components become more susceptible to multiple permanent faults [Mut 2016]. As buffers and crossbar are the biggest components of a router [DeOrio *et al.* 2012], they have higher probability of accumulating faults due to radiation effects, manufacturing defects or other intrinsic failures. For the same reasons, interconnections are often impacted by permanent faults, usually stuck-at, short or bridge faults [Dally & Towles 2004], addressed similarly by the proposed method.

For this work, we assume that the positions of the faults are provided by methods such



Symbol	Definition
S_{msg}	Message size
S_{pck}	Payload size
S_F	Sife of flit
S_{SF}	Size of a subflit
$N_P = \frac{S_{msg}}{S_{pck}}$	Number of packets
$N_F = \frac{S_{pck}}{S_F}$	Number of flits
$N_{SF} = \frac{S_F}{S_{SF}}$	Number of sub-flits

Figure 3.1: Message formatting: Packets, flits and subflits, and associated notations.

as Built-In Self-Test (BIST) techniques [Mohammed *et al.* 2019], which diagnose faults in interconnections and routers using Test Pattern Generator (TPG) and Output Response Analyzer (ORA) blocks. In these techniques, TPGs send test packets through the NoC, while ORAs analyze the received packets to identify if faults occurred between these two blocks, providing their positions and the fault type. Further details can be found in [Bhowmik *et al.* 2019]. Other techniques available in the literature can be used to diagnose the permanent faults in a NoC [Xiang *et al.* 2016]. As these techniques are largely studied in the literature, they are not detailed in this work.

As the objective of the proposed approach is to reduce the impact of multiple faults, instead of correcting them, the targeted domains concern error resilient applications used in approximate computing and communication fields, i.e. applications that can tolerate errors until a certain level, such as image processing and machine learning [Ahmed *et al.* 2018].

2.3.1 Principle

We consider classic NoC routing messages of S_{msg} bits. Fig. 3.1 illustrates the organization of such a message into packets and flits. A message is decomposed into N_P packets of S_{pck} payload bits, each packet contains N_F flits of S_F data bits and includes a header flit for the control of routing. As depicted in Fig. 3.1, we further decompose each flit into N_{SF} SubFlits (SF) of S_{SF} bit size to enable the proposed bit-shuffling technique.

The BiSuT applies shuffle and de-shuffle functions that switch, at run-time, two or more Sub-Flits (SFs) within the same flit, in order to transfer the impact of errors on LSBs. Fig. 3.2 illustrates the principles of our approach. In this example we consider flits crossing a faulty router from north to south, as shows the purple arrow of Fig. 3.2-(a). For simplification reasons, the illustration example considers a single buffer, but BiSuT is also applicable with virtual channels. The example focuses on payload flits, while header flits are discussed in Section 2.3.3. As depicted in Fig 3.2-(b), we consider $S_F = 8$ bits and a SF size equal to $S_{SF} = 2$. Therefore,

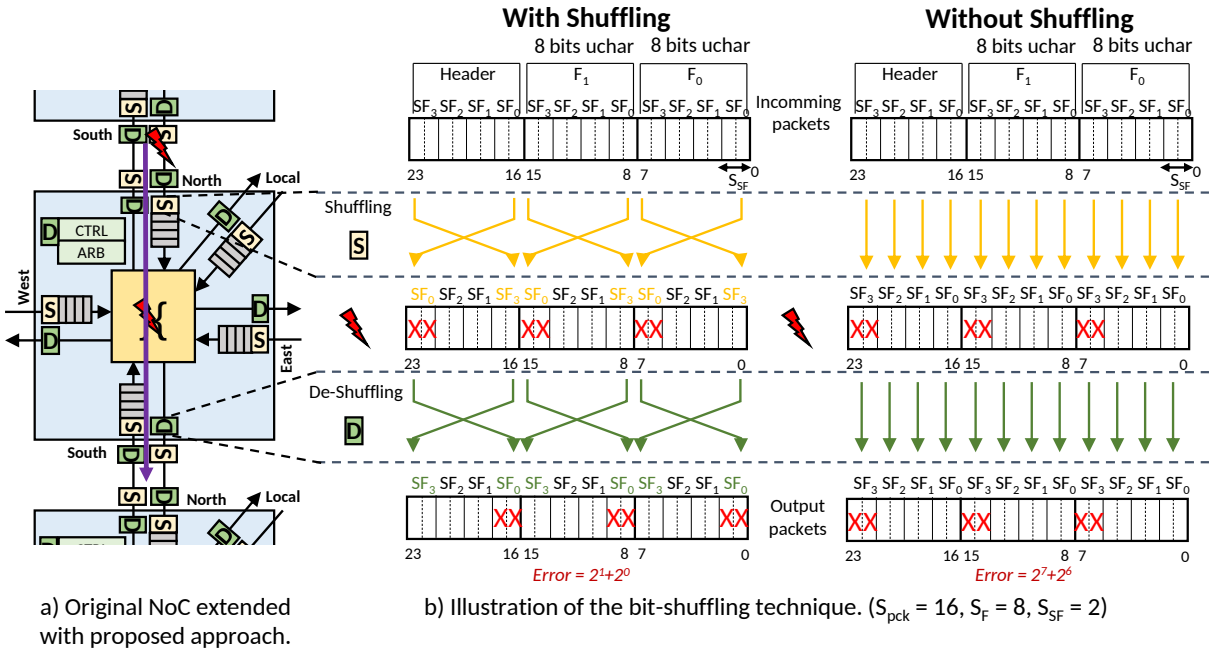


Figure 3.2: Illustration example: Classic NoC extended with BiSuT.

the number of SFs (N_{SF}) in a flit is equal to $N_{SF} = 4$ (SF_0 to SF_3). When no fault occurs, shuffle and de-shuffle functions are disabled and flits cross the NoC router without modification.

Let consider now that two permanent faults occur in the input buffer, affecting the MSBs, i.e., bits 7 and 6 of all incoming flits. The right part of the Fig 3.2-(b) illustrates the crossing of packets without any modification in the flit. The bits 6 and 7 of the two payload flits are affected, leading to errors within the range $\{0, \pm 64, \pm 128, \pm 192\}$, depending on the initial value of the affected bits. The left part of the Fig 3.2-(b) illustrates the proposed bit-shuffling method. The bit-shuffling technique is enabled in the input ports of the router. Then, before crossing the faulty path, the SFs are re-organized by swapping LSBs and MSBs of data bits to allocate MSBs at non-faulty hardware path, i.e. SF_0 and SF_3 are swapped inside each flit. Hence, the impact of the faults is reduced to the range $\{0, \pm 1, \pm 2, \pm 3\}$, depending on the values of the LSBs. Before the flit leaves the router, the SFs are brought to their initial position, and the flit is sent to the output port.

2.3.2 BiSuT implementation

To implement the proposed BiSuT, a NoC router is extended with extra hardware blocks, i.e., Shuffler (S) and De-shuffler (D) blocks. The goal of the shuffler block is to re-organize the SFs with the objective of minimizing the impact of the faults. The goal of the de-shuffler block is to bring back the initial order of the SFs. To deal with the targeted faults, BiSuT is applied i) between two routers, to mitigate errors on the interconnection bus, and ii) between the input and output ports, to mitigate errors inside the router. To achieve that, the aforementioned paths integrate S and D hardware blocks, as depicted in Fig. 3.2-(a). As flits are shuffled when they cross the router, the routing controller (CTRL) needs to re-organize the header of the current

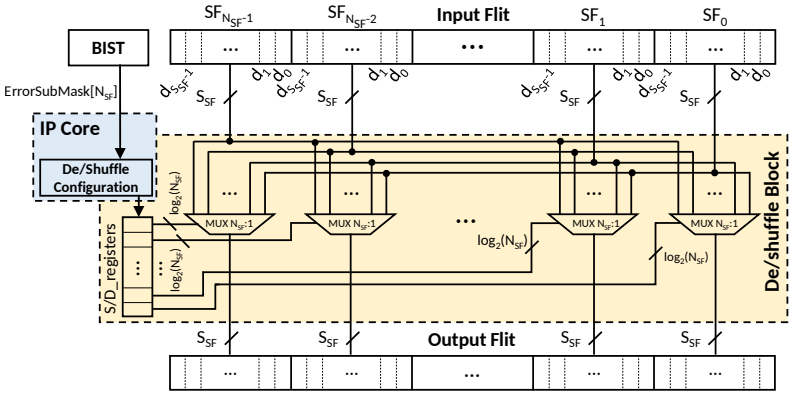


Figure 3.3: Architecture of shuffler (S) and de-shuffler (D) blocks.

packet to read the routing information and propagate the packet towards the expected neighbor or the associated IP. Then, one extra D block is added in the routing controller.

The S and D blocks have similar hardware architecture, presented in Fig. 3.3. It is composed of N_{SF} multiplexers of N_{SF} S_{SF}-bit inputs to one S_{SF}-bit output and registers, which contain the configuration of the multiplexers. The only difference between S and D blocks is the value of the registers, which configures the multiplexers. These values are named S_registers and D_registers for a S and D block, respectively.

2.3.3 Header and sensible data protection

Header flits contain control information, in particular for packet routing. Hence, errors cannot be tolerated in these flits. To handle that, the proposed approach is extended as follows: For NoCs using large data buses (i.e., 64 bits), header flits usually include several unused bits, which are placed on the LSBs. For example, for an architecture with 256 cores, i.e. 16×16 mesh, and packet size equal to 32 flits, the header contains 8 bits for source core, 8 bits for destination core, and 4 bits for packet size. When faults occur on MSBs, the method transfers the faults on the unused SFs, removing any errors.

However, header flits with small data buses (i.e., 32 bits) do not usually include enough unused bits. To address this, we distribute the header flit information into 2 flits, in order to artificially insert unused bits by duplicating flits. Hence, half of the new header flits can be used to tolerate errors, with a small impact on the NoC latency, i.e., adding a single flit in a packet. The same technique is used for sensible data, for example for instructions, with a reduced overhead.

Notice that, today's NoC are typically based on large buses. Hence, the distribution on two flits of headers is a solution that requires to be applied only when the shuffling technique cannot guarantee protection of the header bits.

2.3.4 Data flit organization

To efficiently protect the communication with the BiSuT, packet organization must be considered. The implementation of the BiSuT must take into account the data size (S_{data}) and the

flit size (S_F) to organize the flits inside the Network Interfaces (NIs) of the NoC. For sake of clarity, we define: i) Most Significant Sub-flit (MSS) as the SF including the MSB of the flit, and ii) Least Significant Sub-flit (LSS) as the SF including the LSB of the flit. Up to now we have considered only one data size type per flit and data-bus width equal to a power of 2. However, when considering different data sizes, three cases can occur, as illustrated in Fig. 3.4:

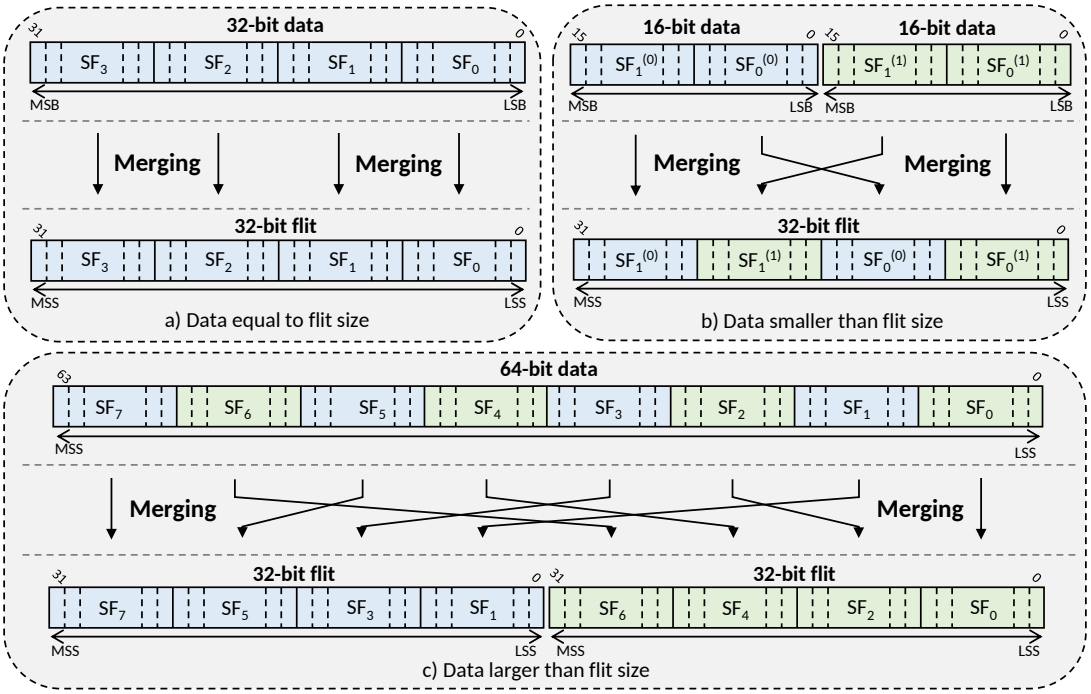


Figure 3.4: Flit organization of 32-bit flit for different data sizes.

a) $S_{data} = S_F$, this is the straightforward case: the data are placed inside the flits without any re-organization, as show in Fig. 3.4-a. The LSBs of the data are placed at the LSSs, and the MSBs of the data are placed at the MSSs.

b) $S_{data} < S_F$, more than one data is sent in one flit. Hence, the data are interleaved within the flit, as shown in Fig. 3.4-b: the MSBs of the data are grouped inside the MSSs, and the LSBs of the data are placed at the LSSs.

c) $S_{data} > S_F$, a data is sent on several flits. The LSBs and MSBs of the data are equally distributed at the flits, as illustrated in Fig. 3.4-c. In this way, all produced flits contain a part of the MSBs and LSBs of the data.

With these re-organizations, the MSSs always hold the important data, compared to the LSSs, making efficient the bit-shuffling method, even when the data-path is impacted by multiple permanent faults. Moreover, packet organization is always included in classic NoC with the help of packetization and de-packetization blocks through the NI, which is the link between an IP core and its router. Then, we only need to insert two additional blocks, called merger and de-merger blocks, that re-organize the data inside flits. As a result, the proposed method does not require high extra hardware.

2.4 Evaluation of the method

In [IJ2] we achieved a comprehensive study of our method by exploring its efficiency at flit-level, regarding data and header protections, at NoC-scale, as well as at application-level with Image Processing and Data Mining Applications. In the following, for sake of conciseness we focus at application-level with an image recognition application based CNN.

2.4.1 Application-level evaluation

The considered application is a Convolutional Neural Network (CNN) developed with Keras API [Gulli & Pal 2017] for image recognizing using Cifar10 [Krizhevsky *et al.* 2009] database. The Cifar10 database is composed of $60,000$ 32×32 colored images split into 10 classes. $50,000$ of these images are used as training data for the CNN learning and the remaining $10,000$ images are used to test the CNN and quantify the accuracy of the classification. Note that, the term *epoch* refers to how many times the training base is used during the learning phase. Theoretically, the validation accuracy is up to 75% for 25 epochs and up to 79% for 50 epochs.

For our experiments, we consider a fault free training phase with 50 epochs to obtain a theoretical validation accuracy of 79%. As shown in Fig. 3.5, each image of the database, used to test the CNN accuracy, is normalized into 16-bit floating-point format and packetized into 32-bit flits, following the approach presented in Section 2.3.4. Fault injection is performed on those flits to simulate the faulty NoC communications. The images are sent to the CNN to compute the approximated classification. The accuracy of the CNN is computed by comparing the approximated classification with the reference, which is computed with free-fault test database. We perform an exhaustive exploration by injecting a MHE to all possible bit positions. We perform a bit-flip on the impacted data to avoid any fault masking, for all elements composing the NoC data-path. We repeat these experiments for MHE sizes equal to 1, 2 and 3 faulty bits considering all possible positions of the fault. The classification accuracy of the BiSuT is compared with the accuracy obtained with Hamming code and the accuracy of unprotected flits. Simulations are implemented in Python 3.6.8 with the help of the Keras and TensorFlow APIs.

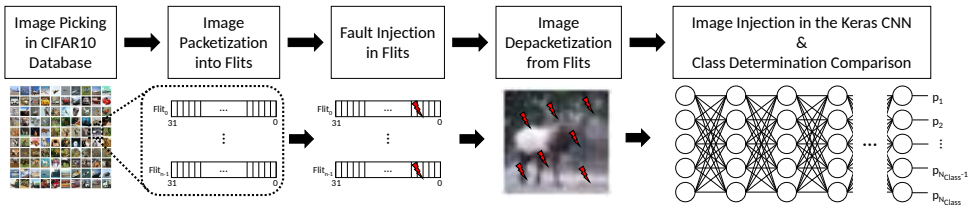


Figure 3.5: Representation of the fault injection in the Keras Convolutional Neural Network (CNN) benchmark.

Fig. 3.6 shows the classification accuracy, i.e., the percentage of correct classification, according to the position of the first faulty bit of the MHE fault.

In Fig. 3.6-a, the fault has a size of one bit. We observe that when the flits are unprotected, the classification accuracy is affected when bits with a position higher than 9 are impacted by the fault. The accuracy can drop to 10%, when MSB are affected. When BiSuT is applied, the class determination accuracy stays at around 79%, which is the theoretical maximum value that

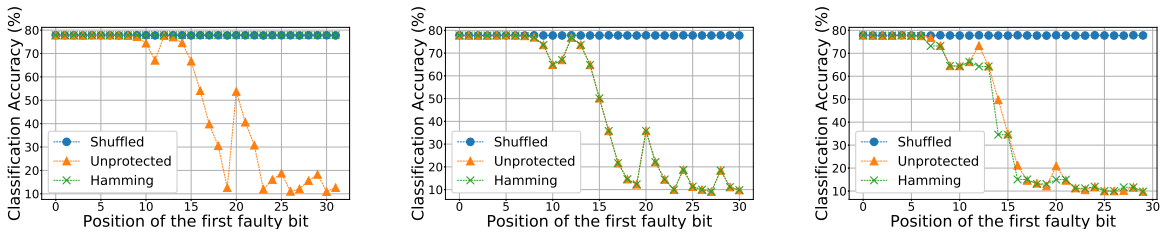


Figure 3.6: Classification accuracy of a CNN on the CIFAR10 database with different fault sizes: a) 1 bit, b) 2 bits, and c) 3 bits.

we can obtain with this CNN. Regarding the Hamming code, the results are similar to BiSuT, since a single bit is faulty, and thus, it can be corrected, providing the maximum theoretical accuracy.

In Fig. 3.6-b, the fault has a size of two bits. Similar to the previous experiment, the accuracy starts to decrease when the bit in position 9 is affected by the fault, i.e., the first faulty bit is the bit number 8, as the burst fault has a size of two bits. Moreover, the classification accuracy for unprotected flits and flits protected with hamming code have similar behavior. On the contrary, when the flits are protected with BiSuT, the accuracy remains at 79% regardless the position of the fault.

Fig. 3.6-c shows the results when the fault has a size of three bits. The behavior is similar to Fig. 3.6-b, except that the accuracy starts to degrade when the first faulty bit is in position 7, since the fault has a size of three bits. Moreover, since a 3-bit fault is considered, the Hamming code is unable to detect the fault. In addition, it can even perform wrong correction, which explains why the classification accuracy with Hamming code is sometimes worst even than the accuracy of the unprotected flits. Nevertheless, BiSuT offers a maximum theoretical accuracy even under a 3-bit fault.

To conclude, BiSuT is more efficient than Hamming code when more than one bit are affected in the incoming flits. Our method maintains the classification accuracy at the maximum theoretical value, when a large number of bits are faulty. On the contrary, conventional methods, such as Hamming code, are not able to preserve the performances. Furthermore, the class determination of the presented CNN is degraded when the bit number 9 is affected by the fault. This means that BiSuT can theoretically handle up to 9 faulty bits by positioning the faults to bits at positions 0 up to 8.

2.4.2 Hardware-level evaluation

In this section, we evaluate the hardware cost of the BiSuT in terms of area cost, power consumption and latency. The shuffler and de-shuffler blocks and the merger and de-merger blocks are synthesized on 28 nm FDSOI technology through a High Level Synthesis (HLS) tools of Mentor Graphic by targeting a clock frequency of 1 GHz. As comparison, we also synthesized an extended Hamming code checker, which is typically used inside NoC routers.

Flit size	Hamming Encoder			Hamming Checker			Hamming Decoder			8 × 8 NoC CONNECT		Overhead	
SF	Area	Power	CP	Area	Power	CP	Area	Power	CP	Area	Power	Area	Power
(bits)	(µm²)	(mW)	(ns)	(µm²)	(mW)	(ns)	(µm²)	(mW)	(ns)	(µm²)	(mW)	(%)	(%)
32 (39 bits bus size)	205	0.26	0.32	519	0.66	0.69	459	0.58	0.58	3, 394, 926	3, 093	24.4	28.8
64 (72 bits bus size)	393	0.5	0.32	1,318	1.7	0.85	971	1.2	0.76	6, 447, 234	4,956	23.5	29.1

Table 3.1: Hamming costs and overhead for a 8×8 NoC CONNECT [Papamichael & Hoe 2012].

Flit size	Sub flit size	Shuffler/De-shuffler block			8 × 8 NoC CONNECT		BiSuT overhead	
SF	SSF	Area	Power	CP	Area	Power	Area	Power
(bits)	(bits)	(µm²)	(mW)	(ns)	(µm²)	(mW)	(%)	(%)
32	4	355	0.23	0.25	2,872,847	2,556	22.3	16.8
(32 bits bus size)	8	188	0.21	0.29			11.8	14.8
64	4	1,274	0.65	0.55	4,585,811	4,021	46.7	27.3
(64 bits bus size)	8	597	0.44	0.25			21.9	18.2

Table 3.2: BiSuT costs and overhead for a 8 × 8 NoC CONNECT [Papamichael & Hoe 2012].

Shuffler and De-shuffler Blocks Table 3.1 and Table 3.2 show the area, the power, and the critical path, respectively for the Hamming technique and our BiSuT method for 32 and 64 data bits. For the Hamming method we performed the evaluation on the encoder, decoder and checker blocks, which are respectively located between the NIs and the routers and within the routers.

As the placement of the blocks of Hamming is not homogeneous in a NoC, and for a fair comparison of the area and power overhead, we performed the comparison between the two techniques on a 8×8 NoC. In this latter, the use of the Hamming technique requires the integration of 64 encoders, 64 decoders, and 288 checkers, which bring the main overheads. For this NoC-scale evaluation, we considered the state-of-the-art CONNECT router [Papamichael & Hoe 2012] based on a 5-ports router, with four virtual channels of 8-flit depth, and a round-robin arbitration. Table 3.2 provides the area and power cost of this NoC, considering 28 nm FDSOI technology. For 32-bits flit, the NoC requires $2,872,847\ \mu m^2$ area and consumes 2,556.99 mW. It has to be noticed that the Hamming method increases the size of the bus as indicated in the Table 3.1. For instance, for subflit of 32 bits, the Hamming technique requires 7 extra bits, hence increasing the size of the bus and buffers to 39 bits. It results an in the area and the power up to $3,394,926\ \mu m^2$ and 3,093.98 mW, respectively.

Regarding the cost of the techniques, for 32-bit flits with 4-bit subflits, the area of one S or D block is only $355.00\ \mu m^2$ and it consumes 0.233 mW. The Hamming checker requires more area and power than the BiSuT, e.g., for 32-bits, it requires $519.00\ \mu m^2$ and consumes 0.663 mW, while it is able to correct only a single error. At the NoC-scale, the BiSuT induces an overhead on area of 22.3% and on power of 16.8%. Compared to Hamming method, this represents a reduction of 9% and 41%, respectively for the area and power overheads, and for a better robustness on fault tolerance. Regarding the BiSuT, we observe that more area is required for smaller subflit, due to the higher number of multiplexers (a smaller S_{SF} means a higher N_{SF}). For instance, for 32-bit flits with 8-bit subflits, the proposed method only add an area overhead of 11.8% and 14.8% on power, which represents a reduction of 2× on area and 1.9x on power

compared to Hamming. Overall, if the flit size increases, the area and power for the S and D blocks also increase, however, they remain smaller compared to the Hamming implementation.

For a configuration with 64-bit flits with 4-bit subflits, the area cost of our method is higher than the area cost of the Hamming code. However, this configuration is a fine grain error mitigation, which is often not required by the application, whereas it mitigates a high number of faults. Despite the area overhead in this case, the hardware gains of the proposed method, compared to the Hamming code, increase drastically when the subflit size increases or when the flit size increases. Moreover, we observe that the critical path of BiSuT is lower than the critical path of the Hamming checker and decoder, regardless the subflit size. For example, if we consider a 32-bit flit with a 4-bit subflit, the critical path is only 0.25 ns for the S and D blocks against 0.69 ns for the Hamming checker, hence not limiting the frequency of the NoC. The Hamming encoder has a lower critical path, but the worst critical paths of the checker and decoders will limit the whole architecture.

Merger and De-merger Blocks The Table 3.3 shows the area, the power and the critical path overheads, which are induced by the merger/de-merger blocks and the header distribution over two flits, when they are implemented in a packetization/de-packetization blocks. For 32-bit flits with 4-bit subflits, the area of the packetization block, which is a part of the NI, is around $7,073\ \mu m^2$ and consumes 7.035 mW. When a merger block is used to organize the data in the flits at the subflit level (M-Packetization) and one shuffler block to apply BiSuT between the NI and the router, we observe an overhead of 5.7% for the area, and of 3.3% for the power consumption. The critical path remains at 1 ns in all cases. When the header distribution on two flits is applied (M/D-packetization), the overhead is increased to 6.0%, for the area, and 3.5%, for the power consumption. The latency is proportional to the distance to reach the destination and the packet size. Then, the addition of one flit due to the header distribution has a small impact on system latency. From the obtained results, we observe that the overhead, introduced by header distribution, is low.

Flit size	**Sub flit size**	**Packetization Hardware Costs**			**M-Packetization Overhead**			**M/D-Packetization Overhead**		
SF	SSF	Area	Power	CP	Area	Power	CP	Area	Power	CP
(bits)	(bits)	(µm²)	(mW)	(ns)	(%)	(%)	(%)	(%)	(%)	(%)
32	4	7,073	7.04	1	5.7	3.3	0	6.0	3.5	0
	8				5.6	3.4	0	6.7	3.5	0
64	4	7,423	6.5	1	3.5	4.2	0	4.3	4.3	0
	8				3.2	3.9	0	4.0	3.8	0

Table 3.3: Packetization overhead with merger blocks and with 2-flit header distribution capabilities.

These results concern only the packetization and de-packetization blocks, being relatively small compared to the NoC and the entire NI. The Table 3.4 displays similar results for the de-packetization blocks.

Global Comparison of the Hardware Costs Fig. 3.7 shows the global overhead for a 8×8 NoC when the BiSuT is used with the merger and de-merger blocks at the NI and the header distribution process. For that, we consider that each router and interconnection are protected with the BiSuT. Moreover, the header distribution and the merger/de-merger blocks for data

Flit size	Sub flit size	De-packetization Hardware Costs			M-De-Packetization Overhead			M/D-De-Packetization Overhead		
SF	SSF	Area	Power	CP	Area	Power	CP	Area	Power	CP
(bits)	(bits)	(μm^2)	(mW)	(ns)	(%)	%	(%)	(%)	%	(%)
32	4	4,374	3.83	1	12.1	1.2	0	13.1	11.9	0
	8				11.3	12.5	0	13.6	11.9	0
64	4	6,928	6.18	1	4.8	2.4	0	5.7	2.3	0
	8				5.8	3.0	0	4.9	2.2	0

Table 3.4: De-packetization overhead with merger blocks and with 2-flit header distribution capabilities.

size management are added to the NI. For the comparison with Hamming (H), we consider that each router and interconnection are enhanced with the Hamming code.

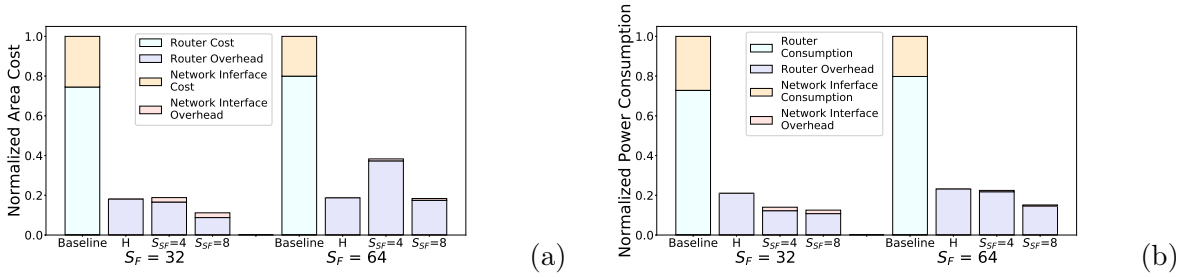


Figure 3.7: Overhead comparison between BiSuT and Hamming code with a) area and b) power.

Results displayed in Fig. 3.7 show that the global overhead of the proposed method depends of the flit size and subflit size. For example, for 32-bit flits with 4-bit subflits, the overhead of the BiSuT is of 18.8% for the area and 14.0% for the power consumption, against 18.2% and 21.0% for the Hamming code. We observe in Fig. 3.7 that the area overhead is correlated with the subflit number that composes the flit. For example, with 8 subflits the costs are similar to the Hamming code, while they are superior with 16 subflits and lower with 4 subflits. However, we note that the power consumption is particularly reduced with the use of the BiSuT. In general, the global overhead of the proposed method is equal or lower than the overhead of the Hamming code.

We show that the performance of the NoC is not impacted by the combinatory blocks used to shuffle and merge the flits. The critical path with the proposed method stays approximately the same as the routing logic, contrary to the Hamming implementation. The proposed method ensures a limited impact on the frequency, with a few additional cycles for the bit-shuffling implementation. Moreover, the register computation is optimized to complete when no inversion is performed and this algorithm is executed only when a fault is detected. These optimizations limit the impact in terms of NoC performance. It has to be noticed that the lightweight configuration of the BiSuT (e.g. 32-bit flits with 8-bit subflits, and 64-bit flits with 8-bit subflits) outperforms the Hamming method on the protection of data as demonstrated in section 2.4, while fine grained mitigation configurations, such as 64-bit flits with 4-bit subflits, consume more area but for an incomparable mitigation efficiency.

2.5 Conclusion and perspectives

The presented technique allows multiple permanent faults to be mitigated through reorganization of the flits. Our proposal swaps bits to transfer permanent faults on LSBs keeping MSBs safe. The approach is extended with merger/de-merger blocks to ensure that MSBs of the data are always located on the MSBs of the flits and that the LSBs of the data are always located on the LSB of the flits. Furthermore, flit redundancy is inserted to handle critical data, such as headers. The comparison with an Hamming code shows that BiSuT is able to mitigate a large number of faults against a low hardware cost in terms of area and power consumption. Moreover, this method has small critical path and latency overhead.

We have extended this work by proposing a Region-based BIt-Shuffling Technique (R-BiSuT) applied at a coarse-grain level, that trades off fault mitigation efficiency to reduce further the hardware costs [IC2]. To achieve the R-BiSuT, the NoC is divided into regions, and the shuffling/de-shuffling technique is then applied at their borders. The obtained results show that the area and power overheads can be reduced by 31% and by 35%, respectively, with a small impact on the MSEs.

This study demonstrates a lightweight solution to handle multiple permanent errors in a NoC based heterogeneous architecture. It opens new research perspectives related to i) improvement of dynamic fault-tolerant routing protocols, and ii) handle process variability issues for emerging on-chip interconnects.

i) Latency improvement in faulty NoCs
As futur works we will explore how the shuffling technique improve the latency in a faulty NoC. Routing algorithm is the key element to determine the path between a source and a destination in a NoC based architecture. Adaptive routing algorithms determine the data path with respect to fault location within the NoC. In the state of the art, faulty components are bypassed by using neighbor routers. These bypasses may become complex in presence of several faulty locations, as multiple bypasses may cause deadlocks. This can be solved by using rectangular bypass area encompassing the faulty routers or by using routing based on Virtual Channels (VCs) [Naghibi Jouybari & Mohammadi 2014, Khichar *et al.* 2017, Charif *et al.* 2020]. These adaptive algorithms increase the latency as bypasses are longer paths, or are area hungry due to VCs. The bit shuffling approach provide new routing paths within a faulty NoC. Indeed, an error-resilient data can cross faulty routers to maintain the packet latency similar to a fault-free NoC. Regarding sensitive data, flit duplication also allows them to cross the faulty zone, hence reducing the routing algorithm complexity, while maintaining the latency low.

ii) Handling process variability issues of nanophotonic links
In ONoCs, the conversion between electric to optical signals are done by using dedicated NIs. Due to architectural constraints, these NIs are shared between several cores, e.g. cluster of cores. Process-variations on MRs has been reported in a photonics platform [Orcutt *et al.* 2011] leading to 4.79nm of process-variation drift. Without countermeasure, this process-variation can lead to inoperative components, hence to a network bandwidth degradation. Methods have been proposed to compensate this problem by adding extra

MRs to compensate faulty components leading to 90% of available bandwidth at the cost of double the MRs [Xu *et al.* 2012]. The resonant shift can also be tuned with temperature adjustment, but at the cost of 2.9mW/nm. Thus, associating ONoC with bit shuffling is an efficient lightweight solution to handle inoperative MR due to Process-variation for error-tolerant applications.

2.6 Research dissemination

The works presented in this section have been disseminated in [IJ2], [IC2], [IC3], [NC1], [NC2].

3 Timing errors in Dynamic Voltage and Frequency Scaling architectures

DVFS became the prominent way to reduce the energy consumption in digital systems by trading the performance of the system to an acceptable margin. Indeed, scaling down supply voltage effects in quadratic reduction in energy consumption of the system, but increase the propagation delay of the gates. Scaling techniques have evolved and been explored in greater extent over the time to unlock the opportunities of higher energy efficiency by operating the transistors near or below the threshold voltage [Kuroda *et al.* 1998, Le Sueur & Heiser 2010]. Effect of scaling coupled with variability issues make highly pipelined systems more vulnerable to timing errors [Stott *et al.* 2013], hence near-threshold computing is still seen as no go zone for conventional sub-nanometer designs.

In this section we present two main contributions:

1. Voltage over-scaling based approximate operators for applications that can tolerate errors. We explore the energy efficiency achievable versus the output errors by tuning transistor operating triad: combination of supply voltage, body-biasing scheme, and clock frequency.

2. Dynamic speculation window used in double sampling schemes for timing error detection and correction in pipelined logic paths. The proposed method employs online slack measurement and conventional shadow flipflop approach to adaptively overclock or underclock the design and also to detect and correct timing errors due to temperature and other variability effects.

3.1 Tolerating error for energy improvement in Near Threshold Computing

3.1.1 Context of the work

Error-resilient computing is an emerging trend in VLSI, in which accuracy of the computing can be traded to improve the energy efficiency and to lower the silicon footprint of the design [Han & Orshansky 2013]. Emerging classes of applications based on statistical and probabilistic algorithms used for video processing, image recognition, text and data mining, machine learning, have the inherent ability to tolerate hardware uncertainty. Such error-resilient applications that can live with errors, void the need for additional hardware to detect and correct errors. Also, error-resilient applications provide an opportunity to design approximate hardware to meet the computing needs with higher energy efficiency and tolerable accuracy loss. In error-resilient applications, approximations in computing can be introduced at different stages of computing and at varying granularity of the design. Using probabilistic techniques, computations can be classified as significant and non-significant at different design abstraction levels like algorithmic, architecture, and circuit levels.

3.1.2 Approximation in Arithmetic Operators

Approximations in arithmetic operators are broadly classified based on the level at which approximations are introduced [Han & Orshansky 2013]. This section reviews methods proposed

in the literature at physical and architectural levels.

In [Tziantzioulis *et al.* 2016], operators of a Functional Unit (FU) are characterized by analyzing relationship between V_{dd} scaling and BER (Bit Error Rate, ratio of faulty output bits over total output bits). Based on the characterization, for every FU in the pipeline, one more *imprecise* FU running at lower V_{dd} is designed. According to the application's need, selected set or all the FUs in the pipeline are accompanied by an *imprecise* counterpart in the design. Based on the user defined precision level, computations are performed either by *precise* or *imprecise* FU in the pipeline. Instead of duplicating every FU with an imprecise counterpart, a portion of the FU is replaced by imprecise or approximate design as discussed in [Han & Orshansky 2013]. For instance, least significant inputs can be processed by *approximate* operator and most significant inputs can be processed by *accurate* operator to increase the energy efficiency at the cost of acceptable accuracy loss. In [Soares *et al.* 2015] n-bit RCA adder based on near-threshold computing is proposed with two parts; k-bit LSBs approximated while $(n-k)$ bits computed by precise RCA. scriptsize

Another class of physical-level approximation is achieved by applying dynamic voltage and frequency scaling to an accurate operator. Due to the dynamic control of voltage and frequency, timing errors due to scaling can be controlled flexibly in terms of trade-off between accuracy and energy. This method is referred as Voltage Over-Scaling (VOS) in [Tziantzioulis *et al.* 2016]. Similar to VOS, clock overgating based approximation is introduced in [Kim *et al.* 2016]. Clock overgating is done by gating clock signal to selected flip-flops in the circuit during execution cycles in which the circuit functionality is sensitive to their state. In all the approximation methods at physical level, in addition to the deliberate approximation introduced, impact of variability has to be considered to achieve optimum balance between accuracy and energy. Decoupling the data and control processing is proposed in [Akturk *et al.* 2015] to mitigate the impact of variation in near-threshold approximate designs. Also, technologies like FDSOI provide good resistance towards the impact of variability.

Approximation at architectural level is discussed in [Li *et al.* 2015], where accuracy control is handled by bitwidth optimization and scheduling algorithms. Also other forms of architectural-level approximations are discussed in [Lingamneni *et al.* 2011], where probabilistic pruning based approximation method is proposed. In this method, design is optimized by removing certain hardware components of the design and/or by implementing alternate way to perform the same functionality with reduced accuracy. In [Kim & Tiwari 2011], a probabilistic approach is discussed in the context of device modeling and circuit design. In this method, noise is added to the input and output nodes of an inverter and the probability of error is calculated by comparing the output of the inverter with a noise-free counterpart. In [Schlachter *et al.* 2015], new class of pruned speculative adder are proposed by adding gate-level pruning in speculative adders to improve Energy Delay Area Product (EDAP). Though there is claim that the pruned speculative adder will show higher gains when operated at sub-threshold region, no solid justification is given in [Schlachter *et al.* 2015]. In general, approximations introduced by pruning methods are more rigid in nature, which lacks the dynamicity to switch between various energy-accuracy trade-off points.

On contrast, voltage scaling based approximations are more flexible, easy to implement and offer dynamic control over energy-accuracy trade-off. Approximation introduced by supply voltage scaling offers *dynamic approximation*, by changing the operating triad (combination

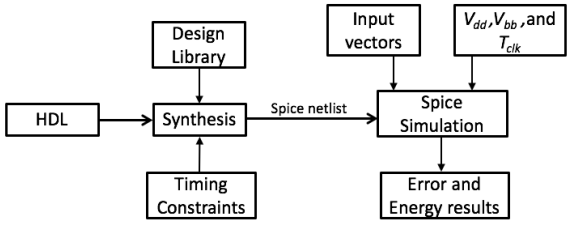


Figure 3.8: Flow of arithmetic operator characterization.

of supply voltage, body-biasing scheme, and clock frequency) of the design at runtime, which makes the user to control the energy-accuracy trade-off efficiently. In [Lingamneni *et al.* 2013], limitations of voltage over-scaling based approximate adders such as need for level shifters and multiple voltage routing lines are mentioned. These limitations can be overcome by employing uniform voltage scaling along the pipeline or at larger granularity.

3.1.3 Characterization of arithmetic operators

In this section, characterization of arithmetic operators is discussed for voltage over-scaling based approximation. Fig. 3.8 shows the characterization flow of the arithmetic operators. Structured gate-level HDL is synthesized with user-defined constraints. The output netlist is then simulated at transistor level using SPICE (Simulation Program with Integrated Circuit Emphasis) platform by varying operating triads (V_{dd}, V_{bb}, T_{clk}), where V_{dd} is supply voltage, V_{bb} is body-biasing voltage, and T_{clk} is clock period. In ideal condition, the arithmetic operator functions without any errors. Also, EDA tools introduce additional timing margin in the datapaths during Static Timing Analysis (STA) due to clock path pessimism. This additional timing prevents timing errors due to variability effects. Due to the limitation in availability of design libraries for near/sub-threshold computing, it is necessary to use SPICE simulation to understand the behaviour of arithmetic operators in different voltage regimes. By tweaking the operating triads, timing errors e are invoked in the operator and can be represented as

$$e = f(V_{dd}, V_{bb}, T_{clk}) \tag{3.1}$$

Characterization of arithmetic operator helps to understand the point of generation and propagation of timing errors in arithmetic operators. Among the three parameters in the triad, scaling V_{dd} causes timing errors due to the dependence of operator's propagation delay t_p on V_{dd}, such as

$$t_p = \frac{V_{dd}.C_{load}}{k(V_{dd} + V_t)^2} \tag{3.2}$$

Body-biasing potential V_{bb} is used to vary the threshold voltage (V_t); thereby increasing the performance (decreasing t_p) or reducing leakage of the circuit. Due to the dependence of t_p on V_t, V_{bb} is used solely or in tandem with V_{dd} to control timing errors. Scaling down V_{dd} improves the energy efficiency of the operator due to its quadratic dependence to total energy. $E_{total} = V_{dd}^2.C_{load}$. Mere scaling down F_{clk} does not reduce the energy consumption, though it will reduce the total power consumption of the circuit. Therefore, F_{clk} is scaled along with V_{dd} and V_{bb} to achieve high energy efficiency.

Benchmarks	Area (μm^2)	Total Power (μW)	Critical Path (ns)
8-bit RCA	114.7	170	0.28
8-bit BKA	174.1	267.7	0.19
16-bit RCA	224.5	341	0.53
16-bit BKA	265.5	363.4	0.25

Table 3.5: Synthesis Results of 8 and 16 bit RCA and BKA

Benchmarks	T_{clk} (ns)	V_{dd} (V)	V_{bb} (V)
8-bit RCA	0.5, 0.28, 0.19, 0.13	1 to 0.4	-2 to 2
8-bit BKA	0.5, 0.19, 0.13, 0.064	1 to 0.4	-2 to 2
16-bit RCA	0.7, 0.53, 0.25, 0.20	1 to 0.4	-2 to 2
16-bit BKA	0.7, 0.25, 0.20, 0.15	1 to 0.4	-2 to 2

Table 3.6: Operating triads used in Spice simulation

3.1.4 Experiments and Energy Efficiency Results

In our experiments, we characterized 8- and 16-bit ripple carry adder (RCA) and Brent-Kung adder (BKA) using 28nm-FDSOI technology. Table 3.5 shows the synthesis results (area, static plus dynamic power, critical path) of different adder configurations. Post synthesis, SPICE netlist of all the adders are generated and simulated using Eldo SPICE (version 12.2a). Table 3.6 shows the different operating triads used to simulate the adders. Clock period (T_{clk}) of the adders is chosen based on the synthesis timing report. Supply voltage (V_{dd}) of all the simulations are scaled down from 1.0V to 0.4V in steps of 0.1V and body-biasing potential (V_{bb}) of -2V, 0V, and, 2V. Pattern source function of SPICE testbench is configured with specific input vectors to test the adder configurations. Circuit under test is subjected to 20K simulations for every different operating triad with same set of input patterns. Energy per operation corresponding to different operating triads is calculated from the simulation results. Output values generated from the SPICE simulation are compared against the golden (ideal) outputs corresponding to the input patterns. Automated test scripts calculate various statistical parameters like BER (ratio of faulty output bits over total output bits), MSE and bit-wise error probability (ratio of number of faulty bits over total bits in every binary position) for all the test cases.

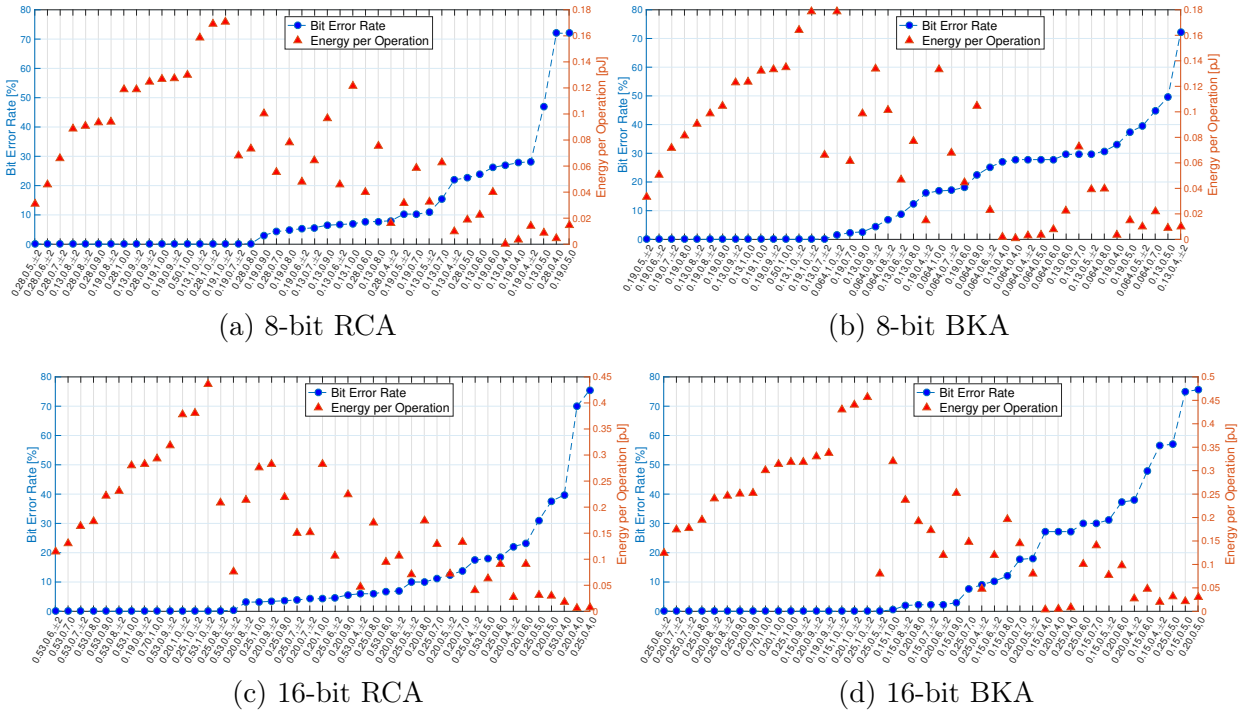
Fig. 3.9-a) and Fig. 3.9-b) show the plots of BER vs Energy/Operation of 8-bit RCA and BKA adders. Likewise plots of BER vs Energy/Operation of 16-bit RCA and BKA adders are shown in Fig. 3.9-c) and Fig. 3.9-d) respectively. The label of *x-axis* of the plots show the operating triads in the format T_{clk} (ns), V_{dd} (V), and V_{bb} (V) respectively. In all the adder configurations, energy/operation decreases and BER increases in sync with the supply voltage over-scaling. Table 3.7 shows the maximum energy efficiency (amount of energy saving compared to ideal test case) achieved by 8-bit and 16-bit RCA and BKA in different BER ranges. Due to the parallel prefix structure, BKA adders show staircase pattern in BER plot shown in Fig. 3.9-b) and Fig. 3.9-d. On other hand, RCA adders based on serial prefix show exponential pattern in BER plot.

Energy/operation curve of all the four plots show two patterns corresponding to 0% BER, and BER greater than 0%. Effect of voltage over-scaling is visible in the left half of the plots, where energy/operation is gradually reduced in sync with reduction in V_{dd} while BER is at 0%.

BER Range	Number of Triads				Max. Energy Efficiency (%)				BER at Max. Energy Efficiency (%)			
	8-RCA	8-BKA	16-RCA	16-BKA	8-RCA	8-BKA	16-RCA	16-BKA	8-RCA	8-BKA	16-RCA	16-BKA
0%	16	14	15	18	76	75.3	60.5	73.3	0	0	0	0
1% to 10%	15	7	15	9	87	65.3	83.6	84	8	8.8	6	9.1
11% to 20%	2	5	6	3	74	89	86.2	73.3	11	16.1	17.5	18.1
21% to 25%	3	2	2	-	92	82.8	90.8	-	22	25	22.1	-

Table 3.7: Energy Efficiency and BER in 8-bit and 16-bit Ripple Carry and Brent-Kung Adders

Another important observation is that the effect of body-biasing is helping to keep the BER at 0% in this region of the plot. Both the 8-bit RCA and BKA adders operated at 0.5V V_{dd} with forward body bias of 2V V_{bb}, achieve maximum energy efficiency of 76% and 75% respectively at 0% BER. Similarly, 16-bit RCA and BKA achieve maximum energy efficiency of 60% and 59% respectively at 0% BER, while operating at 0.6V for V_{dd} with forward body bias V_{bb} of 2V. This set of operating triads provides high energy efficiency without any loss in accuracy of the computation by taking advantage of near-threshold computing and body-biasing technique.



(a) 8-bit RCA (b) 8-bit BKA

(c) 16-bit RCA (d) 16-bit BKA

Figure 3.9: Bit-Error Rate vs. Energy/Operation for 8-bit and 16-bit adders

On the right half of the BER vs Energy/Operation plots, where the BER is greater than 0%, energy curve starts in three branches and tapers down when the BER reaches 40% and above. In those three branches, operating triads with body-biasing are most energy efficient followed by triads without body-biasing and finally triads with overclocking. In 8-bit BKA adder, 28 out of 43 operating triads operate within 0% to 25% BER. Similarly, 36 triads operate within 0% to 25% BER in 8-bit RCA adder. In 16-bit BKA and RCA adders, correspondingly 30 and 38 operating triads operate within BER range of 0% to 25%. For an application with acceptable error margin of 25%, 8-bit RCA, 8-bit BKA, 16-bit RCA, and 16-bit BKA can be operated at

0.4V V_{dd} with forward body bias V_{bb} 2V to achieve maximum energy efficiency of 92%, 89%, 90.8%, and 84%, respectively.

Approximation in arithmetic operators based on voltage over-scaling, provµm²ides *dynamic approximation*, which makes the user to control the energy-accuracy trade-off efficiently by changing the operating triad of the design at runtime. In this method, *dynamic approximation* can be achieved without any design-level changes or addition of extra logic in the arithmetic operators unlike accuracy configurable adder proposed in [Kahng & Kang 2012]. Dynamic speculation techniques like in section 3 can be used to estimate the BER at runtime to switch between different triads to achieve high energy efficiency with respect to user defined error margin. 8-bit RCA and BKA can be dynamically switched from accurate to approximate mode by merely scaling down V_{dd} from 0.5V to 0.4V at the cost of 8% BER to increase energy efficiency from 76% to 87%. Similarly in 8-bit RCA, switching from accurate to approximate mode is possible by reducing V_{dd} from 0.5V to 0.4V at the cost of 16% BER to increase energy efficiency from 75% to 89%. BKA adder configuration records more BER compared to RCA because of more logic paths of same length due to parallel prefix structure. Likewise in 16-bit RCA, accurate to approximate mode can be switched by scaling V_{dd} from 0.6V to 0.4V at the cost of 6% BER to increase energy efficiency from 60% to 84%. In 16-bit BKA, accurate to approximate mode can be switched by scaling V_{dd} from 0.6V to 0.4V at the cost of 9% BER to increase energy efficiency from 59% to 84%. Both 16-bit adders provide leap of 24% increase in energy efficiency at the maximum cost of 9% BER compared to accurate mode.

3.1.5 Discussion

The use of voltage over-scaling is an efficient way to improve the energy efficiency of operators and to provide a trade-off with approximation level of outputs that can be used for error-resilient applications. However, with process and temperature viariabilities, the use of voltage over-sclaing can be tough to deploy on real system without control on the output quality, especially for near-threshold computing. In the following section we present an online technique allowing to measure the timing slack and to adapt the clock frequency with respect to the targeted output errors.

3.2 Timing error detections for adaptive over-/under-clocking

3.2.1 Context of the work

Effect of scaling coupled with variability issues make highly pipelined systems more vulnerable to timing errors [Stott *et al.* 2013]. Such errors have to be corrected at the cost of additional hardware for proper functioning [Nicolaidis 2015]. The Razor method proposed in [Ernst *et al.* 2004] is a very popular error detection and correction architecture for timing errors in digital systems, inspired from the more general double-sampling technique [Nicolaidis 2015]. In the double-sampling architecture, an additional sampling element, known as shadow register, is used along with the main output register to sample the output at different timing instances. The shadow registers are clocked by a delayed version of the main clock known as the shadow clock. The timing interval between the rising edges of the main clock and the shadow clock is termed as the speculation window (ϕ).

There is a trade-off between width of the speculation window and fault coverage. For a wider speculation window, more errors could be detected and corrected, but this requires more buffer insertion for logic paths that are shorter than the speculation window. And vice-versa for a narrower speculation windows. Designs with fixed speculation window optimized for certain operating condition might suffer from an increased error rate under variability issues such as thermal effects. In order to tackle these limitations, double-sampling technique with dynamic speculation window is required to adpatively vary the speculation window with respect to the variability effects.

Razor-like fault detection and correction techniques are used in reconfigurable architectures like FPGAs (Field-Programmable Gate Arrays) and CGRAs (Coarse-Grain Reconfigurable Arrays) due to moderate clock frequencies and function-specific pipelines [Stott *et al.* 2014], [Brant *et al.* 2013]. All these existing methods use a fixed speculation window for the double sampling and overclocking of the system without any adaptive feedback to tackle the variability effects. Due to the fixed speculation window, there is no dynamicity in these methods, which cannot adaptively change the frequency of the design in both directions (overclocking and underclocking) at run time.

In this work, we propose to use a dynamic speculation window in double-sampling schemes for error detection and correction in pipelined logic paths. An FPGA prototype is used as proof of concept, though the idea can be extended to any type of datapaths. **This technique is suitable for voltage-over scaling, however we focus on a FPGA demonstrator with frequency scaling as it is more easily tunable, while still producing similar timing errors**. The proposed method is based on double-sampling and online slack measurement [Levine *et al.* 2013] to overclock and underclock the design adaptively according to temperature and other variability effects, and therefore to detect and correct timing errors. Addition of buffers for shorter path is not required, since the speculation window is dynamically changed.

3.2.2 Proposed Method for Adaptive Overclocking and Error Detection in FPGAs

In this work, we propose to use a dynamic speculation window combined with the double-sampling method for error detection and correction and accordingly to adaptively overclock and underclock the design based on variability effects. The following sections present the proposed

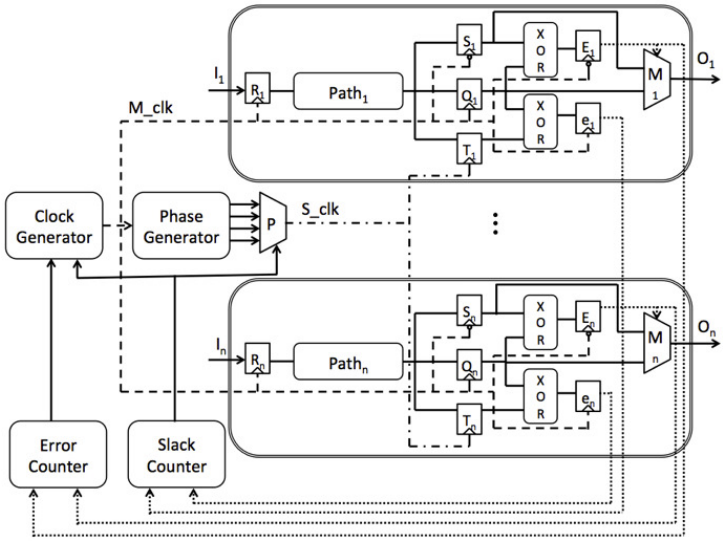


Figure 3.10: Principle of the dynamic speculation window based double-sampling method. Solid lines represent data and control between modules, dashed lines represent main clock, dash-dot-dash lines represent shadow clock, and dotted lines represent feedback control

architecture of a dynamic speculation window and then our adaptive feedback loop for over-/under-clocking the design.

Dynamic Speculation Window Architecture Fig. 3.10, shows the schematic of the proposed dynamic speculation window architecture. A design composed of n logic paths is annexed with two shadow registers. The number of logic paths n is pre-determined as the percentage of the critical delay margin. In the rest of this section, we consider one path $Path_1$ out of n paths. As shown in Fig. 3.10, logic path $Path_1$ is constrained between input register R_1 and output register Q_1. Two shadow registers S_1 and T_1 are added to sample the output of the $Path_1$, in contrast to the one shadow register approach in [Stott *et al.* 2014], [Brant *et al.* 2013], and [Levine *et al.* 2012]. T_1 is used to measure the available slack in the path which is used to overclock the design, while S_1 is used to sample the valid data, when the delay of the $Path_1$ is not meeting the timing constraints of the main clock M_clk due to the variability effects. As like double-sampling architectures, it is assumed that the shadow register S_1 always samples the valid data. In this setup, all three registers S_1, Q_1 and T_1 use the same clock frequency but different phases to sample the output. The main output register Q_1 samples the output at the rising edge of M_clk, while shadow register S_1 samples the output at the falling edge of M_clk. Shadow register T_1 samples the output in the rising edge of the shadow clock S_clk generated by the phase generator. Two XOR gates compare the data in main and shadow registers S_1 and T_1, the corresponding error values are registered in E_1 and e_1. The registers E_1 and e_1 are clocked by falling and rising edge of the M_clk respectively.

Fig. 3.11 shows the timing diagram of the proposed method. In Scenario 1, the shadow clock (S_clk) leads the main clock (M_clk) by phase ϕ_1. Since the $Data$ (output of $Path_1$) is available before the rising of the shadow clock, the valid data is sampled by T_1. Registers Q_1

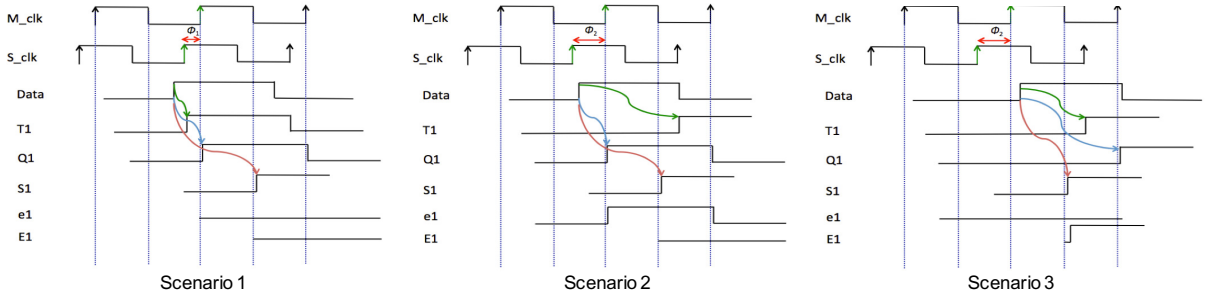


Figure 3.11: Timing diagram of proposed method. Scenario 1: Slack measurement phase. Scenario 2: Maximum overclocking margin. Scenario 3: Impact of temperature at maximum overclocking frequency.

and S_1 also sample the valid data in the rising and falling edges of the main clock respectively. Since the main and shadow registers sampled the same data, e_1 and E_1 are held at logic zero representing no timing error. In Scenario 2 shown in Fig. 3.11, the shadow clock leads the main clock by a wider phase ϕ_2, which results in invalid data being sampled by T_1, while Q_1 and S_1 sample the valid data. In this case, the error signal e_1 is asserted at the rising edge of the main clock, while E_1 is still at logic zero, which indicates the overclocking using speculation window greater than or equal to ϕ_2 will result in errors. In Scenario 3 shown in Fig. 3.11, due to temperature or other variability effects, the logic delay of the $Path_1$ violates the timing constraints of both the shadow and main clocks. This results in wrong data getting sampled by the shadow register T_1 and main register Q_1. The shadow register S_1 samples the valid data since *Data* is valid before the falling edge of the main clock. Since both T_1 and Q_1 sample wrong data, e_1 is not asserted, while E_1 is asserted due to the difference between Q_1 and S_1 registers. Fig. 3.10 shows the schematic of error correcting mechanism in the proposed method. Here the outputs of the main register Q_1 and the shadow register S_1 are connected to the multiplexer M_1. When the error signal E_1 is equal to zero, multiplexer output M_1 passes output of Q_1. When E_1 is held high, the output of S_1 is passed to the next stage of the pipeline.

Adaptive Over-/Under-clocking Feedback Loop Fig. 3.10 shows the feedback loop of the proposed method. The error signals E_i and e_i from all the monitored paths are connected to *Error Counter* and *Slack Counter* respectively. For different speculation windows ϕ_i, *Error Counter* and *Slack Counter* count the discrepancies in the paths being monitored. The *Slack Counter* margin defines the amount of errors that can be tolerated while overclocking the design and the *Error Counter* margin defines the amount of error that can be tolerated due to variability effects. Tolerable error margin of *Error Counter* and *Slack Counter* are determined from the timing reports after synthesis and placement of the design. After, the synthesized bitstream is programmed into the FPGA and the design is intially clocked at the maximum frequency (F_{max}) estimated by Vivado.

In this work, we have implemented error-free overclocking, therefore *Slack Counter* margin is set to zero. At the ambient core temperature of around 28°C, when *Error Counter* and *Slack Counter* are equal to zero, the speculation window ϕ is increased for every clock cycle

from 0° to 180° at discrete intervals of 25° until the *Slack Counter* records an error. Beyond 180° up to 360°, the speculation window width repeats due to periodic nature of the clock. Once the slack counter records an error, the system is put on halt and safely overclocked by trading the available slack. While the speculation window ϕ is swept, the pipeline functions normally since the output register Q_i and the shadow register S_i sample the same valid data. Regarding error correction due to variability effects, if *Error Counter* records more than 2% of the monitored paths not meeting the timing constraints, the system is put on halt and the operating frequency of the system is decreased in multiples of a clock frequency step δ to reduce the errors (in our test platform the frequency step is $\delta = 5MHz$). Since correcting more errors will reduce the throughput of the design, decreasing the operating frequency is a necessary measure. Multiplexer M_i, as shown in Fig. 3.10, corrects the error by connecting the output of the shadow register S_i to the next stage of the pipeline. Once the temperature falls back, the operating frequency of the design is again increased based on the slack measurement as described earlier.

3.2.3 Experiments and results

The proposed Dynamic Speculation Window in the double-sampling method is implemented in a Xilinx Virtex VC707 evaluation board. Vivado tool flow is used to synthesize and implement the RTL benchmarks into the FPGA. A testbench is created for all the benchmark designs with an 80-bit LFSR pattern generator for random inputs to the design under test. Onboard IIC programmable LVDS oscillator is used to provide clock frequency for the pattern generator and the design under test. As shown in Fig. 3.10, control signals from the *Error Counter* and the *Slack Counter* determine the output frequency of the LVDS oscillator. Different phases of the generated clock frequency are obtained from the inbuilt MMCM (Mixed-Mode Clock Manager) module in the FPGA [Xilinx 2015]. The core temperature is monitored through XADC system monitor, and the temperature is varied by changing the cooling fan speed.

To highlight the adaptive overclocking and timing error detection and correction capability of the proposed method, we have implemented a set of benchmark designs in both Razor-based method and the proposed dynamic speculation window based method. Benchmarks used in this experiments are common datapath elements like FIR filter (8 taps, 8 bits), and various versions of 32-bit adder and 32-bit multiplier architectures. After implementation, timing reports are generated to spot the critical paths in the design. Table 3.8 shows the area footprint and the maximum operating frequency determined by the Vivado tool flow. Timing and location constraints are used to place the shadow registers S_i and T_i close to the main output register Q_i for all the output bits of the design. For the Razor-based method, all the monitored paths are annexed with one shadow register S_i in contrast to the proposed method with two shadow registers, and the logic paths that are shorter than speculation window are automatically annexed with buffers by the synthesis tool. For comparison purpose, both the Razor-based method and the proposed method are subjected to identical test setup. In the Razor-based method, the design is overclocked beyond the maximum frequency estimated by Vivado and the FPGA's core temperature is varied to introduce variability effects in the design. Online slack measurement is not used in the Razor-based method to show how the proposed method with slack measurement can adaptively overclock as well as underclock under identical test setup.

In the proposed method, after the synthesized bitstream is programmed into the FPGA, the

design is clocked at F_{max} estimated by Vivado. At the ambient core temperature of around 28°C, the speculation window ϕ is increased for every clock cycle and if the *Slack Counter* is zero at the widest speculation window, then the clock frequency can be increased up to 40%, because at 180°, ϕ corresponds to half of the clock period. That implies all the critical paths in the design have positive slack equals to half of the clock period. Overclocking is done by halting the design and increasing the clock frequency from the LVDS oscillator. The LVDS oscillator is programmed through the IIC bus with the pre-loaded command word for the required frequency. After changing the frequency, again the phase sweep starts from 0° until the *Slack Counter* records an error. This way, the error-free overclocking margin of each design is determined with respect to the maximum operating frequency estimated by Vivado.

While operating at the safe overclocking F_{max} at 28°C, the core temperature of the FPGA is varied by changing the speed of the cooling fan. Due to the increase in temperature, the critical paths of the design starts to fail. When the *Error Counter* records error in more than 2% of the monitored paths, the design is put on halt and the frequency is reduced in multiples of 5MHz until the *Error Counter* margin is below 2%. Since the shadow register S_i latches the correct output, errors are corrected without re-executing for that particular input pattern. The tolerance margin is used to reduce the overhead in error correction mechanism which can affect the throughput of the design.

Table 3.8 provides synthesis results on the benchmarks without any error detection technique, with razor flip-flops [Stott *et al.* 2014], and with proposed error detection and correction technique. The area overhead of the proposed method compared to the original design is kept between 3.6% to 5.4%, which demonstrates the applicability of the technique in real designs. Table 3.8 also provides the maximal frequency F_{max} estimated by Vivado. The Look-Up Table (LUT) overhead of the proposed method is on average 0.4% less compared to the Razor implementation, since no buffers are needed for the shorter paths. However, Flip-Flop (FF) overhead of the proposed method is 0.8% more compared to the Razor method due to the two shadow flip-flops used for slack measurement and error correction.

Benchmarks	Area w/o error detection		Overhead for proposed method		Razor overhead		Estimated
	LUTs	Flip-Flops	LUTs	Flip-Flops	LUTs	Flip-Flops	Fmax (MHz)
8-tap 8-bit FIR Filter	1279	1547	2.5%	2%	2%	1%	167
Unsigned 32-bit Multiplier	7270	4511	1.9%	1.7%	2.3%	0.9%	76
Signed 32-bit Wallace Tree Multiplier	5997	4458	2.8%	1.8%	3.2%	1%	80
2-bit Kogge-Stone Adder	3944	4117	3.7%	1.7%	4%	1.2%	130
32-bit Brent-Kung Adder	3753	4053	3.3%	1.9%	3.8%	1%	135

Table 3.8: Synthesis results of different benchmark designs

Fig. 3.12, shows the plot of safe overclocking frequency of all the benchmark designs that can be reached by the proposed error detection method for different FPGA's core temperature. As an example, for the FIR filter design, Vivado estimated F_{max} is 167MHz. However this design can then be overclocked at the ambient temperature of 28°C by more than 55% up to 260MHz, without any error, and up to 71% for the other benchmarks. At the safe overclocking frequency, the temperature is increased from 28°C to 50°C at which *Error Counter* records 12% of the paths failed. Since the percentage of failing paths is more than the determined margin of 2%, the clock frequency of the LVDS oscillator is brought down to 180MHz, as shown in Table 3.9, which results in a percentage of failing paths below 2%. Once the temperature falls back to

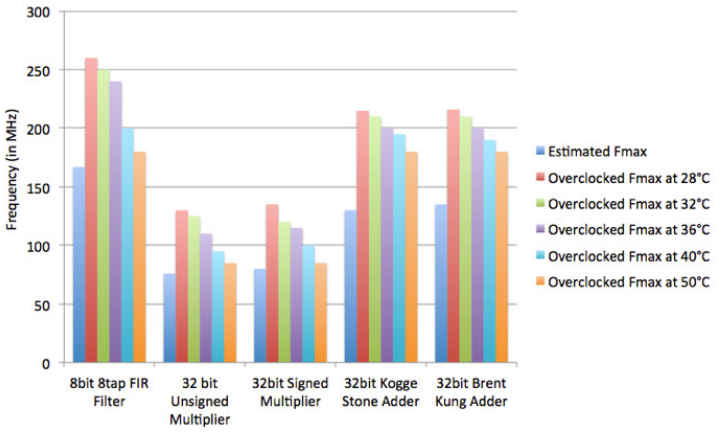


Figure 3.12: Overclocking versus temperature for different benchmarks.

Benchmarks	Safe overclocking F_{max} at 28°C (in MHz)	Safe overclocking margin at 28°C	% of monitored paths fails at 50°C for F_{max} at 28°C	Safe overclocking F_{max} at 50°C (in MHz)	Safe overclocking margin at 50°C
8-tap 8-bit FIR Filter	260	55%	12	180	8%
Unsigned 32-bit Multiplier	130	71%	22	85	12%
Signed 32-bit Wallace Tree Multiplier	135	69%	26	85	7%
32-bit Kogge-Stone Adder	215	64%	13	180	38%
32-bit Brent-Kung Adder	216	60%	21	180	33%

Table 3.9: Impact of temperature while overclocking in benchmark designs

28°C, the available slack is measured and the clock frequency is increased again to 260MHz.

Table 3.9 lists the impact of temperature and the safe overclocking margin at the ambient temperature of 28°C and high temperature of 50°C for all the benchmarks. The percentage of monitored paths that fails at 50°C shows the impact of the temperature while overclocking the design. The operating frequency has to be scaled down to limit these errors which can be eventually corrected by the error correction technique in the proposed method. Even at the higher temperature of 50°C, the FIR filter is safely overclocked up to 8% compared to the maximum frequency estimated by Vivado. Similarly, at 50°C, both unsigned and signed multiplier architectures can be safely overclocked up to 12% and 7% respectively. A maximum safe overclocking margin of 38% is achieved for the Kogge-Stone adder and 33% by the Brent-Kung adder while operating at 50°C. These results demonstrate the overclocking and error detection / correction capability of the proposed method with a limited area overhead in the FPGA resources.

Fig. 3.13, shows the comparison of the proposed dynamic speculation window and the Razor-based overclocking for the 8-bit 8-tap FIR filter design in Virtex 7 FPGA. As shown in Fig. 3.13, the core temperature of the FPGA is varied by controlling the cooling fan of the FPGA. The FPGA's core temperature is increased from the ambient room temperature of 28°C to 50°C and then decreased again back to the room temperature. When the temperature increases, critical paths in the design starts to fail, which makes the proposed method and the Razor-based method to scale down the frequency to limit the error below 2% of monitored paths. Initially, at the room temperature, the design is running at 260 MHz. At the temperature of

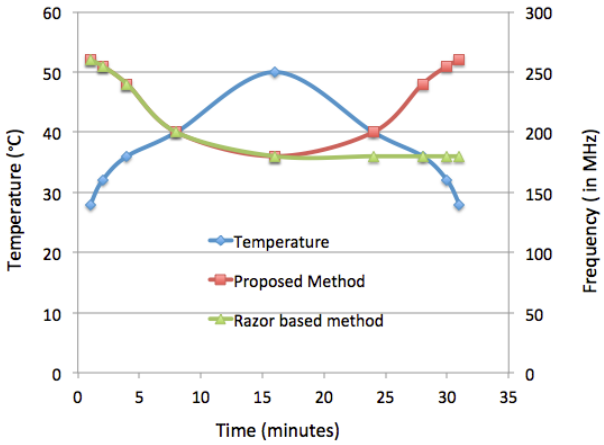


Figure 3.13: Comparison of Razor-based feedback look and the proposed dynamic speculation window. Curve in blue diamond represents the FPGA's core temperature. Curve in red square represents the frequency scaling by the proposed method. Green triangle curve represents the frequency scaling by the Razor-based method.

50°C, the frequency is scaled down to 180 MHz. When the temperature falls back to the room temperature, the proposed method is able to measure the available slack in the pipeline and to increase the frequency to overclock the design and therefore to increse performance. Under similar test setup, the Razor-based feedback loop scales down the frequency as the temperature increases. However, when the temperature falls back, there is no change in the frequency. This is because the Razor implementation does not have the slack measurement in place to adaptively change the frequency when the temperature is reducing. Due to the additional shadow register placed in the proposed method, the system is able to measure the slack available in the pipeline which is traded to overclock the design according to temperature fluctuations. From Table 3.12, we can observe a 0.78% difference in terms of area overhead between the proposed method and the Razor implementation, on average. This gives more upper hand for the proposed dynamic speculation window based error detection method over the classical Razor-based timing error detection.

3.2.4 Discussion

In this work, we have proposed to use a dynamic speculation window in double-sampling for error detection and correction, and implemented this method in an FPGA prototype. The proposed design uses double-sampling and slack measurement to adaptively overclock the design. The maximum of 71% safe overclocking margin at ambient temperature has been achieved at the cost of shadow registers, XOR gates, and counters, which results in a maximum area overhead of 1.9% LUTs and 1.7% FFs over the original design, as shown in Table 3.8. Instead of merely overclocking the design this method detects timing errors and corrects it in real time. This method can be easily expanded to other form of datapaths and also reconfigurable architecture like CGRAs.

3.3 Conclusions and perspectives

We have proposed to use voltage over-scaling to highlight possible trade-off between energy efficiency and approximation in arithmetic operators that can be used for error-resilient applications. In [IC10], we characterized different configurations of adders using different operating triads to generate statistical model for approximate adder. We have achieved maximum energy efficiency of 76% in 8-bit RCA while operating at 0.5V V_{dd} without any accuracy loss. By increasing the effect of voltage over-scaling from 0.5V to 0.4V, energy efficiency is increased to 87% at the cost of 8% BER. All the adder configurations have shown maximum energy gains of up to 89% within 16% of BER and 92% within 22% of BER. Behaviour of arithmetic operators during voltage over-scaling in near/sub- threshold region can be characterized by SPICE simulations. However, SPICE simulators take long time to simulate exhaustive set of input patterns needed to characterize arithmetic operators. In [IC10], we have laid down the framework to construct statistical model by characterizing approximate operators based on voltage over-scaling. These models allow to simulate the behavior of faulty arithmetic operators at functional level. Dynamic approximation is enabled by employing dynamic speculation methods to monitor the errors at runtime [IC11] at a reasonnable area overhead. The proposed method is generic enough to be applied to any datapath, hence are promising for any hardware accelerators, and it has been validated on a FPGA demonstrator.

3.4 Research dissemination

The works presented in this section have been disseminated in [IC10] and [IC11].

Chapter 4

Research perspectives

Contents

1 **Research perspectives** **79**

1.1 Fast design space exploration of multi-technologies NoC for manycore architectures 80

1.2 Cache coherence protocols for emerging on-chip interconnects 81

1.3 Fault tolerant accelerator for AI 81

1.4 Arithmetic logic circuit in 2D semiconductors for approximate computation 82

1 Research perspectives

My research perspectives aim at contributing to build the next generation of computing architectures. My research emphasis is on energy efficiency of heterogeneous manycore architectures and is driven by the three following challenges:

- **End of Moore's law:** the scaling limit of the transistors is known to be nearly reached, bringing with it the end of increasing amount of resources for computing architectures [Theis & Wong 2017]. However, the industry market keeps demanding for better computing architectures. I am convinced that hardware specialization is a key to tackle this challenge: it requires dedicated hardware accelerators to be designed to handle part of computation [Borkar & Chien 2011], while being tailored to achieve energy efficiency constraints, such as hardware accelerators for neural networks [Venkataramani *et al.* 2021], or Matrix Multiplication in High Performance Computing [Asri *et al.* 2021].

- **Emerging technologies:** they are essential to build tomorrow's computing architectures. Recent integration technology advances offer opportunities for manycore architectures and computing paradigms [Vivet *et al.* 2021]. For instance, photonic on silicon and wireless-on-chip enable new interconnect topologies [Shacham *et al.* 2008], efficient broadcasting for cache coherency [Karkar *et al.* 2016], and photonic Multiply-Accumulate (MAC) for neural networks [Nahmias *et al.* 2020].

- **Fault tolerance:** Compared to CMOS, disruptive technologies suffer from higher variability due to still maturing fabrication process. This prevents from their deployment [Wachter *et al.* 2017], which calls for optimization methods and dedicated fault-tolerant hardware designs to improve their robustness. Moreover, as we approach the limit of

CMOS scaling, it becomes increasingly unlikely for a computing device to be fully functional due to various sources of faults [Srinivasan *et al.* 2004, Oates 2016], especially in harsh environment such as in space [Sec 2016]. This call to provide fault tolerant techniques to enhance robustness or to limit fault impacts on error resilient applications, e.g. neural networks or approximate computing [Torres-Huitzil & Girau 2017].

Naturally, part of these perspectives takes place by benefiting from my current research activities and are mid-term research perspectives, such as "Fast design space exploration of multi-technologies NoC for manycore architectures", "Cache coherence protocols for emerging on-chip interconnects", while "Fault tolerant accelerator for AI" and "Arithmetic logic circuit in 2D semiconductors for approximate computation" are long-term research perspectives.

1.1 Fast design space exploration of multi-technologies NoC for manycore architectures

System-on-chips become more and more complex as they integrate an increasing and various number of Intellectual Properties (IP) within the same chip such as a large variety of cores, memories, interconnects, and hardware accelerators. This heterogeneity and complexity, increase dramatically the effort to explore the design space of a manycore architectures. Current manycore simulators are time consuming. For instance, Sniper manycore simulator [Carlson *et al.* 2014] is known to provide a good tradeoff between simulation time and accuracy of results (10%) as it simulates instructions on time intervals instead of each clock cycle. To simulate the execution of a 64-core architecture running the x264 application with a 13-second video from the PARSEC benchmark suite [Bienia *et al.* 2008], it requires around 5 days of simulation with a Dual Intel Xeon 4214 processor (2x12 cores at 3.5GHz) and 64GB of memory (2.4GHz), hence clearly limiting the exploration if we target several parameters evaluation or different thread mappings. It is worth mentioning that with a cycle accurate simulator such as Gem5 it takes weeks [Binkert *et al.* 2011]. In this latter full-system simulator, 40 to 70% of simulation time is dedicated for the network [Mandal *et al.* 2019]. On the other hand, mathematical modeling [Mandal *et al.* 2021, Vijaya Bhaskar & Venkatesh 2021] provides an interesting tradeoff between accuracy of results and computation time. It has been shown that to test a 10x10 manycore architecture it only requires 38min of computation instead of 12h of simulation with Booksim simulator, at the cost of 15% of errors [Vijaya Bhaskar & Venkatesh 2021].

In this work, we will aim at proposing an analytical model for the exploration of heterogeneous on-chip interconnect technologies in the context of manycore architectures. The major challenge to face will be the consideration of different interconnect technologies that provide various performances and packets management. Moreover, evaluating heterogeneous architectures along with errors is not well addressed in the literature. We plan to integrate models considering reliability parameters such ash Bit Error Rate of communication, and error resilience of network-interfaces to evaluate the future manycore architectures.

Means: this mid-term project is part of my researches done through my young research ANR SHNoC project. I am currently supervising a PhD student on analytical modeling of emerging on-chip interconnects who started his thesis on October 2020, and a postdoctoral researcher on optical network-interface error mitigations who started on January 2022.

1.2 Cache coherence protocols for emerging on-chip interconnects

Parallelism capabilities of manycore architectures obviously generate an enormous amount of data exchanges making the communication medium a key element of the overall performance of the system. However, because of the difference of speed between the processors and the main memory, fast and small dedicated hardware-controlled memories containing copies of parts of the main memory (a.k.a caches) are used [Martin *et al.* 2012]. To keep up-to-date these distributed copies of data and to synchronize their accesses, it requires to share information between the nodes. This leads to an increase in the number of data transfers that must be supported by the interconnection media. Furthermore, these specific data transfers concern generally one source and several destinations, which correspond to multicast/broadcast communication schemes. In parallel, technology evolution has allowed for the integration of silicon photonics and wireless communications on chip, thus leading to the Wireless Network-on-Chip (WiNoC) [Karkar *et al.* 2016, Chang *et al.* 2008] and Optical Network-on-Chip (ONoC) [[Shacham *et al.* 2008, Sosa *et al.* 2018] paradigms. These emerging technologies are showing significant advantages for broadcasting data (WiNoC) and low-latency communications (ONoC).

In this context, we will explore how combining these emerging technologies can improve the efficiency of on-chip interconnection systems of shared-memory manycore architectures based on cache coherence protocols. We will study new cache protocols which may benefits from the broadcast efficiency of WiNoC and the low latency on long distance if the ONoC. Model of communications for each media/technology will be defined, in terms of data transfer latency, power consumption, etc., and these models will be used at the operating system level to select the best media for each type of transfer. For that, on-line mechanisms will be developed to evaluate the network traffics, and to estimate the optimal path for each new communication.

Means: this mid-term project is part of ANR Allopticall2 and Rakes projects. I am currently supervising a postdoctoral researcher on cache coherence protocols for wireless and optical NoCs who started on december 2021.

1.3 Fault tolerant accelerator for AI

Deep Neural Networks (DNNs) are amongst the most intensively and widely used predictive models in machine learning [LeCun *et al.* 2015]. Nonetheless, increased computation speed and memory resources, along with significant energy consumption, are required to achieve the full potentials of DNNs, hence calling for dedicated hardware accelerators. However, with the increasing connection density of DNNs, NoC is key interconnect for such architectures [Krishnan *et al.* 2022].

Similarly to traditional computing hardware, hardware accelerators are subject to faults, occurring due to process, aging and environmental reliability threats. Faults occuring after the training phase can seriously affect DNN inference. As a result, DNN prediction failures appear [Torres-Huitzil & Girau 2017, Ruospo *et al.* 2020, Lotfi *et al.* 2019], seriously affecting the application execution, and become a major issue for safety-critical and mission-critical applications, such as robotics, aerospace, smart healthcare, and autonomous driving. In [Bosio *et al.* 2019], a fault injection campaign on different DNNs has shown that the fault impacts depend on the DNN itself, but also on the layer where the faults appear.

In this work we will study heterogeneous fault-tolerance techniques for a NoC-based DNN hardware accelerator. By considering fault impacts between the layers, we will adapt the level of data protection with mapping optimization technique. This mapping of the different layers will allow to optimize the communication paths through the routers with respect to the data protection to guarantee and the throughput to reach.

Means: For this long-term project a collaboration with the French Directorate General of Armaments (In french "Direction générale de l'armement (DGA)") is in progress to fund a PhD thesis starting on october 2020 on this topic. This work will also benefits from the ANR Re-Trusting which aims at providing confidence and trust in decision-making based on AI by explaining the hardware wherein AI algorithms are being executed. This ANR is a consortium between INL (Project Coordinator), LIP6, Inria, and Thales.

1.4 Arithmetic logic circuit in 2D semiconductors for approximate computation

With the end of More's Law, several CMOS substitutes appears as challengers, such as organic semiconductors [Myny *et al.* 2012], carbon nanotubes [Kreupl 2013], or two-dimensional (2D) materials [Zeng *et al.* 2018]. Researchers show more and more interests for this latter as two-dimensional materials highlight an ultrathin channel thickness. This leads to reduced short-channel effects, and an improved electrostatic gate control, hence scaling and power consumption are improved [Wachter *et al.* 2017]. Moreover, they are one of the main candidates for tunnel field-effect transistor (FET) allowing a swing better than 60mV per decade. However, before being mature, numerous problematics have to be solved for this technology. For instance, a 1-bit processor demonstrator has been made where only 80% of the ALUs were working [Wachter *et al.* 2017].

We will propose the design of approximate arithmetic circuits based on 2D-FETs. The first step is to model the behavior of the 2D-FET, then we will build airthmetic logic circuits to study how to use them efficiently for approximate computation. This evaluation methodology has been adopted to evaluate the efficiency of 2D-FET for FPGA fabrics [Baskaran & Sampson 2021]

Means: This is a long-term perspective project where I have not started working on yet. For this work, collaborative project is a necessity as it is a cross-level challenge. I believe that our knowledges on low-power design, fault-tolerant architecture, and approximate computing, are a key to build next generation of hardware accelerator with emerging 2D-FET.

Bibliography

[Agrawal *et al.* 2014] Yash Agrawal, Rajeevan Chandel and Rohit Dhiman. *Design and analysis of efficient multilevel receiver for current mode interconnect system.* In 2014 IEEE Students' Conference on Electrical, Electronics and Computer Science, pages 1–6, 2014. (Cited on page 44.)

[Agrawal *et al.* 2015] Yash Agrawal, Rajeevan Chandel and Rohit Dhiman. *High-performance Current Mode Receiver Design for On-chip VLSI Interconnects.* Advances in Intelligent Systems and Computing, vol. 343, pages 527–536, 01 2015. (Cited on page 44.)

[Ahmed *et al.* 2018] A. B. Ahmed, D. Fujiki, H. Matsutani, M. Koibuchi and H. Amano. *AxNoC: Low-power Approximate Network-on-chips Using Critical-path Isolation.* In IEEE/ACM Int. Symp. on Networks-on-Chip (NOCS), number 6, pages 1–8, Oct. 2018. (Cited on page 53.)

[Akturk *et al.* 2015] Ismail Akturk, Nam Sung Kim and Ulya R Karpuzcu. *Decoupled Control and Data Processing for Approximate Near-Threshold Voltage Computing.* IEEE Micro, vol. 35, no. 4, pages 70–78, 2015. (Cited on page 65.)

[Alexoudi *et al.* 2019] T. Alexoudi, N. Terzenidis, S. Pitris, M. Moralis-Pegios, P. Maniotis, C. Vagionas, C. Mitsolidou, G. Mourgias-Alexandris, G. T. Kanellos, A. Miliou, K. Vyrsokinos and N. Pleros. *Optics in Computing: From Photonic Network-on-Chip to Chip-to-Chip Interconnects and Disintegrated Architectures.* Journal of Lightwave Technology, vol. 37, no. 2, pages 363–379, Jan 2019. (Cited on page 32.)

[Asri *et al.* 2021] Mochamad Asri, Dhairya Malhotra, Jiajun Wang, George Biros, Lizy K. John and Andreas Gerstlauer. *Hardware Accelerator Integration Tradeoffs for High-Performance Computing: A Case Study of GEMM Acceleration in N-Body Methods.* IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 8, pages 2035–2048, 2021. (Cited on page 79.)

[Atabaki *et al.* 2018] Amir H. Atabaki, Sajjad Moazeni, Fabio Pavanello, Hayk Gevorgyan, Jelena Notaros, Luca Alloatti, Mark T. Wade, Chen Sun, Seth A. Kruger, Huaiyu Meng, Kenaish Al Qubaisi, Imbert Wang, Bohan Zhang, Anatol Khilo, Christopher V. Baiocco, Miloš A. Popović, Vladimir M. Stojanović and Rajeev J. Ram. *Integrating photonics with silicon nanoelectronics for the next generation of systems on a chip.* Nature, vol. 556, no. 7701, pages 349 – 354, 2018. (Cited on pages 1 and 21.)

[Bahadori *et al.* 2016a] M. Bahadori, S. Rumley, D. Nikolova and K. Bergman. *Comprehensive Design Space Exploration of Silicon Photonic Interconnects.* Journal of Lightwave Technology, vol. 34, no. 12, June 2016. (Cited on page 28.)

[Bahadori *et al.* 2016b] Meisam Bahadori, Sébastien Rumley, Hasitha Jayatilleka, Kyle Murray, Nicolas A. F. Jaeger, Lukas Chrostowski, Sudip Shekhar and Keren Bergman. *Crosstalk*

Penalty in Microring-Based Silicon Photonic Interconnect Systems. J. Lightwave Technol., vol. 34, no. 17, pages 4043–4052, Sep 2016. (Cited on pages 26 and 37.)

[Bahadori *et al.* 2016c] Meisam Bahadori, Sébastien Rumley, Dessislava Nikolova and Keren Bergman. *Comprehensive Design Space Exploration of Silicon Photonic Interconnects.* Journal of Lightwave Technology, vol. 34, no. 12, pages 2975–2987, 2016. (Cited on page 39.)

[Balasubramanian & Naayagi 2017] P. Balasubramanian and R. T. Naayagi. *Redundant Logic Insertion and Fault Tolerance Improvement in Combinational Circuits.* In Proc. Int. Conf. on Circuits, Syst. and Simul. (ICCSS), pages 6–13, July 2017. (Cited on page 51.)

[Baskaran & Sampson 2021] Saambhavi Baskaran and Jack Sampson. *Evaluation of Tradeoffs in the Design of FPGA Fabrics Using Electrostrictive 2-D FETs.* IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 29, no. 4, pages 691–701, 2021. (Cited on page 82.)

[Bhowmik *et al.* 2019] B. Bhowmik, S. Biswas, J. K. Deka and B. B. Bhattacharya. *A Low-Cost Test Solution for Reliable Communication in Networks-on-Chip.* J. of Electron. Testing, vol. 35, no. 2, pages 215–243, Apr. 2019. (Cited on page 53.)

[Bienia *et al.* 2008] Christian Bienia, Sanjeev Kumar and Kai Li. *PARSEC vs. SPLASH-2: A quantitative comparison of two multithreaded benchmark suites on Chip-Multiprocessors.* In International Symposium on Workload Characterization, pages 47–56, 2008. (Cited on pages 40 and 80.)

[Binkert *et al.* 2011] Nathan Binkert, Bradford Beckmann, Gabriel Black, Steven K. Reinhardt, Ali Saidi, Arkaprava Basu, Joel Hestness, Derek R. Hower, Tushar Krishna, Somayeh Sardashti, Rathijit Sen, Korey Sewell, Muhammad Shoaib, Nilay Vaish, Mark D. Hill and David A. Wood. *The Gem5 Simulator.* SIGARCH Comput. Archit. News, vol. 39, no. 2, page 1–7, aug 2011. (Cited on page 80.)

[Boeuf *et al.* 2013] F. Boeuf, S. Crémer, N. Vulliet, T. Pinguet, A. Mekis, G. Masini, L. Verslegers, P. Sun, A. Ayazi, N.-K. Hon, S. Sahni, Y. Chi, B. Orlando, D. Ristoiu, A. Farcy, F. Leverd, L. Broussous, D. Pelissier-Tanon, C. Richard, L. Pinzelli, R. Beneyton, O. Gourhant, E. Gourvest, Y. Le-Friec, D. Monnier, P. Brun, M. Guillermet, D. Benoit, K. Haxaire, J. R Manouvrier, S. Jan, H. Petiton, J. F. Carpentier, T. Quémerais, C. Durand, D. Gloria, M. Fourel, F. Battegay, Y. Sanchez, E. Batail, F. Baron, P. Delpech, L. Salager, P. De Dobbelaere and B. Sautreuil. *A multi-wavelength 3D-compatible silicon photonics platform on 300mm SOI wafers for 25Gb/s applications.* In 2013 IEEE International Electron Devices Meeting, pages 13.3.1–13.3.4, 2013. (Cited on page 21.)

[Bohr 2018] M. T. Bohr. *Logic Technology Scaling to Continue Moore's Law.* In IEEE Electron Devices Technol. and Manuf. Conf. (EDTM), pages 1–3, Mar. 2018. (Cited on pages 2 and 50.)

[Borkar & Chien 2011] Shekhar Borkar and Andrew A. Chien. *The Future of Microprocessors.* Commun. ACM, vol. 54, no. 5, page 67–77, may 2011. (Cited on page 79.)

[Bosio *et al.* 2019] Alberto Bosio, Paolo Bernardi, Annachiara Ruospo and Ernesto Sanchez. *A Reliability Analysis of a Deep Neural Network.* In 2019 IEEE Latin American Test Symposium (LATS), pages 1–6, 2019. (Cited on page 81.)

[Brant *et al.* 2013] Alexander Brant, Ahmed Abdelhadi, Douglas HH Sim, Shao Lin Tang, Michael Xi Yue and Guy GF Lemieux. *Safe overclocking of tightly coupled CGRAs and processor arrays using razor.* In Proc. IEEE Int. Symp. on Field-Programmable Custom Computing Machines (FCCM), pages 37–44, 2013. (Cited on pages 70 and 71.)

[Carlson *et al.* 2014] Trevor E. Carlson, Wim Heirman, Stijn Eyerman, Ibrahim Hur and Lieven Eeckhout. *An Evaluation of High-Level Mechanistic Core Models.* ACM Transactions on Architecture and Code Optimization (TACO), 2014. (Cited on pages 40 and 80.)

[Chang *et al.* 2008] M. Frank Chang, Jason Cong, Adam Kaplan, Mishali Naik, Glenn Reinman, Eran Socher and Sai-Wang Tam. *CMP network-on-chip overlaid with multi-band RF-interconnect.* In 2008 IEEE 14th International Symposium on High Performance Computer Architecture, pages 191–202, 2008. (Cited on page 81.)

[Chang *et al.* 2019] Y.-C. Chang, C.-S. A. Gong and C.-T. Chiu. *Fault-Tolerant Mesh-Based NoC With Router-Level Redundancy.* J. of Sign. Process. Syst., vol. 92, pages 345–355, Sept. 2019. (Cited on page 51.)

[Charif *et al.* 2020] Amir Charif, Alexandre Coelho, Nacer-Eddine Zergainoh and Michael Nicolaidis. *A Dynamic Sufficient Condition of Deadlock-Freedom for High-Performance Fault-Tolerant Routing in Networks-on-Chips.* IEEE Transactions on Emerging Topics in Computing, vol. 8, no. 3, pages 642–654, 2020. (Cited on page 62.)

[Chen & Joshi 2013] Chao Chen and Ajay Joshi. *Runtime Management of Laser Power in Silicon-Photonic Multibus NoC Architecture.* IEEE Journal of Selected Topics in Quantum Electronics, vol. 19, no. 2, pages 3700713–3700713, 2013. (Cited on page 43.)

[Chen *et al.* 2019] Z. Chen, Y. Zhang, Z. Peng and J. Jiang. *A Deterministic-Path Routing Algorithm for Tolerating Many Faults on Wafer-Level NoC.* In Des. Automat. Test in Europe Conf. Exhib. (DATE), pages 1337–1342, Mar. 2019. (Cited on page 51.)

[Chen *et al.* 2021] Xuanqi Chen, Jun Feng, Jiang Xu, Jiaxu Zhang and Shixi Chen. *Simultaneously Tolerate Thermal and Process Variations Through Indirect Feedback Tuning for Silicon Photonic Networks.* IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 40, no. 7, pages 1409–1422, 2021. (Cited on pages 39 and 45.)

[Chittamuru & Pasricha 2015] Sai Vineel Reddy Chittamuru and Sudeep Pasricha. *Improving crosstalk resilience with wavelength spacing in photonic crossbar-based network-on-chip architectures.* In 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS), pages 1–4, 2015. (Cited on page 24.)

[Choi *et al.* 2018] Wonje Choi, Karthi Duraisamy, Ryan Gary Kim, Janardhan Rao Doppa, Partha Pratim Pande, Diana Marculescu and Radu Marculescu. *On-Chip Communication Network for Efficient Training of Deep Convolutional Networks on Heterogeneous*

Manycore Systems. IEEE Transactions on Computers, vol. 67, no. 5, pages 672–686, 2018. (Cited on page 47.)

[Dally & Towles 2004] W. J. Dally and B. P. Towles. Principles and Practices of Interconnection Networks. Elsevier, 2004. (Cited on pages 50 and 52.)

[Dave *et al.* 2013] Marshnil Dave, Mahavir Jain, Maryam Shojaei Baghini and Dinesh Sharma. *A Variation Tolerant Current-Mode Signaling Scheme for On-Chip Interconnects.* IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 21, no. 2, pages 342–353, 2013. (Cited on page 44.)

[De Dobbelaere *et al.* 2017] P. De Dobbelaere, A. Dahl, A. Mekis, B. Chase, B. Weber, B. Welch, D. Foltz, G. Armijo, G. Masini, G. McGee, G. Wong, J. Balardeta, J. Dotson, J. Schramm, K. Hon, K. Khauv, K. Robertson, K. Stechschulte, K. Yokoyama, L. Planchon, L. Tullgren, M. Eker, M. Mack, M. Peterson, N. Rudnick, P. Milton, P. Sun, R. Bruck, R. Zhou, S. Denton, S. Fath-pour, S. Gloeckner, S. Jackson, S. Pang, S. Sahni, S. Wang, S. Yu, T. Pinguet, Y. De Koninck, Y. Chi and Y. Liang. *Advanced silicon photonics technology platform leveraging a semiconductor supply chain.* In 2017 IEEE International Electron Devices Meeting (IEDM), pages 34.1.1–34.1.4, 2017. (Cited on page 21.)

[DeOrio *et al.* 2012] A. DeOrio, D. Fick, V. Bertacco, D. Sylvester, D. Blaauw, J. Hu and G. Chen. *A Reliable Routing Architecture and Algorithm for NoCs.* IEEE Trans. on Comput.-Aided Des. of Integr. Circuits and Syst. (TCAD), vol. 31, no. 5, pages 726–739, May 2012. (Cited on page 52.)

[Dinechin & Graillat 2017] B. D. Dinechin and A. Graillat. *Feed-Forward Routing for the Wormhole Switching Network-on-Chip of the Kalray MPPA2 Processor.* In Proc. IEEE/ACM Int. Workshop Network-on-Chip Architectures (NoCArc), Oct. 2017. (Cited on page 50.)

[Dong *et al.* 2010] P. Dong, W. Qian, S. Liao, H. Liang, C. C. Kung, N. N. Feng, R. Shafiiha, J. Fong, D. Feng, A. V. Krishnamoorthy and M. Asghari. *Low loss silicon waveguides for application of optical interconnects.* In IEEE Photonics Society Summer Topicals 2010, pages 191–192, July 2010. (Cited on pages 28 and 39.)

[Dubrova 2013] E. Dubrova. Fault-Tolerant Design. Springer-Verlag New York, 2013. (Cited on pages 50 and 51.)

[Duzellier *et al.* 2002] S. Duzellier, S. Bourdarie, R. Velazco, B. Nicolescu and R. Ecoffet. *SEE In-Flight Data For Two Static 32KB Memories on High Earth Orbit.* In IEEE Radiat. Effects Data Workshop, pages 1–6, July 2002. (Cited on page 50.)

[Ebrahimi *et al.* 2013] M. Ebrahimi, M. Daneshtalab, J. Plosila and H. Tenhunen. *Minimal-Path Fault-Tolerant Approach Using Connection-Retaining Structure in Networks-on-Chip.* In IEEE/ACM Int. Symp. on Networks-on-Chip (NOCS), pages 1–8, Apr. 2013. (Cited on page 51.)

[Ernst *et al.* 2004] Dan Ernst, Shidhartha Das, Seokwoo Lee, David Blaauw, Todd Austin, Trevor Mudge, Nam Sung Kim and Krisztián Flautner. *Razor: circuit-level correction of timing errors for low-power operation.* IEEE Micro, vol. 24, no. 6, pages 10–20, 2004. (Cited on page 70.)

[Flich *et al.* 2017] José Flich, Giovanni Agosta, Philipp Ampletzer, David Atienza Alonso, Carlo Brandolese, Etienne Cappe, Alessandro Cilardo, Leon Dragić, Alexandre Dray, Alen Duspara, William Fornaciari, Gerald Guillaume, Ynse Hoornenborg, Arman Iranfar, Mario Kovač, Simone Libutti, Bruno Maitre, José Maria Martínez, Giuseppe Massari, Hrvoje Mlinarić, Ermis Papastefanakis, Tomás Picornell, Igor Piljić, Anna Pupykina, Federico Reghenzani, Isabelle Staub, Rafael Tornero, Marina Zapater and Davide Zoni. *MANGO: Exploring Manycore Architectures for Next-GeneratiOn HPC Systems.* In 2017 Euromicro Conference on Digital System Design (DSD), pages 478–485, 2017. (Cited on page 1.)

[Fu *et al.* 2014] B. Fu, Y. Han, H. Li and X. Li. *ZoneDefense: A Fault-Tolerant Routing for 2-D Meshes Without Virtual Channels.* IEEE Trans. on Very Large Scale Integration (VLSI) Syst., vol. 22, no. 1, pages 113–126, Jan. 2014. (Cited on pages 50 and 51.)

[Fusella & Cilardo 2016] Edoardo Fusella and Alessandro Cilardo. *PhoNoCMap: An application mapping tool for photonic networks-on-chip.* In 2016 Design, Automation Test in Europe Conference Exhibition (DATE), pages 289–292, 2016. (Cited on pages 24 and 31.)

[Gulli & Pal 2017] A. Gulli and S. Pal. Deep Learning With Keras. Packt Publishing Ltd, Apr. 2017. (Cited on page 57.)

[Gunn 2006] C. Gunn. *CMOS Photonics for High-Speed Interconnects.* IEEE Micro, vol. 26, no. 2, pages 58–66, 2006. (Cited on page 21.)

[Hamedani *et al.* 2014] P. K. Hamedani, N. E. Jerger and S. Hessabi. *QuT: A low-power optical Network-on-Chip.* In 2014 Eighth IEEE/ACM International Symposium on Networks-on-Chip (NoCS), pages 80–87, 2014. (Cited on page 32.)

[Han & Orshansky 2013] Jie Han and Michael Orshansky. *Approximate computing: An emerging paradigm for energy-efficient design.* In 18th IEEE European Test Symposium (ETS), pages 1–6, 2013. (Cited on pages 64 and 65.)

[Han *et al.* 2017] M. Han, Y. Han, S. W. Kim, H. Lee and I. Park. *Content-Aware Bit Shuffling for Maximizing PCM Endurance.* ACM Trans. on Des. Automat. Electron. Syst. (TODAES), vol. 22, no. 48, pages 1–26, May 2017. (Cited on page 52.)

[Hosseinabadi & Ansari 2014] Mina Taheri Hosseinabadi and Nirwan Ansari. *Multi-power-level energy saving management for passive optical networks.* IEEE/OSA Journal of Optical Communications and Networking, vol. 6, no. 11, pages 965–973, 2014. (Cited on page 43.)

[Jain *et al.* 2019] A. Jain, V. Laxmi, M. Tripathi, M. S. Gaur and R. Bishnoi. *S2DIO: An Extended Scalable 2D Mesh Network-on-Chip Routing Reconfiguration for Efficient Bypass of Link Failures.* The J. of Supercomput., vol. 75, no. 10, pages 6855–6881, 2019. (Cited on page 51.)

[Kahng & Kang 2012] Andrew B Kahng and Seokhyeong Kang. *Accuracy-configurable adder for approximate arithmetic designs.* In Proc. of the 49th Annual Design Automation Conference, pages 820–825, 2012. (Cited on page 69.)

[Karkar *et al.* 2016] A. Karkar, T. Mak, K. Tong and A. Yakovlev. *A Survey of Emerging Interconnects for On-Chip Efficient Multicast and Broadcast in Many-Cores.* IEEE Circuits and Systems Magazine, vol. 16, no. 1, pages 58–72, 2016. (Cited on pages 1, 79 and 81.)

[Kennedy & Kodi 2017] M. Kennedy and A. K. Kodi. *Laser Pooling: Static and Dynamic Laser Power Allocation for On-Chip Optical Interconnects.* Journal of Lightwave Technology, vol. 35, no. 15, pages 3159–3167, Aug 2017. (Cited on page 28.)

[Khalil *et al.* 2019] K. Khalil, O. Eldash, A. Kumar and M. Bayoumi. *Self-Healing Hardware Systems: A Review.* Microelectron. J., vol. 93, page 104620, 2019. (Cited on page 50.)

[Khichar *et al.* 2017] Jyoti Khichar, Sudhanshu Choudhary and Rohit Mahar. *Fault tolerant dynamic XY-YX routing algorithm for network on-chip architecture.* In 2017 International Conference on Intelligent Computing and Control (I2C2), pages 1–6, 2017. (Cited on pages 51 and 62.)

[Kim & Tiwari 2011] Jaeyoon Kim and Sandip Tiwari. *Inexact computing for ultra low-power nanometer digital circuit design.* In IEEE/ACM International Symposium on Nanoscale Architectures, pages 24–31, 2011. (Cited on page 65.)

[Kim *et al.* 2016] Younghoon Kim, Swagath Venkataramani, Kaushik Roy and Anand Raghunathan. *Designing approximate circuits using clock overgating.* In Proc. of the 53rd Annual Design Automation Conference, page 15, 2016. (Cited on page 65.)

[Kreupl 2013] Franz Kreupl. *The carbon-nanotube computer has arrived.* Nature, vol. 501, no. 7468, pages 495–496, 2013. (Cited on page 82.)

[Krishnan *et al.* 2022] Gokul Krishnan, Sumit K. Mandal, Chaitali Chakrabarti, Jae-Sun Seo, Umit Y. Ogras and Yu Cao. *Impact of On-Chip Interconnect on In-Memory Acceleration of Deep Neural Networks.* J. Emerg. Technol. Comput. Syst., vol. 18, no. 2, dec 2022. (Cited on page 81.)

[Krizhevsky *et al.* 2009] A. Krizhevsky, G. Hinton*et al.* *Learning Multiple Layers of Features From Tiny Images.* Univ. of Toronto, May 2009. (Cited on page 57.)

[Kundu & Chattopadhyay 2014] S. Kundu and S. Chattopadhyay. Network-on-Chip: The Next Generation of System-on-Chip Integration. Taylor & Francis, 2014. (Cited on page 50.)

[Kuroda *et al.* 1998] Tadahiro Kuroda, Kojiro Suzuki*et al.* *Variable supply-voltage scheme for low-power high-speed CMOS digital design.* IEEE Journal of Solid-State Circuits, vol. 33, no. 3, pages 454–462, 1998. (Cited on page 64.)

[Lan *et al.* 2017] F. Lan, R. Wu, C. Zhang, Y. Pan and K. t. T. Cheng. *DLPS: Dynamic laser power scaling for optical Network-on-Chip.* In 2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), pages 726–731, Jan 2017. (Cited on page 43.)

[Le Beux *et al.* 2011] Sébastien Le Beux, Jelena Trajkovic, Ian O'Connor, Gabriela Nicolescu, Guy Bois and Pierre Paulin. *Optical Ring Network-on-Chip (ORNoC): Architecture and design methodology.* In 2011 Design, Automation Test in Europe, pages 1–6, 2011. (Cited on page 24.)

[Le Beux *et al.* 2014] Sébastien Le Beux, Hui Li, Ian O'Connor, Kazem Cheshmi, Xuchen Liu, Jelena Trajkovic and Gabriela Nicolescu. *Chameleon: Channel efficient optical network-on-chip.* In Proceedings of the conference on Design, Automation & Test in Europe, page 304. European Design and Automation Association, 2014. (Cited on pages 1 and 23.)

[Le Sueur & Heiser 2010] Etienne Le Sueur and Gernot Heiser. *Dynamic voltage and frequency scaling: The laws of diminishing returns.* In Proc. of the 2010 intl. conf. on Power aware computing and systems, pages 1–8, 2010. (Cited on page 64.)

[LeCun *et al.* 2015] Yann LeCun, Yoshua Bengio and Geoffrey Hinton. *Deep Learning.* Nature, vol. 521, no. 7553, pages 436–444, 2015. (Cited on page 81.)

[Levine *et al.* 2012] Joshua M Levine, Edward Stott*et al.* *Online measurement of timing in circuits: For health monitoring and dynamic voltage & frequency scaling.* In Proc. IEEE Int. Symp. on Field-Programmable Custom Computing Machines (FCCM), pages 109–116, 2012. (Cited on page 71.)

[Levine *et al.* 2013] Joshua M Levine, Edward Stott, George Constantinides, Peter YK Cheung*et al.* *SMI: Slack measurement insertion for online timing monitoring in FPGAs.* In Proc. Int. Conf. on Field Programmable Logic and Applications (FPL), pages 1–4, 2013. (Cited on page 70.)

[Li *et al.* 2015] Chaofan Li, Wei Luo, Sachin S Sapatnekar and Jiang Hu. *Joint precision optimization and high level synthesis for approximate computing.* In Proc. of the 52nd Annual Design Automation Conference, page 104, 2015. (Cited on page 65.)

[Lingamneni *et al.* 2011] Avinash Lingamneni, Christian Enz*et al.* *Energy parsimonious circuit design through probabilistic pruning.* In Design, Automation & Test in Europe, pages 1–6, 2011. (Cited on page 65.)

[Lingamneni *et al.* 2013] Avinash Lingamneni, Christian Enz, Krishna Palem and Christian Piguet. *Designing energy-efficient arithmetic operators using inexact computing.* Journal of Low Power Electronics, vol. 9, no. 1, pages 141–153, 2013. (Cited on page 66.)

[Liva *et al.* 2016] G. Liva, L. Gaudio, T. Ninacs and T. Jerkovits. *Code Design for Short Blocks: A Survey.* Comput. Res. Repository (CoRR), Oct. 2016. (Cited on pages 50 and 52.)

[Lotfi *et al.* 2019] Atieh Lotfi, Saurabh Hukerikar, Keshav Balasubramanian, Paul Racunas, Nirmal Saxena, Richard Bramley and Yanxiang Huang. *Resiliency of automotive object detection networks on GPU architectures.* In 2019 IEEE International Test Conference (ITC), pages 1–9, 2019. (Cited on page 81.)

[Mahajan *et al.* 2015] D. Mahajan, K. Ramkrishnan, R. Jariwala, A. Yazdanbakhsh, J. Park, B. Thwaites, A. Nagendrakumar, A. Rahimi, H. Esmaeilzadeh and K. Bazargan. *Axilog: Abstractions for Approximate Hardware Design and Reuse.* IEEE Micro, vol. 35, no. 5, pages 16–30, 2015. (Cited on page 32.)

[Mandal *et al.* 2019] Sumit K. Mandal, Raid Ayoub, Michael Kishinevsky and Umit Y. Ogras. *Analytical Performance Models for NoCs with Multiple Priority Traffic Classes.* ACM Trans. Embed. Comput. Syst., vol. 18, no. 5s, oct 2019. (Cited on page 80.)

[Mandal *et al.* 2021] Sumit K. Mandal, Raid Ayoub, Micahel Kishinevsky, Mohammad M. Islam and Umit Y. Ogras. *Analytical Performance Modeling of NoCs under Priority Arbitration and Bursty Traffic.* IEEE Embedded Systems Letters, vol. 13, no. 3, pages 98–101, 2021. (Cited on page 80.)

[Mandorlo *et al.* 2012] Fabien Mandorlo, Pedro Rojo Romeo, Nicolas Olivier, Lydie Ferrier, Régis Orobtchouk, Xavier Letartre, Jean Marc Fedeli and Pierre Viktorovitch. *Controlled Multi-Wavelength Emission in Full CMOS Compatible Micro-Lasers for on Chip Interconnections.* Journal of Lightwave Technology, vol. 30, no. 19, pages 3073–3080, 2012. (Cited on page 32.)

[Martin *et al.* 2012] Milo M. K. Martin, Mark D. Hill and Daniel J. Sorin. *Why On-Chip Cache Coherence is Here to Stay.* Commun. ACM, vol. 55, no. 7, page 78–89, jul 2012. (Cited on page 81.)

[Mittal 2016] Sparsh Mittal. *A Survey of Techniques for Approximate Computing.* ACM Comput. Surv., vol. 48, no. 4, March 2016. (Cited on pages 2 and 32.)

[Mohammed *et al.* 2019] H. J. Mohammed, W. N. Flayyih and F. Z. Rokhani. *Tolerating Permanent Faults in the Input Port of the Network on Chip Router.* J. of Low Power Electron. and Appl., vol. 9, pages 1–11, Feb. 2019. (Cited on pages 51 and 53.)

[Mukherjee & Dhar 2019] A. Mukherjee and A. S. Dhar. *Triple Transistor Based Triple Modular Redundancy With Embedded Voter Circuit.* Microelectron. J., vol. 87, pages 101 – 109, May 2019. (Cited on page 51.)

[Musumeci *et al.* 2019] Francesco Musumeci, Cristina Rottondi, Avishek Nag, Irene Macaluso, Darko Zibar, Marco Ruffini and Massimo Tornatore. *An Overview on Application of Machine Learning Techniques in Optical Networks.* IEEE Communications Surveys Tutorials, vol. 21, no. 2, pages 1383–1408, 2019. (Cited on page 31.)

[Mut 2016] *Single Event Effects Mitigation Techniques Report.* Federal Aviation Admin., William J. Hughes Tech. Center, Feb. 2016. (Cited on pages 50 and 52.)

[Myny *et al.* 2012] Kris Myny, Erik van Veenendaal, Gerwin H. Gelinck, Jan Genoe, Wim Dehaene and Paul Heremans. *An 8-Bit, 40-Instructions-Per-Second Organic Microprocessor on Plastic Foil.* IEEE Journal of Solid-State Circuits, vol. 47, no. 1, pages 284–291, 2012. (Cited on page 82.)

[Naghibi Jouybari & Mohammadi 2014] Hoda Naghibi Jouybari and Karim Mohammadi. *A low overhead, fault tolerant and congestion aware routing algorithm for 3D mesh-based Network-on-Chips.* Microprocessors and Microsystems, vol. 38, no. 8, Part B, pages 991–999, 2014. (Cited on page 62.)

[Nahmias *et al.* 2020] Mitchell A. Nahmias, Thomas Ferreira de Lima, Alexander N. Tait, Hsuan-Tung Peng, Bhavin J. Shastri and Paul R. Prucnal. *Photonic Multiply-Accumulate Operations for Neural Networks.* IEEE Journal of Selected Topics in Quantum Electronics, vol. 26, no. 1, pages 1–18, 2020. (Cited on page 79.)

[Najafi *et al.* 2019] A. Najafi, L. Bamberg, A. Najafi and A. Garcia-Ortiz. *Integer-Value Encoding for Approximate On-Chip Communication.* IEEE Access, vol. 7, pages 179220–179234, Dec. 2019. (Cited on page 52.)

[Narayan *et al.* 2021] Aditya Narayan, Yvain Thonnart, Pascal Vivet and Ayse K. Coskun. *PROWAVES: Proactive Runtime Wavelength Selection for Energy-Efficient Photonic NoCs.* IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 40, no. 10, pages 2156–2169, 2021. (Cited on pages 31 and 47.)

[Neel *et al.* 2015] Brian Neel, Matthew Kennedy and Avinash Kodi. *Dynamic Power Reduction Techniques in On-Chip Photonic Interconnects.* In Proceedings of the 25th Edition on Great Lakes Symposium on VLSI, GLSVLSI '15, page 249–252, New York, NY, USA, 2015. Association for Computing Machinery. (Cited on page 33.)

[Nicolaidis 2015] Michael Nicolaidis. *Double-Sampling Design Paradigm—A Compendium of Architectures.* IEEE Trans. on Device and Materials Reliability, vol. 15, no. 1, pages 10–23, 2015. (Cited on page 70.)

[Nikdast *et al.* 2015] Mahdi Nikdast, Jiang Xu, Luan Huu Kinh Duong, Xiaowen Wu, Xuan Wang, Zhehui Wang, Zhe Wang, Peng Yang, Yaoyao Ye and Qinfen Hao. *Crosstalk Noise in WDM-Based Optical Networks-on-Chip: A Formal Study and Comparison.* IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 23, no. 11, pages 2552–2565, 2015. (Cited on page 24.)

[Oates 2016] Anthony S. Oates. *Interconnect reliability challenges for technology scaling: A circuit focus.* In 2016 IEEE International Interconnect Technology Conference / Advanced Metallization Conference (IITC/AMC), pages 59–59, 2016. (Cited on page 80.)

[Orcutt *et al.* 2011] Jason S. Orcutt, Anatol Khilo, Charles W. Holzwarth, Milos A. Popović, Hanqing Li, Jie Sun, Thomas Bonifield, Randy Hollingsworth, Franz X. Kärtner, Henry I. Smith, Vladimir Stojanović and Rajeev J. Ram. *Nanophotonic integration in state-of-the-art CMOS foundries.* Opt. Express, vol. 19, no. 3, pages 2335–2346, Jan 2011. (Cited on page 62.)

[Pan *et al.* 2009] Yan Pan, Prabhat Kumar, John Kim, Gokhan Memik, Yu Zhang and Alok Choudhary. *Firefly: Illuminating Future Network-on-Chip with Nanophotonics.* In Proceedings of the 36th Annual International Symposium on Computer Architecture, ISCA

'09, page 429–440, New York, NY, USA, 2009. Association for Computing Machinery. (Cited on page 33.)

[Papamichael & Hoe 2012] M. K. Papamichael and J. C. Hoe. *CONNECT: Re-examining Conventional Wisdom for Designing Nocs in the Context of FPGAs.* In ACM/SIGDA Int. Symp. Field Program. Gate Arrays (FPGA), pages 37–46, Feb. 2012. (Cited on page 59.)

[Postman *et al.* 2013] J. Postman, T. Krishna, C. Edmonds, L. Peh and P. Chiang. *SWIFT: A Low-Power Network-On-Chip Implementing the Token Flow Control Router Architecture With Swing-Reduced Interconnects.* IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 21, no. 8, pages 1432–1446, 2013. (Cited on page 32.)

[Radetzki *et al.* 2013] M. Radetzki, C. Feng, X. Zhao and A. Jantsch. *Methods for Fault Tolerance in Networks-on-Chip.* ACM Comput. Surv., vol. 46, no. 8, pages 1–38, July 2013. (Cited on page 50.)

[Radi *et al.* 2021] Bahaa Radi, Mohammadreza Sanadgol Nezami, Mohammad Taherzadeh-Sani, Frederic Nabki, Michaël Ménard and Odile Liboiron-Ladouceur. *A 22-Gb/s Time-Interleaved Low-Power Optical Receiver With a Two-Bit Integrating Front End.* IEEE Journal of Solid-State Circuits, vol. 56, no. 1, pages 310–323, 2021. (Cited on page 37.)

[Reza & Ampadu 2019] M. F. Reza and P. Ampadu. *Approximate Communication Strategies for Energy-Efficient and High Performance NoC: Opportunities and Challenges.* In Proc. ACM Great Lakes Symp. on VLSI (GLSVLSI), pages 399–404, May 2019. (Cited on page 52.)

[Ruospo *et al.* 2020] Annachiara Ruospo, Alberto Bosio, Alessandro Ianne and Ernesto Sanchez. *Evaluating Convolutional Neural Networks Reliability depending on their Data Representation.* In 2020 23rd Euromicro Conference on Digital System Design (DSD), pages 672–679, 2020. (Cited on page 81.)

[Rusu 2010] C. Rusu. *Multi-Level Fault-Tolerance in Networks-on-Chip.* PhD thesis, Inst. Nat. Polytechnique de Grenoble-INPG, 2010. (Cited on page 50.)

[Sampson *et al.* 2011] Adrian Sampson, Werner Dietl, Emily Fortuna, Danushen Gnanapragasam, Luis Ceze and Dan Grossman. *EnerJ: Approximate Data Types for Safe and General Low-power Computation.* SIGPLAN Not., vol. 46, no. 6, pages 164–174, June 2011. (Cited on page 32.)

[Sampson *et al.* 2015] Adrian Sampson, Werner Dietl, Emily Fortuna, Danushen Gnanapragasam, Luis Ceze and Dan Grossman. *ACCEPT: a programmer-Guided compiler Framework for Practical Approximate Computing.* University of Washington Technical Report UW-CSE-15-01, vol. 1, no. 6, June 2015. (Cited on pages 32 and 40.)

[Sánchez-Macián *et al.* 2014] A. Sánchez-Macián, P. Reviriego and J. A. Maestro. *Hamming SEC-DAED and Extended Hamming SEC-DED-TAED Codes Through Selective Shortening and Bit Placement.* IEEE Trans. on Device and Mater. Rel, vol. 14, no. 1, pages 574–576, Mar. 2014. (Cited on page 52.)

[Satya Sai Lakshmi *et al.* 2020] K. A. Satya Sai Lakshmi, A. M. Keerthi, K. M. Sri and M. Vinodhini. *Code With Crosstalk Avoidance and Error Correction for Network on Chip Interconnects.* In Int. Conf. on Trends in Electron. and Inform. (ICOEI), pages 75–79, June 2020. (Cited on page 52.)

[Schaller 1997] R. R. Schaller. *Moore's Law: Past, Present and Future.* IEEE Spectrum, vol. 34, no. 6, pages 52–59, June 1997. (Cited on page 50.)

[Schlachter *et al.* 2015] Jeremy Schlachter, Vincent Camus and Christian Enz. *Near/Sub-Threshold Circuits and Approximate Computing: The Perfect Combination for Ultra-Low-Power Systems.* In IEEE Computer Society Annual Symp. on VLSI, pages 476–480, 2015. (Cited on page 65.)

[Sec 2016] *Space Product Assurance: Techniques for Radiation Effects Mitigation in AASIC and FPGAs Handbook.* Technical report, ESA Requirements and Standards Division, Sept. 2016. (Cited on pages 4, 49, 50 and 80.)

[Sexton 2003] F. W. Sexton. *Destructive Single-Event Effects in Semiconductor Devices and ICs.* IEEE Trans. on Nucl. Sci., vol. 50, no. 3, pages 603–621, June 2003. (Cited on page 50.)

[Shacham *et al.* 2008] Assaf Shacham, Keren Bergman and Luca P. Carloni. *Photonic Networks-on-Chip for Future Generations of Chip Multiprocessors.* IEEE Transactions on Computers, vol. 57, no. 9, pages 1246–1260, 2008. (Cited on pages 79 and 81.)

[Shi *et al.* 2004] Y. Q. Shi, X. M. Zhang, Z.-C. Ni and N. Ansari. *Interleaving for Combating Bursts of Errors.* IEEE Circuits and Syst. Mag., vol. 4, no. 1, pages 29–42, Aug. 2004. (Cited on page 52.)

[Soares *et al.* 2015] LB Soares, Sergio Bampi*et al. Near-threshold computing for very wide frequency scaling: Approximate adders to rescue performance.* In IEEE New Circuits and Systems Conference (NEWCAS), pages 1–4, 2015. (Cited on page 65.)

[Solihin 2015] Yan Solihin. Fundamentals of parallel multicore architecture. Chapman & Hall/CRC, 1st édition, 11 2015. (Cited on page 38.)

[Sosa *et al.* 2018] Joel Ortiz Sosa, Olivier Sentieys and Christian Roland. *A Diversity Scheme to Enhance the Reliability of Wireless NoC in Multipath Channel Environment.* In 2018 Twelfth IEEE/ACM International Symposium on Networks-on-Chip (NOCS), pages 1–8, 2018. (Cited on page 81.)

[Srinivasan *et al.* 2004] J. Srinivasan, S. V. Adve, P. Bose and J. A. Rivers. *The Impact of Technology Scaling on Lifetime Reliability.* In International Conference on Dependable Systems and Networks (DSN), pages 177–186. IEEE, June 2004. (Cited on pages 2, 4, 49, 50 and 80.)

[Stojanović *et al.* 2018] Vladimir Stojanović, Rajeev J. Ram, Milos Popović, Sen Lin, Sajjad Moazeni, Mark Wade, Chen Sun, Luca Alloatti, Amir Atabaki, Fabio Pavanello, Nandish

Mehta and Pavan Bhargava. *Monolithic silicon-photonic platforms in state-of-the-art CMOS SOI processes, Invited.* Opt. Express, vol. 26, no. 10, pages 13106–13121, May 2018. (Cited on page 21.)

[Stott *et al.* 2013] Edward Stott, Zhenyu Guan, Joshua M Levine, Justin SJ Wong and Peter YK Cheung. *Variation and reliability in FPGAs.* IEEE Design & Test, vol. 30, no. 6, pages 50–59, 2013. (Cited on pages 4, 49, 64 and 70.)

[Stott *et al.* 2014] Edward Stott, Joshua M Levine, Peter YK Cheung and Nachiket Kapre. *Timing Fault Detection in FPGA-Based Circuits.* In Proc. IEEE Int. Symp. on Field-Programmable Custom Computing Machines (FCCM), pages 96–99, 2014. (Cited on pages 70, 71 and 74.)

[Sunny *et al.* 2020] Febin Sunny, Asif Mirza, Ishan G. Thakkar, Sudeep Pasricha and Mahdi Nikdast. *LORAX: Loss-Aware Approximations for Energy-Efficient Silicon Photonic Networks-on-Chip.* CoRR, vol. abs/2002.11289, 2020. (Cited on pages 32, 35, 42, 43, 44 and 45.)

[Swarbrick *et al.* 2019] I. Swarbrick, D. Gaitonde, S. Ahmad, B. Gaide and Y. Arbel. *Network-on-Chip Programmable Platform in VersalTM ACAP Architecture.* In Proc. ACM/SIGDA Int. Symp. Field Program. Gate Arrays (FPGA), pages 212–221, 2019. (Cited on page 50.)

[Thakkar *et al.* 2016] Ishan G. Thakkar, S. V. R. Chittamuru and S. Pasricha. *Run-time laser power management in photonic NoCs with on-chip semiconductor optical amplifiers.* 2016 Tenth IEEE/ACM International Symposium on Networks-on-Chip (NOCS), pages 1–4, 2016. (Cited on page 46.)

[Theis & Wong 2017] Thomas N. Theis and H.-S. Philip Wong. *The End of Moore's Law: A New Beginning for Information Technology.* Computing in Science Engineering, vol. 19, no. 2, pages 41–50, 2017. (Cited on page 79.)

[Thomas & Bala 1999] TE Thomas and K Bala. *Multiwavelength optical networks: a layered approach*, 1999. (Cited on page 24.)

[Thonnart *et al.* 2020] Yvain Thonnart, Stéphane Bernabé, Jean Charbonnier, Christian Bernard, David Coriat, César Fuguet, Pierre Tissier, Benoît Charbonnier, Stéphane Malhouitre, Damien Saint-Patrice, Myriam Assous, Aditya Narayan, Ayse Coskun, D. Dutoit and P. Vivet. *POPSTAR: a Robust Modular Optical NoC Architecture for Chiplet-based 3D Integrated Systems.* In 2020 Design, Automation Test in Europe Conference Exhibition (DATE), pages 1456–1461, 2020. (Cited on pages 23 and 46.)

[Torres-Huitzil & Girau 2017] Cesar Torres-Huitzil and Bernard Girau. *Fault and Error Tolerance in Neural Networks: A Review.* IEEE Access, vol. 5, pages 17322–17341, 2017. (Cited on pages 4, 49, 80 and 81.)

[Tziantzioulis *et al.* 2016] G Tziantzioulis, AM Gok*et al.* *Lazy Pipelines: Enhancing quality in approximate computing.* In Design, Automation & Test in Europe Conference & Exhibition (DATE), pages 1381–1386, 2016. (Cited on page 65.)

[Venkataramani *et al.* 2021] Swagath Venkataramani, Vijayalakshmi Srinivasan, Wei Wang, Sanchari Sen, Jintao Zhang, Ankur Agrawal, Monodeep Kar, Shubham Jain, Alberto Mannari, Hoang Tran, Yulong Li, Eri Ogawa, Kazuaki Ishizaki, Hiroshi Inoue, Marcel Schaal, Mauricio Serrano, Jungwook Choi, Xiao Sun, Naigang Wang, Chia-Yu Chen, Allison Allain, James Bonano, Nianzheng Cao, Robert Casatuta, Matthew Cohen, Bruce Fleischer, Michael Guillorn, Howard Haynie, Jinwook Jung, Mingu Kang, Kyu-hyoun Kim, Siyu Koswatta, Saekyu Lee, Martin Lutz, Silvia Mueller, Jinwook Oh, Ashish Ranjan, Zhibin Ren, Scot Rider, Kerstin Schelm, Michael Scheuermann, Joel Silberman, Jie Yang, Vidhi Zalani, Xin Zhang, Ching Zhou, Matt Ziegler, Vinay Shah, Moriyoshi Ohara, Pong-Fei Lu, Brian Curran, Sunil Shukla, Leland Chang and Kailash Gopalakrishnan. *RaPiD: AI Accelerator for Ultra-low Precision Training and Inference.* In 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA), pages 153–166, 2021. (Cited on page 79.)

[Vijaya Bhaskar & Venkatesh 2021] A. Vijaya Bhaskar and T.G. Venkatesh. *Performance analysis of network-on-chip in many-core processors.* Journal of Parallel and Distributed Computing, vol. 147, pages 196–208, 2021. (Cited on page 80.)

[Vivet *et al.* 2021] Pascal Vivet, Eric Guthmuller, Yvain Thonnart, Gael Pillonnet, César Fuguet, Ivan Miro-Panades, Guillaume Moritz, Jean Durupt, Christian Bernard, Didier Varreau, Julian Pontes, Sébastien Thuries, David Coriat, Michel Harrand, Denis Dutoit, Didier Lattard, Lucile Arnaud, Jean Charbonnier, Perceval Coudrain, Arnaud Garnier, Frédéric Berger, Alain Gueugnot, Alain Greiner, Quentin L. Meunier, Alexis Farcy, Alexandre Arriordaz, Séverine Chéramy and Fabien Clermidy. *IntAct: A 96-Core Processor With Six Chiplets 3D-Stacked on an Active Interposer With Distributed Interconnects and Integrated Power Management.* IEEE Journal of Solid-State Circuits, vol. 56, no. 1, pages 79–97, 2021. (Cited on pages 23 and 79.)

[Vlasov 2012] Yurii A. Vlasov. *Silicon CMOS-integrated nano-photonics for computer and data communications beyond 100G.* IEEE Communications Magazine, vol. 50, no. 2, pages s67–s72, 2012. (Cited on page 21.)

[Wachter *et al.* 2017] Stefan Wachter, Dmitry K. Polyushkin, Ole Bethge and Thomas Mueller. *A microprocessor based on a two-dimensional semiconductor.* Nature Communications, vol. 8, no. 1, Apr 2017. (Cited on pages 79 and 82.)

[Wang & Cheng 2019] Yuyang Wang and Kwang-Ting Cheng. *Task Mapping-Assisted Laser Power Scaling for Optical Network-on-Chips.* In 2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pages 1–6, 2019. (Cited on pages 31 and 47.)

[Wang *et al.* 2015] Xiaolu Wang, Huaxi Gu, Yintang Yang, Kun Wang and Qinfen Hao. *RPNoC: A Ring-Based Packet-Switched Optical Network-on-Chip.* IEEE Photonics Technology Letters, vol. 27, no. 4, pages 423–426, 2015. (Cited on pages 1 and 24.)

[Wang *et al.* 2016] Zhehui Wang, Jiang Xu, Peng Yang, Luan Huu Kinh Duong, Zhifei Wang, Xuan Wang, Zhe Wang, Haoran Li and Rafael Kioji Vivas Maeda. *A Holistic Modeling and Analysis of Optical–Electrical Interfaces for Inter/Intra-chip Interconnects.* IEEE

Transactions on Very Large Scale Integration (VLSI) Systems, vol. 24, no. 7, pages 2462–2474, 2016. (Cited on page 33.)

[Wang *et al.* 2020a] L. Wang, Y. Wang and X. Wang. *An Approximate Multiplane Network-on-Chip*. In Des. Automat. Test in Europe Conf. Exhib. (DATE), pages 234–239, Mar. 2020. (Cited on page 52.)

[Wang *et al.* 2020b] Xinyu Wang, Tsan-Ming Choi, Xiaohang Yue, Mengji Zhang and Wanyu Du. *An Effective Optimization Algorithm for Application Mapping in Network-on-Chip Designs*. IEEE Transactions on Industrial Electronics, vol. 67, no. 7, pages 5798–5809, 2020. (Cited on page 47.)

[Werner *et al.* 2016] S. Werner, J. Navaridas and M. Luján. *A Survey on Design Approaches to Circumvent Permanent Faults in Networks-on-Chip*. ACM Comput. Surv., vol. 48, no. 59, pages 1–36, Mar. 2016. (Cited on page 50.)

[Werner *et al.* 2017] Sebastian Werner, Javier Navaridas and Mikel Luján. *A Survey on Optical Network-on-Chip Architectures*. ACM Comput. Surv., vol. 50, no. 6, December 2017. (Cited on page 23.)

[Wolf *et al.* 2008] W. Wolf, A. A. Jerraya and G. Martin. *Multiprocessor System-on-Chip (MP-SoC) Technology*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 27, no. 10, pages 1701–1713, 2008. (Cited on page 1.)

[Wu *et al.* 2017] Chen Wu, Chenchen Deng, Leibo Liu, Jie Han, Jiqiang Chen, Shouyi Yin and Shaojun Wei. *A Multi-Objective Model Oriented Mapping Approach for NoC-based Computing Systems*. IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 3, pages 662–676, 2017. (Cited on page 47.)

[Xia *et al.* 2007] Fengnian Xia, Mike Rooks, Lidija Sekaric and Yurii Vlasov. *Ultra-compact high order ring resonator filters using submicron silicon photonic wires for on-chip optical interconnects*. Opt. Express, vol. 15, no. 19, pages 11934–11941, Sep 2007. (Cited on page 39.)

[Xiang *et al.* 2016] D. Xiang, K. Chakrabarty and H. Fujiwara. *A Unified Test and Fault-Tolerant Multicast Solution For Network-on-Chip Designs*. In IEEE Int. Test Conf. (ITC), pages 1–9, Nov. 2016. (Cited on page 53.)

[Xiao *et al.* 2007] Shijun Xiao, Maroof H. Khan, Hao Shen and Minghao Qi. *Modeling and measurement of losses in silicon-on-insulator resonators and bends*. Opt. Express, vol. 15, no. 17, pages 10553–10561, Aug 2007. (Cited on page 28.)

[Xilinx 2015] Xilinx. *VC707 Evaluation Board for the Virtex-7 FPGA User Guide*, v1.6 édition, April 2015. (Cited on page 73.)

[Xu *et al.* 2005] J. Xu, W. Wolf, J. Henkel and S. Chakradhar. *A Methodology for Design, Modeling, and Analysis of Networks-on-chip*. In IEEE Int. Symp. Circuits and Syst. (ISCAS), volume 2, pages 1778–1781, May 2005. (Cited on page 50.)

[Xu *et al.* 2012] Yi Xu, Jun Yang and Rami Melhem. *Tolerating process variations in nanophotonic on-chip networks.* In 2012 39th Annual International Symposium on Computer Architecture (ISCA), pages 142–152, 2012. (Cited on page 63.)

[Xu *et al.* 2016] Q. Xu, T. Mytkowicz and N. S. Kim. *Approximate Computing: A Survey.* IEEE Design Test, vol. 33, no. 1, pages 8–22, Feb 2016. (Cited on pages 2 and 32.)

[Yang *et al.* 2019] Lei Yang, Weichen Liu, Nan Guan and Nikil Dutt. *Optimal Application Mapping and Scheduling for Network-on-Chips with Computation in STT-RAM Based Router.* IEEE Transactions on Computers, vol. 68, no. 8, pages 1174–1189, 2019. (Cited on page 47.)

[Yu & Ampadu 2008] Q. Yu and P. Ampadu. *Adaptive Error Control for NoC Switch-to-Switch Links in a Variable Noise Environment.* In IEEE Int. Symp. on Defect and Fault Tolerance of VLSI Syst., pages 352–360, 2008. (Cited on page 52.)

[Yu & Ampadu 2011] Q. Yu and P. Ampadu. Transient and Permanent Error Control For Networks-on-Chip. Springer Sci. & Business Media, 2011. (Cited on page 50.)

[Zang & Jue 2000] Hui Zang and Jason P. Jue. *A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks.* Optical Networks Magazine, vol. 1, pages 47–60, 2000. (Cited on page 24.)

[Zeng *et al.* 2018] Mengqi Zeng, Yao Xiao, Jinxin Liu, Kena Yang and Lei Fu. *Exploring Two-Dimensional Materials toward the Next-Generation Circuits: From Monomer Design to Assembly Control.* Chemical Reviews, vol. 118, no. 13, pages 6236–6296, 2018. PMID: 29381058. (Cited on page 82.)

[Zhou & Kodi 2013] L. Zhou and A. K. Kodi. *PROBE: Prediction-based optical bandwidth scaling for energy-efficient NoCs.* In 2013 Seventh IEEE/ACM International Symposium on Networks-on-Chip (NoCS), pages 1–8, April 2013. (Cited on pages 32 and 46.)