



HAL
open science

Contributions to the Test, Fault Tolerance and Approximate Computing of System on a Chip

Mounir Benabdenbi

► **To cite this version:**

Mounir Benabdenbi. Contributions to the Test, Fault Tolerance and Approximate Computing of System on a Chip. Micro and nanotechnologies/Microelectronics. Université Grenoble Alpes, 2023. tel-04254489

HAL Id: tel-04254489

<https://hal.science/tel-04254489>

Submitted on 23 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir l'Habilitation à Diriger des Recherches (HDR) de

L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : EEATS - Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)

Spécialité : Nano électronique et Nano technologies

Unité de recherche : Laboratoire TIMA (Techniques de l'Informatique et de la Microélectronique pour l'Architecture des systèmes intégrés)

Contributions au test, à la tolérance aux fautes et au calcul approché des systèmes sur puce

Contributions to the Test, Fault Tolerance and Approximate Computing of System on a Chip

Présentée par :

Mounir BENABDENBI

Thèse soutenue publiquement le **07/02/2023**, devant le jury composé de :

Giorgio DI NATALE Directeur de Recherches CNRS Laboratoire TIMA (Grenoble)	Président
Vincent BEROLLE Professeur, LCIS (Grenoble)	Rapporteur
Patrick GIRARD Directeur de Recherches CNRS LIRMM (Montpellier)	Rapporteur
Matteo SONZA REORDA Professeur, Politecnico di Torino (Italy)	Rapporteur
Alberto BOSIO Professeur, Ecole Centrale de Lyon	Examineur
Frédéric ROUSSEAU Professeur, Université Grenoble Alpes Laboratoire TIMA (Grenoble)	Examineur



Remerciements

Je voudrai tout d'abord remercier M. Berouille, M. Patrick Girard et M. Matteo Sonza Reorda d'avoir accepté de rapporter ce travail couvrant deux décennies. Merci pour cet honneur et pour le temps qui a été consacré à l'étude de ce document.

Je remercie M. Bosio, M. Di Natale et M. Rousseau pour avoir accepté d'examiner ce travail et de participer aussi au jury de HDR.

Je remercie les différents directeurs de laboratoire qui m'ont accueilli dans leurs locaux : M. Alain Greiner au laboratoire LIP6, Mme Dominique Borrione, M. Salvador Mir et M. Giorgio Di Natale au laboratoire TIMA.

Je remercie M. François Pêcheux, Professeur à l'université Paris Sorbonne, mon compagnon de route et collègue pendant mes années parisiennes. Merci pour ton enthousiasme contagieux en recherche et en enseignement.

Je remercie particulièrement Mme Lorena Anghel et M. Régis Leveugle, pour leur soutien, leur aide et leurs conseils lors de ma mutation de Paris à Grenoble. Grâce à eux l'intégration dans mon nouveau laboratoire a pu se faire naturellement. Je leur exprime ici ma profonde gratitude.

Je remercie aussi M. Leveugle pour ses conseils en tant que superviseur de cette HDR.

Les travaux présentés dans ce document sont le résultat de nombreuses interactions avec différents chercheurs confirmés et en devenir, merci donc aux collègues, aux doctorants et aux nombreux stagiaires avec qui j'ai collaboré, aussi bien au LIP6 qu'à TIMA.

Je n'oublie pas le personnel administratif, au LIP6 et à TIMA, qui a aussi contribué d'une certaine manière à atteindre les travaux présentés ici.

Enfin, je voudrai remercier ma famille, en particulier mon épouse pour son soutien toutes ces années et surtout pour sa patience lors de la rédaction de ce document.

Table of Contents

Preamble	8
----------------	---

Chapter I: Summary

1.	Curriculum Vitae.....	10
1.1	Personal Information.....	10
1.2	Cursus	10
1.3	Research Interests.....	10
1.4	Summary of the hot points.....	10
2.	Summary of scientific activities.....	11
2.1	Preamble.....	11
2.2	Research activities.....	11
2.3	Students	15
2.4	Scientific Projects.....	15
2.5	Teaching activities.....	16
2.6	Dissemination of knowledge	16

Chapter II: Research activities (details)

1.	Context and motivations.....	19
1.1	Design of integrated circuits: impacts of the new technologies and methodologies.....	19
1.2	Test.....	20
1.3	Diagnosis and fault tolerance.....	21
1.4	From fault tolerance to approximate computing	22
2.	System on a Chip Testing.....	24
2.1	Introduction.....	24
2.2	Test Access Mechanisms IEEE 1500 and 1149.1 compliant.....	25
2.3	Software Based Self-Test (SBST).....	28
2.3.1	SBST to test the IPs.....	28
2.3.2	SBST of the embedded microprocessor	31
3.	On-line monitoring for fault tolerance in a manycore SoC.	35
3.1	The database	37
3.2	The monitoring application.....	38
3.3	Outputs.....	40

4.	Test and fault tolerance of Network on a Chip	41
4.1	Network on Chip with deterministic routing for permanent faults tolerance	41
4.1.1	Initialization Stage (HW).....	43
4.1.2	Pre-configuration Stage (SW)	43
4.1.3	Configuration Stage (SW)	44
4.1.4	Outputs.....	45
4.2	Network on Chip with adaptive routing for permanent and transient faults tolerance	45
4.2.1	Router architecture and routing algorithm	46
4.2.2	Network traffic regulation/Congestion management.....	47
4.2.3	Packet recovery scheme	48
4.2.4	Outputs.....	49
4.3	Test and Fault tolerance of 3D Network on Chip.....	49
4.3.1	Off-line Interconnect self-test (IBIST)	50
4.3.2	Enhance the 3D NoC resilience in case of failures.....	51
4.3.3	Single and cross-layer resilience strategies assessment.....	51
4.3.4	ERX a tool for resilience evaluation at design time.....	52
4.3.5	Outputs.....	52
5.	Test and diagnosis of FPGAs.....	53
5.1	Research context	53
5.2	Overview of the novel architecture proposed by FLEXRAS	54
5.3	Test and Diagnosis Methodology.....	57
5.3.1	BIST	57
5.3.2	Logic and interconnect testing (and diagnosis).....	58
5.3.3	Test of the FPGA with fault tolerance.....	58
5.3.4	Developed tools for the CAD flow.....	60
5.3.5	Outputs.....	61
6.	Approximate Computing.....	63
6.1	Introduction.....	63
6.2	Adjustable precision using redundant arithmetic.....	63
6.2.1	Introduction	63
6.2.2	Basic operators.....	64

6.2.3	Outputs.....	65
6.3	Adjustable precision in Floating Point Unit.....	66
6.3.1	Introduction	66
6.3.2	The proposed approach	67
6.3.3	Software implementation: AxQEMU, the impact simulator.....	68
6.3.4	Hardware implementation: The multi-precision VPT FPU.....	69
6.3.5	Self-adaptive precision for iterative algorithms.....	70
7.	Perspectives.....	72
7.1	Preamble.....	72
7.2	Introduction.....	72
7.3	Perspectives on Design, Test and Fault Tolerance of Neuromorphic applications.....	73
7.3.1	Design of AI HW accelerators using emerging Non-Volatile Memories (NVMs).....	73
7.3.2	Test and Fault Tolerance of AI accelerators	73
7.4	Perspectives on Approximate Computing	74
7.4.1	Relationship between Approximate Computing and Fault Tolerance.....	74
7.4.2	Approximate computing for CMOS based architectures.....	75
7.4.3	Approximate computing for NVM based architectures	75
7.5	Design of reliable hardware using non-traditional arithmetic	76

Chapter III: CV (detailed)

1.	Students	80
1.1	Master 2 students.....	80
1.2	PhD Students.....	81
1.3	PostDoc Students	84
2.	Teaching activities.....	84
2.1	Most significant courses	85
2.2	Creation of new courses (from my research activities)	86
2.3	Teaching administration.....	88
3.	Funded Projects and Cooperations.....	89
3.1	Private contracts	89
3.2	Projects in response to selective call for proposal	89
3.3	Cooperations.....	93

4.	Dissemination of knowledge and scientific excellence	94
----	--	----

Chapter IV: Publications

1.	Book chapters	97
2.	Journals	97
3.	Conferences, Symposium, Workshops, Presentations.....	98
4.	Selection of 5 best papers	102

TABLE OF FIGURES

Figure 1 The CAS-BUS Test Access Mechanism.....	26
Figure 2 The CAS BUS TAM architecture with the expansion/decompression mechanism	27
Figure 3 T-Proc, an embedded micro-tester.....	29
Figure 4 Register File testing overview	34
Figure 5 Self adaptability based on application remapping	36
Figure 6 Architecture of a generic cluster in the ADAM project	37
Figure 7 The 2D Mesh multi-cluster MPSOC.....	37
Figure 8 Mapping of both functional and monitoring applications on the SoC	38
Figure 9 Monitoring application task graph	39
Figure 10 A 4x4, DSPIN based, shared memory, MP2SoC architecture	42
Figure 11 The reconfigurable routing algorithm routes packets to bypass the faulty router	44
Figure 12 Router architecture with all the new features.....	46
Figure 13 Packet routing with congestion awareness.....	48
Figure 14 KAF-based BIST for Through Silicon Vias.....	50
Figure 15 The Error Resilience eXploration tool (ERX).....	52
Figure 16 Mesh of clusters FPGA.....	54
Figure 17 Architecture of a Cluster with depopulated crossbars	55
Figure 18 Structure of a crossbar down	55
Figure 19 Configurable logic block and a 4-input LUT	56
Figure 20 Architecture of a Switch Box	56
Figure 21 BIST sessions for a 4x4 clusters MESH	57
Figure 22 The original and the hardened version of the Look Up Table	59
Figure 23 Crossbar Down hardened with two redundant levels.....	60
Figure 24 Hardening through additional Distributed feedback signals.....	60
Figure 25 BIST integrated into the VTR FPGA CAD flow	61
Figure 26 The verification and evaluation Test CAD flow	62
Figure 27 Adder and subtractor using the redundant arithmetic.....	64
Figure 28 Two-Input SBDs parallel adder/subtractor and two versions of a serial adder	64
Figure 29 The proposed approach.....	67

Figure 30 The approach for multiple precisions.....	68
Figure 31 Heat Map obtained using AxQEMU.....	68
Figure 32 The proposed hardware multi precision FPU.....	69
Figure 33 Platform for the project LEARNV.....	87

Preamble

This document is written in order to obtain the diploma of “Habilitation à diriger la recherche” from the University of Grenoble Alpes.

It summarizes the scientific activities done during the last two decades, since the beginning of my PhD thesis at the LIP6 laboratory in Paris in 1998. This document gives an overview of the VLSI issues for which I proposed new solutions, with the help of students I co-supervised and colleagues belonging to different institutes.

This document includes research work done at the University Pierre and Marie Curie (now Paris Sorbonne) from 1998 to 2009 and at Grenoble Institute of Technology (University Grenoble Alpes) from 2009 to 2022.

In the following, only research developments are detailed. However, as an Assistant Professor, I concurrently carried out very substantial teaching and administrative activities. The teaching activities benefit sometimes from the research developments, I hereafter show how the two activities (research and teaching) are entangled.

This document is split in four chapters. The first chapter summarizes all my scientific activities as a short curriculum vitae. The second chapter details my research activities, from what has been done on test, fault tolerance and approximate computing to the perspectives for the coming years. The third chapter corresponds to the curriculum vitae with more details. Finally, the last chapter lists all the papers published so far.

Chapter 1: Summary

1. Curriculum Vitae

1.1 Personal Information

- Name: Mounir BENABDENBI
- Date and Place of Birth: 1st of July 1971, Meknes (Morocco)
- Nationality: French
- Current position: “Maître de Conférences”, Associate Professor at Grenoble Institute of Technology (Grenoble-INP-Université Grenoble Alpes)
- Work address: TIMA laboratory 46 Avenue Felix Viallet 38031 Grenoble, Cedex France
- Work phone: +33 4 76 57 48 08
- Fax : +33 4 76 57 49 81
- Email: Mounir.Benabdenbi@univ-grenoble-alpes.fr

1.2 Cursus

- 1998: Master degree in Electrical Engineering (DEA SyAM) at Université Montpellier 2, France
- 2002: Ph.D. degree in Computer Engineering at Université Pierre et Marie Cury (Paris 6), now Paris-Sorbonne. Title: “Design for Testability of Systems on a Chip (SoC)”
- 2002-2003: Post-doc position, Topic: “Design for Testability of Systems on a Chip (SoC”, Laboratoire d’Informatique de Paris 6 (LIP6)
- 2003-2009: Associate Professor at Université Pierre et Marie Curie, Paris, France.
- From 2009: Associate Professor at Grenoble Institute of Technology (Grenoble-INP-Université Grenoble Alpes), Grenoble France

1.3 Research Interests

- Digital Testing and Design for Testability
- Dependability of Multiprocessors System on a Chip (MPSoCs)
- Memory testing (RAM, caches)
- Test, diagnosis and fault tolerance of 2D and 3D networks-on-chip (NOCs)
- Test and Fault tolerance of FPGA designs
- Approximate Computing
- Emerging non-volatile memories for Artificial Intelligence

1.4 Summary of the hot points

- Co-author of 2 book chapters, 12 articles published in highly selective and reputable international and national journals and 38 articles in reputable international IEEE conferences, symposiums and workshops.
- Co-supervisor of 6 PhD students, 1 post-Doc student and 23 internships at Master 2 level
- Contracts/(co)responsibilities/participation in research projects: one joint ST-LIP6 laboratory, two ANR projects, one European EUREKA CATRENE project, two IDEX IRS Université Grenoble Alpes projects, two actions in the PIA IRT Nano-Elec program.

- Co-head of the Integrated Electronic Systems (SEI) department at Grenoble INP's PHELMA engineering school for six years (45 students per year over two years' cursus).
- Set up and management of two educational projects funded by the PIA IRT Nano-Elec program
- From research to education: creation and management of three new teaching modules at the Master 2 level, two at Paris and one at Grenoble: courses, labs and projects
- Ad hoc expert for the French ANR
- Scientific member of the GDR SoC² (Ex GDR SoC-SIP) of CNRS since 2012: PC member and co-leader at the national level of the Test & Reliability topic of GDR SoC² from 2012 to 2016. Scientific co-animation of the Test & Reliability topic since 2016
- Member of several program committees of IEEE conferences and symposiums.
- Reviewer for international journals and conferences: IEEE journal on Transaction on VLSI, IEEE journal on Transaction on Computers, IEEE Conference in Design And Test in Europe (DATE), ...
- Occasional moderator/session leader in renowned IEEE international conferences

2. Summary of scientific activities

2.1 Preamble

My research activities took place in Paris then in Grenoble. I did my PhD thesis, followed by a one-year postdoctoral position, in the field of SoC Testing at the LIP6 laboratory in Paris. As an Associate Professor, in the same laboratory, I went on working on the same field but I enlarged my scope to the fault tolerance topic. My move to the TIMA laboratory in Grenoble in 2009 gave me the opportunity to capitalize my experience on test and fault tolerance of digital circuits with a focus on Network on Chips. I enriched there my knowledge on FPGA testing and on Approximate Computing. Many of the new proposed approaches involved a combination of hardware and software techniques.

2.2 Research activities

With an ever-increasing integration density of transistors and nanometric dimensions, the design of integrated circuits continues to face many challenges. Among the problems to be solved, the reliability, fault tolerance, resilience and low power aspects have become more and more important over the last two decades. Thus, during my career, my research has been focused on two main topics: fault testing of micro-nano-electronic integrated circuits on the one hand and fault tolerance in these circuits on the other hand.

With my PhD students, interns and colleagues, we have developed methods, architectures and software tools to facilitate the design and testing of integrated circuits and systems (ASIC, SoC and FPGA).

In parallel, I participated in the development of design techniques for massively multiprocessor circuits with the objective of improving the resilience of these circuits. I focused on fault tolerance in Network On Chip (NoC) for 2D and 3D (multiple stacked chips) circuits.

In 2016-2017 I started to make a shift in my research topics, while remaining interested by the reliability of integrated circuits. I am particularly interested in the static and dynamic management of precision in computational results for applications that allow it. This field of research known as "Approximate Computing" aims at judiciously reducing the data precision to drastically reduce the power consumption of circuits and systems.

From 2018 to 2021 I led an open source project which aimed at providing a hardware simulation platform for SoCs, RISC V based, for educational and research purposes.

Finally, I opened recently my interest to new emerging non-volatile memory technologies in the context of reliability, approximate computing and non-Von Neumann architectures for AI.

- **System on a Chip Testing (my postdoc followed by Mathieu Tuna's thesis 2002-2007)**

The complete test (at the time of manufacturing) of the various hardware blocks constituting a system-on-chip can be very long and very expensive (using very expensive industrial tester for example). In order to overcome this problem and to allow also an online self-test after manufacturing, we have proposed to apply the concept of Software Based Self-Test to the testing of SoCs (execution of embedded code to test internal hardware blocks). We have thus proposed different (software) routines to test the small memories embedded in the processors such as the register banks and the cache memories. We then designed (up to the layout), validated and simulated a specific hardware co-processor to be embedded in the SoC, the objective being to transfer part of the functionalities of the industrial tester in the chip to allow: i) its self-test in production and ii) in field test once deployed in its environment.

- **On-line monitoring for fault tolerance in a manycore SoC. (Postdoc of Etienne Faure at LIP6 2009-2010)**

During the last two decades, we have seen a growing number of embedded cores in a single chip, with potential malfunctioning. It became then necessary to find solutions to locate and handle a misbehavior (due to defects, overheating, ...). At the same time, the technology allowed more and more sensors to be embedded within the SoC. Collecting and extracting all the valuable data and processing them to trigger the adequate action was then a challenging issue.

Thus, to prevent the MPSoC from a complete stop once a defect has been identified, practical means had to be found to deactivate, substitute/reconfigure and adapt the application to the downgraded hardware resources.

To allow more flexibility, and enable embedded on-line monitoring and actuating in the field, we have proposed a solution combining hardware and software approaches.

The targeted architectures were Multi Processor SoCs (more than one hundred of processing elements) containing homogeneous functional blocks (all blocks can perform the work requested by the application), but structurally heterogeneous (with different performances and different electrical and timing characteristics). The research project objectives were threefold: non-intrusive online monitoring, online diagnosis/test, and online remapping (through task migration). The main contribution of this work was to make these three steps finely interact through the use of on chip embedded dynamic event databases: local databases of formatted and historized events and a global database describing the whole architecture instant map. Making use of these databases, we provided a distributed SoC self-test and diagnosis approach launched at power on or after a reset.

- **Test and fault tolerance of Network on a Chip (thesis of Z. Zhang 2007-2011, Vladimir Pasca 2009-2012 and Yi Gang 20012-2016)**

Many core architectures usually contain a large number of replicated identical components, such as processor cores and embedded memory banks. The abundance of identical resources makes such architectures able to degrade gracefully, at the condition that faulty components do not hamper the global behavior of the chip.

Unfortunately, the global Network On Chip is the only centralized resource, that cannot easily be duplicated at an affordable cost, and its recovery mechanism in case of failure is not straightforward.

The fault tolerance strategy depends on the NoC topology, the type of targeted faults (permanent or transient) and the routing scheme (deterministic or adaptive). Strategies can be applied at network level, router level and link level. During several years my research activities were then dedicated to propose solutions for permanent and transient faults and for deterministic and adaptive routing.

For issues related to the deterministic routing we have designed and validated a new distributed fault tolerance mechanism targeting 2D Mesh NoCs. This distributed mechanism consists of three steps: 1) detection of the faulty NoC components, 2) deactivation of the faulty components and 3) reconfiguration of the routing function. This mechanism is executed each time the system is rebooted or when the chip is powered on. This mechanism also has the merit of improving the chip's manufacturing yield, by avoiding throwing away the chip because of a defective component.

For issues related to adaptive routing, we have provided methods and architectures for the design, testing and fault tolerance of 3D NoCs, as well as software tools to estimate the reliability of a system architecture during the design phase. We designed also new routing algorithms and implemented a mechanism that allows online adaptation to the traffic load (to avoid network congestion) and to the state of the NoC routers (faulty components). With our dynamic routing, we have shown that the network remains reliable even in the presence of many faulty routers.

- **Test and fault tolerance of FPGAs (thesis of Saif Ur Rehman 20012-2016)**

Beside the manycore reliability studies, I had the opportunity to target another kind of architectures, more regular and also based on a distributed interconnect, namely the Field Programmable Gate Array (FPGA). This device has seen a growing interest and commercial success for many applications, from prototyping to emulation and even as an end product when embedded in critical systems. Through the years, FPGAs had also taken benefit from technology improvements and their test and reliability remained a key issue. Their design, testing and fault tolerance are architecture specific. Most of the architectures and CAD flows belongs to major companies.

In the framework of a French ANR project for a new FPGA architecture we developed a new test and diagnosis approach, BIST-based, aiming the detection and location of faulty components. One major output of this approach is that it allows to remap the application on healthy resources.

In this work we analysed the impact of various architectural parameters on the FPGA testability and evaluate the respective test features in terms of test time and fault coverage. We have extended the proposed strategy and experiments to architectures incorporating defect tolerant techniques at logic and interconnect level. As we cannot count on existing commercial hardware/software flows, we developed a set of tools for automated test and diagnosis implementation and their integration into an academic standard design flow for bitstream generation.

- **Approximate Computing (thesis of Nouredine Ait Said 2018-2021)**

As mentioned in the preamble, my research interest focused since 2017 in the direction of approximated computation in systems-on-chips. This voluntary approximation aims to reduce power consumption which is favorable in the context of IoT and connected objects. Indeed, many applications do not need to have a precise result, it just has to be acceptable (video/image processing, shape recognition in AI, ...)

In my previous works (test and fault tolerance of SoCs) the targeted applications are those requiring both very high reliability and exact outputs. This is mandatory for many critical applications (health, transports, ...) where a high quality of service must be guaranteed. However, these requirements can be relaxed for some classes of application.

AxC is based on the intuitive observation that, while performing exact computation requires a high amount of resources, allowing selective approximation or occasional violation of the specification can provide gains in power consumption, up to one magnitude order. Or, for the same amount of consumption, performances can be enhanced.

This new paradigm is of special interest since in the last decade the rapid rise of the IoT and the increasing energy inefficiency of silicon devices have made the power (and bandwidth) issue a very hot topic.

AxC is often used to trade accuracy of application outputs for a reduction in power or energy consumption.

However, these very last years, the efficiency of the AxC techniques is discussed in the scientific community.

Indeed, the proposed solutions are either too complex to implement (or control) or often dedicated to a SoC sub-block. Moreover, most of the proposed approaches select and implement only one level of accuracy and at design time (after a more or less complex design space exploration for the operands' format).

To circumvent these drawbacks, we introduced the idea of adjustable precision where, depending on the available energy or the computation needs, the computing may be processed with different operand precisions, varying during the execution time.

We explored this idea through two different approaches: one targeting the design of new arithmetic blocks and the other targeting the floating-point hardware/software management.

In the first work we designed arithmetic operators (adders, MACs, ...) using the non-classical redundant arithmetic. These blocks allow the Most Significant Bits (MSBs) to be produced serially in the first place. The precision is thus adjustable overtime, at will, following the application needs.

In the second work we proposed an approach dedicated to the optimization of the floating-point operands' bit width. For that we developed first a new simulator that can be used as a very powerful design space exploration tool to evaluate the impact of the different input sizes on application outputs' precision. Secondly, we designed a multi-precision hardware Floating-point Unit (FPU) to enable Variable Precision in Time (VPT). We showed that the approach is very well suited to application based on stationary iterative algorithms

The both works showed very significant energy consumptions savings.

2.3 Students

As can be seen in table I, during my research activities, i co-supervised one post-doc student (in orange), 6 PhD students (in blue) and 23 master students, not listed here for sake of clarity. The supervising activities are detailed in chapter III.

2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	Current situation		
	Mathieu TUNA																			Senior Designer @ Mentor Graphics		
				Zhen ZHANG																	Senior Designer @ Huawei	
					Etienne FAURE																Senior Designer @ Synopsys	
							Vladimir PASCA														Senior Designer @ Mentor Graphics	
									Yi GANG												Senior Designer @ ALTRAN	
									Saif Ur REHMAN													Senior Designer @ NXP
																		Nouredinne AITSAID			Senior Designer @ SIFIVE	

TABLE 1 Summary of supervised students

2.4 Scientific Projects

During my career I have been involved in several national and international projects. Here follows a summary of the projects, with details given in chapter III.

- 2003-2006 Common Laboratory between LIP6 and STMicroelectronics: scientific leader for the test topic
- 2008-2011 ANR - project ADAM: co-design of the project, co-leader for the LIP6 laboratory and responsible for several deliverables. The project aimed at providing design and reliability solutions for adaptive and dynamic architectures for many core SoCs, through online monitoring
- 2009-2012 European EUREKA CATRENE - project 3DIM3 (3D-TSV Integration for Multimedia and Mobile applications): participation, responsible for several deliverables, presentation for some project reviews. The project objective was to provide new tools methods and architecture (design and reliability for TIMA) solutions to take benefits from the new 3D technology, based on Through Silicon Via integration.
- 2011-2015 ANR project ROBUST FPGA: co-design of the project and scientific leader for TIMA laboratory. The objective of the project was to provide test and fault tolerance solutions for a new hierarchical and mesh-based FPGA.
- 2017-2018, IDEX Grenoble program, exploration and emerging projects - project SysAX: design and management of the project. The project aimed at providing a modeling and simulation platform for approximate computing in SoCs.
- 2017-2018, IDEX Grenoble program, exploration and emerging projects - project CADI: design and management of the project. The project targeted the management of the precision in multi tasked applications running in a many core SoC.

- 2018-2020, PIA IRT Nanoelec program - project "Embedded Vision": co-design and management of the project. The project was targeting educational and research purposes. The objective was to develop an open source co-design (HW/SW) and simulation platform of SoCs based on RISC V processor cores. The SoCs are meant to be mapped on a FPGA to run an image recognition application. In addition, another objective was to develop new labs for master and post master students.

2.5 Teaching activities

As an Associate Professor my teaching activities take a significant part of my day work. In this document, I summarize my past teaching activities including those done in Paris at Université Pierre et Marie Curie (UPMC) and those done in Grenoble at Grenoble INP PHELMA school engineer. I faced diverse students from all levels, from Licence 1 to Master 2. I gave courses on computer engineering and electrical engineering. Most of the courses are directly related to the microelectronic digital design. The list of courses and labs are detailed in chapter III.

Beside the direct teaching, I was strongly involved in the management of the education.

I list here after a summary of my responsibilities in education.

- Co-head of the SEI (Systèmes Electronique Intégrés) department at Grenoble INP PHELMA: pedagogical and administrative responsibility for 6 years
- Creation of three new courses and labs for master students
- Responsible for the Master internships in the SEI department for two years
- Teaching coordinator for 6 years of the module "Hardware architecture of computers" at UPMC involving 6 lecturers for 150 students/year
- Pedagogical coordination for the "design" platform of the CIME Nanotech Grenoble-INP (a common development platform for education and research including all CAD tools and instrumentation) for 6 years: coordination between the CIME staff and the teachers of Phelma w. r. t. their needs on design and simulation tools
- Teaching and coordination of one action of the Nano@School program for 6 years: high school students discovering micro(nano)electronics

2.6 Dissemination of knowledge

Expertise missions

- Adhoc expert/reviewer for the ANR
- Expertise mission at STMicroelectronics Crolles (Grenoble) in 2002: integration of the IEEE 1500 standard in their design flow

Program committee and reviewer

- Reviewer for the following journals: IEEE Transactions on Very Large-Scale Integration (VLSI) Systems, IEEE journal on Transaction on Computers, Elsevier journal on Microprocessors and Microsystems, ACM Transactions on Embedded Computing Systems, Elsevier Microelectronics Reliability Journal, Elsevier Integration, the VLSI Journal, Elsevier Engineering Science and Technology: an International Journal

- Reviewer for the following conferences: IEEE International European Test Symposium (ETS), IEEE International Conference on ReConfigurable Computing, IEEE International Online Test Symposium (IOLTS), IEEE Latin American Test Workshop (LATW) now LATS (Symposium), IEEE Design and Technology of Integrated Systems in Nanoscale Era (DTIS), Symposium on Integrated Circuits and Systems (SBCCI)
- Program committee member: IOLTS (since 2011), LATW (2010-2013), DTIS (since 2019), SBCCI (since 2017), ETS (since 2021)

Jurys

- Jury member for the PhD thesis defense of M. Tu on asynchronous NoC LCIS laboratory in Valence (France) in 2008
- Reviewer of 8 PhD follow up: jury member of the « Comité de Suivi Individuel » (CSI)

GDR SoC²

- Scientific member of the GDR SoC² (Ex GDR SoC-SIP) of CNRS since 2012: PC member and co-chair at the national level of the Test & Reliability topic from 2012 to 2016.
- Scientific co-animation of the Test & Reliability topic since 2016

Organizing Committee

- Local chair of IEEE European Test Symposium 2012 at Annecy
- Finance chair IEEE Online Test Symposium 2014
- Adhoc session moderator for the following conferences and symposia: DATE, ETS, IOLTS, LATS

Chapter II

Research activity: details

1. Context and motivations

1.1 Design of integrated circuits: impact of the new technologies and methodologies

During the period 1997-2022, corresponding to my period of activity in the domain, the semiconductor industry has experienced many (r)evolutions. Hereafter I pinpoint those that triggered challenges for the design and test community.

Technology scaling

The first evolution relates to the size (and shape) of transistors: from 180 nm to 5 or even 3 nm today. This growing increase in integration density has enabled the design of connected embedded systems with high computing performance and low power consumption. Applications taking advantage of these characteristics, with the advent of the Internet, have been rapidly democratized and have had a major societal impact: smartphones and social networks for example. The number of applications that are considered critical boomed and are not anymore limited to the aerospace or transportation domain. How to test and guarantee the in-field good behavior of billion-transistors circuits, at a reasonable cost, was/is an issue.

Design methodologies

The second (r)evolution, induced by the miniaturization of transistors, concerns the new methodology of integrated circuit design. Over time it was no longer possible to quickly design from scratch billion-transistors circuits. From the end of the 1990s, "full custom" design began to be replaced by "design reuse" and hierarchical design. This triggered the advent of SoCs (System on a Chip) and Network on a chip (NoCs), later multi/many core integrated systems on chip began to appear by 2013. To enable design efficiency, the industry had to restructure itself with on one side IP providers and on the other side IP integrators, both types of designers being sometimes in the same company. To enable unambiguous exchanges, a set of standards for the design, integration and testing of proprietary blocks had to be put in place. Testing methods had to evolve to take this new design methodology into account.

Increase in quality costs

In the last decade, Moore's law has been challenged and the rate of transistor miniaturization has slowed down. However, the integration density continues to grow at a steady pace. Progress in terms of transistor types and materials (e.g., FinFET, FDSOI) have contributed to this densification of transistors on silicon. Similarly, the introduction of the third dimension in integrated circuits with vertical links (3D TSV), has also compensated for the progressive obsolescence of Moore's law. We are currently in the era of More Moore, More Than Moore and Beyond CMOS, which reflect a continuous desire to move towards even more integration, particularly with 3D Monolithic circuits (also called 3D sequential) which, according to the IRDS roadmap, should take off in 2024.

This constant integration has allowed great progress in terms of performance but had also required increased attention to everything related to the quality and reliability of the manufactured circuits, especially for circuits intended for critical applications (e.g., aerospace, automotive, health).

Densification exposes the circuit to more defects, defects that can appear at the time of manufacture or later when the circuit is in its final environment. And the more embedded transistors in a single chip, the more test vectors have to be applied at manufacturing time (10x more vectors from a technology to the following).

Thus, concurrently with the densification, the device test costs have boomed and often have corresponded to more than half of the overall chip manufacturing cost.

To compensate for this, the techniques for testing circuits have had to evolve and adapt to allow the developments we have experienced. With the growth of defect density and the need of reliability for critical applications, there were a growing demand for integrating fault-tolerant mechanisms.

My research work is embedded in this general context and is focused on the testing of circuits (production test and in the field test) and on the tolerance to all kind of defects that may occur during or after circuit manufacture (such as Single Event Errors (SEE) and defects due to aging). My main research goal is to improve the robustness of the circuits.

1.2 Test

Defects in an integrated circuit can be modeled by so-called faults and these ones can be classified into three categories according to their behavior: permanent, transient and intermittent.

Beside permanent faults, that permanently damage the chip, transient and intermittent faults can be eliminated by resetting. However, this procedure can take a long time and during this period the system is not available. The application has to be stopped, and all previous results are lost if mechanisms for saving and restoring the system have not been provided, not to mention that these mechanisms also increase the unavailability time of the system as well as its complexity.

As mentioned before, the last twenty years have seen many advances in the design and manufacture of integrated circuits. But to allow this progress, it has been necessary to remove certain obstacles related to the quality and reliability of the manufactured circuit, which will be discussed in this document:

1. The first difficulty is related to the first (r)evolution: increasing density of transistors. How to detect defects (faults) in a context of increasing density, silicon area and sensitivity? How to ensure that the manufactured circuit is testable and tested before being delivered to the customer? The answers have mainly consisted in adapting or improving fault models and existing DFT (Design for Testability) techniques (scan, BIST). However, the transition from one generation of circuit to another has a strong impact on the total cost and in particular on the cost of testing. This can account for 50 to 70% of the total cost for some products. How to prevent the cost of quality from becoming prohibitive? What do we need to sacrifice to maintain a certain level of quality?

2. The second difficulty is related to the second r(e)volution: new design methodology. Since the design of a system is based on the reuse of IPs, the question arises of how to manage the system's block hierarchy from a testing perspective. How to interface IP providers and system integrators for everything related to testing (DFT, access protocol, vectors to apply etc)? Indeed, a major difference with on-board systems is that for a SoC, the whole chip (IPs and glue logic) is tested at once. The industry has partially solved this issue by defining the IEEE 1500 test standard. But we will see in section 2 that it only solves part of the problems encountered.

During my research period, I focused on the test of 2D and 3D SoCs, trying to alleviate the issues related to the new design methodologies and the continuous integration density. I proposed, with my students and colleagues,

new approaches to improve both manufacturing test and online test, taking advantage of the processing capabilities the chip can embed.

A great part of my researches was dedicated to the embedded Network on Chip (NoC), as this component is critical within a SoC. In a manycore architecture, its testability is crucial and require special attention.

For consumer mass-production, SoCs are deployed on dedicated ASIC circuits, however for small series, this kind of design is often mapped on boards embedding one or multiple Field Programmable Gate Arrays (FPGAs). FPGAs are regular structures that have also evolved with the continuous technology improvements. Most of FPGAs designs are proprietary and belongs to companies such as Xilinx or Intel, very few are open source. To understand the test challenges and propose appropriate solutions, I have been involved in a French cooperative project, dedicated to the design and test of a new and open FPGA architecture.

1.3 Diagnosis and fault tolerance

With circuits that can integrate several billion transistors, with complex and heterogeneous architectures, with increasingly sophisticated manufacturing processes, it is not only becoming more difficult to detect defects, but it is also necessary to locate them, off-line or online, either to remove the defective chip, to rectify the manufacturing process, or to activate the mechanisms allowing to tolerate these defects. Thus, the diagnosis of faulty chips has also grown in complexity and dedicated environment has to be set. I focused then on the diagnosis issues at system level, where the term diagnosis is restricted in this document to the identification and localization of a defective gate or block of gate.

One goal of the fault localization is to trigger a corresponding fault tolerance scheme. The adequate fault mechanism depends on several factors such as the criticality of the application. Do we need to tolerate faults AND guarantee the initial functionalities and performances? Or do we accept a certain amount of graceful degradation?

Beside the fault masking and fault containment techniques, fault tolerance is usually implemented with the introduction of redundancy within the chip. We can find four classes of redundancy:

- Hardware redundancy: addition of multiple instances of a gate set and a majority voter. The NMR (N Modular redundancy) illustrates this concept.
- Information redundancy: addition of implicit information in conjunction with the initial data. Error detecting and correcting codes are among the mostly used techniques.
- Time redundancy: a task is repeated several times to guarantee the well behavior of the application.
- Software redundancy: the application is run on different processing elements with different version of the software. This implies a behavior checker monitoring the different software executions.

As SoCs grow in complexity, fault tolerance techniques had been deployed among the different system stack, from the hardware to the application level.

For SoCs, the problem is that once a fault is detected and located, an alert must be launched to the user or to the system, and the appropriate fault tolerant mechanism must be run: deactivation of the faulty block and/or reconfiguration of the hardware and/or the software. These mechanisms must be implemented at an affordable cost and must comply with the criticality specifications.

In addition to the test and diagnosis of SoCs, part of my work has been dedicated to fault tolerance. As we will see, I have proposed a set of mechanisms involving hardware and software solutions. I focused on NoC fault tolerance since this component raises dedicated problems. For example, how can we reconfigure the NoC in case of malfunction of a router? Indeed, depending on the routing scheme, a faulty router can make the whole SoC unusable.

To sum up, the guideline thread for my research work is the triptych Test-Diagnosis-Fault tolerance of SoCs, with a close look up on Network on Chips and on FPGAs.

1.4 From fault tolerance to approximate computing

Beside the fault tolerance studies, during the very last years I put my attention on the new emerging design trends, especially on the Approximate Computing (AxC) paradigm, hereafter introduced.

During the last decades, the semiconductor industry showed many improvements in terms of energy efficiency of produced devices: multi-core designs, (ultra) low power components, etc. However, the energy consumption of computer systems is still growing at an alarming rate. In addition, a large number of applications usually referred as Recognition, Mining and Synthesis (RMS), have emerged and are gaining more and more popularity. From mobile and Internet of Things (IoT) to large-scale data centers, they account now for a significant and growing portion of global computational resources and energy consumption.

It is then mandatory to break the power wall by improving the energy efficiency of silicon devices. This is actually more critical since semiconductor technology continue to scale down to nanometer transistors: better performances but at the price of an increasing energy inefficiency.

Hopefully, many applications, such as the RMS ones, often feature intrinsic error-resilience properties. Indeed, these applications do not have to provide a unique or exact result. For example, in video applications, some types of errors such as locally adaptive quantization can be easily tolerated, as long as the error remains below visual threshold of human visual perception.

Based on these observations, in the last decade, a very promising solution known as "Approximate Computing" (AxC), has gained more and more interest in the scientific community both in the industry and in academia. AxC is based on the intuitive observation that, while performing exact computation requires a high amount of resources, allowing selective approximation or occasional violation of the specifications can provide gains in power consumption, more than one magnitude order. Or, for the same amount of consumption, performances can be enhanced. Various applications of AxC were surveyed such as data analytics, scientific computing, multimedia, signal processing, machine learning and so forth.

Looking at the state of the art, the proposed AxC techniques can be classified as: AxC software (e.g., reduced algorithm iterations, OS task management), AxC architecture (e.g., processors with approximate arithmetic blocs) or AxC circuits (e.g., reduced supply voltage, imprecise logic).

When looking at the potential power savings, the shift towards AxC paradigm is very promising. However, this paradigm is intended to be applied to a subset of non-critical applications.

Thus, depending on the criticality of the application to be implemented, the designer has to deal with two different design strategies with regards to the output precision:

No Error Tolerance

The application is critical and requires exact outputs: the design has to be optimized to allow the application to behave as initially intended.

Error Tolerance

The application is not critical and can tolerate small deviations of its outputs: the design can be optimized to take benefits from the relaxation of the application constraints. The allowed deviations are specified during the design phase. The expected benefit of such strategy is essentially power consumption but as a side effect, improvements on performance and design area can be reached too.

During the last five years I focused on the error tolerance issues and proposed with my colleagues and students several AxC techniques to reduce power consumption in modern SoCs.

In the following more details are given on my research activities. In section 2 I review the proposed approaches to test IPs and the system within an SoC. In section 3 I describe a hardware/software framework enabling the online monitoring of a manycore SoC. Section 4 is dedicated to 2D and 3D NoC testing and fault tolerance. In section 5 I give an overview of the work done to detect and diagnose faults within a new hierarchical FPGA. In section 6 I will briefly describe the proposed approaches to design more power efficient circuits, using the approximate computing paradigm. To complete this chapter, I will give some perspectives on my future research activities.

2. System on a Chip Testing

2.1 Introduction

The increasing complexity of circuits induces many challenges and problems for SoC testing [1]. I describe below those for which we have proposed solutions.

Growing gap between external and internal bandwidth

External bandwidth is defined as the product of the number of pins and the switching frequency of external signals. The internal bandwidth is defined as the product of the number of transistors (or gates) in the circuit and the switching frequency of the internal signals.

These last two parameters increased much more rapidly than the previous ones. For example, during the democratization phase of System-on-Chip (SoC), between 2000 and 2010, there was a three-order-of-magnitude growth in the internal bandwidth while the external bandwidth only increased by a ratio of 10, a trend that was confirmed thereafter. This clearly shows that access to internal elements from the circuit pins was considerably more difficult for SoCs, reducing the expected efficiency of external testing, and increasing the test application times by external equipment to unacceptable proportions.

Moreover, the increasing difference between internal and external switching frequencies makes it impossible for external equipment to apply a test at nominal speed.

Controllability and observability issues

As mentioned before, the design of a system-on-chip involves a massive integration of pre-designed blocks called cores or IPs. These blocks must be delivered to the integrator with the set of test vectors and/or any information related to the testability of the core: initialization values for a logical BIST, March algorithm for the self-test of a memory, fault coverage rate, ...

As there are many companies providing pre-designed blocks, it was necessary to standardize the way this test information is provided, which led to the creation of the IEEE 1450.6 standard called CTL (Core Test Language). Nevertheless, the question still remains "how to convey a test vector from the SoC pins to the inputs and outputs of the core to be tested?" In other words, how to solve the problem of controllability and observability of the inputs/outputs of the core, knowing that the cores can be numerous and come from different suppliers

To harmonize practices, to solve this problem at the core level and to enable CAD tools to automate the generation of testable architectures, the industry created the IEEE 1500 standard in 2005. This standard defines a shell, a test interface, to be added to each SoC core. Based on the IEEE 1149.1 standard, this standard defines an architecture and a set of rules and methods to apply a sequence of test vectors. The two standards are similar on many points: an instruction register, a boundary register equivalent to the boundary scan register, an instruction set, test-specific ports, etc. However, the IEEE 1500 standard differs in some aspects and their interoperability is not straightforward.

Increase in test costs

With the advent of SoCs, the cost of production testing has risen sharply in the first decade. As industrial testers have to be more and more powerful (frequency, accuracy, ...) and provide many functionalities (digital, analog, RF, ...), their cost has strongly increased. The test time has also strongly increased. The number of test vectors

to be applied increased by a factor of 10 when moving to the next technology node, with a bandwidth that did not evolve in the same proportions.

In parallel to the test performed at manufacturing time, the complexity of testing the system-on-chip in its application environment increased. How to perform an efficient and cost-effective online (self)test?

In order to solve the problems mentioned (bandwidth, controllability/observability, manufacturing and in the field test costs), the solutions have generally consisted in transferring as much as possible the functionalities of the external tester to the chip to simplify or accelerate the production test and to allow online testing. My thesis, post-doctoral and early career work is part of this trend, with the particularity of reusing embedded functional software resources to take advantage of the flexibility and genericity they can provide.

2.2 Test Access Mechanisms IEEE 1500 and 1149.1 compliant

My PhD thesis at LIP6 laboratory

During my PhD thesis I focused on the Design for Testability (DFT) of SoCs. Especially, I targeted the bandwidth issue and the test time optimization. The IEEE 1500 standard was in its early development phase and its interoperability with other standards such as IEEE 1149.1 was not solved yet.

The first efforts of the IEEE working group have resulted in the IEEE P1500 standard proposal [2] of a SoC test architecture, which main elements are:

- Test Sources to generate the core test stimuli and Test Sinks to compare the test responses to the expected ones.
- A Test Access Mechanism (TAM) in charge of transporting test data between sources, cores and sinks.
- A Core Test Wrapper which is the interface between the embedded core and the TAM. Through different modes, it provides test functions at the core terminals.

The major effort of the working group focused on the wrapper standardization problem, while source, sink and TAM design is left to the system designer. So new test architectures had to be developed to access core I/Os from the SoC primary I/Os, some cores being deeply embedded in the circuit. TAM architectures can be based on the use of the system bus or on a specific test bus.

The SoC test integrator should be provided with different TAM architectures, in order to select the one that best fit the given core-based SoC. Various criteria may be considered, such as:

- The level or core access and test preparation. The core may be wrapped or not, or include Boundary Scan features easing its accessibility (TAPed cores). It may also include different internal DFT capabilities.
- The complexity of the whole test control.
- The amount and types of test patterns to be applied to the SoC. With the integration density growth comes the increase of the number of patterns needed to test the whole SoC and test patterns at core level must be translated at SoC level (pattern retargeting). This obviously leads to a test time increase.
- The packaging of the SoC. The number of IC pins is sometimes a constraint since it defines the number of pins dedicated to test capabilities.
- The area overhead induced by test feature inclusion.

- The external test equipment that will be used to test the SoC. Depending on the Automated Test Equipment (ATE) capabilities (memory, bandwidth...), the SoC test integrator will select the best TAM architecture for the SoC design.

These constraints led the test engineer to make multiple trade-offs regarding the choice of the appropriate TAM and its characteristics.

Positioning over state of the art

- Some TAMs are designed to access cores with standard or proprietary wrappers ([3], [4], [5]). We have developed a scalable and reconfigurable TAM architecture, named CAS-BUS, which falls into this category. The main benefit of this TAM over the other approaches is that it is dynamically reconfigurable. As others it allows test time optimizations during its design by optimizing the bus width. However, what is different is that it provides test time optimization capability during the test pattern application step by changing the test data routing scheme, from one test session to the other.
- Some others TAMs access TAPed cores, where the Boundary Scan features of the core are used instead of a wrapper to access the core I/Os [5]. But contrarily to our proposal, to our knowledge, none of the existing TAMs deals with both wrapped cores and TAPed cores integrated in the same SoC. Indeed, a system integrator may need to integrate some TAPed cores together with 1500 wrapped cores, when these TAPed cores have primarily been designed as stand-alone chips. This need may arise in order to keep the SoC time-to-market as short as possible, thus avoiding the redesign of already TAPed cores to suppress their boundary scan features and replace them with P1500 wrappers

Thus, I proposed a new TAM allowing the full use of the N bits bandwidth of a dedicated test bus. As can be seen in figure 1 and 2, beside the test bus wires, each core is connected to a configurable TAM based on programmable Core Access Switches (CAS). Each core can be connected or partially connected to the test bus or bypassed.

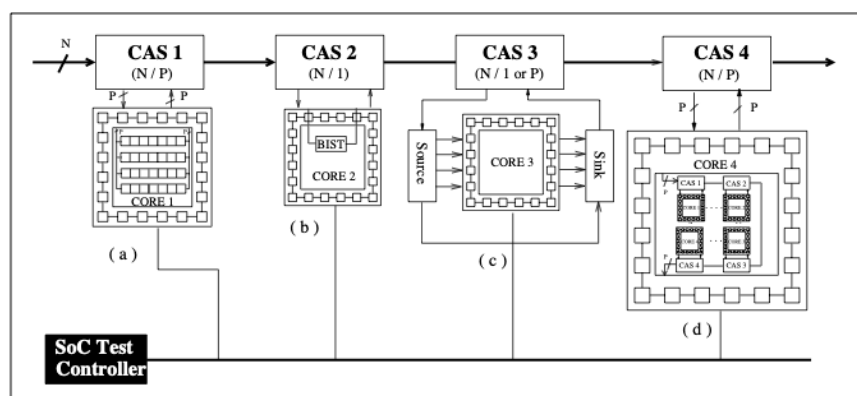


Figure 1 The CAS-BUS Test Access Mechanism

The CAS-BUS TAM is flexible, scalable and dynamically reconfigurable. It allows multiple trade-offs. The architecture is both IEEE 1500-compliant at core level and 1149.1-compliant at SoC level. By correctly configuring the CASes, the test programmer can choose during each test session which core scan chains must be serialized in order to optimize their total length. The goal is to have for each of the N wires the same test length. The control of the global architecture is easy through a simple test access port.

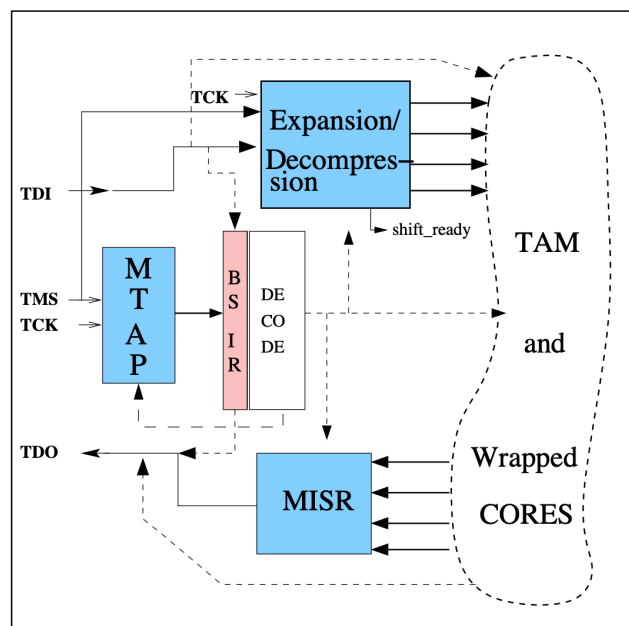


Figure 2 The CAS BUS TAM architecture with the expansion/decompression mechanism

The architecture offers also a complete solution for testing both wrapped and TAPed cores within a SoC, concurrently if needed. With the same test access mechanism, wrapped and TAPed cores can be tested, thanks to a hierarchical test controller.

In addition to the initial architecture I proposed a new lossless compression-decompression/expansion method. Even if it does not provide spectacular compression performances with correlated test patterns like Huffman/similar or run-length coding methods, it is very useful with non-correlated test patterns.

Compression degree varies between 33% (minimum average) and 66% (maximum) when compared to simple 1 to N expansion. Particularly, we proposed this method to enhance test performances in term of test time when expanding a test serial input to more than one SoC internal test chain. A simple modification has been introduced in the IEEE 1149.1 TAP controlling the CAS-BUS architecture. The modified TAP (MTAP) behaves exactly as a classic TAP, except when it is an expansion/decompression process. The method has been tested on the real test patterns of a circuit made in the LIP6 laboratory (PCIDDC: a Network Interface Component) and 35% of compression degree has been reached. The use of other compression methods was studied for the same patterns, but due to their feeble correlation, compression degrees were not significant.

Independent and modular test architecture generators have been developed in the C language. Depending on the SoC features and constraints, the user can generate the appropriate TAM architecture in synthesizable format. These pieces of software make available a tool-box environment for automatic design of plug-and-play components, thus easing the design of SoC test architectures based on the CAS-BUS TAM and its developed instances.

The different components had been synthesized and validated through gate level simulations. Using the ITC99 and ITC2002 benchmarks we showed that the compression ratio was in line with the theory.

This work has been done with the help of Walid Maroufi, postdoctoral researcher at LIP6, Le Yin Keu , in the framework of his DEA internship (Master 2), and Zakaryae Bekkouri and Gilles Richard, in the framework of their final undergraduate study project (Master 1).

The corresponding results have been published in a [B01], [J01], [C01], [C02], [S01] and [W01], referred in the last section of this document.

2.3 Software Based Self-Test (SBST)

My Postdoc followed by Mathieu TUNA PhD thesis at LIP6

With the growing demand for in the field testing, especially for safety critical applications, a new test type has emerged in the early 2000's, the Software Based Self-Test (SBST).

Software based testing implies the presence of a microprocessor to execute a program. The targeted circuits are therefore essentially systems-on-chip (SoC) and microcontrollers, which are now present in most everyday objects for more or less critical applications.

The software based self-test is a functional test that aims to detect structural defects present either within the microprocessor or within the peripheral components to which it is connected.

The test program is a series of specific routines written in assembler. This level of programming is used because it allows with the smallest functional granularity (bit level) to control and observe the different memory elements of the circuit. These routines are therefore finely targeted to detect certain faults present in the microprocessor and in the memory mapped blocks: embedded memories and peripherals.

SBST has many advantages: it allows on site self-testing, as functional resources are used for testing, it can alleviate the need of expensive dedicated DFT features and finally as it is software based, it is very flexible: the test program can be modified even after the chip is delivered to the customer.

However, SBST is often associated with BIST (Built In Self-Test) hardware blocks to enhance the test fault coverage.

During my postdoctoral research and during the PhD thesis of Mathieu Tuna, I used the SBST approach to empower the CAS BUS TAM architecture developed during my PhD thesis. The goal was to define an architecture that is compliant with the IEEE 1500 standard and that can be used both at manufacturing time and when the circuit is shipped to the customer.

In this context we developed two approaches, one to test the embedded IPs and one to test the microprocessor and its caches.

2.3.1 SBST to test the IPs

The first proposed SBST approach is based on the use of a dedicated test co-processor, driven by the embedded microprocessor, to test the different IP cores, at manufacturing time or later when the chip is running in its final environment. The idea is to transfer ATE functionalities within the SoC, at the cost of area overhead.

Positioning over state of the art

SBST to test microprocessors [6] [7] [8], system interconnects [9] [10] and memories [11], [12] have been well studied and convenient SBST solutions have been provided. However, there were a lack of solutions to test other

2.3.1.2 Micro-tester architecture

Besides this SoC scheme, a single new hardware component is introduced, entirely dedicated to SoC testing: the micro-tester. The micro-tester is a processor with two interfaces. On one side, it is connected to the system interconnect. As an initiator it can directly address the external RAM and thus, read test programs stored in. Each tested IP cores has its own test program, available in a format called HTC, specific to this micro-tester. As a memory-mapped target, it receives commands emitted by the GPP. On the other side, it is a IEEE 1500 test controller and a pattern delivery TAM for wrapped cores.

As can be seen in figure 3 the micro-tester includes two major kinds of components namely a Prefetch-Buffer (PB) and TPIUs (standing for Test Program Interpreter Unit). The TPIU is the component that executes the HTC test program of the IP core. There is one dedicated TPIU per tested IP core. The role defined for the PB is to provide an interface between the system interconnect and the TPIUs.

2.3.1.3 Test programs

The test is performed by the GPP/micro-tester pair. The micro-tester is in charge to process test programs. Each tested IP has its own HTC test program. Each HTC file is stored in the external memory. HTC test programs are fetched by the Prefetch-Buffer and executed by a TPIU itself. They contain scan data (test patterns) as well as test control informations. Test control consists on (among others) controlling the scan shifting, applying test patterns and capturing test responses. Test responses are shifted into the TPIU and compacted by the way of an internal MISR. Since the compaction capability can be turned off, two modes are available: the Go/NoGo mode (compaction is turned on) and the Diagnosis mode (compaction is turned off). As one TPIU is dedicated to one IP core, the micro-tester enables the concurrent testing of cores in order to minimize the test application time.

The Master Test Program executed by the GPP supervises the execution of the HTC programs. The MTP can be written in assembler or high-level languages such as C. It is (cross-) compiled and the resulting binary is loaded into an internal memory or can be stored in the external memory. Easiness of writing in high-level languages allows designing a Master Test Program as complex as desired: smart test plan scheduling (considering IPs' power consumption constraints for example), restarting test program on special part, etc. However, this test program should, at least, start the test of the IP cores, wait for interruptions sent by TPIUs, collects test informations (test passed/failed etc.) and finally stores these informations out from the chip.

2.3.1.4 At speed testing

Testing for delay faults (transition faults or path delay faults) requires the application of at speed vectors. In the proposed approach, the micro-tester drives standard wrappers (IEEE 1500) and deals with deterministic test patterns. Thus, the micro-tester handles the two standard types of two-pattern delay tests that are generated by ATPG: Launch-On-Last-Shift (LOLS) and Launch-On-Capture (LOC).

A set of software tools were designed to generate the test hardware features and to generate the HTC programs from the test patterns available in the STIL standard format.

2.3.1.5 Outputs

Several ITC'02 benchmarks were used to evaluate the methodology on the test application time and on the test data volume. The results are compared to those of a dedicated classical bus-based 32-bits TAM driven by an external ATE.

Our approach requires two to three times more cycles to achieve the complete test and introduce a 100 % overhead in data volume (instructions + test patterns), as expected. However, the test program is executed on-chip at SoC speed and the test data is stored in a cheap external memory.

The test process is then not bound to the tester frequency, accuracy and memory capabilities, allowing the use of low cost ATEs at manufacturing time and allows also on the field testing.

A VHDL model of the micro-tester has been realized. Synopsys tools were used to synthesize the model. The model used for the layout generation contains a Prefetch-Buffer with eight TPIUs. Thus, it can test eight different wrapped IP cores. The layout has been generated using the Silicon Ensemble CAD tools with a 0.13 um standard cell library. The overall micro-tester area is 0,53mm. Since it can test eight different IP cores, the additional area to test one IP is thus 0,066 mm. As a reference, in the same technology, the area of a MIPS R3000 and its caches is 2,5 mm. For cores of such size the test area overhead is about 4% which is acceptable.

This work was initiated during my postdoctoral research and completed with Mathieu Tuna during his PhD thesis. The results have been published in [C03], [C04], [W02], [C05], [S04] and [S05]

2.3.2 SBST of the embedded microprocessor

Beside the test of the IP cores, the microprocessor must be tested, either with the classic scan based external test methods or with the SBST approach. We decided to apply the SBST methodology to evaluate its feasibility and its advantages and drawbacks.

Positioning over state of the art

The existing SBST strategies ([6] [7] [8]) for the microprocessor test in the early 2000's was component oriented and only consider the microprocessor logic gates netlist as a whole. Therefore, the efficiency of each method is measured according to a fault coverage targeting only the Stuck-At fault model. However, as some microprocessor components such as Register Files or Memory Caches contain SRAM memory cell arrays, specific tests that consider appropriate fault models must be developed. Using only the Stuck-At fault model leads to consider only 50% of all the faults occurring in memory devices.

Moreover, even though memory BIST is very useful to test memories, the area overhead for small memories (few Kbytes) is not negligible. SBST can then be an attractive strategy to test these small memories, since no area overhead is added.

Thus, in order to achieve a microprocessor test of high quality, we have presented some methodologies to develop self-test programs for Register Files and Memory Caches relying on specific memory tests: the March tests. We first tackled the self-test program development implementing March algorithms for the Register File. Then, methodologies to test memory cell arrays of the Memory Cache has been proposed.

2.3.2.1 Register File testing

A RISC-architecture processor uses a large number of general-purpose registers (GPRs). Those GPRs are grouped together in one component: the Register File (RF). De facto, the Register File of the processor is one of the largest components and has an inherent regularity. Most RFs of pipelined RISC processor cores have three ports, one for writing, two for reading. If we consider a 32-bit architecture microprocessor, the general case is a RF including 32 registers of 32-bits wide. Therefore, this RF contains 1024 memory elements accessed for a read

or a write through a 32-bit word. In order to shrink size and consumption, the RF is designed as a multi-port memory.

The self-test program must implement a March algorithm in order to detect any type of faults. Globally, the testing process consists in filling the entire RF (in increasing or decreasing order) with a Data Background (DB) and reading in the RF (also in increasing or decreasing order) the expected data background. Any programs need some registers to store informations such as variables, counters, addresses, function arguments or the data backgrounds. Thus, the RF must be divided into two parts: a set of registers called the reserved area where the informations needed by the programs are stored. This part (called part-B) does not undergo the test, whereas, the other part (called part-A), undergo the test. At the end of the test of the part-A, we switch these two parts to test the part-B not tested yet, and the part-A becomes the reserved area. Thus, the test of the entire RF is done in two steps.

We have shown that the bi-partitioning of the register bank does not allow to detect some address and coupling faults, especially coupling faults between two cells belonging to different parts. To overcome this issue, we have proposed to tri-partition the register bank and to perform the test in three steps. At each stage, two parts are tested at the same time, the last one being used as a reserved area.

2.3.2.2 Memory Caches testing

There are no instructions allowing to access directly the memory cache. The only strategy to test the cache is by side-effect. Since the cache consists in copies of portions of the main memory, the idea is to execute by the microprocessor a carefully crafted program stored in the main memory. Reading and writing in a specific room in the memory cache is done when the microprocessor accesses a word in the main memory at a specific address, for instructions as well as for datas, using the hit and miss mechanism. Thus, if the instructions and the datas of the program are precisely stored in the main memory, the execution of this program can implement by side-effect a March test on the memory cache arrays. However, since the data cache and the instruction cache are independent and differ from their contents as well as their functionalities, different strategies must be applied to test each of them.

Memory Caches self-test programs must be tailored according to the characteristics of the memory cache. Thus, we have proposed the implementation of self-test programs for a memory cache with the following characteristics: small and simple memory cache containing 1 KBytes of memory. It is a direct-mapped cache array (i.e. each entry in main memory can go in just one place in the cache). The data cache implements a write-through policy (i.e. the information is written to both the block in the cache and to the block in the main memory).

Data Cache

For the data cache the write and read of data backgrounds are respectively implemented by triggering a miss and a hit. The challenge is to initialize the main memory with the good data backgrounds to be stored at the right addresses. This mapping depends on the data cache architecture (number of blocks and block size).

Instruction Cache

The test of the ICache is more complex than the DCache since there is less degrees of freedom in the ICache. Indeed, this cache array contains exclusively instructions executed by the microprocessor. Thus, to manually craft a test program, three different constraints have to be considered:

1. Observability

In the DCache we can read a word and compare it to the expected one. In the ICache this operation is not possible. The ICache poses an observability problem. When the microprocessor read an instruction from the ICache this instruction is immediately executed. Thus, the only observable point is the result of the executed instruction. If the instruction is altered by its traveling through the ICache, the error can occur in different instruction fields: opcode, register coding bits and immediate bits. Thus, the instruction must be carefully chosen in order to guarantee that the altered executed instruction gives another result than the expected one. In order to guarantee that the instruction is the exactly expected one, we have to set an execution context that guarantee the observability. If the executed instruction differs from the one stored in the main memory (differs from at least one bit everywhere in the instruction) the execution of this erroneous instruction lead to an erroneous final context. An execution context can be the register file contents. Thus, with the execution context we can guarantee that an executed instruction is erroneous or not.

2. Instruction set Architecture

In the ICache the instructions are the data backgrounds (DBs). Depending on the number of opcodes available, in RISC architecture this number is limited (by definition), the realization of a DB and the corresponding inverted DB is in general impossible. Thus, in order to realize a March algorithm, we must divide the 32-bit DB into partial DB (pDB). We have to realize the different couple of pDBs as needed to cover the whole 32 bits of the instruction.

3. Operation Sequence

The execution of a program is sequential. A block of instruction is loaded in the ICache, the first instruction is executed followed by the next one and so forth. This sequential execution imposes the following March element : “(wa, ra)” where a is the instruction and thus the DB. According to this constraint we have chosen to build a March algorithm we called March IC (Instruction Cache): $\{\uparrow(wa, ra); \uparrow(\bar{w}\bar{a}, \bar{r}\bar{a}); \uparrow(wa, ra)\}$. According to the literature this sequence ensures the detection of all stuck-at faults as well as all transition faults.

2.3.2.3 Outputs

In order to validate our method and to make the corresponding measurements, a simulation platform in SystemC has been realized (see figure 4). The simulation is accurate to the cycle and to the bit, in order to measure the number of cycles required to execute the test program, as well as its exact size in memory. The microprocessor used was a MIPS R3000 with a pipeline of 5 stages. The microprocessor is connected to a cache memory. An interconnect, more precisely a cross-bar, respecting the VCI standard is used to connect the microprocessor and its cache with a memory. The latter contains the binary which comes from the assembly of the test program. A RAM component is used as a main memory, storing self-test programs. The different SystemC components come from the SOCLIB library. A set of tools has been developed to take as an input any March test, with the corresponding data backgrounds, to generate the appropriate assembly code to be processed. The RAMSES march test simulator (coming from Tsing Hua University Taiwan) is used to process the trace files and evaluate the achieved fault coverage for the different fault types.

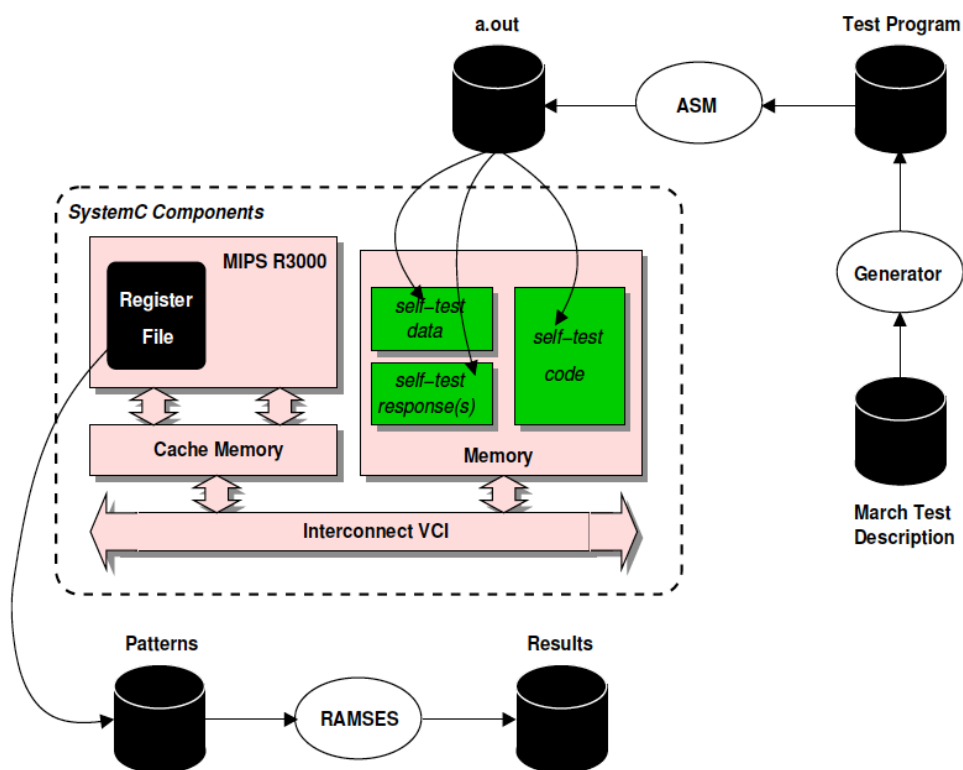


Figure 4 Register File testing overview

Several March tests were implemented for each component (RF and memory caches) leading to different self-test programs. For the RF test as well as for the DCache test, the MATS, MATS++, MARCH X and MARCH C- have been implemented. Since the generation of the March tests for the components is systematic, a set of tools has been developed to take as an input any March test, with the corresponding data backgrounds, to generate the appropriate assembly code to be processed. Whereas the self-test program implementing the March IC algorithm that test the ICache was hand crafted.

The results show that the proposed development methodologies lead to small-sized self-test programs, while a high fault coverage for numerous fault models is achieved.

For the register file, generated self-test programs are smalls, at the range of 1KB. The test execution time is also very small, few thousand cycles, as for the state-of-the-art strategies but here considering all type of memory faults.

For the DCache, self-test program sizes are also very small on the range of few hundred bytes. The test execution times are however much longer than for the register file and depends on the Dcache size (about 400 Kcycles for a 8Kbits Dcache).

For the Dcache and the Icache we compared the test time with a classical hardware-based memory BIST. It results that our approach consumes 6 times more cycles.

For the Icache, the test program size is not negligible, about 9 times the cache size.

At manufacturing time, the test program size is the key-point to reduce the global test application time. Indeed, the self-test program is first downloaded into the embedded memory from the external test equipment, and then, it is executed at the SoC operating frequency. The key issue is that the download phase usually predominates the overall test application time. For an on-site test, this is less an issue.

This work has been done in collaboration with Mathieu Tuna during his PhD thesis and with Olivier Garcia during his Master 2 internship. More detailed results and discussions have been published in [W04] and [W05].

3. On-line monitoring for fault tolerance in a manycore SoC.

Postdoc of Etienne Faure at LIP6

With the growing number of embedded cores, with potential malfunctioning, it became necessary to locate and handle a misbehavior. At the same time, the technology allowed more and more sensors to be embedded within the SoC. Collecting/extracting all the valuable data and processing them to trigger the adequate action was then a challenging issue. To avoid the wide spreading of too many dedicated solutions, IEEE proposed the creation of the 1687 standard [13] (also called IJTAG for Instrument JTAG) to set the hardware architecture to access/control and observe the different embedded registers (scan registers, sensors, ..., all called instruments). However, up to now, the standard only defines the hardware architecture and its controlling protocol. The management of the IJTAG infrastructure can be done by dedicated engines, embedded or external (such as an external ATE). It does not set how to process or use the data in higher software layers.

Positioning over state of the art

Many approaches providing adaptive performance and self-healing capabilities can be found in the literature such as [14-17]. Reliability is achieved thanks to runtime diagnosis/monitoring and online repairing based on the use of DVFS techniques and spare components replacement. However most of the proposals are ad-hoc solutions depending on the embedded processors and sensors, and rely on the use of centralized Diagnostic and Adaptivity Processor unit (DAP). These approaches offer then poor scalability and genericity. Moreover, they focus on a core or a small set of cores and did not offer online testing and repair solutions for more complex chips such as MPSoCs (Massively Parallel Multiprocessor systems on chip), at a global level.

Finally, the proposed solutions cannot take benefits of the IJTAG infrastructure since they are not compliant with this new standard (a proposal at this time period). Thus, they don't allow complex decision computing using values coming both from hardware sensors and software counters.

For that, in addition to the monitoring hardware/software infrastructure, we need to implement and embed a structured database storing the relevant monitoring informations and their history. The database we developed pertain to the class of In-Memory Database System (IMDS). Its functionalities take inspiration from the principles elaborated by known IMDS like BerkeleyDB [18], or Metakit [19]. Such IMDS fill the gap between flat-file (containing only one table) and full-fledged relational database (containing many tables). However, these IMDB have a too big memory footprint and provide too many features for our purpose. Thus, we implemented a simpler and lighter solution.

To prevent the MPSoC from a complete stop once a defect has been identified, practical means had to be found to deactivate, substitute and adapt the software application graph to the downgraded hardware resources.

To allow more flexibility, and above all, enable embedded on-line monitoring and actuating in the field, we proposed a solution combining hardware and software approaches. This work has been done in the framework of the ADAM ANR project (see project description in chapter 3 section 3). ADAM stands for Adaptive and Dynamic Architecture for MPSoC.

ADAM was a research project joining researchers belonging to LIP6, LIRMM and CEA LETI laboratories. It addressed the topic of MPSoC self-adaptability to altered operating conditions, like loss of performance, appearance of faulty hardware resources, and temperature overhead (see figure 5). The architectures targeted by ADAM were MPSoC (more than one hundred of processing elements) containing homogeneous functional blocks (all blocks can perform the work requested by the application), but structurally heterogeneous (with different and process dependant performances, electrical and timing characteristics and reliability). Considering a 2D mesh hardware architecture with a Network on Chip interconnecting clusters containing processing elements and local memories, the project objectives were threefold: non-intrusive online monitoring, online diagnosis/test, and online remapping. The main contribution of the ADAM project was to make these three steps finely interact through the use of on chip dynamic event databases: local databases of formatted/historized events and a global database of architecture instant maps, or architecture audited views classified by event types.

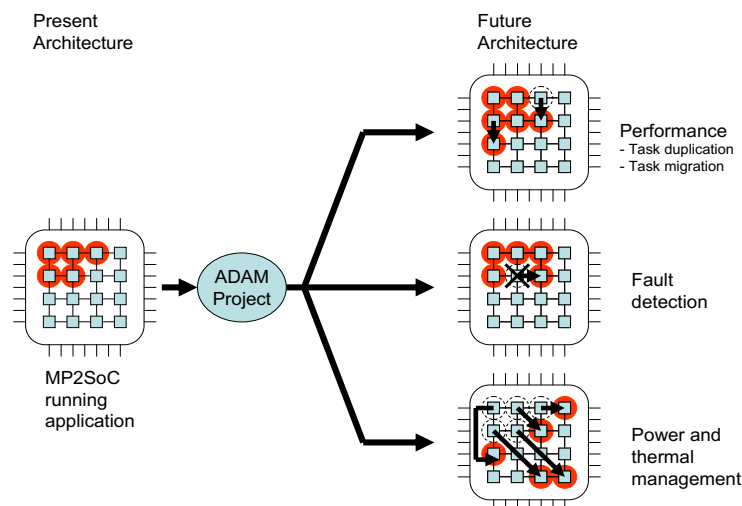


Figure 5 Self adaptability based on application remapping

Self-adaptability can be seen as a closed-loop monitoring problem. Monitoring of a system can be defined as the tuple (observation, diagnosis, reaction). Observation consists in retrieving raw response data corresponding to physical phenomena from the software and hardware sensors distributed over all the system, diagnosis corresponds to their analysis and comparison with expected values, and reaction can be defined as the activation

of some control commands to make the system react appropriately. According to the nature of the measured values, reaction can either be immediate (i.e. when a critical threshold is crossed) or the consequence of a thorough analysis on the evolution of a phenomenon integrated in time. In the first case, an emergency protocol is used, and the accumulation of the previous values of the measures is not needed. In the second case, it is likely that a history of the measured values should be maintained somewhere so that subsequent computations (mean value, max, min, time integration) can be performed.

Thus, it is mandatory for MP2SoC architectures:

- to be as reactive as possible to events that can eventually lead to the destruction of the chip.
- to keep an accurate vision of the evolution of these measures with respect to time. In this particular case, a memory containing measured values is needed, as well as appropriate means to store, handle, and retrieve these accumulated data as simply as possible.

The first point can be realized by hardware sensing and hardware actuating. However, to address the second point, we had to develop an in-memory database engine that fulfills these needs, as well as its associated API (Application Program Interface).

In this project I contributed to the specifications and development of the database and the monitoring infrastructure, with the help of Francois Pêcheux, associate Professor at LIP6 and Etienne Faure postdoctoral researcher attached to the project at LIP6 laboratory. I briefly summarize hereafter the monitoring and database contributions.

The targeted architecture can be seen in figure 6 and 7

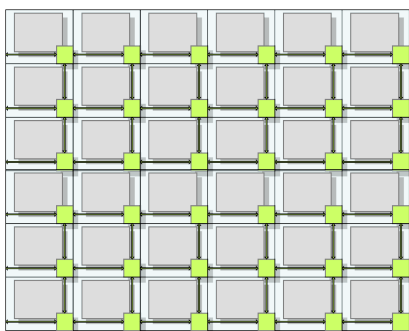


Figure 7 Architecture of a generic cluster in the ADAM project

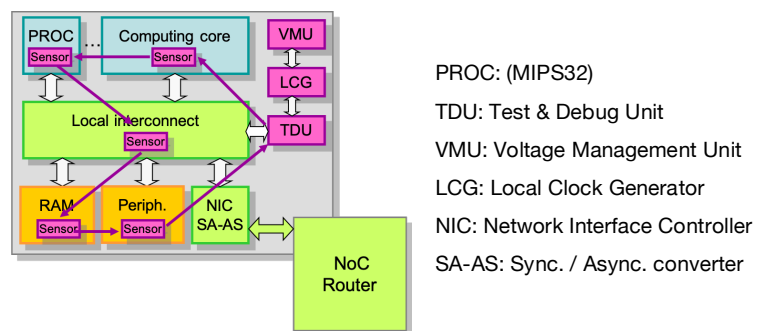


Figure 6 The 2D Mesh multi-cluster MP²SOC

Applications suited to be run on this hardware architecture are based on the KPN (Kahn Process Network) paradigm. The application can be described as a finite set of tasks interconnected by FIFOs. A given task can either be performed in hardware (with a coprocessor) or in software, in which case it becomes a thread executed by a processor. Hence, the embedded applications can contain hundreds of threads executed in parallel by the hardware resources. The performance of the system is directly related to the quality of the application mapping, i.e. the placement of threads on processors and the placement of FIFOs on chip memory banks.

3.1 The database

The DRET (Distributed Event Raw table) is a distributed in memory database that is physically located in a specific part of the RAM memory of each cluster. Its purpose is to contain the monitoring data extracted from both hardware and software. The DRET of a cluster may contain several tables like traditional database systems and

each table contains several formatted events retrieved from the HW/SW sensors. The database has been designed to be an efficient data structure that can be used to store and retrieve information easily from the viewpoint of the monitoring process. The used structure and the API are designed to keep the memory footprint overhead as small as possible. All the information concerning the cluster and the subpart of the functional application running on it is maintained in a specific cluster DRET. The DRET is accessed by means of a set of explicit functions, called the DRET API. The DRET can thus be seen as a uniform repository for the events whatever their origin (hardware sensor, software probe, local or external to a given cluster). While a DRET is mainly related to a given cluster, it can also be used to store a synthetic view of the state of neighboring cluster and, in the case of permanent failures, a list of located faults in the rest of the system. In order to prevent the DRET from occupying too much memory space, a maximum size is associated to each table created in a DRET. A DRET table can also be cyclic. In that case, it behaves like a fixed-size cyclic buffer: inserting a new event in a full table discards the oldest one.

3.2 The monitoring application

The main targeted scenario aims at detecting permanent failures occurring in the chip during the execution of an embedded functional application. The key point of the approach is to perform fault detection by having a second application running in parallel with the functional one, and called the monitoring application. It is very important that the functional application is unaware of the monitoring one. This prevents the user from developing a too complex application that would contain both functional and monitoring code. The purpose of the monitoring application is mainly to collect hardware activity counters provided by embedded sensors and to verify the liveness of the functional application. The monitoring application is by essence distributed on the hardware. It is also a KPN composed of a set of tasks, called monitoring tasks, each of which runs the same code. Each task is responsible for monitoring a subset of the chip hardware. There is one monitoring task per cluster, and each of these tasks monitors the cluster it is running on. Figure 8 shows the mapping of the tasks and communication channels of both functional and monitoring applications on the MPSoC hardware architecture.

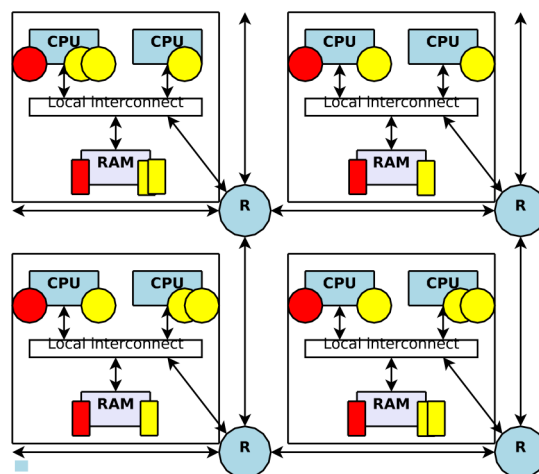


Figure 8 Mapping of both functional and monitoring applications on the SoC

The yellow circles and rectangles represent the tasks and communication channels of the functional application. The red circles and rectangles represent the tasks and communication channels of the monitoring application. On this figure, one can see that the monitoring application and the functional one share the same CPU and RAM banks.

The monitoring tasks are connected to their closest cluster neighbors (up to 4) according to the mesh topology through FIFO channels. The communication channels used are FIFO channels which are used by tasks to read and write their messages. Each channel can be accessed by many readers and writers connected to it (at least one of each). The resulting monitoring task graph is summarized in Figure 9.

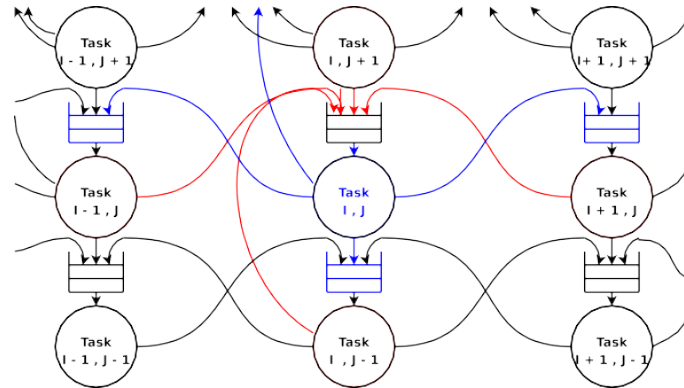


Figure 9 Monitoring application task graph

Monitoring a tile/cluster requires to collect its hardware activity counters. Each component in the tile embeds an activity counter that is increased according to the performed action (initiating a request, answering a request, etc). The counter values are periodically sampled by the test and debug unit (TDU) found in each tile, and then read by the software monitoring task. The hardware counters are connected together using an JTAG scanpath and connected to the test and debug unit. We designed and validated the memory mapped TDU co-processor; the details of its architecture can be found using the references cited at the end of this section.

The monitoring task compares the newly sampled values with the previously collected ones, stored in the DRET. No increment between the new value and the previous one means that there was no activity in the component since last iteration. No activity during several iterations (a configurable number of data samples) means that the component has failed. In this case, an alert is raised and sent to all the neighboring tasks. The alert is broadcasted to all the tasks of the monitoring application. To ensure the robustness of the monitoring, the tasks periodically send messages to their neighbors, expecting an explicit answer. The semantic of this message is comparable to the ping request. Getting an answer means that the neighbor is alive. If no answer is received after a (configurable) timeout delay, the sending task assumes that the receiving one is unable to answer, and concludes that a fatal failure has occurred in the addressed tile. In this case too an alert is raised and broadcasted to the whole monitoring application.

Each task that receives the alert stops running the functional application, stores its DRET into a non-volatile memory, that will be used for the diagnosis step. Once all tiles have received the alert, the system is ready for reboot. To reboot the system, one must ensure that the alert propagation is complete throughout the whole chip. Each monitoring task that receives an ALERT message forwards it to all its neighbors, and sends an acknowledgment message to the sender.

An acknowledgement mechanism is used for two reasons. The first one is to avoid live-lock, as tasks would keep forwarding alert messages each time they receive a new copy. The second reason is to make one task aware of the completion of the alert propagation, making it able to perform the required processing.

In this monitoring application, all tasks use the DRET. Each task creates two DRET tables as it starts. The first one is dedicated to the event storage. The second one contains processed data about the neighboring tiles liveness.

3.3 Outputs

To evaluate the feasibility and the penalties induced by the software monitoring we made several simulations. The monitoring application has been implemented on a multi cluster MP²SoC. To keep it simple, each cluster contains two MIPS32 CPUs, a local RAM block and a Test and Debug Unit (TDU) connected to some dummy registers (using the JTAG protocol). The hardware architecture description is written in SystemC, using Bit-Cycle-Accurate components. All results are measured in simulation cycles. The monitoring application was coded in plain C and compiled with gcc for a MIPS32 target. Its code size is 11KB. The size of the DRET library is 8KB. Each task of the monitoring application uses 15KB for its local data, and an extra 15KB for the DRET tables. The execution time overhead measures are done for two functional applications, each one running in parallel with the monitoring one. The difference between the execution times (without/with monitoring) gives the overhead caused by the monitoring.

The first functional application is a 10 tasks parallelized MJPEG (Motion-JPEG) decoder, uncompressing 200 images. The second one is a Smith&Waterman DNA sequencing application, parallelized in 26 tasks, looking for 6 sequences in a library of 200 or 300 samples. The MJPEG decoder runs on a 4 clusters chip, and the Smith & Waterman runs on a 6 clusters chip. The experiment consists in measuring the execution time overhead due to the concurrent running of both applications (functional and monitoring) on the same platform. In these scenarios, there are no fault injections and the monitoring application does not raise any alert (which should be the common case). The tasks of the monitoring application are periodically wakened up by an interrupt sent by the TDU. The simulations show that the monitoring for the two applications induces about 2% overhead on the execution time. This overhead can be reduced when tuning the periodicity of the data collection by the monitoring tasks (here 2.000.000 cycles). Considering the memory size being very limited in embedded system, the amount of memory required for implementing the DRET is still acceptable. The extra memory required for storing data will depend on the amount of monitored information. Since it is possible to have cyclic table in the DRET, developers can tune the maximum amount of information that must be stored in order to avoid memory overflow.

To summarize this work, we have proposed a scalable, distributed and non-intrusive approach enabling online monitoring data processing and actuating. We demonstrate the feasibility of the approach and evaluated the impact on the execution time and on the extra memory. As a proof of concept, the online testing was here reduced to the comparison of register values. However, thanks to its software modularity, a monitoring task can be used to launch more thorough online tests such as SBST routines as described in the previous section.

As mentioned previously, this work has been done in collaboration with François Pêcheux, associate Professor and Etienne Faure during his postdoctoral research period. Montassar Bougattaya also helped on the design of the Test and Debug Unit during his Master 2 internship. More detailed results and discussions have been published in [C06], [S06] and, [W06].

4. Test and fault tolerance of Network on a Chip

Chips integrating hundreds of cores have already been marketed, and 1000s-core chips are likely to arise in a near future. One of the most important issues, tightly linked to this exponential complexity, is the chip reliability. Zero-defect manufacturing of a many-core architecture on a chip, with 100% fault-free components, is a challenge that is very difficult, not to say impossible to achieve in practice. To improve this intrinsic low yield, manufacture defects must be located and isolated as soon as possible, hence faulty chips can be reconfigured and eventually sold at a lower cost. Moreover, during the chip running time life, numerous permanent, transient and intermittent faults occur because of wear-out effects, particles' striking and environment perturbations. To solve this reliability issue, on-the-field fault tolerant mechanisms must be integrated into the architecture, and activated during chip runtime.

Many core architectures usually contain a large number of replicated identical components, such as processor cores and embedded memory banks. The abundance of identical resources makes such architectures able to degrade gracefully, with faulty components that do not hamper the global behavior of the chip. Unfortunately, the global Network On Chip is the only centralized resource, that cannot easily be duplicated at an affordable cost, and its recovery mechanism in case of failure is not straightforward.

For a given technology, the fault tolerance strategies mainly depend on the NoC topology, the type of targeted faults (permanent or transient) and the routing scheme (deterministic or adaptive). Strategies can be applied at network level, router level and link level. During several years my research activities were dedicated to propose solutions for both permanent and transient faults and for both deterministic and adaptive routing. These activities were done in the framework of three PhD studies, the ADAM ANR project and the 3DIM3 Catrene European project.

With my colleagues and students, for our case studies we selected NoCs with a mesh topology, since this regular topology was the most commonly used. It saves silicon area, and minimize the network latency when compared to other topologies. We targeted the fault tolerance of massively multicore SoCs to take benefit from the numerous identical hardware resources allowing the development of simple hardware and software strategies.

In that context, we adopted a coarse grain deactivation strategy: when a fault is detected and located in a hardware block, the whole block is considered as faulty.

4.1 Network on Chip with deterministic routing for permanent faults tolerance

Positioning over state of the art

Usually the error control process is based on detection, containment, and recovery [20]. However, state of the art solutions often addressed only part of this process and were often based on CRC/ECC techniques [21-22] or on BIST techniques [23]. Moreover, there were no uniform solution allowing to run an on-the-field fault-tolerant mechanism with detection, containment and recovery for permanent faults and for deterministic routing, especially in a scalable and distributed way.

In the framework of the ADAM project, targeting self-adaptive MP2SoC, we needed to rely on healthy NoC component. However, as a shared communication mean, online test and online reconfiguration of the NoC is not possible at an affordable cost. Regarding the faulty processor cores or embedded memory banks, once the erratic

behavior has been detected (detection), the component can eventually be deactivated (containment), and the software application can be remapped on the remaining operational hardware (recovery). But for the NoC, a faulty sub-component impacts the behavior of all the NoC. Therefore, we proposed a solution to be run **at power on or at each reset time**.

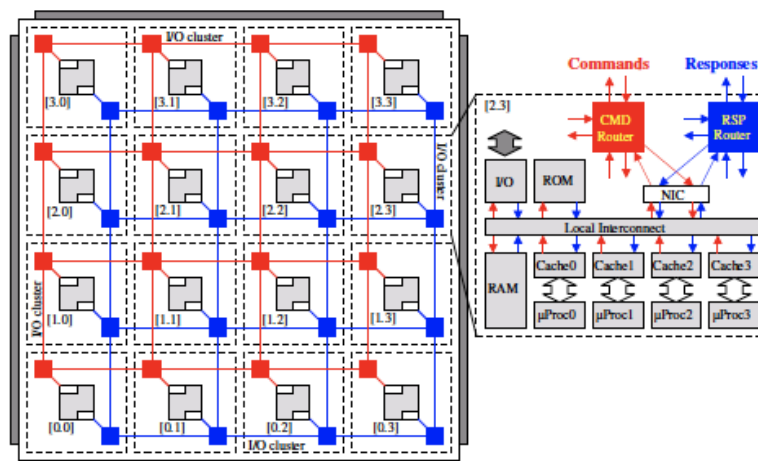


Figure 10 A 4x4, DSPIN based, shared memory, MP2SoC architecture

The hardware NoC

For the case studies the NoC used was a generic predictable scalable network named DSPIN (Distributed Scalable Predictable Interconnect Network) developed at the LIP6 laboratory. It is a typical 2D-Mesh NoC, designed to support many-core architectures and the GALS (Globally Asynchronous Locally Synchronous) approach. It implements all the services and exhibits all the properties one can expect to find in state-of-the-art NoCs.

As shown in Figure 10, a DSPIN based, shared memory, many-core architecture is typically composed of a set of tiles, each of which running in its specific clock domain. A tile may contain one or several programmable processors, a local interconnect, an embedded RAM, an embedded ROM and two routers: in order to avoid deadlocks in command/response traffic, each tile contains two independent routers implementing two separated sub-networks for commands and responses. To avoid significant area overhead, the DSPIN NoC is based on routers with a pair of FIFOs at each port (no virtual channels). For the same reason, the routing algorithm used is a basic turn-based X-first algorithm, without routing tables.

In addition, some special tiles contain I/O ports controllers, used to access external mass storage devices. In order to support “on-the-field” reconfiguration, we suppose that each programmable processor core manages a timeout mechanism that can be activated on demand: when the processor executes a memory load/store operation that is likely to fail, the timeout mechanism is triggered. If the transaction fails, the timeout generates an interrupt; the processor enters its exception mode and thus is able to proceed with the execution of the next instructions.

To deal with on-the-field NoC fault tolerance, in a shared memory many-core architecture, three identified problems must be solved:

- Test Infrastructure: aiming at detecting/deactivating faulty components (routers and channels) in the early stage of NoC initialization, a built-in self-test and self-isolation (BIST/SI) strategy must be defined.
- Configuration Infrastructure: a global configuration (observation) infrastructure must be realized. The purpose is to diagnose/locate the faulty/deactivated components, and then to identify accurately the modification in the NoC

topology, and finally to distribute the reconfiguration parameters to enable the NoC to function in a normal mode but with less resources. Therefore, this infrastructure must supply a robust global configuration bus that is used to propagate the reconfiguration parameters to the NoC components subject to topology changes.

- Routing Algorithm: A new deadlock-free, livelock-free, and reconfigurable routing algorithm must be implemented to support any modification of the topology

To solve these problems, we proposed a three stages mechanism:

1. Initialization stage: distributed NoC self-test
2. Pre-configuration stage: selection of a reliable communication infrastructure among the healthy NoC components
3. Configuration stage – reconfiguration of the routing algorithms using the reliable communication infrastructure

With these three stages, a damaged 2D-Mesh NoC can structurally self-test, partially self-disable, globally self-reconfigure and functionally self-recover, after a simple system reboot or at chip power-on.

I give hereafter a very brief summary of each stage features. The corresponding hardware, software and algorithms can be found in the references at the end of this section.

4.1.1 Initialization Stage (HW)

A fully distributed off-line hardware BIST dedicated to 2D-Mesh NoC is proposed in a GALS context. This BIST is implemented as a distributed initialization procedure, and executed at each system reboot or chip power-on. With the help of this procedure, the fault-free NoC components are activated, while the faulty components are detected, de-activated and configured to behave as “Black Hole”. For that we developed a BIST engine for each router and its channels, targeting permanent faults. If a router test fails, the router is considered as faulty, and all corresponding I/O channels are deactivated. Otherwise, the test of all associated communication channels is started in parallel. If a channel test fails, the channel is considered as faulty, and thus is deactivated. Otherwise, the channel is activated. Each de-activated channel is configured to behave as a “Black Hole”, that discards any incoming data, and produces no outgoing data. With this configuration, when a packet is routed to a faulty channel or a faulty router, the packet is discarded, which will trigger a timeout exception at emitter side (a processor core). These exceptions will be used in the next stage to locate the faulty routers and channels.

Therefore, this procedure avoids the failure propagation (flooding of fake packets for example) and cleans the malfunctions of the overall Network-on Chip through a simple reboot of the chip. It makes the NoC ready for pre-reconfiguration.

The initialization procedure can be performed not only “on the field”, but also at fabrication time. It can thus help to improve the yield by avoiding to throw the whole chip when one single component is faulty.

4.1.2 Pre-configuration Stage (SW)

During the usual (software) initialization of a many-core architecture, the boot code that performs various tests and configuration tasks is generally located in a unique ROM. The Configuration Firmware (called CF in the following) proposed in this work is similar to the BIOS in a multi-core PC, but with a strong assertion: the chip is possibly damaged, and the hardware resources cannot be trusted anymore. Chip initialization takes place in an uncertain world where a processor, a memory bank, a Network Interface Controller (NIC), a router, a channel or

the boot ROM itself may be defective. To remove this uncertainty, the key idea is to have one CF with a small memory footprint per tile and to support a fully distributed approach, where a tile is able to communicate and exchange informations with its neighbor CFs. This results in a Distributed Cooperative Configuration Infrastructure (DCCI) with no predefined master responsible for managing resources. During initialization, the role of the DCCI is to progressively build - only relying on local communications between neighbor tiles - a temporary trusted tree of operational tiles, where each operational tile contains one operational processor running the CF. This tree is built from bottom to up, starting with operational tiles as leaves. This communication tree uses a limited part of the routing capabilities of the NoC (that has not yet been fully configured). It uses only the local communication channels between neighbor tiles for software-based CF-to-CF communication through dedicated mailboxes. The root of the tree is a tile determined by a distributed election process. The most important criterion in this election process is the capability of this root tile to access the external mass storage where more exhaustive test programs and the final operating system itself are available, and can be loaded in the embedded RAM of the root tile. The software-based communication tree can thus be seen as a slow and temporary communication infrastructure, dynamically constructed during the boot stage, using the NoC resources that have been identified as fault-free by the hardware BIST/SI.

As soon as this tree is constructed, and the root is elected, this very unique master processor can access the external mass storage containing a virtually unlimited set of application software outside the chip. That way, it is the chip that adapts itself to the environment and not the opposite. The master processor can use this communication tree to make any processor in the tree execute any specific software task (debug, fine-grain test), to propagate any configuration command to any child tree node, or read any status information.

This communication tree can then be used to complete the many-core architecture reconfiguration, and especially the configuration of the NoC itself that is done on the next stage. The DCCI is a software communication tree, and corresponds to a task graph. We do not need to configure any router or any routing table during the DCCI phase.

4.1.3 Configuration Stage (SW)

By collecting the information stored in each tile on successful and unsuccessful transactions (through the DCCI communication tree), the master processor (root of tree) is able to localize the “Black Holes”.

Once all the “Black Holes” have been identified and localized, the DCCI master (tree root) has knowledge of any topology modification, this description will be then used to configure the NoC, to remap the application tasks and any extra work.

An ultra-low-cost reconfigurable routing algorithm has been proposed to handle the modification of the topology.

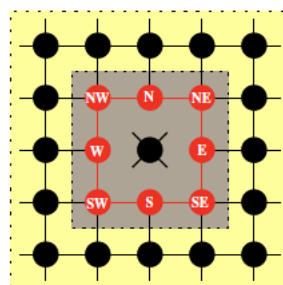


Figure 11 The reconfigurable routing algorithm routes packets to bypass the faulty router

For sake of simplicity, in this work we assumed that, when any input channel of a router is detected as faulty, the entire router is considered as faulty.

As shown in figure 11, with the proposed routing algorithm, a faulty topology is split into two regions: a normal region and a faulty region. In the normal region (yellow), a router does not need to be reconfigured, it uses the default routing algorithm: X-First. In the faulty region (gray), the border routers are configured to create a contour, in order to bypass the faulty router. The main challenge was here to avoid that the new paths introduce a deadlock. Based on channel dependency graph analysis, the developed routing algorithm has been formally proven to be deadlock-free for all single-faulty router topologies. To support this reconfigurable routing algorithm, in each router, a simple 4 bits register for storing the configuration information is needed. Thus, to finalize the configuration phase, all registers of routers in the faulty regions are reset with the adequate values.

4.1.4 Outputs

The costs from the points of view of the silicon area, timing and power consumption were evaluated through a synthesizable VHDL model of a complete router (with BIST and deactivation features) in the Synopsys synthesis environment. The whole router footprint is increased by 39.05%. This cost is caused by the implementation of the reconfigurable routing algorithm and the BIST/SI infrastructure. It is an affordable cost, since the DSPIN NoC is a very small and compact design and it represents typically less than 3% of the silicon area in a typical many-core architecture. The proposed BIST scheme allowed for the whole router to reach a 91% of global stuck-at fault coverage and 71% of global transition delay fault coverage.

All the stages were simulated and validated using many-core architectures modeled as SystemC virtual prototypes, in the framework of a cycle accurate and bit accurate simulation environment.

Using MIPS32 processors in the above-described platform, we measured the application code size. In total, it is 7.5 Kbytes including 5 Kbytes for the DCCI (CF) and 2.5 Kbytes for the software “Black Hole” detection/location. Embedding this application in a many-core architecture is thus affordable in terms of memory footprint.

The execution time to run the full process is also affordable. For example, for a 16x16 NoC with 3 faulty routers, from the power on to a fully reconfigured NoC, the procedure requires about 1.4×10^6 clock cycles (0.14 second with a 1Ghz frequency).

This work was done at LIP6 laboratory, in the framework of the PhD thesis of Zhen Zhang, with the help of François Pêcheux, associate professor (now full professor) and Alain Greiner Professor.

This work led to several publications where the reader can find more details: [J04], [S07] and [S10]

4.2 Network on Chip with adaptive routing for permanent and transient faults tolerance

In order to have a global and better understanding of many core SoCs’ fault tolerance, I studied also NoCs sensitive to transient faults and implementing adaptive routing function. This kind of routing is more flexible and more adapted to online self-adaptability to runtime failures, but the NoC architecture induces a significant area overhead when compared to the deterministic routing. The NoC architecture is often based on the usage of virtual channels (to avoid deadlocks conditions) and/or routing tables.

Positioning over state of the art

Most of the relative works presented in literature focuses mainly on permanent faults and few like [24], study the influence of permanent and transient or permanent and intermittent [25]. In this respect, an added value of our study is to provide some insight about the efficiency of a fault tolerant routing algorithm under the co presence of an unconstrained number of different types of faults: permanent, transient and intermittent.

Another issue is that some routing algorithms targeted traffic optimization but very few consider the fault tolerance. In this context we proposed a new global approach, which works by exploiting local information, based on an existing adaptive fault tolerant routing algorithm named CAFTA (Congestion Aware Fault Tolerant Algorithm) [26], developed at TIMA laboratory. In order to improve transmission latencies and to effectively guide the routing decisions towards load-balanced network traffic, we enhanced this algorithm with a new performance metric. Moreover, to empower a commonly used packet retransmission scheme, we developed new packet recovery mechanisms that mitigate runtime errors that may corrupt (split) a packet during its travel time (due to permanent and transient faults). The approach had been assessed for high failure rates.

4.2.1 Router architecture and routing algorithm

The router architecture, which is depicted in figure 12, is based on buffered inputs with a credit-based wormhole control flow. A typical 5-stage pipeline router architecture is used with some additional enhancements to support the mechanisms that will be described latter on. The five stages are Buffer Write (BW), Route Computation (RC), Virtual Channel Allocation (VA), Switch allocation (SA) and finally the link traversal (LT).

The routers in a NoC are connected through pairs of unidirectional links (bi-directional connection) and each unidirectional link is composed by a group of parallel wires. The fault model we are using here is a coarse-grain fault model under which a whole unidirectional link could be faulty. Partially faulty links where only a partial set of link wires could be faulty were not considered in this study. With this model we also modeled faults in router's ports. Faults can be permanent (either pre-existing due to manufacturing/design errors or runtime due to wear-out, aging effects, etc.), transient (due to Soft Errors, Cross-Talk, etc.) and intermittent.

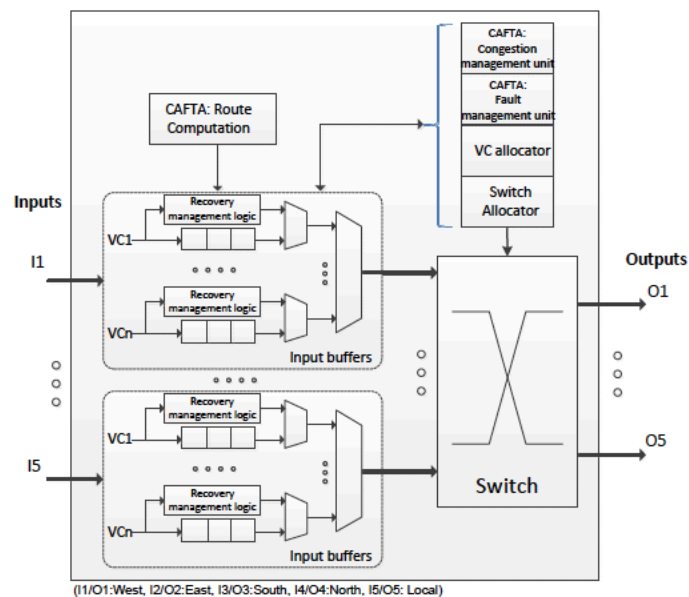


Figure 12 Router architecture with all the new features

The proposed routing algorithm is based on a fault tolerant routing algorithm developed at TIMA laboratory by another group of researchers (developed by F. Chaix, M. Nicolaidis and al., not detailed here). This algorithm combines turn-based North-Last and South-Last adaptive routing algorithms. For each routing algorithm, a separate virtual network (VN) is used. Four virtual channels are used (VCs) per port and two of them are assigned to each VN.

To correctly choose a healthy output, each router maintains in a fault status register the health status of its links.

As our focus here is not on the fault detection mechanisms, for our fault tolerant scheme we made the following assumptions:

- The messages are protected by ECC, and in case of error, flit retransmission occurs from the upstream router.
- Software or hardware self-checking is used to detect faults at processing nodes and links.

In this work we have empowered the basic routing algorithm with:

- Empowered neighbor fault awareness: each router not only knows the health status of its own links but also the links of its neighboring routers (routers at a distance one hop away).
- Network traffic regulation by the use of a new congestion avoidance metric.
- Packet-retransmission combined with new recovery schemes.

I give here after a brief overview of the two last items.

4.2.2 Network traffic regulation/Congestion management

Various metrics have been presented in the literature on how to dynamically regulate the network traffic flow in order to have improved performance (low latency). Thus, there are metrics which make use of local information, like the number of free VCs, the buffer availability of adjacent nodes, and metrics which utilize regional information. These metrics have been proposed and used mainly in a fault-free environment. However, their usefulness can be exploited to guide the selection of propagation paths also in faulty environments. In a faulty environment, the primary goal for a fault-tolerant routing algorithm is the careful selection of fault-free paths towards the destination. The congestion metric is only used as a secondary criterion for path selection in case of several available outputs available.

The improved fault tolerant algorithm CAFTA we developed relies on local (limited region) information, the safe region of applicability of a congestion metric is the region of fault awareness, here one hop as a maximum.

We introduced a simple congestion metric, namely Flits-remain (FR). In a wormhole-based control flow, the header flit of a packet defines the path to be used while it traverses the network from the source towards the destination, and is then followed by the packet data (payload). This path remains active until the tail flit of the packet passes and releases the resources. The measure of occupancy of the resources (buffers, VCs) we had chosen in a router is the number of flits that remain to pass through (*flits-remain* or FR). Usually, the header flit carries the size of the packet which is attached to it. This size represents the number of flits that will follow the header flit. In our router architecture, a counter is added at each port for keeping track of the FR value. When the header flit of a packet leaves a router from one of its output ports, it increases the FR value of that port by an amount equal to the value of its packet-size (in flits). After that, each time a data flit of the packet exits the router through one of its ports, the FR value in the respective exit port is decreased by one.

Using this metric and the fault tolerant algorithm, in a NoC with faulty links, a packet can follow a different and more efficient route from source to destination, as can be seen in figure 13: $\{(0, 1), (1, 1), (2, 1), (3, 1), (3, 2), (3, 3)\}$ instead of $\{(0, 1), (0, 2), (0, 3), (1, 3), (1, 2), (2, 2), (3, 2), (3, 3)\}$ with the initial fault tolerant routing scheme.

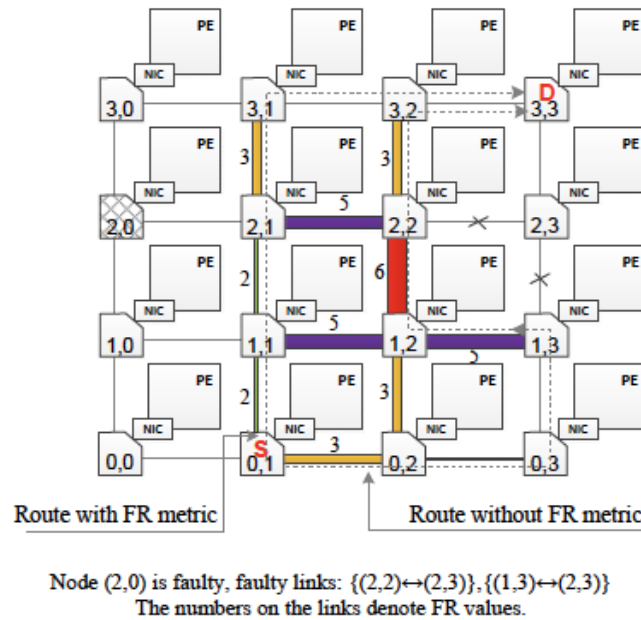


Figure 13 Packet routing with congestion awareness

4.2.3 Packet recovery scheme

During the operation of the NoC, when a fault occurs (permanent, transient or intermittent), different situations may arise depending on the nature of the fault and its location: corruption of a packet, interblocking of packets, loss of the packet, ...

Our main goal was to allow a graceful degradation of the NoC. For that we targeted both static and dynamic, permanent and temporary faults, and we proposed to integrate in each router:

1. A retransmission mechanism with k retries for the output port selected by the routing calculation (RC) stage. This solves the problem of temporary dynamic faults not corrected by ECC techniques (multi-bit faults).
2. An adaptive fault-tolerant routing algorithm linked to the retransmission mechanism. After k unsuccessful retransmission attempts (here for practical reasons we set $k=2$), the algorithm selects another output port to route the packet. This partially solves the problem of nodes or links that contain the static permanent fault, and allows the packet to bypass the faulty node/link.
3. A packet fragmentation/assembly mechanism. This mechanism aims to tolerate dynamic faults (permanent or temporary) that modify the validity of the path taken during the transit of the packet. For example, a router or a link suddenly becomes a hole in the middle of the path while not all the flits of the packets have passed through. This creates sub-packets that can no longer reach their destination and more critically the involved resources along a path cannot be released. This last issue may quickly lead to a complete freeze of the NoC. The proposed schemes (bypass packet recovery and fragment packets

joining) may be proved useful in the cases of high-volume data transfer between nodes e.g. in multimedia applications.

4.2.4 Outputs

CAFTA, with the router's microarchitecture described in figure 12 and the necessary framework to perform the fault injection campaigns, have been implemented in C++ on top of IRIS, a largely used cycle-accurate network simulator. Three different 2D-mesh networks sizes (10x10, 12x12, 16x16) were studied. CAFTA's performances were analyzed under three synthetic traffic patterns: uniform random, transpose and bit complement, that are commonly used. Two different fault types were examined, unidirectional link faults and node faults. The fault patterns considered are: permanent faults, transient faults and intermittent faults. More specifically, we have mixed the type of fault injected, thus for a specific number N of injected faults the ratios of permanent, transient and intermittent faults are: 15% permanent faults (at T0) and 5% permanent runtime faults to account for failures due to wear out, electro migration, aging effects etc., 40% transient faults (due to single event effects, cross-talk, etc.) with duration of one cycle and 40% intermittent faults with random duration from 200 cycles up to 800 cycles.

Several different fault injection ratios (fir) were used and for each fir the chosen network size was simulated until 5000 injected packets either reach their destination or are dropped.

Under different traffic patterns, it was observed that using the FR metric results in a latency improvement (at the saturation point) of 53.95% on the average with respect to the baseline case where a congestion metric is not utilized and 19.9% with respect to the state-of-the-art free virtual channel metric. Furthermore, using the FR metric allows the network to saturate at higher traffic injection rates.

The experiments have shown that our approach allowed high reliability property. With reliability we mean here the ratio between packets reaching their destination among all sent packets. This ratio is about 98% for a 2D mesh network of 16x16 in the presence of 384 simultaneous link faults. For the extreme scenario of 103 routers being simultaneously faulty, the reliability attained by our approach is about 93%.

This work was done at TIMA laboratory, in the framework of the PhD thesis of Yi Gang, with the help of Michael Dimopoulos (Postdoctoral researcher supervised by Michael Nicolaidis Research Director at CNRS), Nacer-Eddine Zergainoh, Associate Professor and Lorena Anghel, Professor at University Grenoble Alpes.

The detailed outputs of this work have been published in [B02], [J05], [S12] and [W11].

4.3 Test and Fault tolerance of 3D Network on Chip

As seen in the introduction, the need for continuous integration for improved performances pushed the designers to consider the third dimension as a significant breakthrough. The power and delay problems of global wires can be alleviated at technological level by stacked 3D integration. However, interconnecting an increasing number of IP blocks, while ensuring high computation and communication throughput, requires a reliable and scalable solution: the 3D Network-on-Chip.

At the time of my research activities, the progresses of Through-Silicon-Via (TSV)-based 3D integration technologies, were not considered mature enough for large-scale production. Testing, manufacturing costs, yield and reliability, thermal management, and heat removal were major challenges. In 3D NoC-based MPSoCs, intra-/inter-die communication reliability was paramount, as transmission faults during network traversal may lead to system failures. Therefore, efficient interconnect test strategies and error resilience techniques were mandatory. In

the framework of the Catrene European project 3DIM³, we addressed the TSV test, yield and reliability of 3D NoC-centric MPSoCs. More specifically we proposed several approaches to:

- test interconnects
- enhance the 3D NoC resilience in case of failures
- explore and assess single and cross layers strategies for resilience. We provided at design time a tool to evaluate the best resilience strategy with regards to the yield and performance requirements

Positioning over state of the art

The test and fault tolerance of TSV based 3D NoCs were at their infancy at the time of this work. Regarding the state of the art ([27], [28]), we proposed interconnect test solutions that offer new features such as reconfigurability and diagnosis capability. But the most preeminent novelties of this work against state of the art are: i) the assessment of cross-layer strategies for resilience and ii) the associated design space exploration tool introduced here after in this section.

4.3.1 Off-line Interconnect self-test (IBIST)

Testing inter-die wires becomes increasingly difficult when the number of TSVs / chip is in the thousands or tens of thousands range. An Interconnect Built-In Self-Test (IBIST) strategy was proposed for testing TSVs of inter-die links in 3D NoCs. Among the advantages of this approach is the capability to sensitize both structural faults like opens and shorts, and parametric faults like delay faults and delay faults due to crosstalk. Using the aggressor-victim scenario, a new fault model was proposed: the Kth Aggressor Fault (KAF) model. The aggressors of each victim wire are the TSVs within a distance given by the aggressor order K (see figure 14 a).

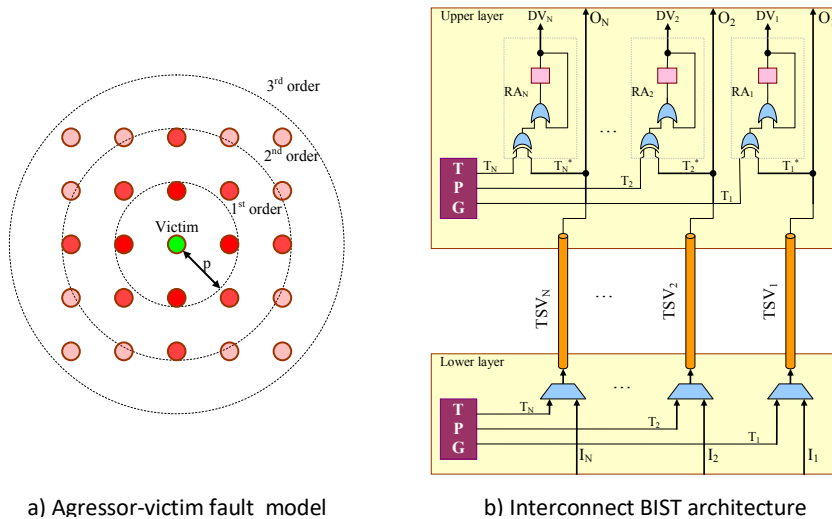


Figure 14 KAF-based BIST for Through Silicon Vias

The proposed architecture is showed in figure 14 b) and is based on dedicated Test Pattern Generators that can be found in the lower and upper layers, with a comparison done on the upper layer. The major advantages of KAF-based testing are shorter test times (e.g. 16 cycles for 1 aggressor order and an arbitrary number of TSVs) and simple hardware implementation, which were up to 50% smaller than existing approaches based on marching-ones and MAF-based tests (Maximum Fault Aggressors) [29]. The proposed BIST solution allows also to locate the faulty TSVs thanks to a diagnosis register that can be retrieved in parallel or serially (DV for Diagnosis Vector,

accessed in parallel in the figure 14 b example). This information is required for fault tolerance techniques to be applied.

A configurable KAF-based IBIST implementation was also proposed. Despite the relative high costs (i.e. almost twice the size of simple KAF-based implementations) of configurable IBIST, being able to modify the aggressor order used for TSV tests has several advantages. First, tests during system lifetime can be performed for different aggressor orders. This strategy can also be used for calibrating TSV technologies before full-scale production.

4.3.2 Enhance the 3D NoC resilience in case of failures

The TSV reliability and yield challenges of 3D NoCs can be addressed by a single-layer or a multi-layer approach. In 3D NoCs, the effort of mitigating permanent and transient faults on TSVs can be shared between the data link and network layers. We wanted in this work to evaluate existing and new strategies for permanent and transient fault tolerance. Solutions were proposed at link level and at network level

Permanent faults:

Different implementations were proposed in order to mitigate permanent defects:

- permanent faults due to manufacturing with TSV Spare-and-Replace and Configurable fault-tolerant Serial Link (CSL). In CSL, links serialize data in multiple cycles, using the remaining functional TSVs.
- permanent faults due to aging/wear-out with an Interconnect Built-In Self-Repair and Adaptive Serialization, IBIRAS. The proposed architecture is dedicated to 3D integrated systems using highly defective vertical wires with not enough functional spare TSVs. It includes an Interconnect BIST, a Reconfiguration Logic block and Serialization and Deserialization modules for an n-bits inter-die link.
- Network failures due to TSV manufacturing and aging are avoided by implementing a new TSV fault-tolerant routing (TSV-FTR) algorithm. Here routers deviate packets on alternative fault-free inter-die links

In a multi-layer approach, the data link serialization-/spare-based strategies were jointly used with TSV-FTR in order to reduce costs and impact on network latency.

Transient faults

Transient faults, which account for most system failures, on intra-/inter-die links are handled by data link strategies based on data coding, error correction (i.e. Forward Error Correction) and retransmission (i.e. Automatic Retransmission Query), but also at a network-level using error correction (NL-FEC).

4.3.3 Single and cross-layer resilience strategies assessment

We implemented and assessed a mix of these techniques to evaluate their efficiency and costs. In order to assess different trade-offs, the error resilience strategies have been implemented on a fully synchronous seven-port router of 3D mesh NoC. The experimental results have shown that, in most cases, traditional solutions like spare insertion (TSV Spare-and-Replace) and data retransmission (Automatic Retransmission Query) are not effective for TSV high failure rates. Error correction codes (ECC)-based solutions for link protection against transients prove to be more efficient in 3D NoCs. Instead of protecting all links in the network, it has been shown that protecting only inter-die links significantly reduces hardware costs, while the network latency penalty is reduced.

Experimental results have shown also that in the current technologies, in more dense technologies the impact of the TSV footprint is larger than that of the serialization / de-serialization circuitry, making serialization-based solutions very attractive.

4.3.4 ERX a tool for resilience evaluation at design time

Although there are many solutions to the TSV reliability / yield challenges, it is difficult to choose the best strategy for a given 3D system. One-fits-all solutions do not exist and each error resilience scheme has its own advantages and disadvantages in terms of costs and impact on system performance. Therefore, an error resilience exploration (ERX) tool has been proposed for 3D NoCs. As seen in figure 15, ERX takes as inputs the NoC architecture (i.e. topology, flit size, etc), the reliability / yield requirements, and component failure rates (i.e. TSV failure rates due to manufacturing / aging and bit error rates on inter-die / intra-die links).

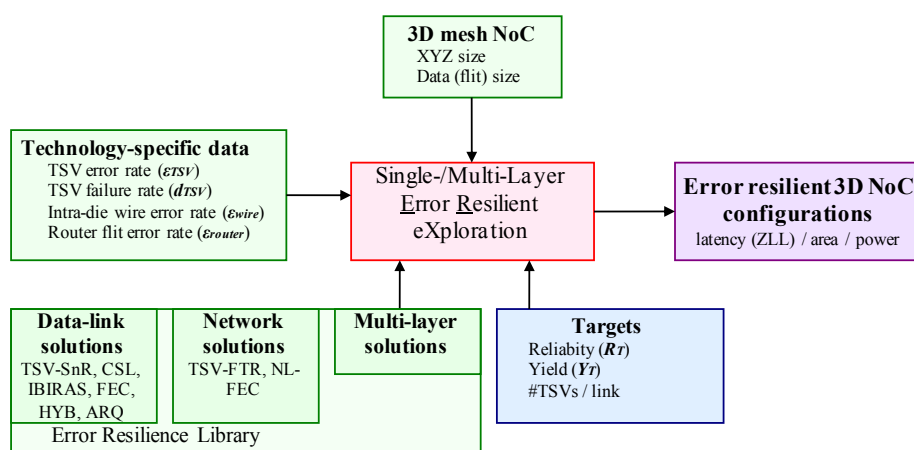


Figure 15 The Error Resilience eXploration tool (ERX)

Using the library of highly configurable error resilience techniques, it automatically implements and assesses different error resilience strategies for the targeted 3D NoC. The result is a set of solutions that satisfy the reliability / yield requirements and a series of constraints such as the number of TSVs / chip. Using these solutions, the system designer decides which solution is better fitted for the system.

4.3.5 Outputs

In an experimental case-study, the capabilities of ERX were evaluated on a 3D massively-parallel MPSoC (MP^2SoC) using a SystemC virtual platform based on the SoCLib library. The 3D MP^2SoC has 64 clusters distributed across four layers and interconnected by a $4 \times 4 \times 4$ 3D NoC.

For two experimental setups, ERX identified several protection strategies, of which one solution (i.e. the optimal one) was implemented in the virtual platform. In both cases, the increased latency of the request and response networks has a negligible impact on the application execution time (i.e. less than 1%). These results show that, despite their non-negligible costs, careful selection of data link and network solutions may not lead to any significant performance overhead.

These different works on 3D NoC reliability were conducted by Vladimir Pasca, in the framework of his PhD thesis. Vladimir was co-supervised with Lorena Anghel with the help of Michael Nicolaidis (Senior Researcher at CNRS from TIMA laboratory). This work was done in collaboration with some academic and industrial partners of the 3DIM³ European project. Saif-Ur Rehman contributed also to the proposed approaches during his internship, co-

supervised with Lorena Anghel. The different solutions and assessments were published in [J02], [J03], [S08], [S09], [S11], [W07], [W08], [W09]

5. Test and diagnosis of FPGAs

5.1 Research context

Beside the MP²SoC reliability studies, I had the opportunity to target another kind of architecture, more regular and also based on a distributed interconnect, namely the Field Programmable Gate Array (FPGA). These devices have seen a growing interest and commercial success for many applications, from prototyping to emulation and even as an end product embedded in critical systems. Through the years, FPGAs had also taken benefit from technology improvements and their test and reliability is a key issue.

To understand the challenges and propose new reliability solutions, I was involved in the ANR project called ROBUST FPGA.

This project was based on the cooperation of the FLEXRAS startup (bought after the project by Mentor Graphics) and three laboratories: LIP6, Telecom Paristech and TIMA. The objective was to provide a set of solutions to make more “ROBUST” a novel FPGA architecture. Among the solutions we can find: i) architecture enhancement with appropriate fault tolerance hardware (Telecom Paristech) and ii) **test and diagnosis for application remapping on healthy resources** (TIMA). In this project LIP6 was responsible for the generation of the hardware architecture and FLEXRAS for the application mapping on the hardware architecture.

We thus focused on the test and diagnosis issues of this new architecture.

Similar to digital ASICs, FPGAs are tested for structural faults (i.e. stuck-at, bridging and delay faults) using DFT techniques such as scan-based design and BIST. These DFT approaches had become quite mature over the years. Optimization of these approaches for new and emergent device architectures as well as the test time reduction were considered as the main challenge. However, BIST was usually preferred over scan-based techniques since it has major advantages for FPGA testing. BIST utilizes device reconfigurability for fault detection. BIST configurations are removed at the end of the test mode and the FPGA gets into a normal application mode on the same resources. Hence, BIST offers an indigenous property of avoiding any area or performance penalty for testing FPGAs. Moreover, parallel testing can be performed to achieve further test reduction time. Finally, BIST allows performing all level testing from wafer to system-level.

Positioning over state of the art

Although BIST is a generic technique, the set of test configurations are architecture specific. For that reason, different BIST approaches are developed for different FPGA architectures. Moreover, since the CAD tools are often proprietary and architecture dependent, it is required to develop new (or adapt) existing tools to evaluate the efficiency of the proposed solutions.

Most of the prior works done target FPGAs having fully populated interconnects [30-34]. These interconnects provide full routability among logic resources but does not exploit FPGA logic equivalence, thus, incur more area and long propagation delay [7]. Therefore, new test schemes are needed for FPGAs having depopulated interconnects, which is the case for our project.

Thus, the main contributions of this work are the followings:

1. Development of test and diagnosis schemes for stuck-at and bridging faults in logic and interconnect resources of the FPGA containing hierarchical interconnect topology.
2. Optimization of the proposed test and diagnosis schemes for test time reduction, avoiding area overhead, hardware modification or loss in diagnostic resolution.
3. Analysis of the impact of various architectural parameters on the FPGA testability and the respective test costs.
4. Extension of the proposed test and diagnosis techniques to the FPGAs architectures incorporating defect tolerant techniques at logic and interconnect level.
5. Development of tools for automated test and diagnosis schemes implementation and their integration into the standard design flow for bitstream generation.

5.2 Overview of the novel architecture proposed by FLEXRAS

The novelty of this architecture lies in two key features: a multilevel interconnect hierarchy and depopulated (sparse) crossbars. As 80% of an FPGA area is due to the interconnect, the proposed technology leads to 24% area saving when compared to an equivalent fully populated architecture (such as the Xilinx Virtex Family), with a very small loss in routability. To better understand the testability challenges, I hereafter depict a short overview of the architecture.

The SRAM based mesh FPGA includes several Configurable Logic Blocks (CLBs) grouped together forming a cluster. Clusters are typically arranged in a grid surrounded by horizontal and vertical routing channels. These clusters are connected together through unidirectional interconnect network composed of wires and switch boxes as shown in figure 16.

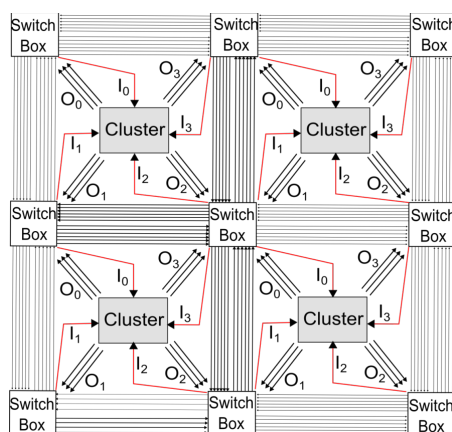


Figure 16 Mesh of clusters FPGA

Cluster architecture

In a clustered FPGA, the number of CLBs in a cluster is known as the cluster size. In each cluster, CLBs are connected together using crossbar structures as shown in Figure 17. Depending on their connectivity, crossbars in the cluster are classified as down-linking (crossbar 'Down') and up-linking (crossbar 'Up') blocks. The crossbar 'Down' offers the flexibility of connecting limited number of cluster inputs to a given CLB. The crossbar 'Up' connects CLBs' outputs to the cluster outputs as well as provides local feedbacks to CLBs through the crossbars

'Down'. It is important to note that these feedback paths are also sparsely distributed among crossbars 'Down'. The crossbars 'Down' in the cluster are sparsely populated meaning that the cluster inputs are uniformly distributed among these crossbars.

The 'Up' and 'Down' crossbars are multiplexer-based structures and an example of a crossbar 'Down' in the cluster of size 12 is shown in figure 18.

Each CLB in a cluster contains a single LUT, a one-bit register and a multiplexer. 4 inputs LUT (LUT-4) is considered as the most area-efficient and therefore used in most of the modern FPGAs. Figure 19 represents a CLB with a 4-input LUT (LUT-4), a Flip-Flop and a multiplexer 2:1 (MUX2) to select either the combinational or the sequential path. A conventional LUT-4 is depicted in figure 19. It is based on SRAM cells followed by a series of MUX2s. The configuration bitstream is loaded in the SRAM, while the MUX2s implement the logic function, according to the values of inputs A, B, C and D.

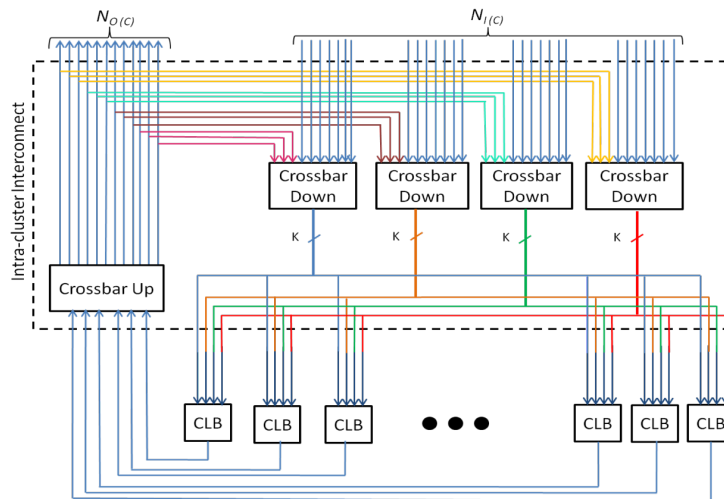


Figure 17 Architecture of a Cluster with depopulated crossbars

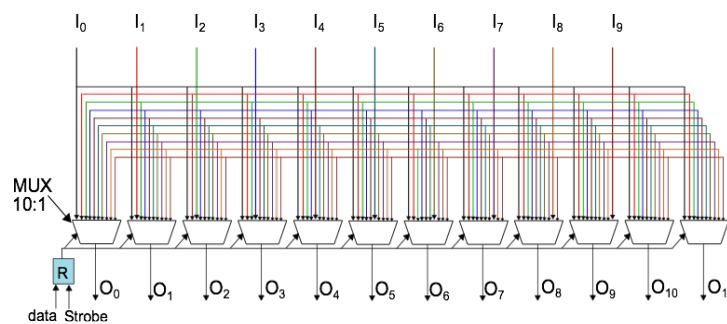


Figure 18 Structure of a crossbar down

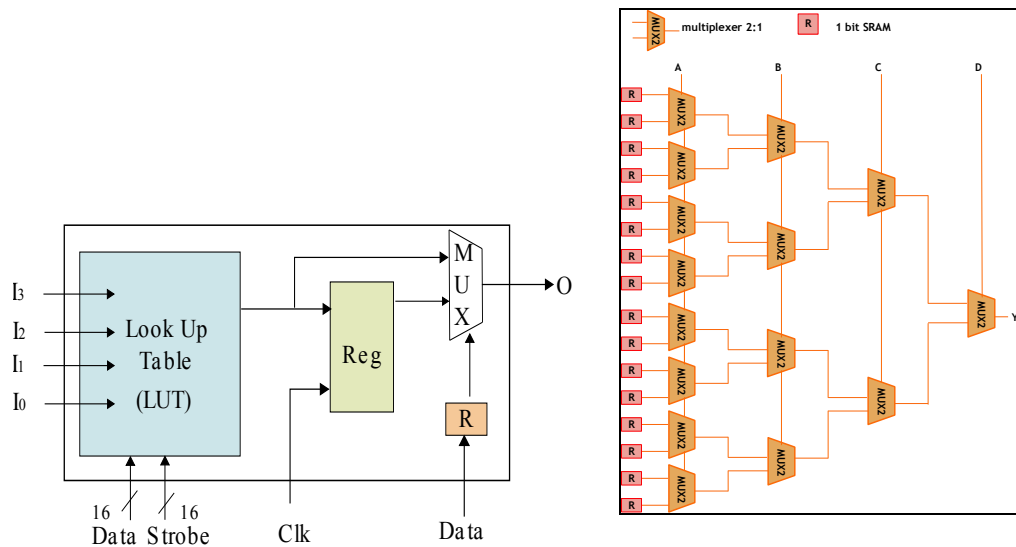


Figure 19 Configurable logic block and a 4-input LUT

Switch Box architecture

In a mesh FPGA, the clusters are connected together through unidirectional interconnect network composed of wires and switch boxes (see figure 16). Along with the connection to the clusters, the switch boxes also provide connection between horizontal and vertical routing channels which are essentially based on unidirectional wires. In this FPGA architecture, the switch box is made of multiplexer-based sparsely populated crossbars similar to those in the cluster.

Inside the switch box, connections are composed of two hierarchical levels. The 'Level 1' relates to connection among adjacent clusters whereas 'Level 2' connects adjacent switch boxes together. Figure 20 shows the internal structure of a central switch box in the mesh FPGA, connected to other four adjacent switch boxes and to four clusters. Being unidirectional in nature, crossbars in the switch box are further classified as downward and upward linking blocks. In this FPGA architecture, these blocks are named Downward Mini Switch Box (DMSB) and Upward Mini Switch Box (UMSB) respectively.

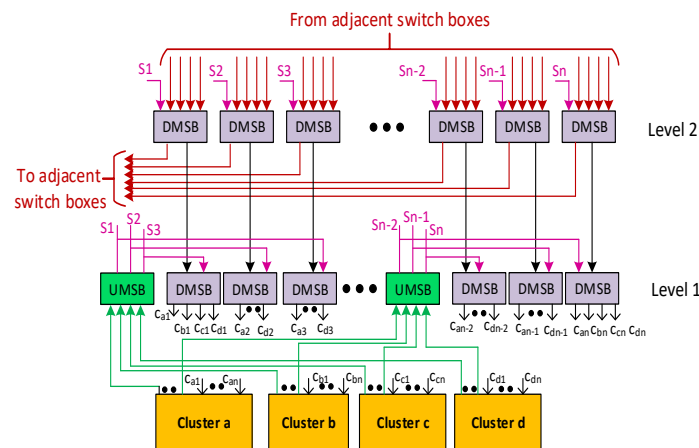


Figure 20 Architecture of a Switch Box

5.3 Test and Diagnosis Methodology

5.3.1 BIST

Our aim was to perform an exhaustive testing of every gate in the CLBs and every MUX in the different interconnect blocks, within the cluster and within the switch boxes. This must be done with the least possible test configurations since in FPGA testing the configuration time largely dominates the test application time. For that purpose, specific offline BIST structures were developed. As can be seen in figure 21, BIST operation is completed in two test sessions. In the first session, some groups of CLBs in the FPGA are configured as Test Pattern Generators (TPGs) and as Output Response Analyzers (ORAs) and some others as Blocks Under Test (BUTs). As BUTs can be configured in a number of ways, several test configurations are required to completely test a BUT. For each configuration, specific test patterns generated by TPG are applied to the BUT and the results are analyzed by the ORA at the end of each test sequence. At the completion of all test configurations, a second test session starts in which CLBs swap their roles (TPG and/or ORA becomes BUT and vice versa). The objective of these two test sessions is to configure every FPGA block as a BUT at least once to complete the test of the whole FPGA.

Similar to the conventional BIST approaches for FPGAs, test and diagnosis is performed here on a modular basis. It means that the logic and interconnect resources are tested separately.

Therefore, we divide our test and diagnosis methodology in two phases.

1. In phase 1, CLBs and crossbars in the cluster are tested and,
2. In phase 2, the switch boxes are tested.

The fault models that are targeted during these test sessions include single stuck-at and pair-wise bridging fault models.

In the TPG cluster, pseudo-exhaustive test patterns are produced by implementing n-bit LFSR, where 'n' is defined by the number of inputs of the targeted module on the BUT cluster. In the ORA, faults are detected by analyzing the output responses of two BUTs. Here, we employ a comparator-based analyzer, where any mismatch between the CLBs outputs of the BUTs is stored and recovered at the end of test sequence in each test configuration. Each CLB of the ORA cluster acts as an independent comparator which compares the output of two CLBs each from different BUTs. The BUTs compared in an ORA, receives identical test patterns but from different TPGs. This strategy helps in avoiding fault masking that may occur due to faulty TPG/ORAs.

The contents of the ORA are readout at the end of the test sequence via a scan-chain mechanism. In our case, a single scan chain is inserted connecting all CLBs of the ORA. This feature of the cluster architecture enables an

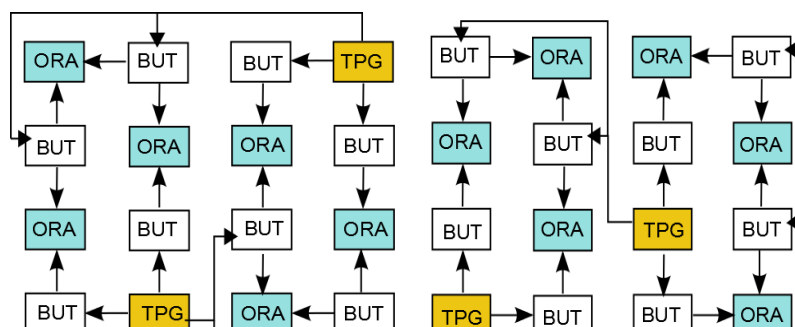


Figure 21 BIST sessions for a 4x4 clusters MESH

enhanced diagnostic resolution. The location of a fault is determined offline by identifying the faulty signal bit-position in the ORA output and the corresponding BUT configuration.

5.3.2 Logic and interconnect testing (and diagnosis)

To detect and locate faults in the FPGA, a set of test configurations is required. To perform an exhaustive testing, every gate, interconnect and multiplexer has to be observed. For this purpose, we can classify the sequence of test configurations with respect to the cluster module they target. In this way, we get test configurations for 1) CLB, 2) Crossbar Up, 3) Crossbars Down and 4) Switch Boxes.

In CLB, the LUT has two modes of operation. 1) RAM mode of operation 2) LUT mode of operation. As we have assumed in this work that the SRAM is fault-free thus, only LUT mode is tested here.

To test the 4-input LUT, XOR function is programmed in its SRAM cells and a 4-bit test pattern is applied at its inputs to test any stuck-at fault. Similarly, in the next configuration, LUT is configured to have XNOR functionality to detect the remaining stuck-at faults that may have escaped to the first configuration.

For the interconnect boxes (intra and extra cluster crossbars), the challenge was to exhaustively detect any possible stuck-at and bridging fault along the multiple wires (path under test). For that we developed several algorithms that enabled to successively apply Logic '0' and '1' at one end of the path under test and to observe the result at the other end (in registers in CLBs).

For the switch boxes the testing strategy was to select a number of Paths Under Test (PUTs) in a test phase using only adjacent logic resources. Using this strategy, any NxN FPGA array can be tested by N parallel 2x2 array procedure which ultimately reduces the test time.

For fault diagnosis, instead of utilizing extra configurations, a method was developed requiring an off line manipulation of the ORA sequence and a fault mapping between the current configuration and the corresponding fault list produced by a tool developed for this purpose.

5.3.3 Test of the FPGA with fault tolerance

Beside the test and diagnosis scheme targeting the application remapping on the fault free resources, in the ROBUST FPGA project the goal was also to propose adequate hardening techniques. LIP6 and Telecom ParisTech proposed some architecture modifications and we wanted to evaluate, from a testability point of view, the impact of such approaches. Indeed, in fault tolerance state of the art, usually, the focus remains on improving the robustness of the FPGA by introducing redundancy without evaluating the test costs of the defect tolerance scheme.

The hardening techniques proposed by our project partners are purely hardware-based, redundancy is applied at logic level and on the intra-cluster crossbars interconnects. On one hand, the techniques are considered hardware-based, they implement fine-grain redundancy and on the other hand, they are software-based because these techniques require remapping and re-routing of the application such that defective interconnect resources can be bypassed. But to do the adequate remapping we need first to locate the faulty resources.

Here follows a very short description of the proposed approaches to better understand the test challenge introduced.

Redundancy in the logic block

The first hardening technique targets the LUT. Such redundant architecture is called *Butterfly* and, according to the authors, has proved to be more robust than the conventional LUT. In traditional *Butterfly* architecture, a LUT-N has N stages, each stage containing 2-Mux2s. The LUT-N output is obtained by a bitwise majority voter in the last stage. In the case of the LUT-4, an 8-input voter is needed, which is obviously more expensive in terms of area, power and delay than a TMR voter. To simplify the original *Butterfly* structure with the purpose of having a mere TMR voter downstream, a modified version has been proposed by Telecom ParisTech as can be seen in figure 22. Consequently, some Mux2s had to be removed. As far as the voter is concerned, a fault-tolerant TMR voter is used. This voter is tolerant to single faults.

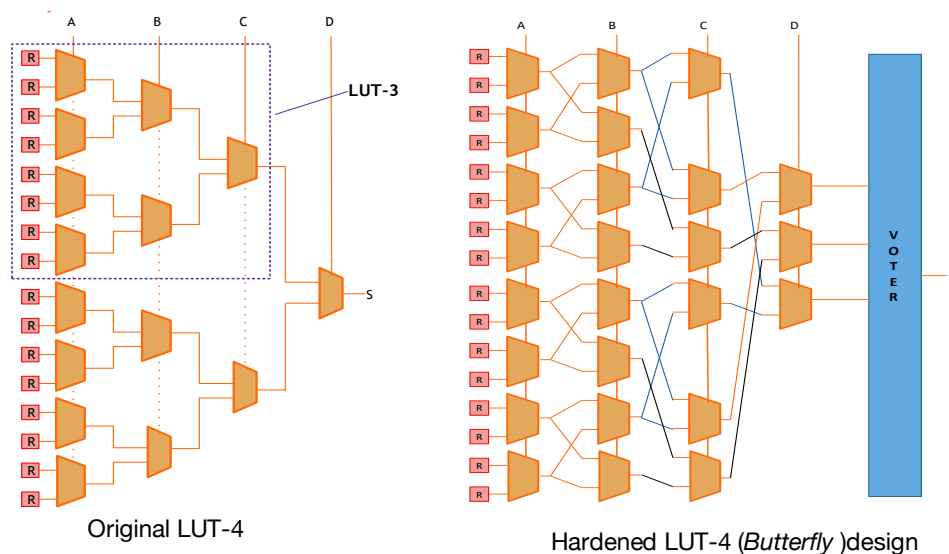


Figure 22 The original and the hardened version of the Look Up Table

Redundancy in the intra cluster interconnect

There are many ways in which redundancy can be applied at interconnect level. Two different techniques have been proposed to target the crossbar 'Down' and the crossbar 'Up':

- Fine Grain Redundancy (FGR) (see figure 23)

In the crossbar Down, four levels of Mux2s are added: two levels of Mux2s upstream to avoid the defect by shifting the signal, and two other levels downstream to restore the signal.

- Distributed Feedback (DF) (see figure 24)

To avoid any kind of problem due to the crossbar up routing signals, a solution has been proposed in which multiple feedback signals are distributed to all crossbars 'Down' (in red in the figure). Thus, a defective connection can be rerouted through another crossbar 'Down'.

To fulfil the objective of being able to locate faults in the defect tolerant FPGA, the proposed BIST schemes developed for the original architecture were extended and analysed to evaluate the impact on overall testability.

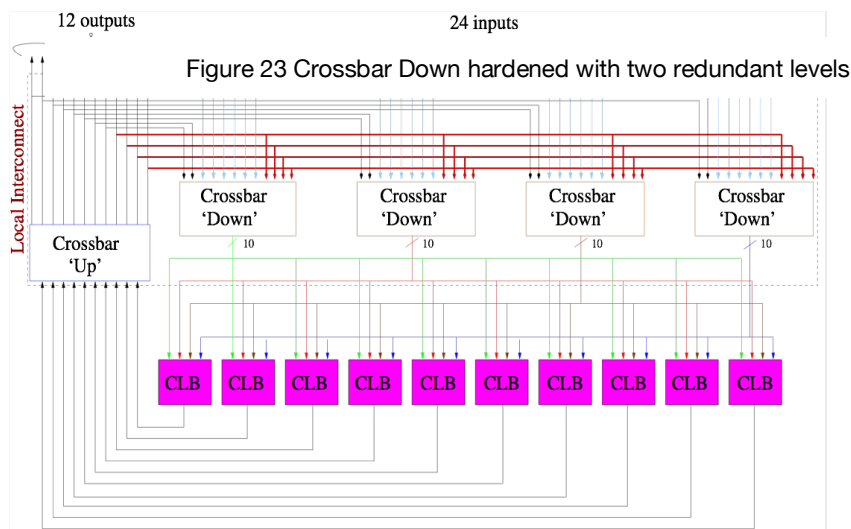
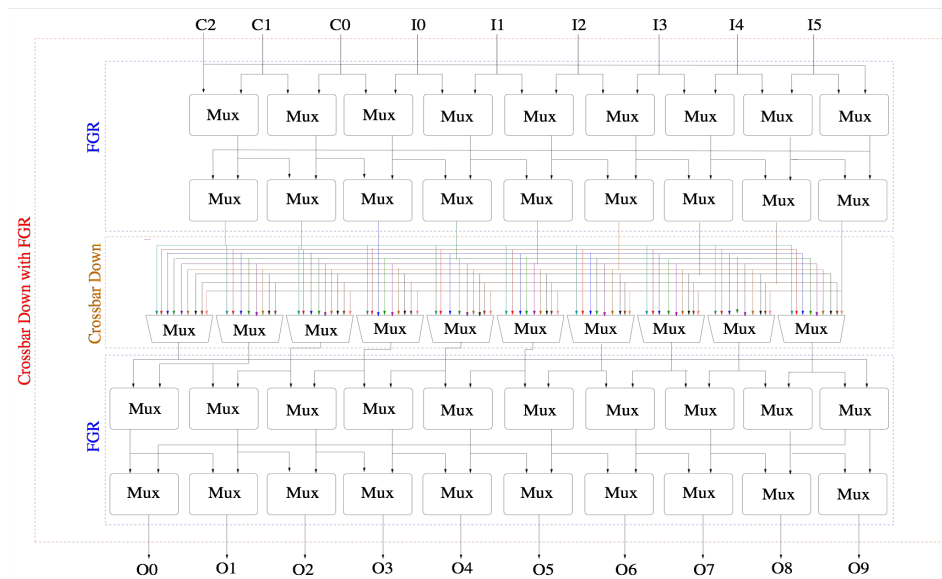


Figure 24 Hardening through additional Distributed feedback signals

5.3.4 Developed tools for the CAD flow

Another major contribution of this thesis is the development of tools to integrate the BIST schemes into the standard FPGA CAD flow. As a matter of fact, the commercially available FPGA tools do not support any FPGA architecture other than their own. Therefore, implementing the proposed BIST schemes using non-commercial tools for a given FPGA comes with challenges. For this purpose, a set of tools were developed which translates the BIST algorithms into files compliant with non-commercial FPGA tools and integrate them into the VTR FPGA flow. VTR (Verilog-To-Routing) Project is a reliable open source platform developed by multiple university research groups for FPGA research purposes. Figure 25 summarize the final bitstream generation flow based on VTR and our tools.

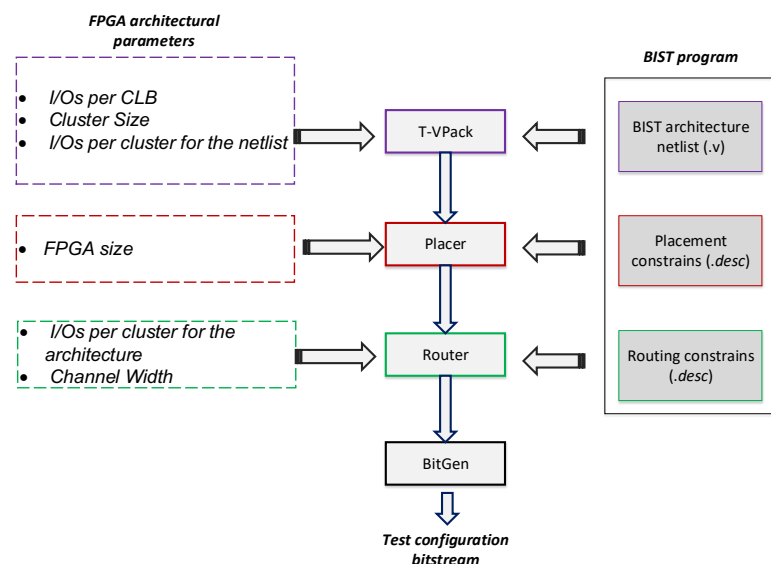


Figure 25 BIST integrated into the VTR FPGA CAD flow

To perform qualitative analysis, another set of tools were developed and integrated into what we called a verification flow, using Synopsys TetraMAX® for fault simulation and Design Compiler® for synthesis. Figure 26 shows an overview of the whole steps involved in this process.

Finally, a last set of tools, not detailed here allows to do the fault mapping. The tool takes as inputs on one side information coming from the fault simulation (Tetramax) and on the other side the BIST configuration bitstreams. At the end, a faulty mismatch detected by an ORA allows to locate a faulty mux or a faulty gate.

5.3.5 Outputs

We made multiple simulations of the BIST algorithms for various FPGA architectures: FPGAs with different cluster sizes as well as FPGAs enriched with different fault tolerant techniques. Testability was evaluated in terms of fault coverage and in test time (number of test configurations). Moreover, the cost of defect tolerant FPGA in terms of routability and defect avoidance was also considered to fully analyze the overall testability. Similar experiments were conducted by performing scan-based simulations to have a comparative analysis of the proposed BIST against standard scan-based technique. As expected, our BIST approach, as for the scan-based approach, allowed to reach a 100% of stuck at and bridging fault coverage for the initial FPGA architecture. As also expected, using the hardened Butterfly LUT, the fault coverage decreased whatever the DFT approach used: 83.68% using scan and 85.6 % using our BIST approach, slightly better. When using the two other fault tolerant techniques (FGR and DF), the fault coverage remains at its maximum (100%). The increase in test time introduced by FGR using scan methodology is 15% while using BIST it just increases to 4%.

As a conclusion of the project, results showed that the cluster of size 12 (12 CLBs per cluster), offers a better routing at a relatively low-test cost along with better robustness with the FGR technique.

The project was a real success, leading to a testable and more robust FPGA architecture.

The here before described works, related to the test and diagnosis, were conducted by Saif-Ur Rehman, in the framework of his PhD thesis. Saif was co-supervised with Lorena Anghel. This work was done in collaboration

with academic (LIP6 and Telecom ParisTech) and industrial partners (FlexRAS). The different solutions and more detailed results were published in [C07], [C08], [S13], [S14], [S15] and [P01].

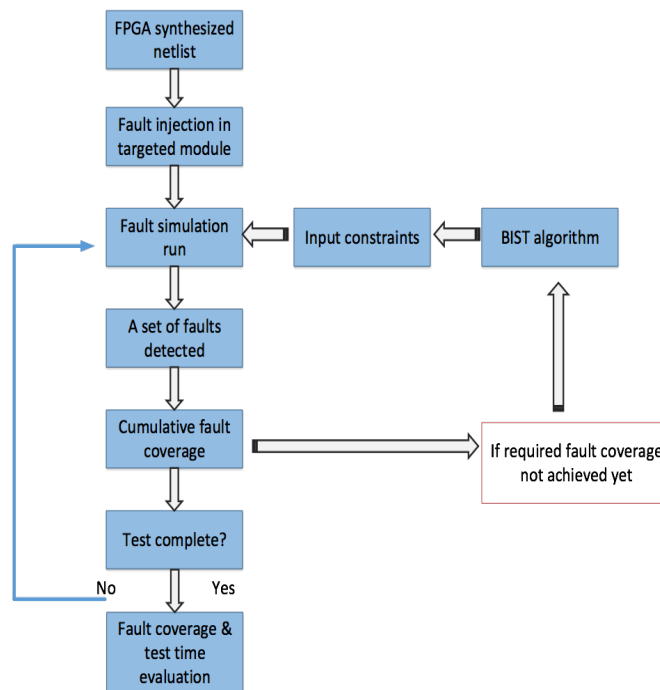


Figure 26 The verification and evaluation Test CAD flow

6. Approximate Computing

6.1 Introduction

In the previous sections the targeted applications are those requiring both very high reliability and exact outputs. This is mandatory for many critical applications (health, transports, ...) where a high quality of service must be guaranteed. However, these requirements can be relaxed for some classes of application.

As explained in the introduction, the Approximate Computing paradigm ([35-36]) is of special interest since in the last decade the rapid rise of the IoT and the increasing energy inefficiency of silicon devices have made the power (and bandwidth) issue a very hot topic.

AxC is often used to trade accuracy of application outputs for a reduction in power or energy consumption.

In 2018, I published with several colleagues a survey on the state of the art of AxC techniques [J06]. Basically, the techniques can be roughly classified as: AxC software (reduced iterations in an algorithm, OS task management, ...), AxC architecture (approximate processors with approximate arithmetic blocs, ...) or AxC circuits (reduced supply voltage of components, imprecise logic, ...).

Positioning over state of the art

These very last years, the efficiency of the AxC techniques is discussed in the scientific community [37]. Indeed, the proposed solutions are either too complex to implement (or control) or often dedicated to a SoC sub-block. Moreover, most of the proposed approaches select and implement only one level of accuracy, at design time, after a more or less complex design space exploration for the operands' format.

To circumvent these drawbacks, we introduced the new idea of adjustable precision where, depending on the available energy, the computing may be processed with different operand precisions, varying during the execution time. This can be very useful for applications where availability is the most important criteria.

We explored this idea through two different works: one targeting arithmetic operands (for dedicated co-processors) and the other targeting the floating-point management (for CPU and FPU). I give hereafter a brief overview of the main ideas of these two research developments.

6.2 Adjustable precision using redundant arithmetic

6.2.1 Introduction

For applications where the power consumption is the first issue (limited battery storage or energy harvesting) and where low latency is not a strict requirement, an idea can be to generate serially the output in multiple cycles chunk by chunk. If full precision is required then we must wait for all the chunks to be computed. If an error tolerance is allowed (here a reduced precision means a reduced number of bits), then the last chunk(s) can be skipped and the processing is stopped sooner. But for that the computing hardware must deliver the most significant bits (MSBs) or chunks at the first place. Using the classical binary system this is not possible or not affordable. However, this is possible using the redundant arithmetic. We thus explored how this arithmetic can be used to design hardware operators allowing to provide an output with adjustable precision, at will.

Redundant arithmetic is based on the usage of another numbering system than the classical binary system. In 2-basis, we talk about Signed Binary Digits (SBD) [38]. An integer A is represented by the equation:

$$A = \sum_i a_i \cdot 2^i \text{ with } a_i = \{-1 = \bar{1}, 0, 1\}$$

Every SBD a is represented by two bits a^+ and a^- such that $a = a^+ - a^-$. Three values are thus possible for an SBD $\{\bar{1}, 0, 1\} = \{01, 00 \text{ or } 11, 10\}$ with two possible codings for 0. A number can have several SBD codings, for instance using 4 SBDs, the decimal value $-5 = \bar{1} 011 = 0\bar{1} \bar{1} 1 = 0\bar{1} 0\bar{1}$ (i.e. $-5 = -8+2+1 = -4-2+1 = -4-1$).

Therefore, this arithmetic is defined as redundant. Passing from A to $-A$ is done easily for example by inverting the bits of each SBD.

6.2.2 Basic operators

Adding (subtracting) an SBD $a = a^+ - a^-$ with one-bit b is done by a bloc named PPM for Plus-Plus-Minus (MMP for Minus-Minus-Plus), shown in figure 27.

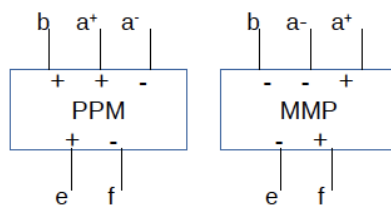


Figure 27 Adder and subtractor using the redundant arithmetic

These are similar to the Full-Adder (FA) block widely used in traditional binary numbering system operators. The two outputs e, f satisfy the relation $2e - f = b + a - a^-$ (resp. $-2e - f = -b + a - a^+$) giving the logic equations:

$$e = \text{Maj}(b, a^+, \bar{a}^-) \text{ resp. } e = \text{Maj}(b, a^-, \bar{a}^+) \text{ and } f = b \text{ xor } a^+ \text{ xor } a^-$$

These basic blocks can then be used to design different arithmetic operators such as the parallel adder/subtractor, or the serial version depicted in figure 28. In the serial version a delay is introduced and implement with flip flops (D).

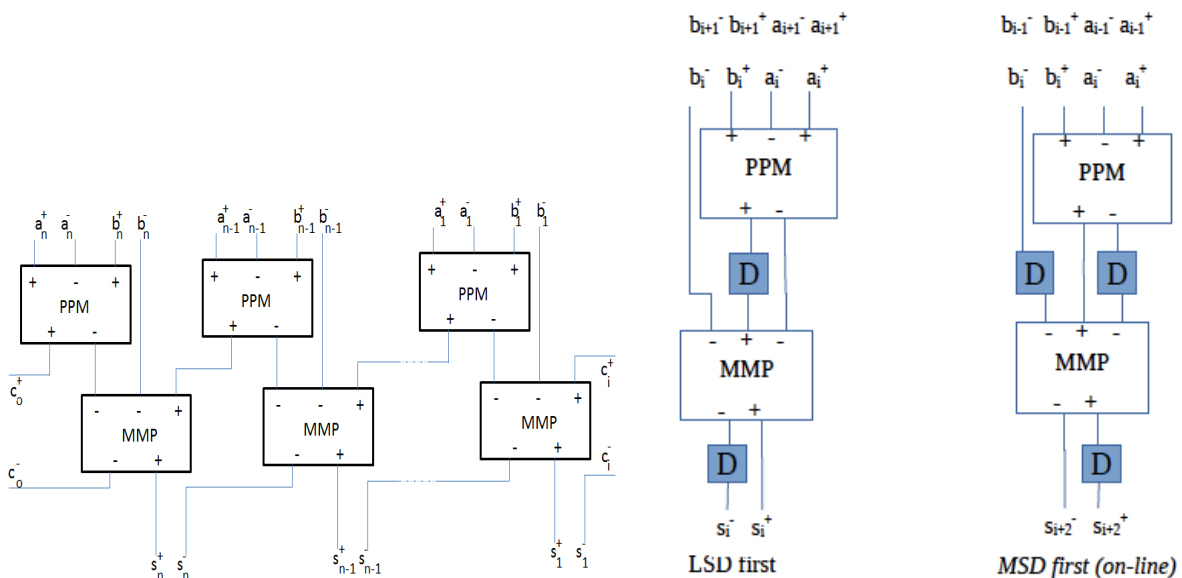


Figure 28 Two-Input SBDs parallel adder/subtractor and two versions of a serial adder

Redundant arithmetic allowed us to conciliate high performance with low consumption, as computations are performed without any global carry/borrow propagation. Indeed, when looking at the 2-input redundant adder/subtractor of figure 28, we see that the carry is only propagated locally. The arithmetic operation is executed thus in a constant time, regardless of the operand representation size. Furthermore, it allowed us to build serial MSD first (on-line) operators.

Using this arithmetic, we designed a library of on-line operators such as multipliers and MACs (Multiply Accumulate), with different input sizes.

For MACs, a precision signal is needed to set the precision of the intermediary multiplications.

6.2.3 Outputs

To demonstrate the advantages of the proposed adjustable operators, we designed a series of versions of the on-line MAC for different operand sizes ranging from 4 to 128 SBDs. The VHDL description was synthesized and mapped into the AMS 350nm CMOS technology. We have also synthesized the equivalent classical parallel MAC for operand sizes from 4 to 128 bits. The comparison of the two sets of implementations, in terms of occupied area, maximum frequency and power consumption showed that above a 16 bits operand size, the redundant version is much more interesting in all aspects. However, one must keep in mind that the output results is not done in one cycle.

An asynchronous version of this MAC has been proposed and developed by the CIS team of TIMA and showed even better power consumption gains since it can benefit from the event driven paradigm.

In summary, we developed a new adjustable precision computing hardware, taking advantages of different features of the redundant arithmetic. The strength of our approach is the adaptation of the architecture to the application. Indeed, the computation can be done in parallel, serially or partially serially (chunks); starting from LSD or MSD.

The proposed approach provides a nice alternative to approximate computing, by allowing the user to dynamically adapt the computation precision to the application environment as the available energy for instance. In addition, the MSDs are first obtained, and can be exploited without waiting for the end of computation as in classical binary operators.

The computation can be stopped when the required precision is reached without the need of performing the whole computation. This would save the corresponding required energy and can be used in IoT applications when the available power supply is at a critical level.

The only subsisting difficulty might come from the cost of the transition between the binary and redundant notation systems. For systems containing a chain of binary arithmetic operators, a recommended utilization scheme would be to rapidly pass from binary to SBD encoding, perform all the computation chain in SBD and move back to binary as a final step.

As a future work we plan to use this approach to build bigger blocks such as digital filters and processors to have a clear evaluation at system level.

This work has been done through the collaboration of two teams of the TIMA laboratory, AMFoRS and CIS. It involved Ali SKAF (senior researcher) and Laurent Fesquet (Associate Professor) for CIS and myself for the AMFoRS side. This work involved also two Master 2 students in the framework of their internship: Mona Ezzadine and Alessandro Fici, I co-supervised with Laurent Fesquet and Ali Skaf.

More details of the approach can be found in [J08], [C09], [W12].

6.3 Adjustable precision in Floating Point Unit.

6.3.1 Introduction

Standard precision floating point computing units (FPU) consume a significant portion of the energy resources in processor-based architectures. But as said before, many modern applications using floating point operands have an inherent ability to tolerate a loss of precision with little or no impact on output results. This characteristic has allowed a sub-field of the Approximate Computing paradigm to emerge, dedicated to floating point management: the Transprecision Computing [39]. It appears that for such applications, IEEE standard floating-point formats are often overkill.

Several tools and techniques to explore the trade-offs between accuracy and power consumption have been proposed in the state of the art. The objective is to help program designers find the most optimized code for a given application.

However, existing techniques require developers to rewrite some or all of their software bricks (applications, libraries, operating systems, etc.), which is often infeasible, complex or at least requires a significant development effort.

In addition to their intrusive nature to the code, most of these tools are only designed for simulation and do not propose the corresponding hardware architectures. This does not allow to evaluate the potential energy gains at hardware level.

Finally, most of these works propose reduced but static precision, i.e., the precision is set once for all during the entire runtime. Therefore, FPU hardware architectures need to have the ability to operate in reduced and variable precision in order to increase their performance and energy efficiency when the context allows error tolerance.

In this work we proposed several contributions that overcome some limitations of the state of the art. First, we propose a quite non-intrusive approach that exploits approximate floating-point computations for power reduction, execution time and energy consumption. This approach introduces approximations at the instruction execution/interpretation level. This allows to handle all types of executable binaries (bare-metal applications, single-/multi-threaded user applications, OS/RTOS, etc.) without the need for specialized compilers or specific instructions. The methodology supports arbitrary reduced precision (ARP) (non-standard floating-point representation) as well as variable precision in time (VPT) which provides the ability for applications to adjust their operational precision at runtime.

Subsequently, a software implementation based on the well-known dynamic binary translator QEMU was developed. It simulates the impact of these approximate calculations on the quality of the result (QoR) of the applications.

Finally, a multi-precision hardware FPU was designed in 28nm FD-SOI ASIC technology to support VPT by instantiating multiple approximate computation blocks in hardware, thus achieving significant energy efficiency gains in hardware.

The proposed methodology and its implementations have been jointly evaluated in software and hardware on a set of benchmark applications, with a focus on iterative methods. The experiments show how very fine adjustments of the accuracy can be achieved with ARP and VPT, without loss of quality of the result, to generate

significant savings in terms of energy consumed during computation. And with an acceptable level of quality loss on the output results, energy efficiency is further increased.

6.3.2 The proposed approach

To address the major limitation of state-of-the-art tools, namely intrusiveness to the source code, we propose to apply the approximations of floating-point calculations at runtime when interpreting/executing the binary instead of manually embedding them directly into the source code.

Figure 29 shows the execution flow for an assembly instruction `fmadd.d`. This RISC-V instruction performs a merged/fused multiply-add (FMA). At the input of the flow, we have the source code of a software application, in our case, in C/C++. This application is then compiled using the cross-compiler of the targeted architecture, in our case, RISC-V. Once the binary is produced, it is either executed on a real RISC-V machine or interpreted on a simulator. In both cases, calculations are either performed exactly using standard IEEE types or using a floating-point format (E, M) with arbitrarily reduced precision (ARP).

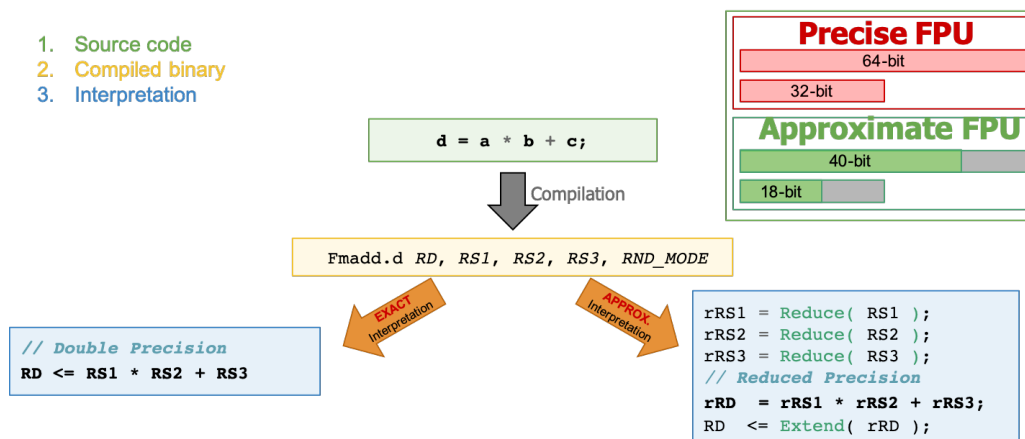


Figure 29 The proposed approach

A floating-point format is defined by the pair (E, M), where E denotes the exponent size and M denotes the mantissa size (in number of bits). The parameter E influences the dynamics of the representation while M impacts the precision of the representation.

We distinguish two modes of execution or interpretation: the exact mode, where the floating-point assembly instructions are executed as usual on a standard IEEE FPU, and an approximate mode, where the instructions are executed in three steps: 1) the inputs are first reduced to the targeted reduced format, 2) the computation is performed on a reduced-precision FMA operator, and 3) the result is converted from the reduced format to the standard format (to fit back the system format). The gain in average power, execution time, and energy consumption is due to the step 2. The counterpart is that the hardware FPU must support both the standard formats and the arbitrary reduced format (as can be seen in figure 29). This will induce a significant area overhead, but as we will see the approach allows to save significant energy consumption.

To support the variation in accuracy at runtime (the VPT), the proposed method can also support multiple approximate modes 1, 2, ..., N with different reduced formats associated (E₁, M₁), (E₂, M₂), ..., (E_N, M_N). This is illustrated in figure 30. In this case, the software application could adjust its calculation precision according to its needs as it goes along.

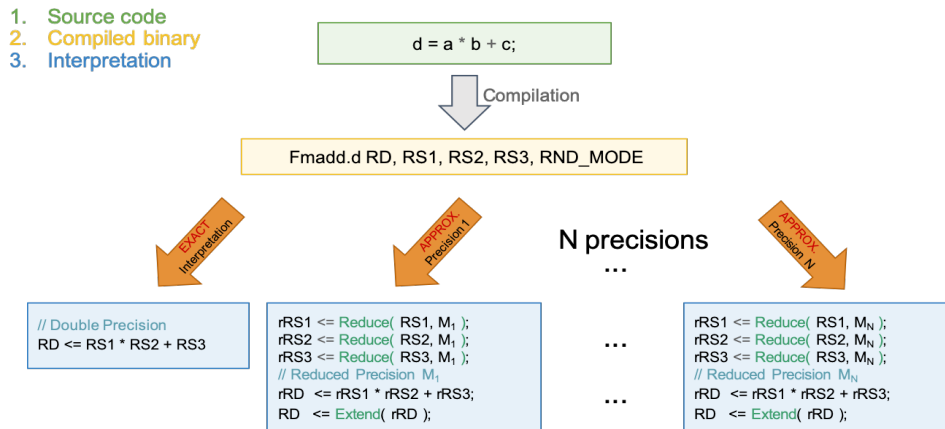


Figure 30 The approach for multiple precisions

6.3.3 Software implementation: AxQEMU, the impact simulator

To implement and evaluate the approach we developed AxQEMU: a minimally intrusive simulator.

The AxQEMU tool is based on the well-known open-source simulator QEMU, a functional simulator that dynamically interprets an executable binary compiled for a certain architecture (in our case RISC-V) on a host machine (in our case x86). It is mainly used to verify the operation and the results of an application in a fast way. Its efficiency, its performance, its design, and its good reputation in the industrial and academic world led us to choose it as the simulation tool for our method. The resulting tool, AxQEMU, has been put in open source.

AxQEMU allows to study the impact of the precision reduction on the numerical results of a software application. Indeed, given an application and a set of inputs, the application is simulated using several reduced precisions. The results are then collected and plotted as a heat map that shows the degradation in the quality of the results (QoR) as a function of the parameters of the chosen (E, M) format.

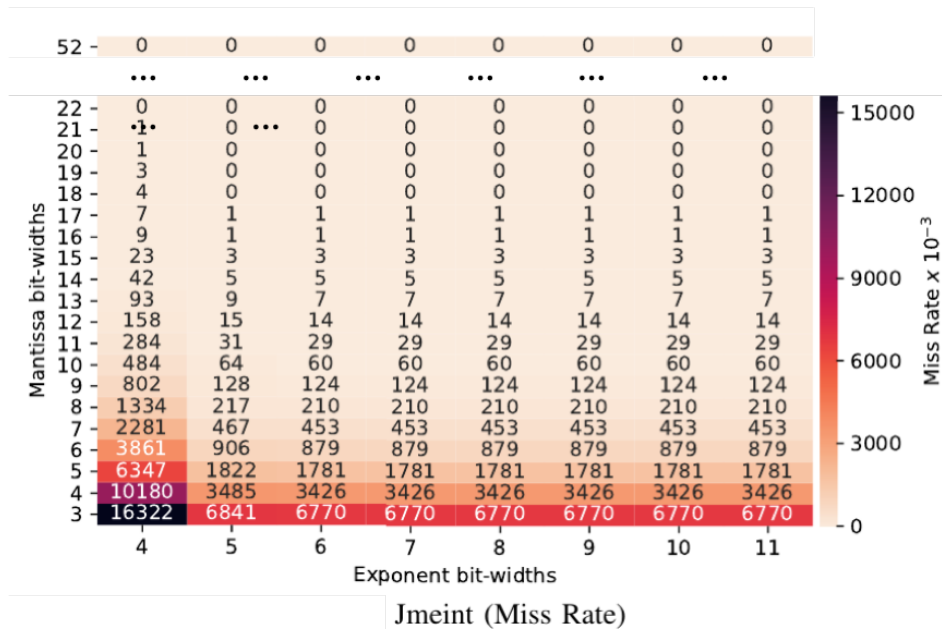


Figure 31 Heat Map obtained using AxQEMU

For example, figure 31 shows the variation in result quality for the application Jmeint often used in video game applications. This application takes as input two triangles in space (defined by the coordinates of their three points), and determines if there is an intersection between them or not. The result being binary, the error rate is then expressed as a Miss Rate. The error rate varies with the size of the exponent E in abscissa and the size of the mantissa M in ordinate. In this example, we showed that with input variables with 24 bits ($E=5 + M=18 + 1$ for the sign) there is no accuracy loss. The standard 64 bits is thus over-designed. We showed also that for an image classification application based on CNNs, 32 bits are also overdesigned, 8 bits are enough.

Thus, AxQEMU allows designers to find the optimal width of the exponent and mantissa for a given error rate or threshold. This design space exploration tool is very powerful and by construction it can be used alone or in conjunction with other state of the art tools and techniques. We have done several experiments on error tolerant applications in the domains of video, gaming and artificial intelligence, showing that most of the time the standard floating-point formats are overestimated.

6.3.4 Hardware implementation: The multi-precision VPT FPU

To evaluate the impact of our approach we designed the new multi-precision FPU, called VPT-FPU, which allows approximate floating-point computations with variable precision at runtime.

The VPT-FPU is described in SystemVerilog, and is synthesized with a 28nm FD-SOI technology. Its architecture and internal blocks are presented in figure 32.

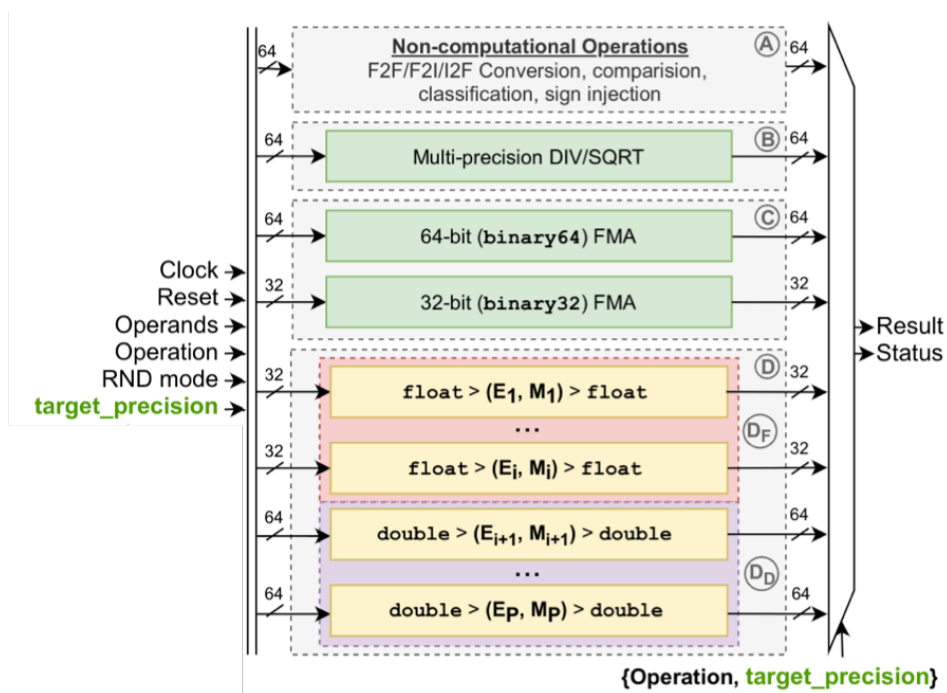


Figure 32 The proposed hardware multi precision FPU

This architecture consists of a part A, where all non-computational operations such as conversions, classification, and comparison are performed. Then there is a block B containing a division and square root unit with adjustable precision, developed by ETH Zurich. Next to these blocks is a block C containing the IEEE standard precision FMA operators (32 and 64-bit formats). These three blocks constitute the standard FPU against which we will compare our FPU. To support dynamic precision variation, another block D has been added to the standard FPU, and it contains two data sub-paths D_F and D_D in which the float and double operations are adjusted,

respectively. Each of these sub paths consists on a precision reduction unit, a precision-reduced FMA unit, and a precision extension unit that converts the result to the standard system format.

The reduced formats supported by the VPT in hardware are configurable during the synthesis stage. Then, at runtime, the software application selects the precision of the calculations from those supported in hardware through a set of programmable control registers, without the need for special instructions or specialized compilers. These registers drive signals such as the “target_precision” signal depicted in figure 32.

To evaluate the gains in terms of power dissipation, execution time, and total energy dissipation, a hardware-level evaluation is performed for several applications by directly applying our methodology.

The power consumptions gains are obtained through the deactivation of non-used data paths (clock gating). The area overhead depends on the number of precision modes and the bit width of the operands.

To give an idea of what to expect, for the Jmeint application, with one additional precision mode (obtained thanks to AxQEMU), the area overhead is of 8.5% for 70% of energy savings with no quality loss.

In this approach the area overhead is the main drawback. But if the standard formats are not required, then the bloc C in figure 32 can be removed, letting only the approximation mode, and thus significant area savings can be reached. This can be the case if the device is an ASIC dedicated to only one application. This can also be the case if the microprocessor and its FPU are mapped on a FPGA for a dedicated optimized application.

6.3.5 Self-adaptive precision for iterative algorithms

To prove the usefulness of our approach for variable precision in time, we targeted scientific applications as a use case.

We focused on two iterative methods, Jacobi and Gauss-Seidel, which solve so-called stationary linear systems of the form $Ax = b$, where A is a $n \times n$ matrix, b and x are vectors of one column and n rows, with x being the unknown.

We proposed two simple modifications of the original algorithms to take advantage of the new multi-precision FPU.

The idea is to start the computations with a very low precision, for example 4 bits of mantissa. The precision is then incremented as the algorithm advances in the calculations. After each iteration, the Euclidean distance between the result of an iteration $k+1$ and the result of the previous iteration k is evaluated. When this distance is small enough, the precision is incremented to 8 bits. Then, the calculations (iterations) continue until the distance falls below another predefined threshold, once this condition is verified, the precision is incremented once again. The process continues until the algorithm converges, thus allowing for self-adaptation of the accuracy without programmer intervention.

This method poses a number of challenges, such as the choice of intermediate thresholds and convergence stagnation. We proposed measures to solve these challenges and we also perform a detailed statistical evaluation on many inputs to study the hardware gains (average power, execution time and total energy dissipated) of the proposed algorithms.

For the Jacobi algorithm, without loss on the result quality, we demonstrate energy consumption savings in the range of 85% to 89 %, for a number of supported precisions varying respectively from 4 to 10. Respectively, the

area overhead varies from 19% to 163% when compared to the reference implementation in double precision. This shows that a small number of precision modes is more energy and area efficient.

This work has been done in the framework of the PhD thesis of Noureddine Aitsaid, co-supervised with Katell Morin Allory, Associate Professor at Grenoble INP-UGA. This work involved also Antoine Maillefert in the framework of his internship.

More details on the approach were published in [J09], [J10], [C10], [S16].

7. Perspectives

7.1 Preamble

Through these two last decades, I acquired a sum of technical knowledge on design and reliability of nano-electronic devices, at bloc and system level. As can be seen in the previous section, i focused my research on improving the quality (test) and the reliability (fault tolerance) of embedded system. For that I had to understand hardware and software issues and how each interact one with another. My intent for the future is to use my expertise and skills to propose new solutions for the design of reliable and energy efficient chips, taking into account new paradigms and technology improvements.

7.2 Introduction

The society is facing multiple challenges such as global warming and resources (energy and materials) shortages. It is thus mandatory for the future to develop sustainable and thrifty electronic devices. At the same time the exponential growth of the IoT in all application domains is pushing towards more energy consumption.

Thus, there is a clear need to make significant breakthroughs in the design of energy efficient devices.

Even with the last technology improvements, the power and memory wall are still the major obstacles faced by design engineers to produce powerful and efficient devices. These last years, in the scientific community, new computing paradigms have arisen with the clear intent to improve the performance and energy efficiency of the electronic devices: stochastic computing, approximate computing, neuromorphic computing, quantum computing which are mostly based on non-Von Neumann computing architectures.

Personally, I am interested in non-von Neumann architectures. Indeed, from my point of view, this paradigm could be a game changer to greatly improve energy efficiency and performance (bandwidth), especially since the emergence of new materials would allow to implement this paradigm in an efficient way. One of the domains that could strongly benefit from this paradigm is the domain of artificial intelligence (AI). Many AI products have been released in the very last years, but the research activities related to their quality and reliability are still in their infancy and there are thus important research perspectives.

Another research path I want to follow is approximate computing because in my opinion many current and future applications do not need to have an exact result. Hardware and software architectures will have to be optimized to improve their energy efficiency. In particular, artificial intelligence applications have certain features that can be exploited to adjust accuracy, at design time or on demand, at run time.

Finally, the third topic that seems promising is the design of hardware accelerators using non-traditional binary representations. Some of them, such as the redundant arithmetic, previously introduced in this document, exhibit properties that may be exploited to build more frugal hardware. However, questions arise with regards to their reliability, issue that has not been tackled yet.

These three research fields, I plan to work on in the coming months and years, are not orthogonal. Merging ideas coming from the non-Von Neuman and Approximate computing paradigms, and taking benefit of other data representations, may lead to efficient and reliable hardware solutions. I hereafter give for each research field an overview of what is forecasted.

7.3 Perspectives on Design, Test and Fault Tolerance of Neuromorphic applications

7.3.1 Design of AI HW accelerators using emerging Non-Volatile Memories (NVMs)

The Von Neuman architectures are based on the strict decoupling of computation, communication and storage functions. Some emerging memory technologies offer the possibility to merge these three functionalities in a single place/component (In Memory Computing), or at least to shorten the distance between their respective implementations (Near Memory Computing). Their use to implement AI applications seems then very promising to boost the performance and reduce the power consumption.

In that context, in the short term I will focus my work on the design of AI hardware accelerators, at circuit level. I will consider NVM technologies such as MRAM, ReRAM and FeFET (Ferro electric FET) technologies to build computing operators and evaluate their efficiency and reliability.

A common work with the INL laboratory at Lyon has started this year on hardware design using their new Ferro electric FET transistor (FeFET) technology. After modeling the behavior of the base transistor, with regards to different functional and electrical parameters, we plan to design and evaluate two kind of basic operators:

- Fully digital (for Logic in Memory)

The right combination of multiple FeFET transistors allows the creation of digital logic gates. These gates can be used to build complex AI computing blocks with weights stored at low cost, thanks to the non-volatility.

- Mixed, analog and digital (for In Memory Computing)

AI applications are often based on convolutional neural networks (CNN) and require calculations involving many MACs (Multiplier Accumulator). These MAC operators can be realized with ReRAM or FeFET memory point arrays, arrays which can be programmed, for example, to store in time the different synaptic weights. The MAC operation is done in the analog domain and requires dedicated circuit to interface to the digital domain.

With the physical and electrical knowledge provided by this first study in addition to the expertise already available in my research team, in the medium term, I plan to implement AI algorithms using the before mentioned operators and compare their performances and power consumption.

Ongoing and forecasted collaborations: EEA/INL Ecole Centrale de Lyon (FR); LIP6 Sorbonne University (FR); ST Microelectronics (FR); Delft University (NL); LIRMM Laboratory (FR)

7.3.2 Test and Fault Tolerance of AI Accelerators

Recent studies showed that AI applications, such as those based on DNNs, are inherently resilient to small perturbations on their inputs or minor errors on the embedded weights. However, medium to large deviations (due to manufacturing induced defects, aging and environment perturbations), may strongly impact the application accuracy and lead to a catastrophic behavior of the system. With the expected density growth provided by the next transistor generations (such as the Gate All Around-GAA) or by future dense non-volatile memories, the susceptibility to errors of such hardware accelerators will grow and will be a major concern for safety and for critical applications. In this context my plan is to carefully evaluate the testability and fault tolerance of such class

of hardware accelerators, with a close look on the machine learning and inference processes since an error will not have the same impact in terms of reliability if it occurs during the training phase or during the inference phase.

Testability

There are many fault models defined especially for AI applications. In many cases a behavioral study is performed before hardware testing to identify the relevant faults – very few faults have a strong effect on the system, most are masked. Thus, dedicated fault modeling can help achieve better coverage during test or diagnosis. There are even more profound differences when dealing with non-Von-Neumann architectures.

Functional testing differs also for AI accelerators since the behavior of the application is more complex to analyze in the presence of a fault. The inherent plasticity and redundancy of AI implementations are blurring the lines between what is acceptable and what is not. Thus, I will lead studies to better understand the impact of defects in DNN applications, for traditional CMOS and for upcoming emerging materials (such as those described in section 7.3.1). Methodologies will be developed for defect modeling, fault modeling, testability evaluation and design for test.

After that I will focus on in the field testing, since AI connected devices are expected to spread at a fast pace in all application domains. As the criticality of the application dictates the requirements for test and reliability, new efficient methods must be developed to allow online and offline self-test of such devices. Exploiting the right AI properties to develop smart functional test (hardware and software based) is a key challenge.

Fault tolerance – towards self-healing AI

Once a fault/error is detected, recovering mechanisms must be put in place. This can be done at different levels of the design layers. My intent is to propose and evaluate new strategies for the different layers.

A first study will be conducted at algorithmic and software level enabling the implementation of new and existing fault tolerant solutions, such as training the DNN in presence of errors, and other Algorithm Based Fault Tolerance (ABFT) techniques. The evaluation of the techniques' efficiency will require the development of fault injection and simulation platforms able to provide the adequate metrics.

DNNs algorithms are naturally resilient to errors, but their hardware implementation may modify this property. The technology, circuit and architecture choices will have an impact on the reliability of the application. The mapping of an algorithm on different hardware types may increase or decrease the fault tolerance. I will explore this issue through the implementation and evaluation of the fault tolerance of AI algorithms mapped on traditional CMOS and on new emerging structures. This will allow to select the best design practices for more reliable applications. Here also an appropriate evaluation infrastructure must be put in place.

Forecasted collaborations: EEA/INL Ecole Centrale de Lyon (FR); Delft University (NL); LIRMM Laboratory (FR); University of California, Irvine (US)

7.4 Perspectives on Approximate Computing

7.4.1 Relationship between Approximate Computing and Fault Tolerance

Fault Tolerance and Approximate Computing target error tolerance but not in the same way, these two paradigms being fundamentally different. One intends to mitigate unexpected faults surging during or after manufacture, the other intends to optimize the design for an expected output error tolerance (QoR). However, we observe that AI applications exhibit also inherent resilience to error. This paves the way for the design of even more efficient

(energy and performance) hardware accelerators. Recent works have shown that data types are often over designed and simplified coding (BNNs and TNNs) can lead to less complex hardware networks, with a smaller footprint, better performance and reduced power consumption, with a very small impact on the outputs' accuracy.

However, reducing the precision of the processed data (reduced operands' bit-width for example) will probably decrease the global reliability of the application since a fault (SEU or aging for instance) will probably have a greater impact on the output correctness. I expect that tuning the precision knob will in an opposite way modify the fault tolerance of the AI hardware but this must be proven and quantified to allow a realistic implementation of approximate techniques without sacrificing too much the reliability. A set of experiments and methods must be developed at all layers of the design flow.

Forecasted collaborations: EEA/INL Ecole Centrale de Lyon (FR); LIRMM Laboratory (FR); TIMA Laboratory (SLS team) (FR)

7.4.2 Approximate computing for CMOS based architectures

When looking at AI algorithms, especially those based on CNNs, we can consider that they are already, naturally approximate. It was shown that some layers don't need a maximum precision to deliver an exact output. The challenge is to finely tune the precision of each layer and/or each arithmetic operator by choosing the right type and size of variables (float, int, or arbitrary non-standard bit-widths). The difficulty is to find the right architectural parameters targeting Quality Of Results (QoR) and for performance levels. This can be done in a first step by a series of trial and error iterations allowing to converge towards a tryptic (performance, QoR, power consumption). Ideally it may also be done during the training phase leading to a one shot final architecture with ideal parameters

In the short-medium term, the first method (trial-error) will be favored by using an architectural exploration tool developed at TIMA in the SLS team. This tool is based on a description of the architecture in Chisel language to simplify and automate the generation and simulation of different versions of the architecture with different custom sizes for the operands. The experience acquired by the SLS team of TIMA these last years on AI architectures and operand optimization will be very helpful. The challenge will be then to add the fault tolerance requirements upon and/or within the optimization flow.

Forecasted collaborations: TIMA Laboratory (SLS team) (FR)

7.4.3 Approximate computing for NVM based architectures

As mentioned in section 7.3.1, AI hardware accelerators can be designed with emerging non-volatile memories. However, contrarily to the previous subsection, the mapping of an algorithm will have to take benefit of the non-Von Neuman paradigm. Recent works have proposed some mapping methods to deploy an algorithm on memory cell arrays implementing near or in memory computing. To optimize the software / hardware resources' usage, approximate computing techniques will surely have a role.

Near/In memory and approximate computing are not mutually exclusive, I expect that taking the best of each will lead to very significant energy savings. This once again must be proved and evaluated. For that, in the medium term, after designing basic operators with memristors and FeFET transistors, i will study how these operators can be assembled and tuned to deliver outputs with different QoRs. The next step will be to propose AI hardware architectures with adjustable static and dynamic precision.

Forecasted collaborations: EEA/INL Ecole Centrale de Lyon (FR); LIRMM Laboratory (FR)

7.5 Design of reliable hardware using non-traditional arithmetic

Design (non-Von Neuman) and technology (NVM) improvements are expected to boost the performance and energy efficiency. However, new data representations for computing are also emerging that aim at the same goals. In a long term I plan to study stochastic (and quantum) computing, but first I plan to push further the developments related to the usage of redundant arithmetic (see section 6.2).

Indeed, the first results showed that this non-traditional arithmetic is fully compatible with the idea of adjustable precision. It enables significant energy savings for applications that tolerates latency or reduced precision.

Moreover, the inherent information redundancy would make it suitable for the design of fault tolerant architectures. However, this redundancy feature may lead to less testability.

Thus, there is a need to evaluate the fault tolerance and testability of architectures based on this redundant arithmetic.

In the short-medium term, beside the optimization of the design features (QoR, performance, energy consumption, area, ...), I will evaluate the advantages and drawbacks of this arithmetic usage, with regards to the reliability and quality, in comparison with designs based on traditional binary arithmetic.

Ongoing and forecasted collaborations: TIMA Laboratory (FR) (CDSI Team)

8. References

Here follow the references for this detailed description of my research activities. For sake of conciseness and clarity, this list is non-exhaustive.

- [1] Y. Zorian. Testing the monster chip. IEEE Spectrum, pages 54–60, July 1999.
- [2] E. J. Marinissen, Y. Zorian, R. Kapur, T. Taylor, and L. Whetsel. Towards a standard for embedded core test: an example. In IEEE International Test Conference (ITC), Atlantic City (NJ), USA, September 1999.
- [3] E. J. Marinissen, R. Arendsen, G. Bos, H. Dingemans, M. Lousberg, and C. Wouters. A structured and scalable mechanism for test access to embedded reusable cores. In IEEE International Test Conference (ITC), Washington (DC), USA, October 1998.
- [4] P. Varma and S. Bhatia. A structured test reuse methodology for core-based system chips. In IEEE International Test Conference (ITC), Washington (DC), USA, October 1998.
- [5] L. Whetsel. Addressable test ports an approach to testing embedded cores. In IEEE International Test Conference (ITC), pages 1055–1064, Atlantic City (NJ), USA, September 1999
- [6] F. Como, M. S. Reorda, G. Squillero, and M. Violante, "On the Test of Microprocessor IP Cores," in Proceedings Design, Automation, and Test in Europe (DATE), Munich, Germany, Mar. 2001.
- [7] L. Chen and S. Dey, "Software-Based Self Testing Methodology for Processor Cores," IEEE Transactions on Computer-Aided Design, Mar. 2001.
- [8] N. Kranitis, G. Xenoulis, D. Gizopoulos, A. Paschalis, and Y. Zorian, "Low-Cost Software-Based Self-Testing of RISC Processor Cores," in Proceedings Design, Automation, and Test in Europe (DATE), Munich, Germany, Mar. 2003.
- [9] L. Chen, X. Bai, and S. Dey, "Testing for interconnect crosstalk defects using on-chip embedded processor cores," in Proceedings of the 38th conference on Design automation. ACM Press, 2001, pp. 317-320.

- [10] W.-C. Lai, J. -R. Huang, and K. -T. Cheng, "Embedded-Software-Based Approach to Testing Crosstalk-Induced Faults at On Chip Buses," in Proceedings IEEE VLSI Test Symposium (VTS), Marina del Rey, CA, May 2001, pp. 204-209.
- [11] S. F. M.H. Tehranipour, Z. Navabi, "An efficient BIST method for testing of embedded SRAMs," in Proceedings International Symposium on Circuits and Systems (ISCAS), vol. 5, May 2001, pp. 73-76.
- [12] R. Rajsuman, "Testing a System-on-a-Chip with Embedded Microprocessor," in Proceedings IEEE International Test Conference (ITC), Atlantic City, NJ, Sept. 1999, pp. 499-508.
- [13] M. Keim, "Thinking About Adopting IEEE P1687?," in *IEEE Design & Test*, vol. 30, no. 5, pp. 36-43, Oct. 2013, doi: 10.1109/MDAT.2013.2278542.
- [14] G. M. ALMEIDA, G. SASSATELLI, P. BENOIT, N. SAINT-JEAN, S. VARYANI, L. TORRES, and M. ROBERT, "An adaptive message passing mp soc framework," *International Journal of Reconfigurable Computing*, vol. Volume october 2009.
- [15] S. Das, D. Roberts, S. Lee, S. Pant, D. Blaauw, T. Austin, K. Flautner, and T. Mudge, "A self-tuning dvs processor using delay-error detection and correction," *IEEE Journal of Solid-State Circuits*, vol. 41, p. 2006, 2005.
- [16] D. Sylvester, D. Blaauw, and E. Karl, "Elastic: An adaptive self-healing architecture for unpredictable silicon," *IEEE Design and Test*, vol. 23, no. 6, pp. 484-490, 2006.
- [17] T. Austin, V. Bertacco, S. Mahlke, and Y. Cao, "Reliable systems on unreliable fabrics," *IEEE Design and Test*, vol. 25, no. 4, pp. 322-332, 2008.
- [18] M. A. Olson, K. Bostic, and M. Seltzer, "Berkeley db," in Proceedings of the FREENIX Track: 1999 USENIX Annual Technical Conference Monterey, California, USA, June 6~A_S11, 1999.
- [19] Metakit web site. [Online]. Available: <http://www.equi4.com/metakit/>
- [20] W. Dally and B. Towles, *Principles and Practices of Interconnection Networks*. San Mateo, CA, USA: Morgan Kaufmann, 2004.
- [21] A. Kohler and M. Radetzki, "Fault-tolerant architecture and deflection routing for degradable NoC switches," in Proc. 3rd ACM/IEEE Int. Symp. NoC, May 2009, pp. 22-31.
- [22] D. Fick, A. DeOrio, J. Hu, V. Bertacco, D. Blaauw, and D. Sylvester, "Vicus: A reliable network for unreliable silicon," in Proc. 46th DAC, Jul. 2009, pp. 812-817.
- [23] S. Lin, W. Shen, C. Hsu, C. Chao, and A. Wu, "Fault-tolerant router with built-in self-test/self-diagnosis and fault-isolation circuits for 2Dmesh based chip multiprocessor systems," in Proc. Int. Symp. VLSI-DAT, Apr. 2009, pp. 72-75.
- [24] Q. Yu, J. Cano, J. Flich and P. Ampadu, "Transient and permanent error control for high-end multiprocessor systems-on-chip", Proc. 6th ACM/IEEE Intl. Symp. on Networks-on-Chip, pp. 169-171, May 2012.
- [25] S. Pasricha and Y. Zou, "NS-FTR: a fault tolerant routing scheme for networks on chip with permanent and runtime intermittent faults," Proceedings of the 16th Asia and South Pacific Design Aut. Conf. (ASPDAC '11), Colorado State Univ., Fort Collins, CO, USA, pp. 443-448, Jan. 2011.
- [26] M. Nicolaidis, L. Anghel, N.-E. Zergainoh, Y. Zorian, T. Karnik, K. Bowman, J. Tschanz, Lu Shih-Lien, C. Tokunaga, A. Raychowdhury, M. Khellah, J. Kulkarni, V. De, D. Avresky, "Design for test and reliability in ultimate CMOS," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2012, pp. 677-682, March 2012.
- [27] Lee H-HS, K. Chakrabarty, "Test challenges for 3D integrated circuits". 2009, journal of IEEE Design and Test of Computers 26(5): PP 26-35
- [28] E.J. Marinissen, Testing TSV-based three-dimensional stacked ICs. 2010, proceedings of Design, Automation and Test in Europe (DATE) Conference, pp 1689-1694

- [29] CuvIELLO M, Dey S, Bai X, Zhao Y, "Fault Modeling and Simulation for Crosstalk in System-on-Chip Interconnects", Proceedings of International Conference on Computer-Aided Design (ICCAD), 1999, pp 297–303
- [30] B. Dutton, C. Stroud, "Built-in Self-Test of configurable logic blocks in Virtex-5 FPGAs", Proc. IEEE Southeastern Symp, on System Theory, pp.230-234, 2009.
- [31] J. Yao, B. Dixon, C. Stroud and V. Nelson, " System-level Built-In Self- Test of global routing resources in Virtex-4 FPGAs", Proc. IEEE Southeastern Symp, on System Theory, pp.29-32, 2009.
- [32] C. Stroud, A Designer's Guide to Built-in Self-Test, Kluwer Academic Publishers, Boston, 2002.
- [33] Z. Jianfeng, H. Hu, W. Dong and P. Liyang, "A cost-efficient self-configurable BIST technique for testing multiplexer-based FPGA interconnect," Journal of Electronic Testing, pp. 647-655, Oct. 2011.
- [34] Suthar, " Built-In Self-Test of FPGAs with provable diagnosabilities and high diagnostic coverage with application to on-line testing, Proc. VLSI Test Symposium, 2006.
- [35] S. Venkataramani, S. T. Chakradhar, K. Roy, and A. Raghunathan, "Approximate computing and the quest for computing efficiency," in Proceedings of the 52Nd Annual Design Automation Conference, DAC '15. New York, NY, USA: ACM, 2015, pp. 120:1–120:6.
- [36] Q. Xu, T. Mytkowicz, and N. S. Kim, "Approximate computing: A survey," IEEE Design Test, vol. 33, no. 1, pp. 8–22, Feb 2016.
- [37] A. Sampson, "Closed Problems in Approximate Computing", blog 2017, <https://www.cs.cornell.edu/~asampson/blog/closedproblems.html>
- [38] A. Skaf and A. Guyot: "VLSI Design of On-line Add/Multiply Algorithms" International Conference on Computer Design (ICCD 93), Cambridge, USA, Octobre 1993.
- [39] A. C. I. Malossi and M. Schaffner and A. Molnos and L. Gammaioni and G. Tagliavini and A. Emerson and A. Tomás and D. S. Nikolopoulos and E. Flamand and N. Wehn, "The transprecision computing paradigm: Concept, design, and applications," in 2018 Design, Automation Test in Europe Conference Exhibition (DATE), Mar. 2018, pp. 1105–1110.

Chapter III : CV details

1. Students

1.1 Master 2 students

I supervised 23 master 2 students in the framework of their internship. $\frac{1}{4}$ of the students was co-supervised with my PhD students or with senior colleagues. I don't mention here the numerous Master 1 students I supervised for sake of clarity.

- Bram Lee (2002): Design and simulation of a test co-processor IEEE 1500 compliant
- Emmanuel Viaud (2003): SoC testing using Software Based techniques
- Mehdi Talei (2004): Testability of the AOC SoC
- Matthieu Carrier (2005): Integration of a stereovision application in an SoC
- Oussama Chtorou (2005): Software Based Self-test of the MIPS R3000 processor
- Olivier Garcia (2007): Software-Based Self-Test Strategies for Memory Caches of RISC Processor Cores
- Montassar Bougattaya (2008): Development of an online monitoring application for MPSoC
- Majdi Allegue (2009): MPSoC monitoring using the IEEE 1687 standard
- Morgan Delahaye (2009): Instrumentation of a multi tasks application on a SoC
- Yi Gang (2011): Distributed checkpoint/Rollback for MPSoCs
- Saif Ur Rehman (2011): 3D NoC fault tolerance
- Anouar Soltani (2013): Design for Testability of a cluster-based FPGA
- Amina Belhaj Messaoud (2013): Investigation of Adaptive Fault-tolerant Task mapping on Distributed Many Core Systems
- Gabriel Villanova (2018): Self-adaptive approximate System on a Chip: Hardware platform development
- Nouredinne Aitsaid (2018): Self-adaptive approximate System on a Chip: Software platform development
- Talha Qureshi (2019): Multi-threaded embedded programming on Berkeley's RISC V platform
- Sohaib Chafik (2019): Mapping an embedded Video application on FPGA using Berkeley's Rocket Chip RISC V platform
- Alessandro Fici (2019): Development of a redundant online MAC for integration in a RISC V SoC
- Antoine Cerf (2020): Approximate Computing and Distributed Computing in SoCs: modelling and simulation
- Xavier Gros (2020): Approximate Computing and Distributed Computing in SoCs: Platform Development
- Antoine Maillefert (2021): Développement d'une Unité de Calcul en Virgule Flottante à Précision Variable et Intégration dans un Processeur RISC-V
- Luis Felipe Camponagara (2022): Design and Evaluation of Logic in Memory Solutions for ReRAMs
- Marcelo Correa (2022): Design and Evaluation of Logic in Memory Solutions with FeFETs

1.2 PhD Students

PhD Students	Years	%
Mathieu TUNA	2003-2007	80 %
Zhen ZHANG	2007-2011	50 %
Vladimir PASCA	2010-2013	30 %
Yi GANG	2011-2015	80 %
Saif Ur REHMAN	2011-2015	80 %
Noureddine AITSAID	2018-2021	60 %
Overall		380 %

Mathieu TUNA 2003-2007

Title: Auto-test logiciel des Systèmes intégrés sur puce (SoC)

Funding: scholarship grant from doctoral school

Co-supervisor: Alain GREINER (Full Professor at Paris Sorbonne)

As technology shrinking allows billions of transistors to be integrated in a single chip, the manufacturing test of such complex systems on a chip induce very expensive test cost. To facilitate the test access to embedded IP blocks, a new standard has been developed, namely the IEEE 1500 standard.

The objectives of this thesis were twofold:

- Propose new solutions to reduce manufacturing test cost by transferring external test capabilities to the chip while taking benefit of the advantages brought by the new IEEE 1500 standard
- Propose new solutions to allow the test of the device when embedded in its operation field

For that, the thesis exploits the Software-Based Self-Test (SBST) paradigm to test the IP Cores on one hand and the embedded processor and its caches on the other hand.

The outputs of this thesis are:

- a new approach for the manufacturing test of IEEE 1500 compliant embedded IP cores, based on the usage of a new embedded test co-processor. This co-processor was designed and synthesized for area evaluation.
- new SBST routines to test both the register file of the embedded processor and its associated memory caches.

The outputs of this thesis were published on eight papers in IEEE international conferences and symposiums.

Zhen ZHANG 2007-2011

Title: Détection de pannes franches et reconfiguration automatique dans un micro-réseau intégré sur puce

Funding: scholarship grant from doctoral school and research contract thanks to the ANR project ADAM

Co-supervisor: Alain GREINER (Full Professor at Paris Sorbonne)

Test of many core SoCs is complex as it involves hardware and software issues. Making them resilient to defects that happens at fabrication time or in their field of operation is challenging. One component is of special concern since it is shared by the whole other embedded blocks namely the Network on Chip.

The thesis objective was to develop a new approach allowing the test (detection) and deactivation of faulty components and the reconfiguration of the NoC according to the healthy resources' topology. The constraint of this procedure is to be run in the operational field of the device.

The output of the thesis is a new distributed procedure, hardware and software based, launched at reset time, allowing the application to be remapped on healthy resources. The thesis allowed to prove and quantify the feasibility of the approach for NoCs based on deterministic routing.

The outputs of this thesis were published on one journal and two renowned IEEE international conferences and symposiums.

Vladimir PASCA 2010-2013

Title : Développement des Architectures HW/SW Tolérantes aux Fautes et Auto-calibrantes pour les Technologies Intégrés 3D

Funding: doctoral research contract from the European project 3DIM³

Co-supervisor: Lorena ANGHEL (Full Professor at Grenoble INP UGA)

The technology improvements allowed the design of 3D Stacked chips using Through Silicon Vias (TSVs). The reliability of this TSVs is of special concerns especially when used to build a 3D NoC.

The thesis goal was to implement and assess different 3D NoC resiliency strategies, for each layer of the NoC implementation. The main idea was to study which cross layer strategy is more efficient depending on the application and the technology constraints.

The output of this thesis was a deep study on resilience of 3D NoCs providing the designer a set of guidelines to design efficient and reliable 3D NoCs. In addition, a tool was developed enabling a fast and accurate design space exploration for reliability assessment.

The outputs of this thesis were published on two journal and six renowned IEEE international conferences and symposiums.

Yi GANG 2011-2015

Title: Conception d'un micro-réseau intégré NoC tolérant les fautes multiples statiques et dynamiques

Funding: scholarship grant from doctoral school

Co-supervisor: Lorena ANGHEL (Full Professor at Grenoble INP UGA)

In harsh environments, many core SoCs must integrate efficient fault tolerant mechanisms dealing with permanent and transient faults. Moreover, the application should not be stopped, inducing a preference for on line strategies.

The thesis objectives were to develop fault tolerant strategies to improve the 2D NoC resilience in the presence of multiple faults. The targeted NoC routing scheme was the adaptive routing one to allow on line adaptation of the system.

The output of this thesis is the development and assessment of new routing algorithms and schemes allowing a high reliability in packet delivery in the presence of a high number of defective channels and routers. Moreover, the proposed techniques allow packet to travel around faulty resources while at the same time avoiding traffic congestion.

The outputs of this thesis were published on a chapter book, one journal and two renowned IEEE international conferences and symposiums.

Saif Ur REHMAN 2011-2015

Title: Development of test and diagnosis techniques for hierarchical mesh-based FPGAs

Funding: doctoral contract from the ANR project ROBUST FPGA

Co-supervisor: Lorena ANGHEL (Full Professor at Grenoble INP UGA)

The Test and fault tolerance of FPGAs are two topics whose implementation is very architecture dependent and that are usually considered separately.

In the framework of a new FPGA architecture developed by the startup FlexRAS we decided to study and evaluate the impact of fault tolerance on the testability metrics.

The main objective of this thesis was to find a method to detect and localize defects to enable a remapping of the application on healthy resources. For that we needed to design a new and efficient BIST based strategy adapted to this new architecture. Another objective was to evaluate the “robustness” of the FPGA when embedding hardware redundancies.

The output of the thesis is a method to test and locate the defects on the FPGA, along with a set of tools allowing the generation of the test application bitstream. Moreover, the architecture embedding fault tolerant solutions has also been evaluated from a testability point of view.

The outputs of this thesis were published on eight papers in IEEE international conferences and symposiums.

Noureddine AITSAID 2018-2021

Title: Toward Floating-Point Run-time Variable Precision in CPU-based Architectures

Funding: scholarship grant from doctoral school

Co-supervisor: Katell MORIN ALLORY (Associate Professor at Grenoble INP UGA)

In order to reduce energy consumption of non-critical applications the approximate computing paradigm has emerged this last decade. In this paradigm significant power consumption savings can be reached if the application outputs' precision is reduced to an acceptable threshold. For such applications, very often the variables in applications are overdesigned. Moreover, in an SoC, among the most consuming processing part, we

can pinpoint the Floating-Point Processing Unit (FPU). Based on these observations, applying the approximate computing to application using floating point formats is of great interest and need to be evaluated.

The thesis objectives were to propose an approach to quickly evaluate at software and hardware level the impact of bit width reduction of the inputs on the application outputs' accuracy and on power consumption.

The thesis outputs are: i) a new approach based on instruction interpretation, quite non-intrusive in the source code; ii) a new simulator, based on QEMU, allowing an automated design space exploration measuring the loss in the quality of results of an application for different inputs' sizes; iii) a hardware FPU allowing the precision of operands to vary during execution time; iv) the assessment of the efficiency of the approach on iterative algorithms.

The outputs of this thesis were published on two journals and two renowned IEEE international conferences and symposiums.

1.3 PostDoc Students

Etienne FAURE

Title: Self adaptive monitoring for test and fault tolerance in many core SoCs

Funding: postdoctoral contract in the framework of the ANR project ADAM

Co-supervisor: Francois Pécheux (Full Professor at Paris Sorbonne)

In the conjunction of Zhen ZHANG thesis, we needed in the ADAM project, a framework and a simulation platform allowing the implementation of the proposed techniques to monitor and reconfigure a massively multiprocessors architecture. We needed also to design efficient embedded databases that fulfill the monitoring needs.

The output of this postdoctoral work was the creation of two type of structures (local and global) to be used for embedding, storing and processing monitored events. This work also contributed to the development of a SystemC based simulation platform enabling the evaluation of the proposed solutions in the project.

Etienne Faure's work strongly contributed to make the ADAM project a success since the designed database was used by all the project partners.

The output of this work was published on one IEEE conference, one symposium and one workshop.

2. Teaching activities

Most of my teaching activities have been dedicated to the design of discrete digital circuits and systems on board or integrated on chip, targeting both hardware and software aspects, from transistor level to system level, dealing with both performance and reliability aspects.

My teaching activities are characterized by their diversity:

- diversity of level: from L1 to M2 for different audiences (university, engineering school, initial training, apprenticeship, continuing education for active engineers, and also to high school students, see Nano@School program described in section 3) with often heterogeneous levels within the same group.
- diversity of teaching types: lectures, tutorials, supervised work on machines, practical work on boards with measuring instruments and project-based learning. Most of the practical work and projects are

done either on a workstation with CAD software or on SoPC (System on Programmable Chip) electronic boards based on microcontrollers and FPGAs.

In addition to my teaching activities in groups, I have been strongly involved in the follow-up and management of internships, at the M1 and M2 levels, both in Paris and in Grenoble: tutoring, supervision, reporting, jury member, jury president, ...

In order to meet the specific needs of the different audiences and to consider the needs of the industry, I have often adapted my teaching, either by partially redesigning it (updating the contents and documents, changing the teaching method, ...), or by creating new modules, some of which are based on my research activities.

Finally, an important characteristic of my activities related to teaching is the taking of more and more important responsibilities as my career progresses: from coordinator of a teaching module when I was recruited at Paris 6 to the pedagogical and administrative responsibility of a program at Grenoble INP PHELMA.

In what follows I will review in a non-exhaustive way some of the points mentioned above: the most significant courses, the creation of courses and the different pedagogical and administrative responsibilities.

2.1 Most significant courses

At Pierre and Marie Curie University (UPMC Paris 6, now Paris Sorbonne)

In addition to some basic computer science courses, here are some keywords for a selection of the most significant and long-lasting courses over the period 2003-2009, for which I was responsible for one or more groups, and for which I participated in the creation and writing of pedagogical documents and exams.

- Machine and number representation - Hardware architecture of computers: L2 and L3 UFR Informatique

Boolean algebra, combinatorial logic, integer representation, internal architecture of a microprocessor (MIPS R3000), assembly language, combinatorial and sequential digital electronics, automata, compilation, cache memories, system architecture, interrupts, task switching, pipeline processor.

- Peripherals management in M1, Master SESI (Electronic and Integrated Systems)

Study of parallel and serial ports, PCI bus, use and creation of Linux drivers driving an LCD display and a thermometer through the parallel port.

- Design for test of integrated circuits and System on Chip Test in M2, Master SESI

Defects, fault models, test vector generation algorithms, DFT (Design For testability), BIST (Built in Self-Test), IEEE JTAG and IEEE 1500 standards, ... (see teaching creations below).

At Grenoble INP - PHELMA engineering school

Since my arrival in Grenoble in 2009, a large majority of my teaching has taken place in the form of practical work and projects, in the first year of the core curriculum (L3), in the SEI department (Systèmes Electroniques Intégrés), at the M1 and M2 levels, and in the MT (Microelectronics and Telecom) apprenticeship department created in 2013. Below are some modules that illustrate these TP (practical work) and project aspects, modules for which I was responsible for one or more groups depending on the year and where I participated in the creation and writing of pedagogical documents and exams.

- Computer and microprocessor - 1st year of core curriculum (L3 level)

Series of lectures-TD-TP on the architecture of microcontrollers based on the assembler of the MSP430 from Texas Instrument with at the end the programming of a FIR filter on a board with GBF and oscilloscope.

- TPs Integrated Systems - 1st year of the SEI program (M1)

This series of practical exercises aims to illustrate the whole design flow of integrated circuits, from the specifications to the drawing of the masks that are sent to the clean room for manufacturing. The various design and simulation software used are those used in the semiconductor industry.

- Embedded Software Systems Implementation (M2, SEOC department and MT apprenticeship programs at PHELMA)

Projects carried out by final year students in pairs or in triads. From a specification, students develop hardware and software architectures that they prototype on a programmable board of FPGA type: "vocoder" application, video processing application, ...

At other teaching institutions

Beside UPMC and Grenoble INP PHELMA, I had the opportunity to give lessons of Design For Testability in several other universities such as the French University of Cairo in Egypt in 2007 and 2008. Since my arrival in Grenoble, each year, I teach DFT lessons and supervise practical works of students belonging to the Master MISTRE of Université Grenoble Alpes. I also give punctually lessons to Mentor (Siemens) engineers on Test and testability aspects, in the framework of the Grenoble INP "Formation continue" department.

2.2 Creation of new courses (from my research activities)

Design for test of integrated circuits - Paris 6, M2 Master SESI (30 hours 30 students)

In Paris, in 2002, I took charge of this module which deals with the architectures, methods and algorithms to be implemented in order to design an integrated circuit that can be tested after fabrication. I did a deep redesign/transformation of the existing course (about 3/4) by transferring knowledge from my research work. I created new courses and with the help of a colleague MCF François Pêcheux, I created five new tutorials using CAD softwares used in the industry. This teaching followed me in Grenoble and evolved throughout my career as I taught and adapted it to different audiences (university, engineering school, initial training, initial training by apprenticeship and G-INP continuing education to active technicians and engineers) and in different places (UPMC, Grenoble INP, Joseph Fourier University (UGA) and French University of Egypt in Cairo)

System-on-Chip Testing (SoCTest) - Paris 6, M2 Master SESI (28 hours 18 students)

To complete the previous module, I created an optional course when I was recruited in Paris in 2003. This course extends the notions of testing to the system level and positions the students who follow it at the level of the international state of the art. I created this course alone (transparencies, handouts), the five machine-based tutorials having been created with the help of Matthieu Tuna, a PhD student under my supervision. I taught this module for 5 years at UPMC, until 2009.

Embedded Systems labs/project - G-INP PHELMA in 3A SEI and 3A MT courses (M2 level), 20 + 20 students, 40 hours of lab/project + 8 hours of lecture

As soon as I arrived in Grenoble in 2009, I took charge of the project "Design of systems on a SoC" whose objective is to run a multi-tasking MJPEG decoding application on a multi-processor architecture, using modeling

and simulation in SystemC language. I made evolve this module to complete it in 2015 with the creation of a new 8-hour lecture on embedded/integrated systems with an emphasis on multi-core and multi-task aspects.

In order to make students better understand the link between hardware and software and to anticipate the RISC-V revolution that was emerging in the industry, I launched in 2018 a large-scale project namely a complete overhaul of this module of 40 supervised hours (+40 h not supervised). Thanks to the help of 4 interns in final year project and Nouredine AITSAID PhD student that I co-supervised, I was able to replace in several times all the TPs + project. To illustrate the link between hardware and software, we used the software platform (free of charge) of hardware-software co-design Rocket Chip developed at Berkeley. This platform is based on a RISC V instruction set processor (which was becoming increasingly important in industry and research).

I obtained funding for 2019 and 2020 from the PIA IRT Nanoelec program which allowed me to recruit some interns to help me in the creation of the tutorials and the project: scripts, source code, development of a demonstrator, and a technical and pedagogical documentation of a hundred pages in English.

A big work of updating documents (installation and VPN connection tutorials), source files and tools has been done in autumn 2020 to allow a remote teaching and to adapt to the covid 19 pandemic.

After learning via the TPs, the students move on to the project (see figure 1). This project consists in deploying on FPGA a bare metal application that reads images on SD card, performs different filtering on them (bypass, Sobel and classification based on CNN network) before displaying them on VGA screen. On the FPGA the students deploy the LowRISC SoC developed at ETH Zurich.

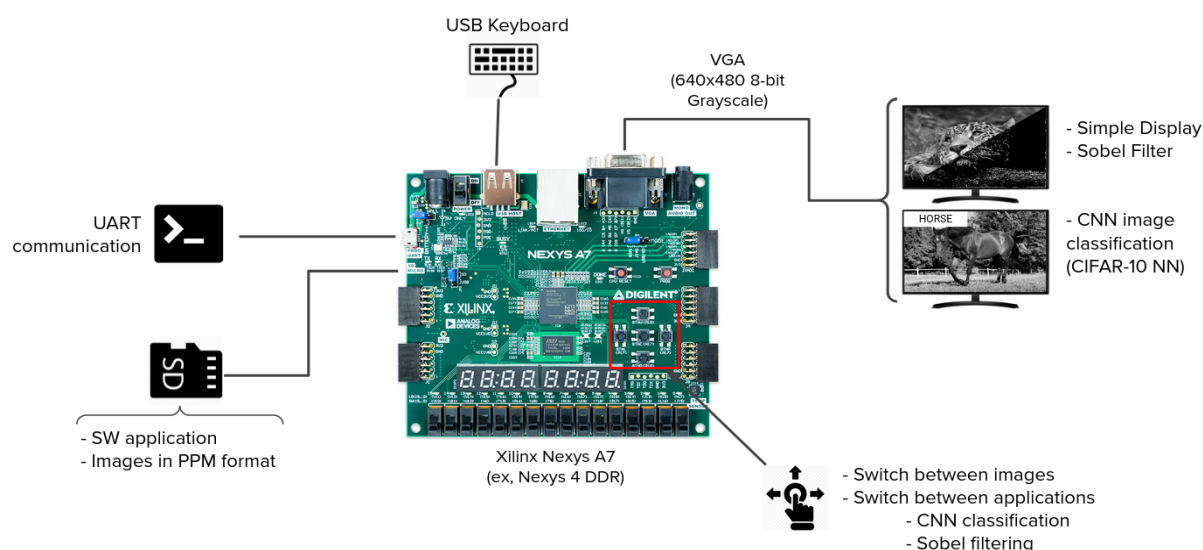


Figure 33 Platform for the project LEARNV

For more informations you can visit the github: <https://tima-amfors.gricad-pages.univ-grenoble-alpes.fr/learnv/>

This new teaching has been widely acclaimed by the students, including those who followed it in difficult conditions in semi-presential in winter 2020-2021. To my knowledge, this type of teaching is rare in Europe and has given rise to several publications and presentations in international workshops. It is available to the international community on the Internet under the name of the "LearnV" project.

It has been demonstrated twice at the University Booth of the IEEE DATE conference, in March 2020 and upon request of the organizers in February 2021. A video was recorded in this context and broadcast on the web: [link to the video](#)

2.3 Teaching administration

I here after list the main responsibilities (in order of importance) that are linked to my educational involvement.

- Co-head of the SEI (Systèmes Electronique Intégrés) department at Grenoble INP PHELMA: pedagogical and administrative responsibility for 6 years: 2010-2013 and 2017-2020

From 2010 to 2013 and from 2017 to 2020, I was co-responsible for the pedagogical and administrative aspects of the SEI department at PHELMA (45 students in 2A and about 30 in 3A). With another Associate Professor, we managed the pedagogical models, the timetables, the recruitment and the follow-up (sometimes individual) of the students and the temporary workers, the relations with the students, the teachers and the industrial partners. We made the link with the services of schooling, the administration... We participated very regularly in the PHELMA partners' days (physical forums with companies) and in the various meetings.

In 2020, in order to cope with the consequences of the covid 19 pandemic, the management of the department was very complicated and time-consuming. Among other things, it was necessary to put in place the mechanisms necessary for pedagogical continuity.

- Set up, pedagogical and administrative management of two educational projects at Phelma G-INP: PIA IRT NanoElec training program. (150 KE + 48 KE)
 - ❖ I did not take part in the design of the project, but I was in charge of its implementation and budget management from 2012 to 2014 for the "Reconfigurable systems for prototyping". With the help of Robin Rolland, technician at CIME, we proceeded with a budget of 150 Keuros to the selection, the purchase (drafting and management of public tenders) and the implementation of programmable FPGA boards and a set of electronic measurement instruments (GBFs oscilloscopes, probes, ...) to meet the needs of the various courses of PHELMA. I participated in the various meetings and presented the progress of the work for this action for which I was responsible. Since then, students in the SEI, MT and SEOC departments at Phelma have been using this equipment regularly.
 - ❖ In 2018-2020, with the help of Stéphane Mancini (Associate Professor at Grenoble INP), I set up and led an "embedded vision" project that aims to implement two new TPs/project modules at PHELMA in the 3rd year (see creation of TP/project Embedded Systems mentioned earlier). We obtained for our respective sub-projects (LearnV project for me and "embedded vision" for him) a global funding of 40 + 8 Keuros for 2019 and 2020 for the recruitment of trainees, research engineer and for the purchase of adequate programmable cards. I ensured the administrative management, the purchase of the programmable cards and the various reports and presentations to the IRT.

- Teaching coordinator for 6 years of the module "Hardware architecture of computers" at UPMC: 2004-2009

I taught and coordinated the pedagogical team of the module "Hardware architecture of computers" L3 in Paris 6 from 2004 to 2009: 150 students on average, 6 teachers. I managed the updating of the subjects of tutorials and practical exercises, the writing of exams, I took care of the timetable, the website, the communication with

the students, ... I participated in the deep reorganization of the module (creation of new tutorials and practical exercises, writing of new handouts)

- Responsible for the Master internships in the SEI department for two years: 2010-2012

Almost as soon as I arrived in Grenoble, I took on the responsibility of management and pedagogical coordination of the internships for 45 students in the SEI program in 2A and about 30 in 3A (interface with teachers, students and industrialists, advice, management of the defenses, ...)

- Pedagogical coordination for the "design" platform of the CIME Nanotech Grenoble-INP for 6 years: 2014-2020

During this period, I was the link between the teachers and the technicians of CIME Nanotech to match the CAD software resources of CIME and the needs of the teachers of PHELMA school. For information, the CIME Nanotech is an entity of Grenoble INP which provides universities, industrialists and researchers with rooms, machines, CAD tools, a clean room and various development and measurement instruments for everything "nano". I was only in charge of the design platform, essentially made of CAD software and rapid prototyping boards.

- Teaching and coordination of the Nano@School program for high school students: 2014-2020

Since 2014 I have been participating as a teacher in the Nano@School program which takes place at CIME Nanotech for high school students in the Grenoble area. A set of practical exercises over 2h30 aims to introduce the microelectronics professions to this young audience (in groups of 8 people). These discovering exercises are then used as a basis for work with their teacher when they return to high school. Until 2019, I was in charge of recruiting and coordinating the interventions (other teachers and PhD students) for the "design" theme that required about ten interventions per year.

3. Funded Projects and Cooperations

3.1 Private contracts

2003-2006 Common Laboratory between LIP6 and STMicroelectronics

The joint laboratory included 5 research themes linking LIP6 and STMicroelectronics. I was in charge of the theme "SoCs testing", a theme involving 6 people (ST+LIP6) for a budget of about 75Keuros/year. The goal of these activities was to propose test solutions for IP cores IEEE 1500 compliant. I made regular presentations of the progress of the work to the managers of the joint laboratory.

2018-2019 CIFRE thesis contract between the TIMA laboratory and the company Dolphin Integration

With my colleague Lorena Anghel we have set up a partnership with this company on the topic: Adaptive techniques for reducing the power consumption of systems-on-chip in fully depleted silicon-on-insulator technology. A PhD student was recruited in December 2017 but unfortunately the health of the student led us to interrupt this thesis in July 2019.

3.2 Projects in response to selective calls for proposal

I regularly respond, almost every year (with local colleagues, in France or abroad), to different calls for projects, most of the time it is national projects (like the ANR), regional (Pack Ambition région AURA) or local (IDEX IRS).

Sometimes the projects stop at the second phase of the selection process, and sometimes they pass all the stages. I describe below in chronological order only the projects where I have been strongly involved during my career.

2008-2011 ANR - project ADAM

Title: Adaptive Dynamic Architectures for MPSoCs

Place and period: LIP6 2008-2011

Participants: LIP6 (project leader), CEA LETI, LIRMM

Funding: ANR, program “Ingénierie Numérique & Sécurité” , 220 Keuros for LIP6

Involvement: I contributed to the setup of the project (proposal co-writing), I co-animated the project and was responsible for severable deliverables.

Special note: I began the project at the LIP6 laboratory in Paris in 2008, but after September 2009, as I moved to Grenoble, I continued to be very active on the project (supervision of Zhen Zhang thesis and Etienne Faure Postdoc, participation to the meetings, the deliverables, the reporting, ...)

Goal of the project:

The project addresses the hot topic of MPSoC self-adaptability to altered operating conditions, like loss of performance, appearance of faulty hardware resources, and temperature overhead. Taking simultaneously these problems into account is mandatory to ensure the reliability and competitiveness of embedded systems. For that the proposal was to provide solutions based on three steps: non-intrusive online monitoring, online diagnosis/test, and online remapping of the application.

Contributions:

The main contribution of the ADAM project was to make these three steps finely interact through the use of new on chip dynamic event databases: local databases of formatted/historized events and a global database of architecture instant maps.

For the LIP6 laboratory, the outputs of the project where a cycle accurate simulation platform, the creation and validation of the event databases structures and the creation and validation of a monitoring test application running on this platform. The experiments and measurements proved the feasibility and efficiency of the approach.

Results of the project have been published in [J04], [C06], [S06], [S07], [S10], [W06].

2009-2012 European EUREKA CATRENE program - project 3DIM3

Title: 3D-TSV Integration for Multimedia and Mobile applications (3DIM3)

Place and period: TIMA 2009-2012

Participants: STMicroelectronics (project leader), Cadence Design Systems, CEA-LETI, EADS DS, Fraunhofer Institute, Infineon Technologies, Lyon Institute of Nanotechnology, NXP Semiconductors, R3Logic, Recore Systems, TIMA Laboratory, TU Delft, University of Erlangen-Nürnberg, Virage Logic

Funding: CATRENE European program CT105, about 600 Keuros for TIMA

Involvement: I co-animated the local research activities and was responsible for severable deliverables.

Goal of the project:

The 3DIM3 project intended to provide novel systems-design methodologies, new design tools and systems architectures to handle emerging three-dimensional chips through silicon via (3D TSV) integration technologies for multimedia and mobile products. The expected outcome of 3DIM3 is to enable the design of 3D TSV integrated multimedia and mobile products with higher performance, lower power consumption and smaller size/form factor, from system and architecture levels to layout, at a lower cost than was currently achieved.

Contributions:

The main contribution of TIMA for this project was the development of algorithms and hardware solutions enabling the test and fault tolerance of TSV based 3D NoCs. Existing and new multi-layer resilience strategies were assessed and a design for reliability exploration tool has been developed.

Results of the project have been published in [S08], [S09], [S11], [W07], [W08], [W09].

2011-2015 ANR project ROBUST FPGA

Title: ROBUST FPGA

Place and period: TIMA 2011-2015

Participants: Telecom Paris (project leader), FLEXRAS (part of Mentor Graphics now), LIP6, TIMA

Funding: ANR, program "Ingénierie Numérique & Sécurité", 200 Keuros for TIMA

Involvement: I contributed to the setup of the project (proposal co-writing), I co-animated the project and was responsible for severable deliverables. I was a work package leader and responsible for TIMA.

Goal of the project:

Starting with a new type of FPGA architecture provided by the FLEXRAS startup, the objective was to provide solutions to make the final device robust to fabrication and lifetime defects. The reliability improvements were based on one hand on an efficient BIST application to be mapped on the FPGA and on the other hand on the integration of carefully chosen redundancy strategies.

Contributions:

For TIMA, we provide a new BIST methodology able to detect and locate permanent defects, enabling the remapping of the functional application on healthy resources, thus enhancing the manufacturing yield.

We extended the BIST approach to take into account the redundancy solutions provided by the partners and evaluated their impact on the testability features (fault coverage and test time). A set of tools was developed to automate the generation of the BIST and diagnosis applications.

Results of the project have been published in [C07], [C08], [S13], [S14], [S15], [P01].

2017-2018 IDEX UGA program, exploration and emerging projects - project SysAX

Title: Systèmes auto adaptatifs pour le calcul approché

Place and period: TIMA 2017-2018

Participants: TIMA

Funding: IDEX (Initiative d'Excellence) program, funded by the Grenoble Alpes University, 15 Keuros for TIMA

Involvement: I setup (writing of the proposal) and managed the project alone

Goal of the project:

The SysAx project aims at developing new methods and techniques to integrate approximations in a system-on-chip to allow its adaptation to its environment. For example, the input signal may become noisier, the battery warning level may be reached, the circuit may show signs of decreasing reliability, ... The system will then have to be able to switch from one precision mode to another automatically, either to reduce power consumption or to extend its lifetime.

Contributions:

This exploratory project led both to the development of a SoC simulation platform RISC-V based (described in section 2.2) and to the setup of ideas on Approximate Computing that served as a basis for the PhD thesis of Nouredine AITSAID

2019-2020, IDEX UGA program, exploration and emerging projects - project CADI

Title: Calcul Approché et Distribué dans les systèmes Intégrés

Place and period: TIMA 2019-2020

Participants: TIMA

Funding: IDEX (Initiative d'Excellence) program, funded by the Grenoble Alpes University, 12 Keuros for TIMA

Involvement: I setup (writing of the proposal) and managed the project alone

Goal of the project:

Almost all the studies and solutions proposed this last decade on Approximate Computing are only aimed at single-processor architectures, while applications based on intensive parallel computing are becoming more and more important.

Parallel RMS applications, being naturally predisposed to relaxations of precision constraints, require to implement an intelligent distribution of tasks/computations on hardware resources. A great potential in terms of energy saving is thus expected if the precision level is reduced to a tolerable threshold.

The CADI project aims at better understanding the issues (and proposing solutions) related to the extension of the Approximate Computing paradigm to multitasking applications that must run on multi-core embedded systems.

Contributions:

The project began in 2020 with the help of two internship students. It allowed to identify the main hardware and software obstacles. However, the project had to be stalled due to the pandemic issues. This project is part of my future work.

2018-2020, PIA IRT Nanoelec program - project "Embedded Vision"

Title: Embedded Vision

Place and period: TIMA 2018-2020

Participants: TIMA

Funding: PIA (Programme Investissement d'avenir) IRT (Institut de Recherche Technologique) Nanoelec, funded by the Grenoble Alpes University, 48 Keuros for TIMA

Involvement: I setup (writing of the proposal) and co-managed the project with Stephane Mancini (TIMA-CDSI team): students management, reporting, reviews, ...

Goal of the project:

The project is in majority an educational project and includes two subprojects: one dedicated to the development of a hardware coprocessor for image classification to be mapped on a FPGA, the other one is the project LearnV described in section 2.2. I managed alone the LearnV project which goal was to provide a design and simulation platform RISC-V based, to serve for training labs but also to be used as an experimentation platform for research development.

Contributions:

The project output is a hardware and simulation platform, based on the Berkeley Rocket Chip platform, running on PC station. An image classification application runs on a FPGA mapping the hardware architecture synthesized using this platform. A set of labs and a project has been developed to better understand the hardware and software issues when designing embedded systems. Master 2 students are using this platform and the FPGA board since 2019.

Results of the project have been published in [J07], [P02], [P03], [P04], [P05].

3.3 Cooperations

Beside the collaborations naturally induced in the projects previously described, it's worth mentioning some national collaborations I had during my research activities.

Collaboration with LIVIC laboratory Paris Saclay in 2005

When I was in Paris, I had the opportunity to collaborate on a project with the LIVIC laboratory (Laboratoire sur les interactions véhicules-infrastructures-conducteurs).

In this non-funded project, we started with a sequential stereo vision based algorithm, provided by the LIVIC, dedicated to the detection of obstacles with embedded cameras. The main goal of the project was to parallelize the algorithm/application and deploy it on a multi-processor hardware architecture. The research and development team was constituted by Professor Alain Greiner, Frederic Petrot (Associate Professor at this time), Roselyne Chotin Avot (Associate Professor) and myself from LIP6 and Raphael Labayrade from LIVIC. Mathieu Carrier, an internship student i co-supervised, led part of the software and hardware development.

The project was a success and led to a publication on a renowned conference and a workshop [S02], [W03].

Collaboration with TIMA's CDSI team 2017-2020

In 2017 I collaborate with Laurent Fesquet (Associate Professor) and Ali Skaf (senior researcher) belonging to the CDSI team of TIMA. The non-funded project objective was to see how redundant arithmetic could be useful to design adjustable and approximate operators.

The results were published in [J08], [C09], [W12].

Collaboration with Alberto Bosio 2018

With our common interest on several research topics, especially on Approximate Computing, I used to collaborate with Alberto Bosio (when he was at LIRMM) and Marcello Traiola a PhD student he was co-supervising. From the main discussions and exchanges we co-published a survey on Approximate Computing on a well-known journal [J06].

4. Dissemination of knowledge and scientific excellence

Expertise missions

- Adhoc expert/reviewer for the ANR

In 2015, 2017 and 2018: expertise of three projects during the first and second phase of the generic call for projects of the French National Research Agency for the evaluation committee "Micro and nanotechnologies for information and communication

- Expertise mission at STMicroelectronics Crolles (Grenoble) in 2002

In June and July 2002, I worked at ST Crolles to help, as an expert, the company to integrate the IEEE 1500 standard (architectures for system-on-chip testing) into the design tool stream.

Program committee and reviewer

- Reviewer for the following journals: IEEE Transactions on Very Large-Scale Integration (VLSI) Systems, IEEE journal on Transaction on Computers, Elsevier journal on Microprocessors and Microsystems, ACM Transactions on Embedded Computing Systems, Elsevier Microelectronics Reliability Journal, Elsevier Integration, the VLSI Journal, Elsevier Engineering Science and Technology: an International Journal
- Reviewer for the following conferences: IEEE International European Test Symposium (ETS), IEEE International Conference on ReConfigurable Computing, IEEE International Online Test Symposium (IOLTS), IEEE Latin American Test Workshop (LATW) now LATS (Symposium), IEEE Design and Technology of Integrated Systems in Nanoscale Era (DTIS), Symposium on Integrated Circuits and Systems (SBCCI)
- Program committee: IOLTS (since 2011), LATW (2010-2013, 2023), DTIS (since 2019), SBCCI (since 2017), ETS (since 2021)

Jurys

- Examiner of Mr. Tu's thesis defence, directed by Vincent Berouille and Chantale Robach, entitled "Test de NoCs asynchrone" at LCIS laboratory in Valence, defended in June 2008
- Jury reviewer for the mid-term thesis defense of Ms. Arwa Ben Dhia on January 22, 2013 at Telecom ParisTech.
- Member of the jury of the individual follow-up committee of Thomas Brouchet in July 2017, PhD student at LabStic (Université Bretagne Sud), co-supervised by Philippe Coussy and Gwénoél CORRE
- Member of 4 individual follow-up committee (CSI): Thiago Leite, Otto Rollof, Francois Bertrand, and Arthur in September 2018, PhD students at TIMA laboratory, CDSI team, supervised by Laurent Fesquet (TIMA)

- Member of the jury of the individual follow-up committee (CSI) of Arthur Vianes in December 2020 and December 2021, PhD student in CIFRE thesis in Kalray company, co-supervised by Frédéric Rousseau (TIMA) and Benoit de Dinéchin (Kalray)

GDR SoC²

- Scientific member of the GDR SoC² (Ex GDR SoC-SIP) of CNRS since 2012:
PC member and co-chair at the national level of the Test & Reliability topic from 2012 to 2016.
- Scientific co-animation of the Test & Reliability topic since 2000
From the early beginning of my PhD thesis I was part of the co-organization/animation of the Test and Reliability topic at the national level, first in what was called AS Test SoC, later in the GDR SoC SIP that finally became GDR SoC². That includes logistic management, scientific program setting, and scientific day animations.

Organizing Committee

- Local chair of IEEE European Test Symposium 2012 at Annecy
- Finance chair IEEE Online Test Symposium 2014
- Adhoc session moderator for the following conferences and symposia: DATE, ETS, IOLTS, LATS, ...

Other Dissemination of Knowledge

- Publication of two synthesis papers on an online national journal for active engineers
I was asked in 2020 and 2021 to take charge of the writing of two large articles (with Quizz) for the online journal "Techniques de l'ingénieur" on the Testing of integrated circuits and systems (my specialty in teaching and research). These papers, co-written with Professor Regis Leveugle, review and synthesize the state of the art on Test and Design for Test, at IC and SoC level, **on about 40 pages each**. The target audience is the French-speaking engineering community. The two articles are online and referenced as [J12] and [J11].
- Publications in pedagogical conferences
With my colleagues and students, we have published our pedagogical innovations around the "SoC Design" and "Embedded Systems" (detailed in section 2.2) at several international and national workshops and conferences referenced as [J07], [W10], [P02], [P03], [P04], [P05].
- Demonstrator at the University Booth of the IEEE Design And Test Conference (DATE) in 2020 and 2021
In 2020 and at the request of the organizers in 2021 we presented online (due to the pandemic issues) a live demo of our research and educational development platform LearnV

Chapter IV:
Publications

1. Book chapters

- [B02] Benabdenbi M., Anghel L., Dimopoulos M., Gang Yi
Adaptive Routing for Fault Tolerance and Congestion Avoidance for 2D Mesh and Torus NoCs in Many-Core Systems-on-Chip
Advances in Microelectronics: Reviews, Sergei Y. Yurish (Eds.) , Ed. IFSA, International Frequency Sensor Association, pp. 405-435, Vol. 1, 2017
- [B01] M. Benabdenbi, W. Maroufi, M. Marzouki,
SOC (System-on-a-Chip) Testing for Plug and Play Test
Edition by Krishnendu Chakrabarty. Kluwer Academic Publishers, Boston Hardbound, ISBN 1-4020-7205-8, September 2002.

2. Journals

- [J12] Benabdenbi, Mounir et Leveugle, Régis.
Test des circuits intégrés numériques - Conception orientée testabilité
Dans : « Électronique », [en ligne], Editions Techniques de l'Ingénieur (T.I.) [Paris, France], 2022, e2461, doi: 10.51257/a-v2-e2461, disponible à l'adresse : <https://www.techniques-ingenieur.fr/base-documentaire/electronique-photonique-th13/architecture-et-tests-des-circuits-numeriques-42276210/test-des-circuits-integres-numeriques-e2461/>
- [J11] Benabdenbi, Mounir et Leveugle, Régis.
Test des circuits intégrés numériques - Notions de base et évolutions
Dans : « Électronique », [en ligne], Editions Techniques de l'Ingénieur (T.I.) [Paris, France], 2021, e2460, doi:10.51257/a-v2-e2460, disponible à l'adresse : <https://www.techniques-ingenieur.fr/base-documentaire/electronique-photonique-th13/architecture-et-tests-des-circuits-numeriques-42276210/test-des-circuits-integres-numeriques-e2460/>
- [J10] Ait Said, N.; Benabdenbi, M.; Morin-Allory, K.
Self-Adaptive Run-Time Variable Floating-Point Precision for Iterative Algorithms: A Joint HW/SW Approach.
MDPI *Electronics* journal, 2021, 10, 2209. <https://doi.org/10.3390/electronics10182209>
- [J09] Ait Said N., Benabdenbi M, Morin-Allory K.
Arbitrary Reduced Precision for Fine-grained Accuracy and Energy Trade-offs
Microelectronics and Reliability Journal, Volume 120, 2021, 114099, ISSN 0026-2714, <https://doi.org/10.1016/j.microrel.2021.114099>.
- [J08] Skaf A., Ezzadeen M., Benabdenbi M., Fesquet L.
Clocked and event-driven redundant adjustable precision computing
Microelectronics Reliability, Ed. Elsevier, Vol. 111, pp. 113729, DOI: 10.1016/j.microrel.2020.113729, août 2020
- [J07] N. Ait Said, M. Benabdenbi, G. Villanova Magalhães
Prototypage Matériel-Logiciel de Systèmes Intégrés avec l'architecture RISC-V
Numéro spécial de la revue J3eA, Journal sur l'enseignement des sciences et technologies de l'information et des systèmes, volume 18, 2019
<https://www.j3ea.org/articles/j3ea/pdf/2019/02/j3ea191016.pdf>

- [J06] Anghel L., Benabdenbi M., Bosio A., Traiola M., Vatajelu I.
Test and Reliability in Approximate Computing
Journal of Electronic Testing: Theory and Applications, Ed. Springer, Vol. 34, No. 4, pp. 375-387, DOI: 10.1007/s10836-018-5734-9, août 2018
- [J05] Dimopoulos M., Gang Yi, Anghel L., Benabdenbi M., Zergainoh N.-E., Nicolaidis M.
Fault-Tolerant Adaptive Routing under an Unconstrained Set of Node and Link Failures for Many-Core Systems-on-Chip
Microprocessors and Microsystems, Ed. Elsevier, Vol. 38, No. 6, pp. 620-635, DOI: 10.1016/j.micpro.2014.04.003, août 2014
- [J04] Zhang Z., Refauvelet D., Greiner A., Benabdenbi M., Pecheux F.
On-the-Field Test and Configuration Infrastructure for 2-D-Mesh NoCs in Shared-Memory Many-Core Architectures
IEEE Transactions on Very Large-Scale Integration (VLSI) Systems, Ed. IEEE, Vol. 22, No. 6, pp. 1364 - 1376, DOI: 10.1109/TVLSI.2013.2271697, juin 2014
- [J03] Pasca V., Anghel L., Benabdenbi M.
Kth-Aggressor Fault (KAF)-based Thru-Silicon-Via Interconnect Built-In Self-Test and Diagnosis
Journal of Electronic Testing: Theory and Applications, Ed. Springer, Vol. 28, No. 6, pp. 817-829, DOI: 10.1007/s10836-012-5322-3, décembre 2012
- [J02] Pasca V., Anghel L., Nicolaidis M., Benabdenbi M.
CSL: Configurable Fault Tolerant Serial Links for Inter-die Communication in 3D Systems
Journal of Electronic Testing: Theory and Applications, Ed. Springer, Vol. 28, No. 1, pp. 137-150, DOI: 10.1007/s10836-011-5260-5, février 2012
- [J01] M. Benabdenbi, W. Maroufi, M. Marzouki
CAS-BUS: A Test Access Mechanism and a toolbox Environment for Core-based System Chip Testing
Journal of Electronic Testing: Theory and Applications, Special issue on Plug-and- Play Test Automation for System on a Chip), pages 455-473, Volume 18, Number 4, August/October 2002.

3. Conferences, Symposium, Workshops, Presentations

I hereafter list the different papers published in **Journals**, **Conferences**, **Symposiums** and **Workshops** with a review process and official proceedings. I also mention papers leading to a presentation or a poster but with informal proceedings. They are referred with the letter **P**. The name of the supervised student is underlined.

- [S16] N. Ait Said, M. Benabdenbi and K. Morin-Allory
FPU Reduced Variable Precision in Time: Application to the Jacobi Iterative Method
2021 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2021, pp. 170-175, doi: 10.1109/ISVLSI51109.2021.00040.
- [C10] N. Ait Said, M. Benabdenbi, and K. Morin-Allory
FPU Bit-Width Optimization for Approximate Computing: A Non-Intrusive Approach
in 2020 15th Design Technology of Integrated Systems in Nanoscale Era (DTIS), 2020, pp. 1-6.
- [C09] Skaf A., Ezzadeen M., Benabdenbi M., Fesquet L.
On-Line Adjustable Precision Computing

- IEEE Conference on Design & Technologies of Integrated Systems (DTIS 2019), Mykonos, GREECE, April 2019, pp. 1-5, doi: 10.1109/DTIS.2019.8735067.
- [W12] Skaf A., Ezzadeen M., Benabdenbi M., Fesquet L.
Adjustable Precision Computing Using Redundant Arithmetic
Workshop on Approximate Computing (AxC'2019), Florence, ITALY, 29 mars 2019
- [P05] N. Aitsaid, M. Benabdenbi
LearnV: A Hardware/Software RISC V Based platform for Research and Education
Colloque National du GDR SoC2 19-21 juin 2019 à Montpellier
- [P04] Liliana Andrare, Mounir Benabdenbi, Olivier Muller, Frédéric Rousseau, Frédéric Pétrot
Teaching basic computer architecture, assembly language programming, and operating system design using RISC-V
CEA-LETI and IRT NanoElec 2nd RISC V workshop, 1-2 octobre 2019 à Paris, organisé par le CEA et l'IRT Nanoelec.
- [P03] N. Aitsaid, M. Benabdenbi
Teaching Hardware/Software co-design using Rocket Chip
Workshop international RISC V 11-13 juin 2019 à Zurich
- [P02] N. Aitsaid, M. Benabdenbi, G. Villanova
Systèmes intégrés : mieux comprendre le lien matériel-logiciel en utilisant l'architecture RISC-V et la plateforme Rocket Chip
15èmes journées pédagogiques de la Coordination nationale pour la formation en micro-électronique et en nanotechnologies (JP-CNFM), décembre 2018 St Malo
- [C08] Rehman Saif-Ur, Benabdenbi M., Anghel L.
Application-independent testing of multilevel interconnect in mesh-based FPGAs
IEEE 10th International Conference on Design and Technologies for Integrated System in Nanoscale (DTIS'15), pp. 1-6, Naples, ITALY, DOI: 10.1109/DTIS.2015.7127383, 21 au 23 avril 2015
- [S15] Rehman Saif-Ur, Blanchardon A., Ben Dhia A., Benabdenbi M., Chotin-Avot R., Naviner L., Anghel L., Mehrez H., Amouri E., Marrakchi Z.
Impact of Cluster Size on Routability, Testability and Robustness of a Cluster in a Mesh FPGA
IEEE Computer Society Annual Symposium on VLSI (ISVLSI'14), pp. 553-558, Tampa, FL, UNITED STATES, DOI: 10.1109/ISVLSI.2014.66, 9 au 11 juillet 2014
- [S14] Rehman Saif-Ur, Benabdenbi M., Anghel L.
Cost-efficient cluster in a mesh SRAM-based FPGA
IEEE 20th International On-Line Testing Symposium (IOLTS'14), pp. 75-80, Platja d'Aro, Girona, SPAIN, DOI: 10.1109/IOLTS.2014.6873675, 7 au 9 juillet 2014
- [W11] Dimopoulos M., Gang Yi, Benabdenbi M., Anghel L.
Efficient Fault-Tolerant Adaptive Routing under an unconstrained Set of Node and Link Failures for Many Cores System On Chip
Workshop on Dependable Multicore and Transactional Memory Systems (DMTM'14), (joint to HIPEAC event), pp. 1-2, Vienna, AUSTRIA, 22 janvier 2014
- [W10] L. Anghel, M. Benabdenbi, F. Petrot
System On Chip Project: Integration of a Motion-JPEG Video Decoder

- 10th European Workshop on Microelectronics Education (EWME 2014), pp. 167-170, doi: 10.1109/EWME.2014.6877419. May 14-16, Tallin, Estonia
- [C07] Ben Dhia A., Rehman Saif-Ur, Blanchardon A., Naviner L., Benabdenbi M., Chotin-Avot R., Mehrez H., Amouri E., Marrakchi Z.,
- A defect-tolerant cluster in a mesh SRAM-based FPGA**
- International Conference on Field-Programmable Technology (ICFPT'13), pp. 434 - 437, Kyoto, JAPAN, DOI: 10.1109/FPT.2013.6718407, 9 au 11 décembre 2013
- [S13] Rehman Saif-Ur, Benabdenbi M., Anghel L.
- BIST for Logic and Local Interconnect Resources in a Novel Mesh of Cluster FPGA**
- IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT'13), pp. 296 - 301, New York, NY, UNITED STATES, DOI: 10.1109/DFT.2013.6653622, 2 au 4 octobre 2013
- [S12] Gang Yi, Dimopoulos M., Benabdenbi M., Anghel L., Zergainoh N.-E., Nicolaidis M.
- Fault-tolerant adaptive routing under static and run-time, permanent and transient failures for many-core systems-on-chip**
- IEEE International On-Line Testing symposium (IOLTS'13), pp. 7 - 12, Chania, CRETE, DOI: 10.1109/IOLTS.2013.6604043, 8 au 10 juillet 2013
- [P01] Rehman Saif-Ur, Benabdenbi M., Anghel L.,
- Cost-efficient Testing of LUT and Intra-cluster Interconnect of a Novel SRAM-based FPGA**
- Colloque National System-On-Chip System-In-Package (GDR SoC-SiP'13), Lyon, FRANCE, 10 au 12 juin 2013
- [S11] Pasca V., Rehman Saif-Ur, Anghel L., Benabdenbi M.
- Efficient link-level error resilience in 3D NoCs**
- IEEE 15th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS'12), pp. 127 - 132, Tallinn, ESTONIA, DOI: 10.1109/DDECS.2012.6219038, 18 au 20 avril 2012
- [S10] Zhang Z., Refauvelet D., Greiner A., Benabdenbi M., Pecheux F.
- Localization of damaged resources in NoC based shared-memory MP2SOC, using a Distributed Cooperative Configuration Infrastructure**
- 29th IEEE VLSI Test Symposium (VTS'11), pp. 229 - 234, Dana Point, CA, UNITED STATES, DOI: 10.1109/VTS.2011.5783726, 2 au 4 mai 2011
- [W09] Pasca V., Anghel L., Benabdenbi M.
- Configurable Thru-Silicon-Via interconnect Built-In Self-Test and diagnosis**
- IEEE Latin America Test Symposium Workshop (LATW'11), pp. 1-6, Porto de Galinhas (PE), BRAZIL, DOI: 10.1109/LATW.2011.5985896, 27 au 30 mars 2011
- [S09] Pasca V., Anghel L., Rusu C., Benabdenbi M.
- Configurable Serial Fault-Tolerant Link for Communication in 3D Integrated Systems**
- International On-Line Test Symposium (IOLTS'10), pp. 115-120, Corfu, GREECE, 3 au 7 juillet 2010
- [W08] Pasca V., Anghel L., Benabdenbi M.
- Fault Tolerant Communication in 3D Integrated Systems**
- DSN Workshop on Dependable Systems and Networks (WDSN'10), pp. 131-135, Chicago, UNITED STATES, 28 juin 2010

- [W07] Pasca V., Anghel L., Rusu C., Benabdenbi M.
Non-regular 3D mesh Networks-on-Chip
DAC Workshop on Diagnostic Services in Network-on-Chips (DSNoC'10), Anaheim, UNITED STATES, 13 juin 2010
- [S08] Pasca V., Anghel L., Rusu C., Benabdenbi M.
Configurable Fault-Tolerant Link for Inter-die Communication in 3D on-Chip Networks
European Test Symposium (ETS'10), pp. 258, Prague, CZECH REPUBLIC, 24 au 28 mai 2010
- [C06] E. Faure, G. Marchesan Almeida, M. Benabdenbi, P. Benoit, F. Clermidy, F. Pêcheux, G. Sassatelli, L. Torres
An in-memory monitoring database for self-adaptive MP2SoCs
IEEE International Conference on Design and Architectures for Signal and Image Processing (DASIP'2010), Edimbourg, Scotland, October 2010, pp 97-104.
- [S07] Z. Zhang, A. Greiner, M. Benabdenbi
Fully distributed initialization procedure for a 2D-Mesh NoC, including off-line BIST and partial deactivation of faulty components
15th IEEE International On-Line Testing symposium 2010, Corfu, Greece, July 2010, pp 194-196
- [S06] E. Faure, M. Benabdenbi, F. Pêcheux
Distributed online software monitoring of manycore architectures
15th IEEE International On-Line Testing symposium (IOLTS 2010), Corfu, Greece, July 2010, pp 56-61
- [W06] M. Benabdenbi, F. Pêcheux, E. Faure
On - Line Test and Monitoring of Multi - Processor SoCs: A Software - Based Approach
IEEE Latin-American Test Workshop, pp. 1-6, doi: 10.1109/LATW.2009.4813798, Buzios, Brazil, March 2009
- [S05] M. Tuna, M. Benabdenbi, A. Greiner
At-Speed Testing of Core-Based System-On-Chip Using an Embedded Micro-Tester
25th IEEE VLSI Test Symposium, Berkeley, pp. 447-454, doi: 10.1109/VTS.2007.16., California, USA, May 2007
- [W05] M. Tuna, O. Garcia, M. Benabdenbi
Software-Based Self-Test Strategies for Memory Caches of RISC Processor Cores
8th IEEE Latin-American Test Workshop, Cuzco, Peru, March 2007, pp. 124-130
- [W04] M. Tuna, M. Benabdenbi
Software-Based Self-Test of Register Files in RISC Processor Cores using March Algorithms
IEEE Latin-American Test Workshop, digest of papers, Buenos Aires, Argentina, March 2006, pp. 67-72
- [S04] M. Tuna, M. Benabdenbi, Alain Greiner
STESOC: A Software-Based Test-Access-Mechanism Controller
11th IEEE European Test Symposium, Southampton, England, May 2006, pp. 91-96
- [W03] Greiner A., Pétrot F., Carrier M., Benabdenbi M., Chotin-Avot R., Labayrade R.
MP-SoC Architecture for an Obstacle Detection Application in Pre-Crash Situation
2nd International Workshop on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC'06), pp. 24-30, Montpellier, FRANCE, 1 juillet 2006

- [S02] Greiner A., Pétrot F., Carrier M., Benabdenbi M., Chotin-Avot R., Labayrade R.
Mapping an obstacles detection, stereo vision-based, software application on a multi-processor system-on-chip
IEEE Intelligent Vehicles Symposium, pp. 370-376, Tokyo, JAPAN, DOI: 10.1109/IVS.2006.1689656, 13 au 15 juin 2006
- [C05] M. Tuna, M. Benabdenbi, A. Greiner
T-Proc: An Embedded IEEE1500-Wrapped Cores Tester
2nd IEEE Conference on Ph.D. Research in MicroElectronics and Electronics, Otranto, Italie, June 2006, pp. 493-496
- [W02] M. Tuna, M. Benabdenbi, A. Greiner
STES: Testing wrapped IP cores using a dedicated Test Processor
IEEE International Workshop on Infrastructure IP, PalmSprings, California, USA, May 2005, pp. 60-66
- [C04] M. Tuna, M. Benabdenbi, A. Greiner
STES: a New Software-Based Strategy for Testing SoCs Containing Wrapped IP Cores.
In Proceedings of the 12th International Conference in Mixed Design of Integrated Circuits and Systems (MIXDES 2005), Krakow, Poland, June 2005, pp.459-464. **Outstanding Paper Award**
- [C03] M. Benabdenbi, A. Greiner, F. Pêcheux, E. Viaud, M. Tuna
STEPS: experimenting a new software-based strategy for testing SoCs containing P1500-compliant IP cores
Design Automation and Test in Europe Conference, Paris, France, February 2004, pp. 712-713
- [C02] M.Benabdenbi, W.Maroufi, M.Marzouki
Testing TAPed Cores and Wrapped Cores with the Same Test Acces Mechanism
IEEE Design Automation and Test in Europe, pp150-155, Munich, Germany, March 2001.
- [S01] W.Maroufi, M.Benabdenbi, M.Marzouki
Solving the I/O Bandwith Problem in System on a Chip Testing
XIII Symposium on Integrated Circuits and System design, Manaus, Brazil, September 2000.
- [C01] M.Benabdenbi, W.Maroufi, M.Marzouki
Cas-bus: A Scalable and Reconfigurable Test Access Mechanism for Systems on a Chip
IEEE Design Automation and Test in Europe, pp 141-145, Paris, France, March 2000
- [W01] W.Maroufi, M.Benabdenbi, M.Marzouki
Controlling the Cas-Bus TAM with IEEE 1149.1 TAP: A Solution for Systems on a Chip Testing
4th IEEE International Workshop on Testing Embedded Core-Based System-Chips, pages 4.5.1-4.5.6, Montreal, Quebec, Canada, May 2000.

4. Selection of 5 best papers

- [J09] Ait Said N., Benabdenbi M, Morin-Allory K.
Arbitrary Reduced Precision for Fine-grained Accuracy and Energy Trade-offs
Microelectronics and Reliability Journal, Volume 120, 2021, 114099,ISSN 0026-2714, <https://doi.org/10.1016/j.microrel.2021.114099>.

[J06] Anghel L., Benabdenbi M., Bosio A., Traiola M., Vatajelu I.

Test and Reliability in Approximate Computing

Journal of Electronic Testing: Theory and Applications, Ed. Springer, Vol. 34, No. 4, pp. 375-387, DOI: 10.1007/s10836-018-5734-9, août 2018

[J05] Dimopoulos M., Gang Yi, Anghel L., Benabdenbi M., Zergainoh N.-E., Nicolaidis M.

Fault-Tolerant Adaptive Routing under an Unconstrained Set of Node and Link Failures for Many-Core Systems-on-Chip

Microprocessors and Microsystems, Ed. Elsevier, Vol. 38, No. 6, pp. 620-635, DOI: 10.1016/j.micpro.2014.04.003, août 2014

[J04] Zhang Z., Refauvelet D., Greiner A., Benabdenbi M., Pecheux F.

On-the-Field Test and Configuration Infrastructure for 2-D-Mesh NoCs in Shared-Memory Many-Core Architectures

IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Ed. IEEE, Vol. 22, No. 6, pp. 1364 - 1376, DOI: 10.1109/TVLSI.2013.2271697, juin 2014

[S05] M. Tuna, M. Benabdenbi, A. Greiner

At-Speed Testing of Core-Based System-On-Chip Using an Embedded Micro-Tester

25th IEEE VLSI Test Symposium, Berkeley, pp. 447-454, doi: 10.1109/VTS.2007.16., California, USA, May 2007

Résumé

L'augmentation croissante de la densité d'intégration a permis de concevoir des systèmes embarqués connectés à haute performance de calcul et à faible consommation d'énergie. Les applications tirant parti de ces caractéristiques, avec l'avènement d'Internet, se sont rapidement démocratisées et ont eu un impact sociétal majeur : smartphones et réseaux sociaux par exemple. Cette intégration constante a permis de grands progrès en termes de performances mais a également nécessité une attention accrue pour tout ce qui concerne la qualité et la fiabilité des circuits fabriqués, en particulier pour les circuits visant des applications critiques (par exemple, aérospatiale, automobile, santé).

La densification expose le circuit à davantage de défauts, défauts qui peuvent apparaître au moment de la fabrication ou plus tard lorsque le circuit se trouve dans son environnement final. Pour compenser cela, les techniques de test des circuits ont dû évoluer et s'adapter pour permettre les développements que nous avons connus. Avec l'augmentation de la densité de défauts et le besoin de fiabilité, il y a également eu une demande croissante d'intégration de mécanismes de tolérance aux fautes.

Dans cette présentation, je ferai un tour d'horizon de mes contributions de recherche axées sur le test de circuits à grande échelle (test de production et test du circuit dans son environnement) et sur la tolérance à tous types de défauts qui peuvent survenir pendant ou après la fabrication du circuit. En guise de résumé, je présenterai les contributions suivantes :

- Tests d'IPs et de systèmes sur puce (SoC) en utilisant des techniques d'auto-test basé sur le logiciel (SBST).
- Plateforme matérielle/logicielle permettant la surveillance en ligne de SoCs massivement multi-processeurs.
- Test et tolérance aux pannes de réseaux sur puce (NoC) 2D et 3D
- Test d'une nouvelle architecture FPGA hiérarchique.
- Approches pour concevoir des circuits plus économes en énergie en utilisant le paradigme du calcul approché.

Mots-clés : système sur puce, test, tolérance aux fautes, réseau sur puce, calcul approché, FPGA

Abstract

The growing increase in integration density has enabled the design of connected embedded systems with high computing performance and low power consumption. Applications taking advantage of these characteristics, with the advent of the Internet, have been rapidly democratized and have had a major societal impact: smartphones and social networks for example. This constant integration has allowed great progress in terms of performance but had also required increased attention to everything related to the quality and reliability of the manufactured circuits, especially for circuits targeting critical applications (e.g., aerospace, automotive, health).

Densification exposes the circuit to more defects, defects that can appear at the time of manufacture or later when the circuit is in its final environment. To compensate for this, the techniques for testing circuits have had to evolve and adapt to allow the developments we have experienced. With the growth of defect density and the need for reliability, there was also a growing demand for integrating fault-tolerant mechanisms.

In this presentation, I will overview my research contributions focused on the testing of large-scale circuits (production test and in the field test) and on the tolerance to all kinds of defects that may occur during or after circuit manufacture (such as Single Event Errors (SEE) and defects due to aging). As a summary, I will present the following contributions:

- IPs and System on a Chip (SoC) testing using Software Based Self-Test (SBST) techniques
- A hardware/software framework enabling the online monitoring of manycore SoCs.
- 2D and 3D Network On Chip (NoC) testing and fault tolerance.
- Test of a new hierarchical FPGA architecture.
- Approaches to design more power efficient circuits using the approximate computing paradigm.

Keywords: SoC, test, monitoring, fault tolerance, NoC, approximate computing, FPGA

