

## 6. Architectures for Robust and complex Integrated Systems (ARIS)

**Group Leaders: M. Nicolaidis and R. Velazco**

(e-mails: [Michael.Nicolaidis@imag.fr](mailto:Michael.Nicolaidis@imag.fr), [Raoul.Velazco@imag.fr](mailto:Raoul.Velazco@imag.fr))

**Members: L. Anghel, M. Ben-Jrad, S. Bergaoui, M. Benabdenbi, G. Bizot, A. Bocquillon, T. Bonnoit, G. Canivet, F. Chaix, D. Fall, I. Fall, J.B. Ferron, G. Foucard, T. Frank, R. Leveugle, V. Maingot, P. Maistri, W. Mansour, G. Marques-Costa, M. Nicolaidis, P. Papavramidou, V. Pasca, P. Peronnard, D. Natalia Ruiz Armador, C. Rusu, C. Sahnine, P. Vanhauwaert, H. Yu, N. Zergainoh**

Research areas	Contracts	Industrial Partners
<ul style="list-style-type: none"> <li>- Design of robust complex integrated systems for nanotechnologies.</li> <li>- Hardware/Software dependability analysis from RT-Level descriptions.</li> <li>- Secure digital implementations</li> <li>- RT-Level design for reliability/safety/availability and/or security.</li> <li>- Methodology, tools and experimentation for the study the sensitivity to radiation of integrated circuits and systems</li> <li>- Methods and tools for fault injection</li> <li>- Mitigation techniques for the effects of radiation in integrated circuits and systems</li> <li>- Design and exploitation of experiments on-board satellites and high altitude balloons</li> <li>- Design for test, yield and reliability in nanometric memories</li> <li>- Concurrent error detection in nanometric designs</li> <li>- Circuit self-regulation under variability, circuit degradation and application constraints</li> <li>- Fault-tolerant, power-aware task scheduling and allocation</li> <li>- Error recovery architectures for massively parallel processors</li> <li>- Design for test, yield and reliability in 3D interconnects</li> <li>- Fault-tolerant 2D and 3D NOC routing algorithms</li> </ul>	ATMEL EADS IW ESA FT-ORANGE STMicroelectronics THALES <b>ALFA</b> NICRON <b>ANR</b> EMAISeCi FME3 <b>CATRENE</b> OPTIMISE TOETS 3DIM3 <b>MEDEA+</b> PARACHUTE <b>MINALOGIC</b> ARAVIS ASTER SOCKET SHIVA	ARTISAN (USA) ATMEL (France) CADENCE (France) CEA/LETI (France) CISCO (USA) CNES (France) EADS IW (France) e2v (France) FT Orange (France) IROC (France) GEMPLUS (France) NASA GFSC (USA) NXP (France) ONERA DESP (France) R3Logic (France) RECORE (Netherlands) ST Microelectronics (France) ST Ericsson (France) THALES COMM. (France)

### 6.1 Summary

The ARIS group was created in 2007 as a continuation and extension of past activities performed in the RIS (Reliable Integrated Systems) and the QLF (QualiFication of circuits) groups of TIMA. Research activities of ARIS deal with two main areas: the design of robust architectures for ultimate technologies and the study of the behavior of integrated circuits and systems under intentional attacks or in harsh environments. Indeed, different types of interferences and parasitic effects affect the reliability of modern electronic systems. Nanometer circuits, micro-electronics, micro-system technology and power electronic systems are already part of our daily life. However, these systems may encounter many problems with natural and artificial interferences coming from various sources (e.g. particle radiation effects, electromagnetic interferences, etc ). Another related area is the new threat on secure systems, related to fault-based attacks.

Recent research areas of the group include fault tolerant architectures for mitigating the flaws of nanometric CMOS (variability, accelerated circuit aging and parasitic effects); fault tolerant

architectures for high defect densities targeting post CMOS nanotechnologies; and Computing architectures for nanotechnologies.

A last research area of the ARIS group concerns the design of robust (high-yield, high-reliability, low-power) architectures for ultimate CMOS and post-CMOS technologies. Silicon-based CMOS technologies are approaching their ultimate limits. By approaching these limits, power dissipation, fabrication yield, and reliability are steadily worsening, thus making nanometric scaling increasingly difficult. In particular, process, voltage and temperature variations (PVT) affect parametric yield and reliability. Reliability is also increasingly affected by soft-errors, electromagnetic interferences, and accelerated circuit aging. Low power, a stringent requirement in nanometric technologies, is increasingly difficult to meet with further nanometric scaling. It is also conflicting with reliability requirements as voltage reduction increases dramatically circuit sensitivity to failures.

These problems are becoming increasingly hard to treat as we move down to the ultimate CMOS nodes. They risk becoming showstoppers, unless a new design paradigm able to maintain acceptable levels of power dissipation, yield and reliability is introduced. These issues are further exacerbated by unprecedented complexity levels as the integration of tera-device ICs is expected to become a reality within one decade. Research in ARIS group concerns a variety of approaches spanning from circuit level up to OS, including: design for test, yield and reliability for embedded memories and interconnects (2D and 3D); fault tolerant 2D and 3D routing algorithms; circuit-level concurrent error detection techniques; self-regulation of circuit parameters in response to variability, aging and varying application constraints; fault-tolerant and power-aware task scheduling and allocation; error recovery in multiprocessor systems. These activities culminate to the development of a comprehensive platform enabling designing robust (high-yield, high-reliability and low-power) single-chip massively-parallel tera-device processors fabricate in ultimate-CMOS and post-CMOS technologies affected by high defect densities. In particular, this approach is expected to achieve reliable operation in massively parallel processor arrays where all links, routers and nodes may include faults affecting their temporal behaviour (e.g. delay faults and clock skews), most of them include faults affecting their logical behaviour, and new destructive failures may affect the array links, routers and nodes with high frequency (e.g. MTBF of the order of few days).

Concerning researches related with the operation of integrated circuits in harsh environment, the main stress considered is radiation of nuclear and space environments, but it is interesting to mention that particles reaching the Earth's surface from the Sun, up to now innocuous for microelectronics circuits, have sufficient energy to flip bits in memories or corrupt the logic of parts manufactured with less than 0.25  $\mu\text{m}$  and supply voltages drop to less than 2.2 Volts. This constitutes a threat to avionics control systems and even to systems operating at sea level.

One of the significant issues of these researches is the prediction of error rates a studied system (circuit, architecture, software ...) will have in the final environment it will operate. The refinement of forecasting error rate strategies needs both to perform ground tests by means of simulated radiation environment (particle accelerators,...), and to compare ground test results to data obtained from experiments aboard of spacecrafts, balloons, satellites or while installed at high altitude.

Another research field concerns the development of innovative methods and tools dedicated to the predictive analysis, validation and qualification of integrated electronic systems using fault injections. The sensitivity predictive analysis platform covers the development of multilevel fault injection methods and tools to be applied at different system abstraction levels from RTL level to gate level descriptions.

The activities of the ARIS group concern both natural and intentional faults in integrated systems. Activities are also on-going on the design of secure circuits protected against fault-based attacks. These activities include the analysis of the circuit robustness against a whole panel of attacks (DFA, DPA, EMA) and the evaluation of protection techniques. Common approaches are developed to analyze the dependability level and protect circuits against both types of faults (natural and intentional), taking into account their different characteristics.

## 6.2 Design of robust integrated systems under high failure rates

*Members:* L. Anghel, M. Nicolaidis, M. Benabdenbi, G. Bizot, T. Bonnoit, F. Chaix, D. Fal, P. Papavramidou, V. Pasca, H. Yu, N. E. Zergainoh

Silicon-based CMOS technologies are approaching their ultimate limits. By approaching these limits, power dissipation, fabrication yield, and reliability are steadily worsening, making nanometric scaling increasingly difficult. In particular, as process parameters variations increasingly spread out with nanometric scaling, the probability that a complex SoC die includes timing faults increases drastically at each new process generation. Timing is also increasingly affected by variations of voltage (due to IR drop) and temperature (due to local hot spots etc). Thus, process, voltage and temperature (PVT) variations may create unpredictable timing behaviour and affect parametric yield and reliability. Reliability is also increasingly affected by soft-errors, electromagnetic interferences (like cross talk), and accelerated circuit aging. Low power, a stringent requirement in nanometric technologies, is increasingly difficult to meet with further nanometric scaling. It is also conflicting with reliability requirements as voltage reduction increases dramatically circuit sensitivity to failures. These problems are becoming increasingly hard to treat as we move down to the ultimate CMOS nodes. They risk to become showstoppers, unless a new design paradigm able to maintain acceptable levels of power dissipation, yield and reliability, is introduced. These issues are further exacerbated by unprecedented complexity levels as the integration of tera-device ICs is expected to become a reality within one decade.

An important research axis in ARIS group concerns the development of techniques for designing robust complex integrated systems. These techniques are spanning from circuit level up to OS, including:

- Low-cost circuit-level concurrent error detection in logic designs covering the whole set of failures affecting nanometric technologies.
- Self-regulation for adapting circuit parameters to variability and circuit degradation.
- New approaches for efficient BIST, self-repair, and ECC implementation in memories.
- Fast memory ECC design.
- Self-test and self-repair for 2D and 3D interconnects.
- Differential Voltage Frequency Scaling at processor- level and array-level.
- Fault-tolerant, variability aware and power-aware task scheduling and allocation algorithms.
- Coherent check-pointing and error recovery at array-level.
- Check-pointing-free error recovery at array-level.
- Fault-tolerant, congestion-free, and deadlock-free 2D and 3D routing algorithms.

Each of these techniques addresses at the most appropriate level some of the major issues concerning yield, power, and reliability in nanometric technologies. Furthermore, several of these techniques are combined into an integrating framework enabling the implementation of robust (high-yield, high reliability, and low-power) single-chip massively parallel tera-device processors, in highly defective technologies.

These activities are supported by three CATRENE projects (OPTIMISE, TOETS and 3DIM3) and by the ARAVIS/Minalogic project.

### 6.2.1 Low-cost highly versatile memory BIST for advanced nanometric nodes and/or 3D systems

In modern SoCs embedded memories concentrate the majority of defects. In addition, defect types are becoming more complex and diverse and may escape detection during fabrication test, leading to field failures due to the use of faulty components in final products. As a matter of fact, memories have to be tested by test algorithms achieving very high fault coverage for a comprehensive set of faults. Fixing the test algorithm during the design phase may not be compatible with this goal, as unexpected failures not covered by this algorithm may occur during production. Also, having the possibility to select the memory test algorithm after fabrication is very important during the initial phase of introduction of a new process node (both process debug and production ramp-up). Finally, in 3D systems, logic (processor) dies can be stacked with memory dies coming from various sources, and using different memory sizes in different versions of a product. Thus, memory BIST integrated in the logic die should be adaptive to the various memory sizes used in the final product and fault models corresponding to the various memory-die sources. In this context, programmable BIST approaches, allowing selecting after fabrication a large variety of memory tests, are desirable but may lead on

unacceptable area cost. In this work we develop a flexible memory BIST approach enabling full programmability in silicon of all three components of memory test (test algorithm, test data, and address sequencing), as well as of the size of the memory under test.

### **6.2.2 Eliminating speed penalty in ECC protecting memories**

One of the major drawbacks in implementing error control codes (ECC) in memories is the performance penalty associated to the code generation and error detection and correction circuitries. In this work, we developed innovative architectures enabling complete elimination of this penalty. The proposed concept and its evaluation to an experimental design is described in a paper presented at DATE 2011. This solution is generic and can be implemented in any SoC architecture. Current work concerns an automation tool able to insert and optimize our solution at RTL for any SoC design, comprising any number of memories.

### **6.2.3 Self-test and self-repair for 2D and 3D interconnects.**

In this work we address Design for Yield of 3D interconnects. These interconnects are realized by means of through silicon vias (TSVs) and represent an important challenge in producing 3D systems, as TSVs exhibit very high failure rates. To cope with this issue we develop efficient self-repair architectures using TSV reconfiguration circuitry that replaces faulty TSVs by fault-free spare ones. However, when we have to repair very large numbers of TSVs and/or when the defect density exceeds a certain level, this approach requires adding a large number of TSVs, resulting in high area penalty and high fabrication cost. In these situations, we have developed an alternative solution in which we transform a parallel transmission link that includes faulty TSVs, into a link that uses only the fault-free TSVs to transmit messages serially. The serialization architectures include two approaches. In the Configurable fault-tolerant Serial Links (CSLs), presented at IOLTS 2010, the serialization circuitry is programmed by an external tester. In the I-BIRAS architecture (Interconnect Built-In Self-Repair and Adaptive Serialization for Inter-Die Communication in 3D Integrated Systems), both the serialization circuitry and its control are integrated in the dies to be stacked. This enables performing serialization not only after fabrication but also in the field to cope with product-life failures. Some results of this ongoing work can be found in our publications at IOLTS 2010 and ETS 2011.

### **6.2.4 Low-cost circuit-level concurrent error detection in logic designs**

Traditional fault-tolerance using massive redundancy (e.g. duplication and TMR) induces very high area and power penalties and has unacceptable impact on hardware resources and power dissipation. In our past work we introduced the double-sampling approach (VTS 1999, DATE 2000), which drastically reduces area and power penalties. While this approach has found significant adoption by both industrial R&D and the academia, it employs redundant sequential elements (a shadow latch per circuit flip-flop) and results in significant power penalty. As a consequence, this approach is often used to protect only the critical paths of a design. Our recent work introduces a combination of the double-sampling technique with latch-based design, in an architecture referred as GRAAL (presented in a chapter of the book “Soft Errors in Modern Electronic Systems”, M. Nicolaidis, Springer 2011). With this architecture, concurrent error detection is achieved by adding just an XOR gate per circuit latch, resulting in a low-area and low-power concurrent error detection scheme. A first implementation of this scheme, using the CSEM icyflex1 processor, shows an area penalty of 17% and a power penalty of 8.4% (ETS 2011) for complete protection of all circuit paths. Current work concerns an improved implementation targeting less than 13% extra area and less than 5% extra power for complete circuit protection.

Work on double-sampling in flip-flop based designs targets a new approach enabling trading error detection capabilities with circuit speed, in a manner that the circuit adapts its operation on the fly to changing reliability requirements and/or severity of environmental conditions.

### **6.2.5 Self-regulation for adapting circuit parameters to variability, circuit degradation and application context**

In this work, we monitor the rate of error detections provided by the concurrent error detection circuitry, in order to determine optimal operating frequencies and voltage levels and adapt accordingly clock frequency and Vdd to PVT variability, circuit degradation during product life, and varying computing-power requirements due to changing application contexts. Power dissipation/reliability constraints are used to select the operating clock frequency and Vdd level of the circuit according to the computational constraints (task deadlines) required by the application. Aggressively low Vdd is used for tight power-dissipation constraints, relaxed Vdd is used in reliability-tight constraints. Interactions between hardware and OS are used, as reliability and power-dissipation constraints, and application deadlines are provided by the OS, while the final selection of Vdd and clock frequency is done at the hardware level.

### **6.2.6 Fault-tolerant, congestion-free and deadlock-free 2D and 3D routing algorithms**

This work targets single-chip, massively parallel processors. Conventional fault tolerant routing uses routing tables and determines fault-free routes in a static manner. Unfortunately, in a complex network, this approach will lead to frequent congestions as statically pre-established routes can not take care of the actual network traffic. To cope with this issue, we have developed a distributed algorithm in which the routing decisions are taken locally and opportunistically (i.e. the routing path is modified according to congested or faulty node/router/link that a message can encounter during its transmission). This approach can tolerate failures of high multiplicity and avoid congestions, but deadlocks become possible. We achieve deadlock freedom by using virtual channels and pertinent routing rules acting at the local level. The algorithm was shown efficient in networks comprising thousands nodes and affected by high defect levels. Some results of this ongoing work can be found in our papers published at 2010 IEEE Network Computing and Applications and at DATE 2011. These algorithms have been extended to cover 3D networks. Some results of this extension can be found in NORCHIP 2009 MICPRO 2010, IOLTS 2010.

### **6.2.7 Variability, aging and power-aware task scheduling and allocation**

In this work we develop algorithms for performing task scheduling and allocation that minimizes power, under constraints related to variability and to aging-induced circuit degradation. In our approach we consider the application to be organized into tasks, task clusters and groups of task clusters. The approach acts in three steps. The two first steps are done off-line.

The first step performs local static allocation: it determines the number of execution cycles of each group of tasks clusters and their sum (giving a rough estimation of time and power). This step is easy to perform (low computation complexity).

The second step performs global static scheduling. It determines a set of optimal operating points (time-power) for each group of task clusters. Due to the high computational complexity, we perform this step by means of a genetic algorithm.

The third step (on-line scheduling), uses the results of the second step to select the points that minimize the energy of the active groups of tasks while respecting their deadline.

This approach targets a global optimization and is computationally intractable for large processor arrays. The results obtained led us to the development of local, non-deterministic approaches using opportunistic decision rules.

### **6.2.8 Coherent check-pointing and error recovery at array-level**

This work concerns check-pointing and rollback recovery in single-chip multiprocessor arrays. The challenges we faced in this domain are twofold. First, when rollback is executed in some parts of the system, we have to maintain coherence with the tasks executed by other parts. Second, check-points have to be saved in reliable (thus external to the chip) memory. But as the single-chip multiprocessor I/Os are limited, they may be easily congested by check-pointing. To cope with these issues we developed innovative coordinated check-pointing algorithms that enable performing coherent rollback in a simple manner. Furthermore, in large processor arrays, intelligent broadcast techniques and partitioning are used to reduce the number of broadcasts and the size of checkpoints. Some results of

this ongoing work have been published in IOLTS 2008, ISCAS 2008, DELTA 2008, and MICPRO 2010.

In a more recent approach, we use a hierarchical (parent-child) task distribution approach and maintain the hierarchy until successful execution. In case of a failure occurring in some node when it executes some part of the application, its parent node distributes this part to other fault-free nodes of the array. This way, error recovery is achieved without performing check-pointing, thus avoiding the associated performance penalties and possible I/Os congestions. The approach is shown to be robust to multiple node failures, as we can go up to the parent-child tree until a fault-free parent node.

The two approaches are complementary as in some applications storing intermediate states for performing check-pointing require less memory space than maintaining the task distribution hierarchy while in some other applications the inverse is true.

### **6.2.9 Integrated fault-tolerant routing, power-aware task scheduling and allocation, rollback recovery and circuit parameter regulation**

In the first part of this work we developed an innovative algorithm integrating fault-tolerant routing, power-aware task scheduling and allocation and rollback recovery in single-chip massively parallel multiprocessors. We resolve the complexity of fault tolerant routing and task scheduling and allocation in complex arrays by adopting a distributed non-deterministic approach, taking local decisions in opportunistic manner. This algorithm uses a hierarchical (parent-child) task distribution approach. It maintains this hierarchy until successful execution of the current tasks. In case of failure in some array node its parent node distributes the failed part of the application to new nodes of the array, as described in the previous section. This way, error recovery is achieved for multiple failed nodes without performing check-pointing, thus avoiding storing check-points to an external memory and the associated performance penalties and I/Os congestions. The fault tolerant and error recovery capabilities of the algorithm were validated in arrays comprising thousands of processing nodes and including up to 20% failed routers /links /nodes, executing streaming applications.

The above algorithm was further extended to perform circuit parameters regulation adaptive to PVT variability, circuit degradation due to aging, and changing reliability, power-dissipation and computing power requirements of the application. To cope with the intractable complexity associated with the management of these requirements in complex processor arrays, we chose a hierarchical approach which adopts the task distribution hierarchy described earlier. Regions, sub-regions and nodes of the array are selected by the different levels of the task allocation hierarchy, by taking into account power-dissipation and speed distributions over the array nodes induced by variability and aging. This approach enables efficient power and deadline aware task scheduling and allocation at reasonable computation time.

Furthermore, in the proposed approach, each task encapsulates deadline constraints and power dissipation/reliability constraints. These constraints are used by the node selected for executing the task to determine its operating frequency and Vdd level.

### **6.2.10 Designing robust single-chip massively parallel tera-device processors**

The above approaches are integrated in a framework aimed at designing robust single-chip massively parallel tera-device processors, comprising thousands of processing nodes and fabricated in ultimate CMOS or post-CMOS technologies affected by high defect densities. Our platform combines several of the approaches described above in an innovative manner that enables their optimal cooperation and allows handling high defectivity while achieving low-power operation. Particularly, this approach is able to achieve reliable operation in arrays where all links, routers and nodes include faults affecting their temporal behaviour (e.g. delay faults and clock skews), most of them include faults affecting their logical behaviour, and new destructive failures may affect with high frequency (e.g. MTBF of the order of few days) the array links, routers and nodes..

### 6.3 Hardware/Software dependability analysis from RT-Level descriptions

Members: R. Leveugle, P. Maistri, P. Vanhauwaert, J.B. Ferron, L. Anghel, M. Ben-Jrad, S. Bergaoui, R. Clavel (VDS), L. Pierre (VDS)

Significant effort was targeted during the last years on developing efficient techniques to analyze, at design time and early in the design flow, the functional consequences of soft errors. The goal is to precisely identify the soft errors leading to unacceptable application disturbances, in spite of all the possible masking effects due to the circuit architecture (redundancy, performance-oriented features, etc.) or to the application characteristics (meaningless computations in a parallel structure, meaningless precision of some data, etc.). Targeted circuits have essentially been synchronous digital circuits. Most of the techniques, developed since more than ten years in the team, start from synthesizable RTL descriptions. Such descriptions are already close to the final hardware in terms of cycle accuracy and in terms of memory cells identification. Higher level descriptions may in some cases be used, with limited representation of soft error locations and reduced accuracy in terms of propagation analysis. Software is also taken into account in the case of systems based on microprocessors. Robustness evaluations may aim at (1) classifying the soft errors with respect to their functional impact (2) identifying error propagation paths (3) identifying critical locations or registers (4) ensure that a given set of behavioural properties always hold for a given set of soft errors (e.g., a given maximum multiplicity of erroneous bits). Classification may be used to compute derating factors on the application failure probability. Critical locations are hardened in priority when trade-offs have to be made. The efficiency of the hardening techniques implemented at RT-Level can be validated as well during such analyses.

A whole set of techniques has been developed to cover the wide range of analysis objectives and circuit characteristics. Most of the studied approaches concern the optimisation of fault injection techniques. One cornerstone has been for a long time the length of the experimental campaigns. Most efforts were therefore targeted to the performance optimisation, but without forgetting the need for flexibility with respect both to the type of circuit and the analysis aims. In addition, techniques have been studied to (1) reduce the number of soft errors to inject (2) analyze the criticality of variables and registers for embedded software and (3) propose alternatives for cases in which fault injection is not adequate, such as the need of guarantee of a given behaviour in spite of all assumed soft errors.

Concerning the acceleration of the fault injection experiments, we proposed more than ten years ago to take advantage of hardware emulation. A few years ago, we proposed efficient injection platforms based on SoPCs (System on Programmable Chip). Such platforms take advantage of the processor core integrated in the SoPC to reduce the quantity of data exchanges with the host computer, thus accelerating the experiments, while maintaining the largest flexibility with respect to the type of possible dependability analyses. The platform has still been improved with new optimizations published in 2010 at LASCAS. Unfortunately, even with such accelerations, exhaustive fault injections often remain unaffordable in a reasonable time.

Another approach has also been used, based on the partial reconfiguration capabilities of some FPGAs. In that case, it is possible to avoid any instrumentation of the circuit under evaluation; soft errors are injected by reconfiguring the contents of flip-flops. The same approach can be used to study the effect of configuration errors in SRAM-based FPGAs. We have also improved the injection environment by adding a database of realistic error patterns in order to improve the precision of the analyses. The patterns can be obtained once from device characterizations under a given perturbation source. The obtained patterns can then be used to analyze the robustness of several versions of a design or several designs, without resorting each time to costly perturbation sources such as particle accelerators or lasers. This is especially powerful to study regular structures since the patterns obtained on a small area of the chip can be abstracted and relocated on similar areas in other parts of the chip. This has been exploited for example in FPGAs to study the effect of configuration errors. The approach has been published at ICECS in 2010.

Due to time limitations, only partial analyses can be performed in most cases. They are based on a randomly (and often arbitrarily) selected set of faults or errors. Such a statistical fault injection (SFI) has been very extensively used in the literature but the margin of error on the results given on such a basis was unknown. We have therefore proposed and validated an approach to quantify the errors on the results with a given confidence level, or conversely to evaluate the number of injections to perform

in order to achieve a given error/confidence level. The method has been presented at the DATE Conference in 2009.

SFI can then be very useful in doing quick classification or derating factor estimations, with a controlled margin of error. Unfortunately, this approach does not address all possible expected outcomes of a fault injection campaign. In the case of a large number of potential errors and workload cycles, only a very small proportion of the registers are actually perturbed at a few cycles. This means that such results cannot help in identifying the most critical registers or clock cycles. Flip-flop grading remains possible when the random selection is only used to reduce the number of injection cycles in each flip-flop; however, in that case, the efficiency of SFI is noticeably reduced and the required number of experiments remains very large. Error propagation paths are also only partially exercised, so decisions on the best hardening positions are hard to make. Finally, SFI cannot guarantee that a given property always holds; this can at best be assessed with a given margin of error. Several complementary approaches have therefore been studied in order to more efficiently obtain some expected outcomes and will be summarized hereafter.

An approach based on Timed Petri Net models has been proposed and evaluated. It has been shown that this type of model can efficiently identify some harmless errors even in the case of architectures difficult to analyze with classical fault pruning techniques. Such models can therefore be used to reduce the number of injection experiments. They can also help in grading the criticality of the flip-flops in a circuit. The results have been presented at IOLTS in 2009.

For embedded software, identifying critical variables or registers is very important before fault injections are started. An improved algorithm for critical register identification has been proposed and implemented in the Gcc compiler; this has been published in the IEEE Transactions on Nuclear Science in 2010. Effects of compilation options on the register and memory criticality have been analyzed and presented at LASCAS 2010 and DELTA 2011. The study also pointed out the strong impact of micro-architectural features (such as dependency management) on the actual criticality of internal processor registers. As a consequence, an algorithm was developed to improve the precision of the lifetime analysis; this algorithm is currently under evaluation for the Leon3 processor.

The specific case of configuration errors in SRAM-based FPGAs was also addressed by developing a tool performing static analyses at design time to evaluate the criticality of the various bits in the configuration file. Such analyses are less precise than fault injections because the dynamic behaviour of the application cannot be captured, but they can be very short (a few minutes, while fault injections may require days or weeks). The SEFEA-ProD tool developed in the team was used to analyze the robustness of designs in Xilinx Virtex II and Atmel AT40K devices. Comparisons have been made with experimental results based on laser fault injections and proton ground tests, showing very good correlations. Results have been published at SCS in 2009.

Another study aims at proving properties even in presence of soft errors. This follows 2005 experiments based on formal property checking and published at IOLTS. Work was done in collaboration with the VDS group in TIMA and the LIP6 laboratory, in the frame of the FME3 project, supported in the period 2008-2010 by the French national research agency (ANR). Model checking techniques and theorem provers (ACL2, then PVS) were used and extended to efficiently model the effect of single or multiple bit errors and prove properties in presence of such errors. The resilience of systems was in particular analyzed, i.e. their capability, after a disturbance due to a soft error, to recover a correct behaviour. The work is continued in the frame of the SHIVA project, to prove the efficiency of some protection techniques against fault-based attacks on cryptoprocessors.

## 6.4 RT-Level design for reliability/safety/availability and/or security

*Members: R. Leveugle, P. Maistri, G. Canivet, V. Maingot, S. Bergaoui*

Protecting a design against natural perturbations or malicious attacks can be done at several levels. We mostly focus here on approaches that can be applied at RT-Level, therefore quite early in the design flow and easy to synthesize on several physical targets (several ASIC technologies, FPGAs ...). Approaches studied in the past also included operating system or software modifications but we focused these last years on hardware protection techniques. Some protections aim at improving reliability, safety and/or availability against natural perturbations (radiations, particles, electromagnetic fields) and other disturbances caused by for example process, voltage and temperature (PVT) variations. Some others are dedicated at improving security against malicious attacks, either passive

(based on power or electromagnetic measures) or active (laser-based, glitch-based or electromagnetic-based perturbations).

The main part of the work is on the development and validation of robust cores including new protection schemes against malicious attacks. Processor cores are extended with dependability-oriented features (Leon2 some years ago, currently Leon3) and cryptographic coprocessors are designed (especially for the AES and RSA encryption/decryption algorithms). This activity is part of several projects, including SHIVA (Minalogic) and EMAISeCi (ANR). The coprocessors developed in the frame of SHIVA have strong throughput expectations in addition to fault-based attack protection. The versions developed within EMAISeCi aim at demonstrating protections against electromagnetic attacks.

After analyzing the realistic error patterns obtained due to laser-based and power glitch attacks onto SRAM-based FPGAs (publications at IOLTS 2008 and VTS 2009), it was shown that an efficient protection or "countermeasure" against fault-based attacks on a ASIC implementing an AES crypto-processor (published in IEEE Transactions on Computers in 2008) was not efficient on SRAM-based FPGAs due to the remanent errors induced in the configuration. An improved countermeasure was therefore implemented and validated; this has been published at ASAP 2010 and in the Journal of Cryptology (Springer). Additional protections have been designed and are under evaluation against electromagnetic attacks.

In addition to the development of hardening techniques and hardened IPs, a study has aimed at evaluating the impact of fault-oriented protections on leakage information. As a matter of fact, in the security context, it is useless to protect a circuit against only one type of attack, since a hacker could use several approaches to obtain secret information. The last results on this study were published at SCS 2009.

Recently, a study was started on the use of Graphics Processing Units (GPUs) to implement cryptographic functions. GPUs are many-core structures that can be well suited to implement high-performance cryptographic applications at low cost. The study is currently focused on performances but will be extended in the near future towards countermeasures for robust implementations.

Due to the increasing spatial multiplicity of error patterns, protecting a circuit with information redundancy is more and more difficult. This is particularly true when malicious attacks are concerned, but the problem exists also for natural perturbations. Another approach consists in using functional checks. In this context, we have developed in 2008 a new control-flow checking technique that has been improved in 2010. This technique is non-intrusive and does not require a modification of the initial microprocessor-based system. Checks include not only the control flow itself, but also the integrity verification of critical data, with several possible trade-offs between overheads and error detection. No assumption is made on the error multiplicity. The approach is compatible with the norms requiring a complete separation between the nominal functions and the checking features (e.g. for automotive applications). A first prototype has been developed including (1) a specific watchdog processor (or infrastructure IP I-IP) and (2) development tools. The watchdog program is automatically generated at compile time by a modified version of the Gcc compiler. Additional tools have been recently developed to cope with linkage constraints. The current prototype is currently optimized and will be validated in 2011 by fault injection campaigns. It will be used as a basis for security management in the SHIVA project and to validate the approach against radiation impacts in the CATRENE OPTIMISE project.

Finally, studies are on-going on the problems related to configuration errors in SRAM-based FPGAs. The goal is to propose specific design techniques to achieve robustness at lower cost than the classical massive redundancy approach. A first approach is under evaluation for detecting configuration errors in Atmel AT40K devices by taking advantage of unused resources for a given design.

## 6.5 Radiation Hardened memory cells

*Members: L. Anghel, M. Nicolaidis, R. Velazco*

Radiation-induced transient effects in silicon CMOS circuits are essentially charge collection and transport phenomena resulting from direct ionization. The collected charges may inadvertently change, for short time intervals, the internal node voltages of the circuit (single event upset). These transients may change the electrical behavior of the MOS transistors in digital and analog circuits.

Design hardening techniques at circuit level can be developed to achieve immunity to upsets. A lot of hardened-by-design memory elements have been proposed in the last years. In 1994 and 1997 were developed two hardened cells in TIMA laboratory, so-called, HIT (Heavy Ion Tolerant cell) and DICE (Dual Interlocked Cell) respectively. The philosophy was to strengthen the feedback of the cell in order to restore the data potentially corrupted by the impact of a charged particle. However, the price to pay is an increase of the cell area and a higher power consumption. In 2010 the HIT cell was included in the DARE (Design Against Radiation Effects) library and is presently considered to be transferred to many industrial partners to be used for the development of commercial radiation hardened circuits.

## 6.6 Study by real life experiments of the effects of natural radiation on the operation of submicronic integrated circuits

*Members: P. Peronnard, W. Mansour, G. Foucard, R. Velazco*

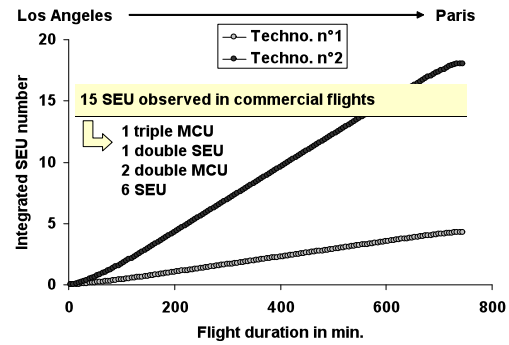
Ionization resulting from charged particles and atoms present in the substrate of CMOS circuits, may modify memory cell's content or provoke a transient pulse within a combinational circuit. This phenomenon which until recently was only considered to be a treat for space applications is nowadays a major concern for avionic equipments and even for any application operating at ground level. Thus it constitutes a potential obstacle to the reliable operation of circuits manufactured from future deep submicronic processes.

We have developed experimental boards, so-called SRAMCheckers including a 1 Gigabit SRAM memory built from successive generations (130 nm and 90 nm) of COTS (Commercial Off The Shelf) SRAM circuits. The boards are exposed to the effects of natural atmospheric radiation both during commercial long-haul flights and in high altitude balloons. The goal of these experiments is to get objective data about error rates and errors features (multiplicity for instance) in the real life environment, this with the final goal of predicting the conjecture will face the future of submicronic components. The sensitive to the considered errors of SRAMs selected for these real life experiments was also predicted using a tool developed at ONERA DESP. This tool, so-called MUSCA SEP<sup>3</sup> (MULTi-SCALEs Single Event Phenomena Predictive Platform (MUSCA SEP<sup>3</sup>)) consists in sequentially modelling all these various physical mechanisms likely to lead to a SEE occurrence in integrated circuits. MUSCA SEP<sup>3</sup> inputs include a device description, i.e., the semiconductor active zones, the passivation metallization layers and the package. When the tested circuit is a SRAM, the elementary cell (layout) is described and so the rules of translation allowing modelling the whole memory plan.

The SRAM Checker boards were activated during many commercial flights and balloons (in the frame of BALLTRAP project with ONERA) putting in evidence the occurrence of MCU (Multiple Cell Upsets) and MBU (Multiple Bit Upsets) which are of high concern for applications requiring high reliability. Indeed, many of the faults detected proved that as the impact of a single neutron may perturb the content of 3 bits of a word. In 2009 the results obtained in flights were presented at IOLTS 09 (International On-Line Test Symposium), and a work confronting predicted error-rate to obtained measures was presented at NSREC and published at IEEE TNS.



Los Angeles – Paris flight

Single and Multiple events errors predicted by MUSCA SEP<sup>3</sup> for a Los Angeles-Paris flight.

**Figure 6.1 Measures vs. predictions issued from MUSCA SEP3 for two long-haul commercial flights**

## 6.7 Development of a generic and flexible test bed suitable for the qualification of integrated circuits devoted to operate in harsh environment

*Members: R. Velazco, G. Foucard, P. Peronnard, W. Mansour*

With the miniaturization, integrated circuits become more and more sensitive to perturbations resulting from the effects of the environment (temperature, radiation, EMC...). This activity concerns the design and HW/SW improvements of a test system which facilitates the realization and exploitation of qualification tests for all kind of circuits, from a simple register bench to complex components such as processors.

Screening tests are mandatory to predict error rates in many fields. In the case of radiation, they consist in exposing the studied parts, eventually operating in vacuum, to particles representative of the ones that will be encountered in the operational environment. The hardware and software developments related with such tests must take into account the random nature of event occurrence, both in time and space. On one hand this entails on-line error detection, on the other hand this makes mandatory the need for development of ad hoc hardware mechanisms related with the detection and recovering from destructive errors, such as SEL (Single Event Latchups), which are power/ground short circuits provoked by the activation of a parasitic thyristor present in bulk-CMOS circuits impact of a single particle; or critical errors (sequencing loss, system crashes) that may occur circuits such as processors, system on chips,...

In the past we have prototyped different versions of a dedicated test system having the following characteristics:

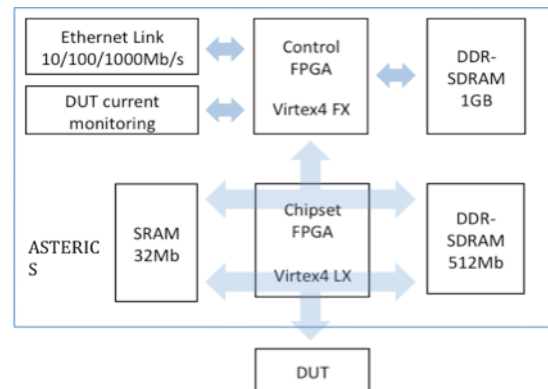
The last version of such a platform, called ASTERICS (Advanced System for the TEst under Radiation of Integrated Circuits and Systems), was designed to deal with the following requirements:

- The whole system must have a size allowing to entirely fitting it in enclosures commonly used for environmental qualification tests.
- The DUT must be tested on-line, operating in nominal conditions.
- The system should allow to exercise as many circuits (among candidates to a given project for instance) as possible, to avoid waste of time/money consequence of delays provoked by operating the facilities used to simulate the studied environment.
- Capability for the tester to be remotely controlled anywhere in the world through the internet network.

The new ASTERICS test platform constitutes a powerful tool with generic capabilities for the qualification under radiation of digital circuits. The idea is to implement the whole DUT board architecture by means of an FPGA whose configuration is obtained from compiling the description of key features of the DUT in a hardware description language such as VHDL or Verilog. In this way, there is only a minor hardware development, limited to wiring the DUT pins to the ones of the tester connector. The architecture of ASTERICS is mainly composed of:

- A Xilinx Virtex4FX FPGA, containing a PowerPC hardware processor, having in charge the following tasks: it controls the circuit under test, runs on and off the test sequences and monitors the latchup circuit. It is also able to transfer test programs and to gather test results from the component being studied to/from the PC user interface via an Ethernet 10/100/1000 link. A second FPGA used as a *chipset* and several static and dynamic memory banks are also included in the board: a 32Mb SRAM memory organized in 2 banks of 512k\*32bits (which should allow the test of advanced 64-bits processors) and 512Mb of DDR-SDRAM (16Mx32bits) to enable fast and efficient processor tests. To cope with a wide range of DUTs, all these memory banks are managed by the FPGA of ASTERICS's motherboard, a Virtex4LX FPGA from Xilinx, which is optimized for high-performance logic. It is multi-volt Input/Outputs compliant, that means it can drive signals in 3.3, 2.5, 1.8, 1.5 and 1.2 Volts, reducing the number of components required to interface the DUT.
- The DUT board comprises exclusively the component to be tested and, in some cases voltage regulators.

The last enhancement of ASTERICS was the proof of its remote use, through the ETHERNET connection. Such a feature may allow significantly reduce the cost of radiation testing campaigns, as they may be performed without requiring the presence of the experimenter. The demonstration of this remote use of ASTERICS was done in the 2009 and 2010 editions of SERESSA (international School on Radiation Effects on Systems for Space Applications) performing on-line laser tests remotely monitored. ASTERICS was transferred to the CRC (Cyclotron Research Center) of UCL (Université Catholique de Louvain-la-Neuve, Belgium) to be used in the future as the basic platform for radiation ground testing performed in various radiation ground test facilities.



**Figure 6.2 ASTERICS: a generic test platform for radiation ground testing and fault injection**

## 6.8 Predicting SEU error rates from Radiation Ground Testing and Fault Injection

*Members: R. Velazco, P. Peronnard, G. Foucard*

When estimating the sensitivity to radiation of an integrated circuit the goal is to evaluate the average number of impinging particles required to provoke a fault. Main considered non destructive faults for advanced ICs are SE (Single Event Upset), MBU (Multiple Bit Upset), MCU (Multiple Cell Upset and SEFI (Single Event Functional Interrupt). The consequences of this in the operation of a circuit or a system can be studied by means of fault injection techniques performed according different strategies, which depend basically of the available circuit model (SPICE, RTL, ...). To evaluate the error rate of a circuit/system it is in all the cases is required a measure of the intrinsic sensitivity, *so-called cross-section*, of the target circuit/system to the considered Single Event.

$\sigma$  is called the interaction cross-section and is a direct measure of the IC sensitivity and is calculated as the number of particles required to provoke one Single Event. Its unit is the  $\text{cm}^2$  or it is expressed in barn ( $1 \text{ barn} = 10^{-24} \text{ cm}^2$ ). Generally  $\sigma$  is given as an interaction cross-section per device (or per bit).

As a consequence, the end-product of a radiation ground testing will be a plot of the interaction cross-section versus particle energy (measured in terms of Linear Energy Transfer (LET) which is the energy transferred to the Silicon by the impinging particle.

Processors are included in most of the architectures devoted to embedded systems. The determination of SEU cross-section of microprocessor's memory elements requires the use of a so-called *static test* which consists in exposing the device to a suitable particle flux while the content of the DUT's registers and memory elements are observed. This is usually attained by executing a test program in charge of initializing these memories and dumping their content after a given period of time. The usually obtained measure, called *cross-section* (the number observed bit flips divided by the number of incident particles) is used to predict the final error-rate of the tested device in a given harsh environment. However it has been shown that the measured static cross-section can significantly overestimate the one the circuit while it executes a real application. The reason is simple: while a test program is written to maximize the number of errors observed, a real application is not. Moreover, many memory bits are not used, or are refreshed so often that SEUs in these regions have no impact on the system's behavior.

Fault injection may help in predicting the behavior of a real program which would be used for the final application. It was demonstrated by many experiments that the final cross section of an application executed by a processor can be obtained by multiplying the static cross-section by the error rate obtained during fault injection. The key point is *how to perform a fault injection campaign* where instants and location of injected faults suitably match the ones of faults occurring when the application will operate in the final environment.

Fault Injection strategies can be classified in two families: software based and hardware based. Among software based ones, depending on the available DUT description level, can be mentioned:

- SPICE based fault injection, if a SPICE net-list is available;
- VHDL / Verilog fault injection when a behavioral or RTL description does exist;
- ISS based fault injection if an Instruction Set Simulator is available.

Hardware based fault injection requires a physical device, and faults can be injected using:

- FPGA based fault injection, if a RTL description is available and mapped to an FPGA;
- The CEU (Code Emulated Upset) method, based in the random activation of interrupt signals.

The CEU approach, initially proposed by ARIS and published in 2000 consists in triggering, at a randomly chosen clock cycle and while the processor is running, an asynchronous interrupt. When the DUT receives the interrupt, it transfers the control from the executed benchmark to a trap handler. This handler then flips the content of a randomly chosen bit and resumes the benchmark execution. Obtained results proved that the predicted error rate fits very well the measured ones. The last years the CEU approach was extended to complex processors, to make possible to target cache memories as well as register files. This new concept was explored in the frame of SCADRI project and applied to two advanced processors: the Power PC 7447 and 7448. The main contribution of these experiments was the validation of the CEU error-rate prediction approach for an advanced and complex processor while it executed a program issued from a real space application, an Attitude and Orbital Control Systems (AOCS) provided by CNES (French government space agency).

Fault injection sessions and radiation ground testing campaigns were performed using the ASTERICS test platform. Radiation ground testing results were gathered during experiments performed at Louvain-La-Neuve heavy ions facility. We used the resulting cross-sections and flux settings as inputs to our fault injection set-up. Outputs of the fault injection experiments were analyzed with *exactly the same tools* used to analyze the radiation ground test results. At the 95% confidence level, there was no disagreement. The test set-up cannot see significant differences between the rates of SEU induced by radiations and those issued from SEU-like faults injected. In the tables 1 and 2 below are provided, for two selected heavy ions, results proving the accuracy of the proposed approach to predicted SEU rates.

**Table 1: Predictions vs. Measures for the PPC 7448 when data cache is deactivated**

Ion	$\sigma_{\text{SEU}}$ Predicted	$\sigma_{\text{SEU}}$ Measured
Argon	1.96E-06	1,84E-06
Krypton	3.82E-06	3,56E-06

**Table 2: Predictions vs. Measures for the PPC 7448 when data cache is activated**

Ion	$\sigma_{\text{SEU}}$ Predicted	$\sigma_{\text{SEU}}$ Measured
Argon	2,12E-05	2,04E-05
Krypton	3,24E-05	3,17E-05

## 6.9 Evaluation of the sensitivity to radiation of SRAM-based FPGAs

*Members: G. Foucard, A. Bocquillon, R. Velazco*

The increasing popularity of low-cost safety-critical computer-based applications in a large scope of areas (such as space and avionics, automotive, biomedical, telecontrol, etc.) requires the availability of new circuits and methods for designing dependable systems. In particular, in the areas where computer-based dependable systems are currently being introduced, the cost (and hence the design and development time) is often a major concern, and the adoption of commercial reconfigurable hardware, such as SRAM-based FPGAs (Field Programmable Gate Arrays) is a common practice. As a result, software implemented fault tolerance is an attractive solution for this class of applications, since it allows the implementation of dependable systems without incurring the high costs coming from designing custom hardware or using hardware redundancy.

Despite these attractive characteristics, designers are reluctant to use these components for critical applications due to their sensitivity to Single Event Upsets (SEUs) provoked by radiation. All SRAM-based FPGA resources are controlled by its configuration memory, an SEU in this area may thus change the original behaviour of the application. Moreover a fault affecting this memory is permanent until the device is configured again.

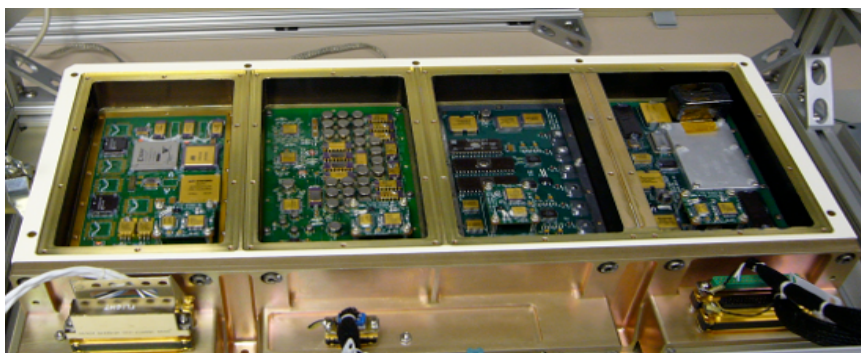
Quantifying the sensitivity of the configuration memory is therefore mandatory in order to evaluate the sensitivity of a specific application. These researches focused on the methods and tools to evaluate the potential weaknesses of fault-tolerant applications when implemented in SRAM-based FPGAs. A TMR (Triple Modular Redundancy) architecture of a crypto-processor was selected and implemented in a Virtex II. The HIF (Heavy Ion Facility) cyclotron of UCL was used to put in evidence the significant contribution of the SRAM configuration memory to the sensitivity of any application implemented on a SRAM-based FPGA. The static cross-section was obtained and combined with the results of fault injection experiments, this to predict the error-rates of the implemented fault-tolerant crypto-processor which were classified in three main families of outputs: error detected, falsely detected error (the result was correct but the TMR output corresponds to detected an error) and undetected errors (the TMR issued an error not seen by the voter).

The final results of these experiments, presented in 2010 at NSREC and published at IEEE TNS, are depicted in Table 3 and show that the maximum underestimation factor being less than 2. This difference might be the result of MBUs not being considered in the fault injection campaign. Falsely detected errors follow a reverse trend: the prediction overestimates the measure by a factor close to 5. This could be explained by the small number of errors of this type observed during radiation ground testing. It is important to note that in such a case, the prediction is certainly closer to the reality than the measure.

**Table 3. Measured vs. predicted error rates for the TMR implemented in the studied FPGA**

Error rate	Particles	Detected errors	Falsely detected errors	Undetected errors
Measured	Carbon	$1.04 \times 10^{-4}$	N/A	N/A
	Argon	$2.84 \times 10^{-3}$	$6.67 \times 10^{-6}$	$7.78 \times 10^{-5}$
Predicted	Carbon	$9.53 \times 10^{-5}$	$1.55 \times 10^{-6}$	$2.09 \times 10^{-6}$
	Argon	$1.94 \times 10^{-3}$	$3.16 \times 10^{-5}$	$4.25 \times 10^{-5}$

The TMR crypto-core application was implemented in an experiment, so-called COTS, accepted to be included in the payload of the LWS (Living With a Star) satellite of NASA. The results obtained in flight, launch scheduled October 2012, will be confronted to those issued from radiation ground testing and fault injection experiments as well as to those predicted by MUSCA SEP3 simulator. **Figure 6.3** depicts the flight version of the SET (Space Environment Testbed), in which the one of the left is the COTS experiment developed at TIMA/ARIS around the TMR crypto-core implemented in the Virtex II FPGA experiment



**Figure 6.3** The SET experiment included in the payload of NASA LWS (Living With a Star) satellite