Jornada de Seguimiento de Proyectos, 2010
Programa Nacional de Tecnologías Informáticas

Microarchitecture and Compilers for Future Processors (TIN2007-61763)

Antonio González
Universidad Politécnica de Catalunya
Intel-UPC Barcelona Research Center

Abstract

This project investigates on the design of next decade microprocessors, taking into account the new opportunities and challenges of future technologies, and the characteristics of future workloads. The project focuses on key areas of microarchitecture and compilers for future microprocessors: (i) power reduction and temperature control, (ii) speculative multithreaded processors, (iii) clustered microarchitectures, (iv) instruction-set architecture design, (v) memory architecture, (vi) variability-aware architectures, and (vii) co-designed virtual machines.
Keywords: Microprocessors, microarchitecture, compilers

1 Project Goals

This project focuses on novel microarchitecture and compiler solutions for future generation microprocessors. The tasks of the project and their specific objectives are detailed below.

1.1. Power Reduction and Temperature Control

- Reduce power and energy for multithreaded applications running on multi-core systems.
- Design new dynamic thermal solutions for multi-core systems.

1.2. Speculative Multithreaded Processors

- Reduce the dynamic size of the pre-computation slice.
- Improve the data cache locality of speculative multithreaded processors.
- Design a novel speculative multithreading paradigm, called Anaphase, for leveraging multiple cores to improve single-thread performance in multi-core systems.

1.3. Clustered Processors

- Design power-effective clustered multithreaded architectures for both single-threaded and multi-threaded workloads.
- Investigate compilation techniques to improve the performance of the proposed architecture.

1.4. ISA Extensions for Dynamically Scheduled Processors

- Develop a microarchitecture simulator that models an out-of-order processor with instruction predication and register windows.
- Investigate new techniques to execute predicated code on out-of-order processors.
- Design an efficient implementation of register windows for out-of-order processors.

1.5. Memory Architecture

- Characterize NUCA caches in a multicore environment. Characterize strengths and weaknesses of the main schemes in the literature.
- Propose novel designs of multicore NUCA caches.
- Implement global replacement mechanisms in distributed multicore caches.

- Evaluate network configurations and coherence protocols for distributed caches in terms of performance, energy consumption and temperature.
- Design new efficient and simple hardware schemes to implement transactional memory.

1.6. Variability-Aware Architectures

- Characterize the main sources of variations and their impact on delay and power.
- Propose adaptive techniques for the main circuits in a microprocessor to operate close to the common case.
- Increase the efficiency (higher performance and lower power) of circuits by proposing new variation-aware designs that improve latency and power.

1.7. Co-Designed Virtual Machines

- Design and develop a research infrastructure to for co-designed virtual machines.
- Propose novel hw/sw co-designed approaches to reduce the overheads of virtual machines.
- Implement and evaluate novel speculative optimizations in simple processors with the aim to reduce power and area.

2 Level of Success of the Project

Most of the objectives proposed have already been accomplished during these first two and a half of the project. The level of success for each of the different research areas is described below.

2.1. Power Reduction and Temperature Control

Multicore processors have become mainstream in all segments. Energy consumption is one of major hurdles in multi-core systems and the workload imbalance among cores is a main source of energy inefficiency.

In this task, we developed a novel mechanism called meeting point [19], which dynamically estimates the criticality of the threads in a parallel execution. Knowing the criticality of each thread has multiple applications. In particular we designed two novel schemes called thread delaying and thread balancing by using the thread criticality information to save energy and improve performance, respectively. Thread delaying combines per-core dynamical voltage-frequency scaling (DVFS) and meeting point thread characterization to reduce energy consumptions on non-critical threads. Our experiments shows that it can achieve up to 40% energy savings without any performance loss for 4-cores and 8-cores systems. Thread balancing gives higher priority in the issue queue of a simultaneous multithreaded (SMT) core to the critical thread. The experiments show that it improves performance from 1% to 20%.

We also proposed a technique called thread fusion [30], which reduces power consumption when a SMT core is executing two threads from a balanced parallel application. Two dynamic instances of the same static instruction, each from a different thread are merged into a single instruction, consuming half of the resources of front-end pipeline stages. Our experiments show an average energy reduction of 10% with less than 1% impact on performance.

In the area of new thermal solutions, we proposed different dynamic thermal management schemes based on Thin-Film Thermoelectric (TFTECs) cooling devices to handle multiple hotspots in multi-core systems [27]. Results show that our TFTEC-based proposal achieves a performance that is within 8% of a thermally unconstrained processor.

We also evaluated different dynamic thermal management techniques such as thread migration, local DVFS and global DVFS for multi-cores to boost performance while controlling the operating

temperature [26]. The experiments show that local DVFS is the most effective scheme in almost all configurations and the combination of global DVFS and thread migration performs quite well compared to local DVFS.

2.2. Speculative Multithreaded Processors

We proposed techniques to improve the cache behavior and the size of the pre-computation slices in speculative multithreaded processors. The first technique, called Selective Replication, mitigates the loss of temporal and spatial locality with respect to a sequential execution. As most of the additional misses correspond to the same static instructions, we proposed a technique that detects which static instructions are likely to be delinquent and in case of miss, the missing line is replicated in all thread units.

Besides, we observed that there are many instructions executed by the pre-computation slices that do not compute any value consumed in the speculative thread, but they are required to compute the control-flow. We proposed a technique that removes those branches of the pre-computation slices that are highly predictable as well as all the instructions just required to compute such branches. The original pre-computation slice is split into multiple shorter slices and a Slice Predictor determines which one is used. Both techniques combined obtain a performance improvement of around 20% on a set of hard to parallelize benchmarks [9].

Finally, we proposed a novel speculative multithreading technique called Anaphase. It features a innovative partitioning algorithm [11], which automatically decomposes applications into speculative threads at instruction granularity, and a set of novel and cost-effective hardware mechanisms that support the execution of the resulting threads in a multi-core system.

Our studies demonstrated the effectiveness of Anaphase for boosting single-thread performance, resulting from exploiting ILP, TLP and MLP, with high accuracy, and low overheads, even in hard to parallelize applications where traditional coarse-grain speculative multithreading schemes do not perform well. In particular, Anaphase boosted Spec2006 applications by 41% on average, and up to 2.6x for some selected applications [10] on a dual core processor.

2.3. Clustered Processors

In the area of clustered processors, we proposed a heterogeneous micro-architecture where voltages and frequencies can be varied at run-time at cluster granularity. This novel approach has better flexibility and adaptability than previous proposals where voltages and frequencies could only be varied when new applications were scheduled [19].

Moreover, we enhanced this microarchitecture by proposing novel compile-time analysis that identifies code regions that may take advantage of similar voltage/frequency pairs [19]. We also proposed schemes for acyclic [28] and cyclic [2] instructions scheduling, cluster assignment and register allocation.

We proposed a novel ROB scheme able to allocate more in-flight instructions than conventional ROB designs without enlarging its size [3][31].

For multithreaded clustered processors, we proposed novel schemes that make a better distribution of the available resources among the running threads significantly improving the processor performance [32].

Finally, we proposed schemes that combine hardware and software to make a more adaptive and more complex-effective assignment of instructions to clusters [20]. In particular, we proposed a scheme that generates an initial partition of the code at compile time, and then it refines this assignment at run time through simple hardware mechanisms.

2.4. ISA Extensions for Dynamically Scheduled Processors

We designed and implemented a simulator that models an out-of-order processor supporting the IPF ISA, with full predication and register windows. It was developed from scratch by using the Liberty framework from Princeton University, and we proposed several new techniques to support predication and register windows on dynamically scheduled processors.

First, we proposed a technique that allows the efficient implementation of predication in dynamically scheduled processors, which is based on the prediction of predicates [47]. Our technique significantly improves branch prediction accuracy by taking advantage of the early resolution of some predicates with respect to the branches that use them [48].

Second, we proposed three techniques for early release of physical registers for systems with register windows, that drastically reduce register requirements, and hence their cost [49]. These techniques exploit several observations on the semantics of subroutine call conventions.

With the proposed technique, the number of required registers is reduced to the bare minimum (i.e. logical registers plus one) without any significant performance degradation with respect to unlimited number of registers [7]. Moreover, its implementation is less complex than previous related approaches.

2.5. Memory Architecture

We proposed several techniques related to transactional memory, non-uniform cache (NUCA) and distributed cache architectures for multicore processors. At the same time, we did some works on design-space exploration and power reduction.

First, we did a detailed analysis of NUCA caches in multicore processors [35],[39]. The results of the analysis pointed out the replacement policy as a critical NUCA policy, and we proposed two novel mechanisms. The Last Bank mechanism [37] includes an extra bank in the center of the NUCA cache that behaves like a Victim Cache. The LRU-PEA [9] is based on a novel prioritization for data blocks in the NUCA cache, which is used to decide the most appropriate data to evict from the NUCA cache.

We also worked on distributed cache architectures for large-scale multicore processors. Our first proposal [38] [41] is based on Cooperative Caching. In this proposal, the coherence engine is redesigned to allow for its partitioning and thus, eliminate the size constraints imposed by the duplication of all tags. At the same time, a global replacement mechanism has been added to improve the usage of cache space. Then, we evaluated several network configurations for distributed caches in terms of performance, energy consumption and temperature [6] [44]. Finally, we proposed a technique to save leakage in distributed caches by selectively switching off the less used lines. This technique relies on analyzing the information of coherence protocols [34]

Another area related to the memory hierarchy of multicore processors that we explored was transactional memory. The first step was to analyze and evaluate the most common techniques proposed in the recent literature [40]. This analysis pointed out a penalty that is introduced when large transactions are executed, particularly, in the presence of contention. Therefore, we proposed FasTM [36] to reduce this problem. FasTM increases the performance significantly, maintaining the simplicity of the design.

2.6. Variability-Aware Architectures

In this area, we studied reliability topics such as NBTI, soft errors, fuses and novel memory cells.

On the memory cell area, we proposed a novel macro-cell [46] that has better characteristics under process variations. When using this macro-cell, memory structures still work at points where

conventional cells would not work. We also analyzed the usage of 3T1D cells for future technologies and we showed that they work better under process variations than conventional 6T cells [4][42][43][44]. We also evaluated the impact on performance predictability of memory cells and techniques to optimize it [7][51]. We proposed a technique to detect erratic bits [17] and modeled the variability on current and future memory technologies [28].

We also worked on the area of detection, confinement and correction of errors caused by variations [1][9][13][14][15][16][17][18]. We proposed a robust design for the control logic of the processors [18][23][24][25]. The technique proposed verifies that data consumed by instructions is correct. This cover potential faults in the datapath, the register file, the selection logic, the issue logic, etc.

Another focus was the design of robust regular circuitry. These designs are much more restrictive than conventional ones but they tolerate a larger amount of variation and they reduce the design time of circuits [46].

2.7. Co-Designed Virtual Machines

One of the challenges we have found when doing research on co-designed virtual machines is the lack of a robust, flexible and complete tool to evaluate such systems. For this purpose, we recently started a task to design and implement a tool chain to be used in our future research activities.

Another focus of our research in this part has been the code of the virtual machine itself, referred to as the optimizer from now on for simplicity. We have started exploring mechanisms to execute the optimizer more efficiently. In particular, we have analyzed what are the start-up overheads of the optimizer. Reducing start-up overheads is mainly important for: (i) short applications and (ii) interactive applications. We found that an important overhead is the constructions of the Data Dependence Graph and have proposed a scheme that reduces he overheads from 19% to 3%.

We have been working on a scheme to efficiently execute 64-bit binaries in narrow (e.g., 16 bit) pipelines. In this area, we proposed two novel speculative optimizations to remove unnecessary narrow 16-bit instructions. The first technique is referred to Global Productiveness Propagation and it is based on speculatively removing those instructions that do not modify the architectural state at 16-bit granularity given a code region (in our case, routines). We have observed that this technique is able to eliminate 7% of the dynamic instruction stream. On the other hand, the technique referred as Local Productiveness Pruning eliminates unnecessary 16-bit instructions at the 64-bit instruction granularity. This reduces the dynamic stream by 20%.

Finally, we designed a specialized functional unit consisting of multiple processing elements and we have integrated it into a general-purpose simple processor, with special emphasis in the interconnection between this unit and memory, the register file and the other functional units. We developed a novel code generation and optimization techniques to make an effective use of this unit in a co-design virtual machine system. Such techniques include instruction scheduling and instruction fusion. The proposed techniques speed up general-purpose applications by 15%.

3 Results Indicators

3.1. Journal Publications

[1] J. Abella, X. Vera, O. Unsal, O. Ergin, A. González, J.W. Tschanz, Refueling: Preventing Wire Degradation due to Electromigration, IEEE Micro, November-December 2008

[2] A. Aletà, J.M. Codina, J. Sánchez, A. González, D. Kaeli. AGAMOS: A Graph-Based Approach to Modulo Scheduling for Clustered Microarchitectures, IEEE Transactions on Computers, vol. 58, n. 6, pp. 770-783, Feb. 2009.

[3] F. Latorre, G. Magklis, J. González, P. Chaparro and A. González, CROB: Implementing a Large Instruction Window through Compression. Trans. on High-Performance Embedded Architectures and Compilers, vol. 3, no. 2 2008.

[4] X.Liang, R.Canal, G.Y. Wei, D. Brooks, Replacing 6T SRAMs with 3T1D DRAMs in the L1 Data Cache to Combat Process Variability, IEEE Micro Micro's Top Picks from Computer Architecture Conferences, v.28 n.1 pp.60-68 Jan-Feb 2008

[5] C. Molina, J. Tubella and A. González, Reducing Misspeculation Penalty in Trace Level Speculative Multithreaded Architectures, Lecture Notes in Computer Sciences, Springer-Verlag Heidelberg, vol 4759, January 2008.

[6] M. Monchiero, R. Canal and A. González,Power/Performance/Thermal Design Space Exploration for Multicore Architectures, IEEE Transactions on Parallel and Distributed Systems v. 19 n. 5 pp.666-681, May 2008.

[7] Eduardo Quiñones, Joan-Manuel Parcerisa and Antonio González. Leveraging Register Windows to Reduce Physical Registers to the Bare Minimum. IEEE Transactions on Computers (accepted for publication).

[8] X. Vera, B. Lisper, J. Xue, Data Caches Locking for Tight Timing Calculation, in ACM Transactions on Embedded Computing Systems (TECS), vol 7(1), 2007

3.2. Symposia Publications

[9] R. Ranjan, P. Marcuello, F. Latorre and A. González, P-slice Effective Speculative Multithreading, in Proceedings of the International Symposium on High-Performance Computing (HiPC'09), Kochi (India), December 2009.

[10] C. Madriles, P. López, J.M. Codina, E. Gibert, F. Latorre, A. Martínez, R. Martínez, A. González, "Boosting Single-thread Performance in Multi-core Systems through Fine-grain Multi-Threading", In Proceedings of the 36th International Symposium on Computer Architecture (ISCA'09), Austin (Texas), June 2009.

[11] C. Madriles, P. López, J.M. Codina, E. Gibert, F. Latorre, A. Martínez, R. Martínez, A. González, "Anaphase: A Fine-Grain Thread Decomposition Scheme for Speculative Multithreading", In the Proceedings of 18th International Conference on Parallel Architecures and Compiler Techniques (PACT'09), September 2009.

[12] J. Abella, J. Carretero, P. Chaparro, X. Vera, A. González, Dynamic Errors: Symptoms and Solutions, Presentation in the Industrial Track of the 14th International European Conference on Parallel and Distributed Computing (EURO-PAR'08), Las Palmas de Gran Canaria (Spain), August 2008

[13] J. Abella, P. Chaparro, X. Vera, J. Carretero, A. González, On-Line Failure Detection and Confinement in Caches, In Proceedings of the 14th International On-Line Testing Symposium (IOLTS'08), Rodhes (Greece), July 2008

[14] J. Abella, P. Chaparro, X. Vera, J. Carretero, A. González, High-Performance Low-Vcc In-Order Core, In Proceedings of the 16th International Symposium on High-Performance Computer Architecture (HPCA'10), Bangalore (India), January 2010

[15] J. Abella, J. Carretero, P. Chaparro, X. Vera, A. González, Low Vccmin Fault-Tolerant Cache with Highly Predictable Performance, In Proceedings of the 42nd International Symposium on Microarchitecture (MICRO'09), New York (New York), December 2009

[16] J. Abella, X. Vera, A. González, Penelope: the NBTI-Aware Processor, In Proceedings of the 40th International Symposium on Microarchitecture (MICRO'07), Chicago (Illinois), December 2007 (a version of this paper has been published in the internal Intel conference Intel Design and Test Technolgy Conference 2007)

[17] J. Abella, X. Vera, O. Unsal, O. Ergin, A. González, Fuse: A Technique to Anticipate Failures due to Degradation in ALUs, In Proceedings of the 13th International On-Line Testing Symposium, Heraklion (Crete, Greece), July 2007

[18] J. Abella, X. Vera, O. Unsal, A. González, NBTI-Resilient Memory Cells with NAND Gates for Highly-Ported Structures, In the 1st Workshop on Dependable and Secure Nanocomputing (WDSN'07) held in conjunction with DSN'07, Edinburgh (UK), June 2007 (also a poster in DSN'07), US Patent 7,447,054

[19] A. Aletà, J.M. Codina, A. González, D. Kaeli, Heterogeneous Clustered VLIW Microarchitectures, In the 2007 International Symposium on Code Generation and Optimization (CGO), March 2007.

[20] Q. Cai, J. M. Codina, J. González, A. González. A software-hardware hybrid steering mechanism for clustered microarchitecture, In the 22nd IEEE Int. Parallel and Distributed Processing Symposium, IPDPS – 2008.

[21] Q. Cai, J. Gonzalez, R. Rakvic, G.Magklis, P. Chaparro and A. Gonzalez. Meeting Points: Using Thread Criticality to Adapt Multicore Hardware to Parallel Regions, International Conference on Parallel Architecture and Compilation Techniques (PACT) ,2008

[22] J. Carretero, P. Chaparro, J. Abella, X. Vera, A. González, End-to-End Register Data-Flow Continuous Self-Test, In Proceedings of the 36th International Symposium on Computer Architecture, Austin (Texas), June 2009 (a version of this paper has been published in the internal Intel conference Intel Design and Test Technolgy Conference 2008)

[23] J. Carretero, X. Vera, J. Abella, P. Chaparro, A. González, A Low-Overhead Technique to Protect the Issue Control Logic against Soft Errors, In the 5th Workshop on Silicon Errors in Logic - System Effects (SELSE'09), Stanford (California), March 2009

[24] J. Carretero, X. Vera, P. Chaparro, J. Abella, On-line Failure Detection in Memory Order Buffers, In Proceedings of the International Test Conference (ITC'08), Santa Clara (California), October 2008

[25] J. Carretero, X. Vera, J. Abella, P. Chaparro, A. González, A Low-Overhead Technique to Protect the Issue Control Logic against Soft Errors, In Proceedings of the XIX Jornadas de Paralelismo, Castellón (Spain), September 2008

[26] P. Chaparro, J. Gonzalez, G. Magklis, Q. Cai and A. Gonzalez. Understanding the Thermal Implications of Multi-Core Architectures. IEEE Transactions on Parallel and Distributed Systems, Special Section on CMP Architectures, 18(8), pp. 1055-1065, IEEE Computer Society, Aug 2007.

[27] P. Chaparro, J. Gonzalez, Q. Cai and G. Chrysler. Dynamic Thermal Management using Thin-Film Thermoelectric Cooling, International Symposium on Low Power Electronics and Design (ISLPED), 2009.

[28] J.M. Codina, J.Sánchez, A. González. Virtual Cluster Scheduling through the Scheduling Graph, In the 2007 International Symposium on Code Generation and Optimization (CGO), March 2007.

[29] S. Ganapathy, R. Canal, A. González, A. Rubio, Circuit Propagation Delay Estimation Through Multivariate Regression-Based Modeling Under Spatio-Temporal Variability, In Proceedings of the 2010 IEEE Design, Automation and Test in Europe Conference (DATE'10), Dresden (Germany) March 2010.

[30] J. Gonzalez, Q. Cai, P. Chaparro, G. Magklis, R. Rakvic and A. Gonzalez. Thread Fusion, International Symposium on Low Power Electronics and Design (ISLPED), 2008.

[31] F. Latorre, G. Magklis, J. González, P. Chaparro and A. González, Building a Large Instruction Window Through ROB Compression, in MEDEA Workshop MEmory performance: DEaling with Applications, systems and architecture; Spetember 2007.

[32] F. Latorre, J. González, A. González. Efficient Resource Assignment Schemes for Clustered Multithreaded Processors. In 22nd IEEE International Parallel and Distributed Processing Symposium, April 2008.

[33] J. Lira, C. Molina and A. González, LRU-PEA: A Smart Replacement Policy for Non-Uniform Cache Architectures on Chip Multiprocessors, In Procs of the 27th International Conference on Computer Design (ICCD'09), Lake Tahoe (California), October 2009.

[34] M. Monchiero, R. Canal and A. González, Using Coherence Information and Decay Techniques to Optimize L2 Cache Leakage in CMPs, In Proceedings of the IEEE 38th International Conference on Parallel Processing (ICPP'09), Vienna (Austria), September 2009

[35] J. Lira, C.Molina and A. González, Performance Analysis of Non-Uniform Cache Architecture Policies for Chip-Multiprocessors Using the Parsec V2.0 Benchmark Suite, In Proceedings of the XX Jornadas de Paralelismo, A Coruña, September 2009.

[36] M. Lupon, G. Magklis, and A. Gonzalez. FASTM: A Log-Based Hardware Transactional Memory with Fast Abort Recovery. In the Proceedings of 18th International Conference on Parallel Architecures and Compiler Techniques (PACT'09). September 2009.

[37] J. Lira, C. Molina and A. González, Last Bank: Dealing with Address Reuse in Non-Uniform Cache Architecture for CMPs, In Procs of the International European Conference on Parallel and Distributed Computing (Euro-Par'09), Delft (The Netherlands) 2009.

[38] E. Herrero, J. González, R. Canal, Distributed Cooperative Caching, In Proceedings of the IEEE/ACM 17th International Conference on Parallel Architectures and Compilation Techniques (PACT'08), Toronto (CA), October 2008

[39] J. Lira, C.Molina and A.González, Analysis of Non-Uniform Cache Architecture Policies for Chip-Multiprocessors Using the Parsec Benchmark Suite, In Procs of the 2nd Workshop on Managed Many-Core Systems (MMCS'09),Washington D.C, (USA),March 2009.

[40] M. Lupon, G. Magklis, and A. Gonzalez. Version Management Alternatives for Hardware Transactional Memory. In the Procs of the 9th ACM Workshop on MEmory performance: DEaling with Aplications, systems and architectures (MEDEA'08). October 2008.

[41] E. Herrero, J. González, R. Canal, A Scalable and Power-Efficient Memory Hierarchy for Multicore Architectures, Intel 2008 European Research and Innovation Conference, Leixlip (Ireland), September 2008.

[42] X.Liang, R.Canal, G.Y. Wei, D. Brooks, DRAM-based On-Chip Cache Architectures to Combat Process Variations, Intel 2008 European Research and Innovation Conference, Leixlip (Ireland), September 2008

[43] X.Liang, R.Canal, G.Y. Wei, D. Brooks, Process Variation Tolerant 3T1D-Based Cache Architectures, The 40th International Symposium on Microarchitecture (MICRO-40), Chicago, (Illinois), December 2007

[44] X.Liang, R.Canal, G.Y. Wei, D. Brooks, Process Variation Tolerant Register Files Based on Dynamic Memories, ASGI'07,Workshop on Architectural Support for Gigascale Integration (In conjunctoin with ISCA 2007), San Diego, (California), June 2007

[45] M. Monchiero, R. Canal and A. González, Design Space Exploration for Multicore Architectures: A Power/Performance/Thermal View, Proc. of the 20th ACM Int. Conference on Supercomputing, Cairns (Australia)

[46] M. Pons, F. Moll, A. Rubio, J. Abella, X. Vera, A. González, Via-Configurable Transistors Array: A Regular Design Technique to Improve ICs Yield, In the 2nd International Workshop on Design for Manufacturability and Yield (DFM&Y) held in conjunction with ITC'07, Santa Clara (California), October 2007

[47] E. Quiñones, J.-M. Parcerisa and A. González. Selective Predicate Prediction for Out-of-Order Processors. In Proc. Of the Int'l Conf. on Supercomputing (ICS'06), pp. 45-54, Cairns (Australia)

[48] E. Quiñones, J.-M. Parcerisa and A. González. Improving Branch Prediction and Predicated Execution in Out-of-Order Processors. In Proc. of the 13th Int'l Symp. on High-Performance Computer Architecture (HPCA'07), pp 75-84, February 2007.

[49] E. Quiñones, J.-M. Parcerisa and A. González. Early Register Release for Out-of-Order Processors with Register Windows. Proc. of Int. Conf. on Parallel Architectures and Compilation Techniques, pp. 225-234, September 2007

[50] A. Valero, J. Sahuquillo, S. Petit, V. Lorente, R. Canal, P. Lopez and J. Duato, An hybrid eDRAM/SRAM macrocell to implement first-level data caches, In Procs of the 42nd Int Symposium on Microarchitecture (MICRO'09), New York (New York), December 2009

[51] X. Vera, J. Abella, Surviving Errors in Multi-Core Environments, Presentation in the Special session on Reconfiguration and Fault Tolerance in Future Massively Parallel Multi-Core Chips of the 13th International On-Line Testing Symposium (IOLTS'07), Hersonissos-Heraklion (Crete, Greece), July 2007

3.3. PhD Students

During these two years an a half, there were 5 PhD students of the team that successfully completed their PhD (Pedro Chaparro, Josep M Codina, Eduardo Quiñones, Fernando Latorre, Alex Aletà), and 21 more that are still working on it (Llorenç Cruz, Carlos Aliagas, Stefan Bieschewski, Rakesh Ranjan, Indu Bhagat, Abhishek Deb, Marc Pons, Enric Herrero, Carlos Madriles, Pedro López, Govind Shenoy, Shrikanth Ganapathy, Aleksandar Brankovic, Javier Carretero, Rakesh Kumar, Javier Lira, Marc Lupón, Demos Pavlou, Martí Torrents, Gaurang Upasani, Fahimeh Yazdanpanah).

3.4. Patents

During the last 2.5 years, the members of the team have (co-)authored 8 inventions that were filed.

3.5. Technology Transfer

The research group has been extremely active in the area of technology transfer. Some of the members of the team have been working on a joint research lab of the Universitat Politècnica de Catalunya and Intel Corporation. The group has collaborated with various product groups in Intel in the design of future microprocessors and compilers.

3.6. Collaboration with Other Groups

In addition to multiple groups in Intel, we have collaborated with other research groups at: Universidad Politécnica de Valencia, University of Wisconsin-Madison, Edinburgh University, Northeastern University, University of Glasgow, IMEC, Harvard University, and other groups in UPC.

3.7. International Projects

We started a new research project funded by the EU FP7 program. The project is codenamed TRAMS: Terascale Reliable Adaptive Memory Systems. UPC and Intel are partners in the project together with IMEC (Belgium) and the University of Glasgow (UK).

3.8. Other Indicators

The members of the team have participated in more than 15 program committees of symposia, have been general chair for MICRO 2008 and program chair for HCPA 2008, have been members of the editorial board of five journals and have given numerous invited talks in symposia, academia and industry.