



ANCORA: Analysis and Compilation Tools for Reconfigurable Architectures

João M. P. Cardoso

Pedro C. Diniz



INSTITUTO
SUPERIOR
TÉCNICO



HiPEAC Computing Systems Week
Barcelona, Spain, June 2, 2008



ANCORA: Analysis and Compilation Tools for Reconfigurable Architectures

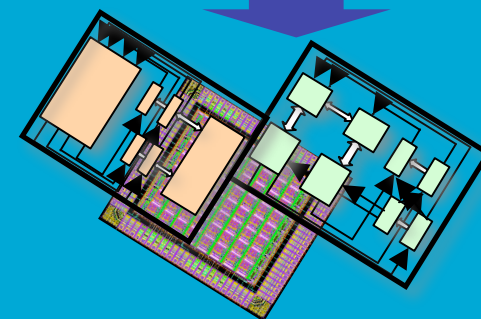


Research:

- Developing compiler techniques for the future computing systems (mainly focusing reconfigurable computing platforms)
- Developing non-traditional computing engines for accelerating algorithms
- Developing Embedded Computing Systems

PROJECTS:

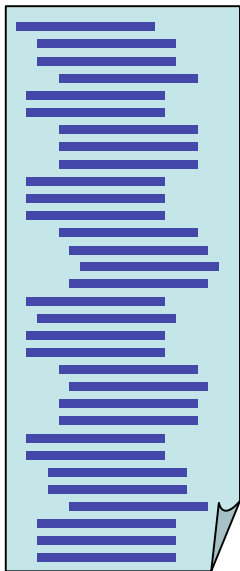
- **VECTOR: Matlab Compilation and Hardware Synthesis of Custom-Vector Processing for Image and Signal Processing Algorithms**, FCT: PTDC/EEA-ELC/71556/2006 [IST/INESC-ID, FEUP]
- **COBAYA: Closing the compilation gap between algorithms and coarse-grained reconfigurable array architectures**, FCT: PTDC/EEA-ELC/70272/2006 [IST/INESC-ID]
- **AMADEUS: Aspects and compiler optimizations for matlab system development**, FCT: PTDC/EIA/70271/2006 [IST/INESC-ID, UNova, UMinho]
- **UPCASE: User-Programmable Context-Aware Services**, [IST + PTInovação, Taguspark]



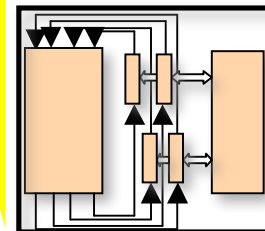
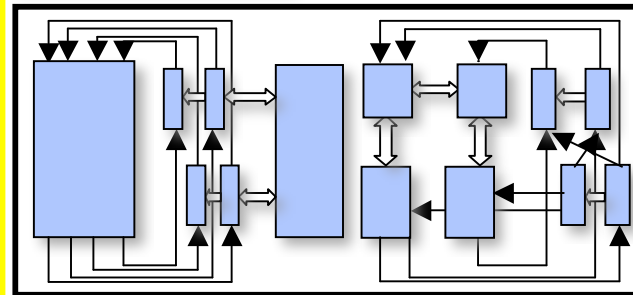
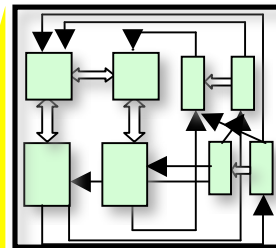
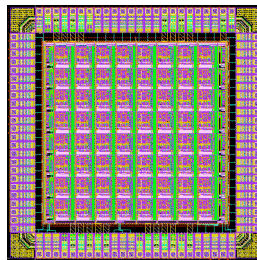
Reconfigurable Architectures Programming

- Future reconfigurable architectures will exacerbate all the programming problems
- Issues:
 - How can programming languages help the compiler?
 - How can architectures help the compiler and tools?

Application Code



Compilation,
Synthesis and
Optimization



Reconfigurable Architectures
and Execution Models

Binary Translation: A Vision

- The Holy Grail: Maintain Binary Compatibility
- How?
 - Use Hot-Spot Compilation Ideas
 - Compiler is very lean - generate instruction-level data-flows
 - Delegate to hardware to “construct” on-the-fly the data-flow
 - Incrementally reconfigure and redo P&R - code improves as it runs.
- Many Challenges ...

