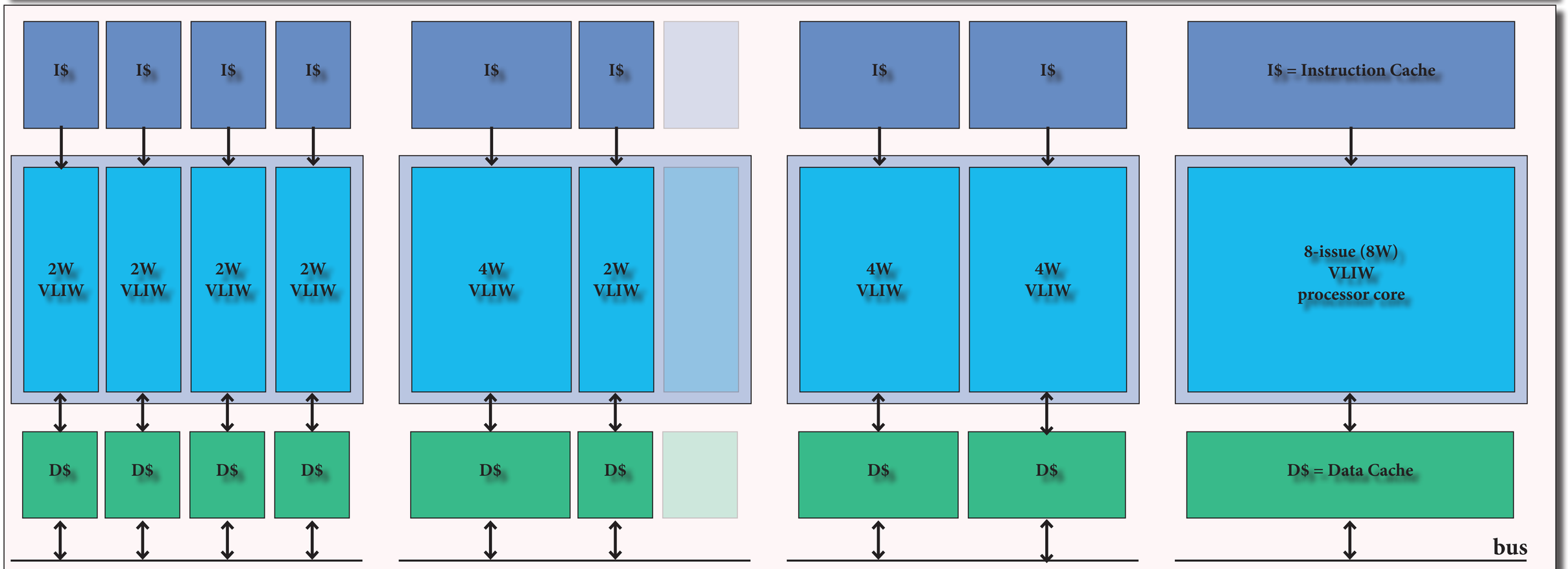
The Liquid Architecture Project
The ρ–VEX Processor

Computer Engineering Laboratory
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, Netherlands

Design of a parameterized and dynamically reconfigurable processor:
* parameterization to allow adaptation to *changing functionalities, program requirements, operating environment*
* run-time adaptation, i.e., during system operation and/or program execution
* current design exchanges between ILP (large core with many parallel units) and TLP (several cores in parallel)
* homegeneous/heterogeneous multi-/many-core within a SINGLE design
* power gating can be supported to save power (see 2nd figure)



4 2W cores + 4 I$ + 4 D$ configuration:
* 4 concurrent 2-issue cores
* 4 I-caches
* 4 D-caches

= low ILP programs & high TLP

1 4W core + 1 2W core + 2 I$ + 2 D$:
* 1 4-issue & 1 2-issue core in parallel
* 2 I-caches (different sizes)
* 2 D-caches (different sizes)

= mix of programs with diff. ILPs
= save power by shutting down a core

2 4W cores + 2 I$ + 2 D$ configuration:
* 2 concurrent 4-issue cores
* 2 I-caches (comprised of smaller ones)
* 2 D-caches (compr. of smaller ones)

= run 2 programs with medium ILP

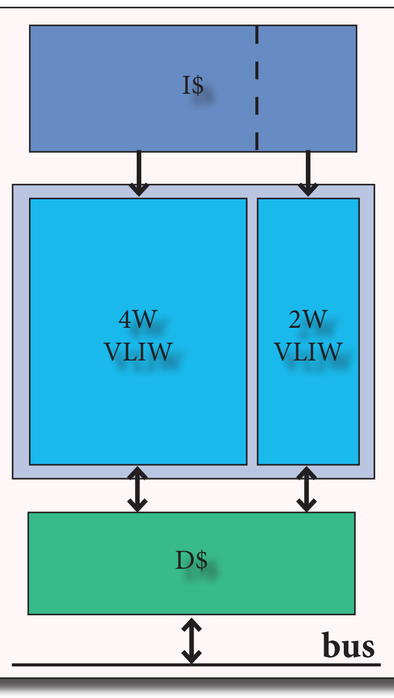
1 8W core + 1 I$ + 1 D$ configuration:
* 1 single 8-issue core
* 1 I-cache (comprised of smaller ones)
* 1 D-cache (compr. of smaller ones)

= run single program with high ILP

Generic Binary (presented at DATE 2013):
* Compile for 8-issue and address them as 2-issue bundles
* Fix dependencies & skip NOPs bundles
* Small change in "update PC" & "skip NOPs" hardware
* Interruptable code (can restart at exactly the same position)
* Dynamic issue-width switching by:
  - programmer, compiler, hardware scheduler, and/or OS
* Updated result: on average 5% performance loss

"Recent" Developments: (published @)
* Dynamically Reconfigurable Register File (@ DATE 2010)
* Multi-ported Register File using BRAMs (@ FPT 2010)
* ρ–VEX 2.0 & Extensions (2 papers @ WRC 2010):
  - pipelining, forwarding, interrupts, exceptions
* Run-time Task Migration (@ ARC 2012)
* Dyn. Issue-width Reconfiguration (@ FPT 2010 & DATE 2011)
* Dynamic Issue-width with 1st level I-cache (@SAMOS 2012)
* Dynamic Fault-Tolerance Support (@ ARC 2013)

Current Demo Explained
* Heterogeneous multi-core
* Design implemented on ML-605
* Dynamic loading of program via host computer
* Single application in generic binary format in I$
* Both 2W and 4W cores run the same code
* When interrupted, execution is moved to the other core
* Context switching (or via RF sharing)
* Output written to frame buffer for display



Current Work: (to be published)
* GRLIB-based SoC close to be finalized
* Run-time reconfigurable design under test
* Linux support under development
* Software-based fault-tolerance under dev.

Other interesting notes:
* Hands-on tutorial available
* Complete lab developed and available
* Work will be released under GPL

Contact information:
Project Leader: Stephan Wong (J.S.S.M.Wong@tudelft.nl)
Project Website: http://rvex.ewi.tudelft.nl/
Lab Website: http://ce.ewi.tudelft.nl