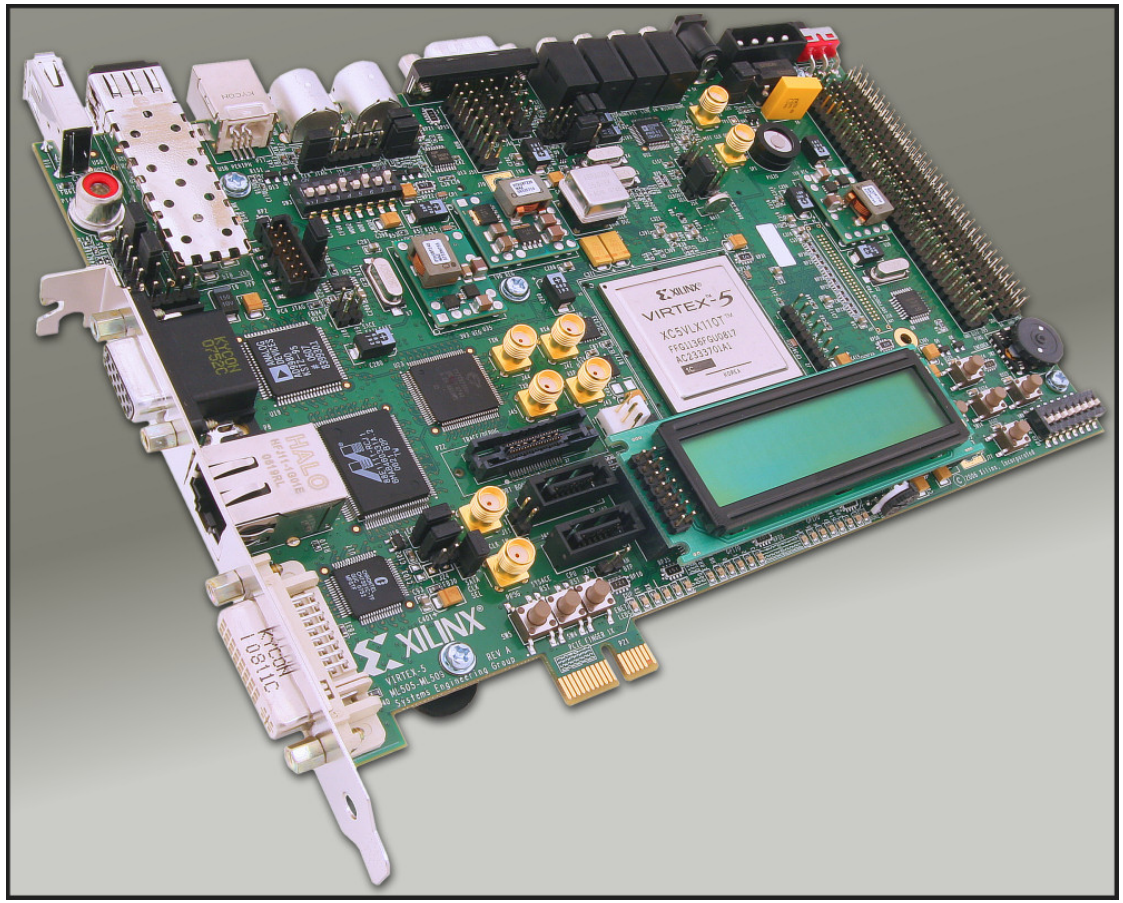
DREAMER v2
(Statically Scheduled Logic Emulation)

Albert Magyar, Richard Lin, Jonathan Bachrach

January 6, 2014

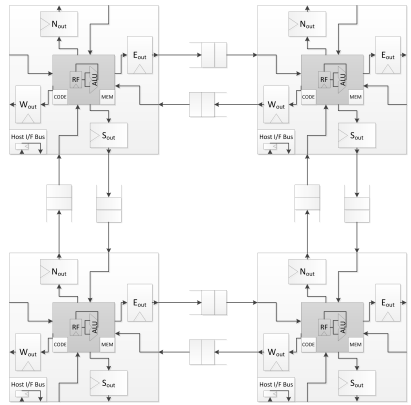
Why design a new reconfigurable logic platform?



- Efficient support for word-width operations, debug visibility
- Enjoy FPGA build times?
- Enjoy recompiling to change wire visibility with ChipScope?

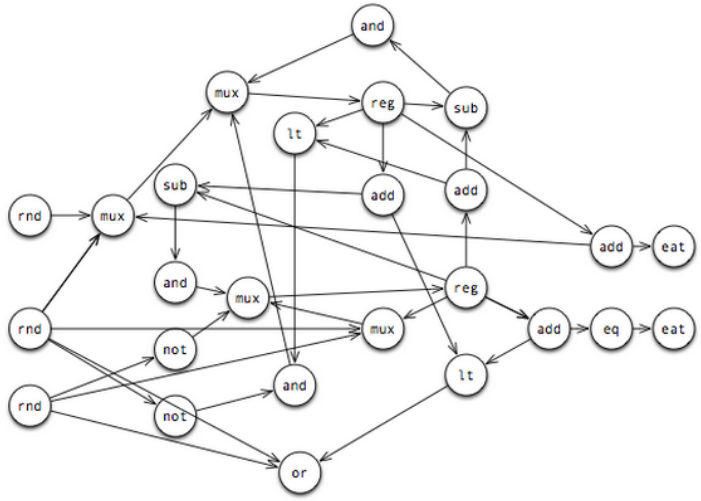
High-level motivation

- We want to build a network of small, parallel processors
- Also supporting tools!



- Occupy a new point in the space between programmability and efficiency
- Explore applications in varied domains
  - Fast logic emulation
  - Providing a new programmable logic target
  - Low-level, low-overhead parallel programming

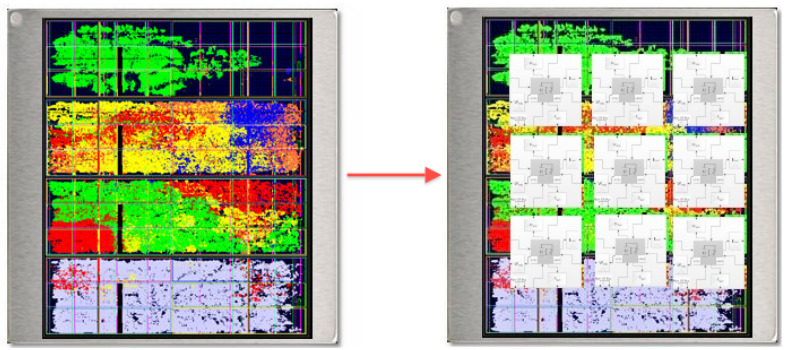
Emulation and prototyping



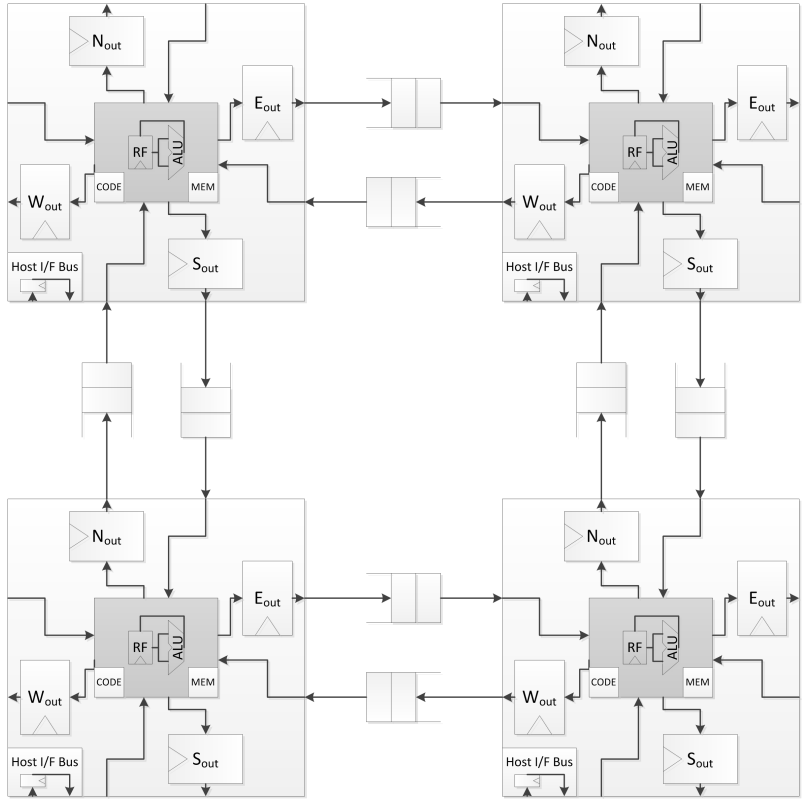
- Close designer productivity loop for hardware creation
  - Provide emulation faster than software simulation
  - Provide better dynamic wire visibility than FPGAs
- Fast compile time is a big win here
  - Chisel designs map to the Chiselator in seconds
  - Overall iteration speed much faster
- Match FPGA in cost
  - Implement array on FPGA for *FPGA overlay*?

Real implementation target

- FPGA alternative
  - Again, beating FPGA tools in compile time is key
  - Hope to beat FPGAs in energy-delay product
    - Multiplex expensive, word-width ALUs in time
- Direct programming
  - Highly-parallel execution in dataflow order
  - Is this better than SIMD? For what applications?
- Maybe an FPGA is the best way to realize this grid $\rightarrow$ *overlay*

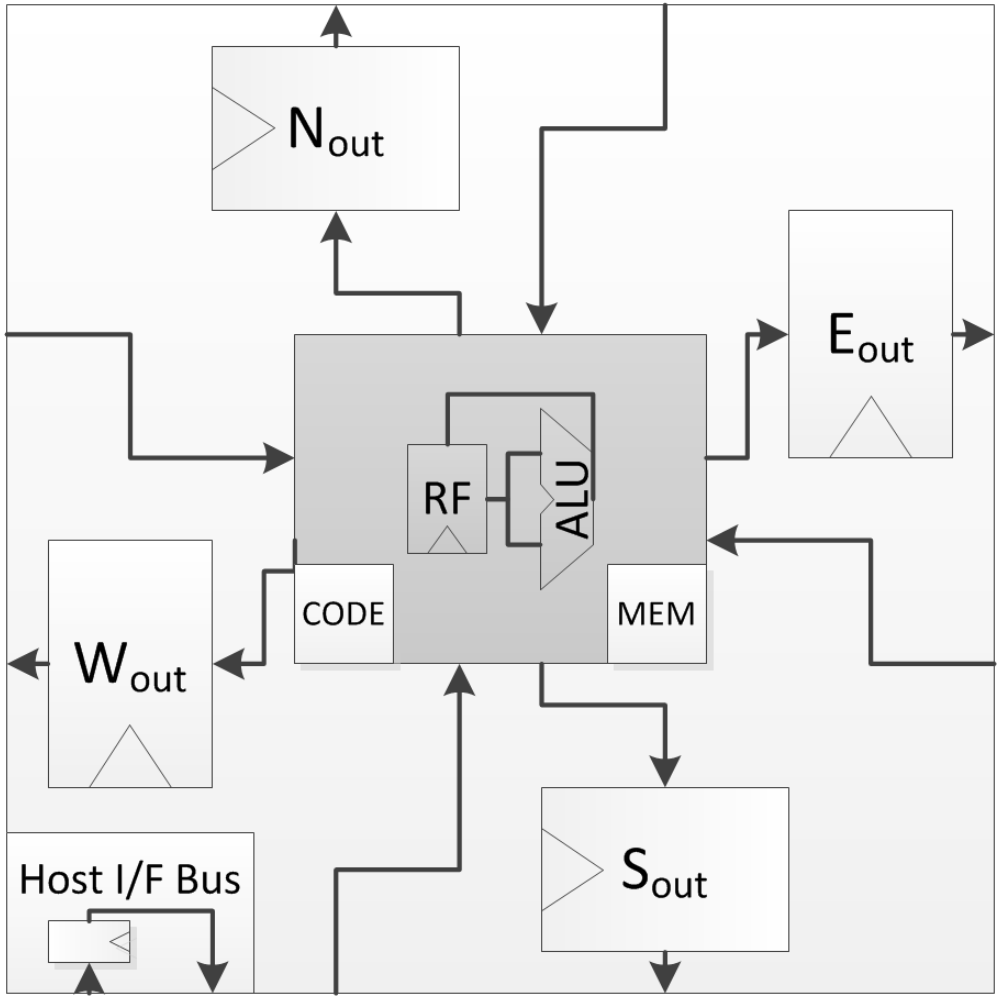


Array architecture



- Massively parallel machine (in theory)
- 4-nearest neighbor connected network
- Static scheduling of operations and data movement
- Ready/valid interface on ports $\rightarrow$ supplants noop storage
- 2-element buffers$\rightarrow$ deadlock less common

Tile architecture



- Extremely reduced ISA meant for emulation
- Operations for logical, arithmetic, and mux nodes
- No branching – simple emulation loop

Initial VLSI results