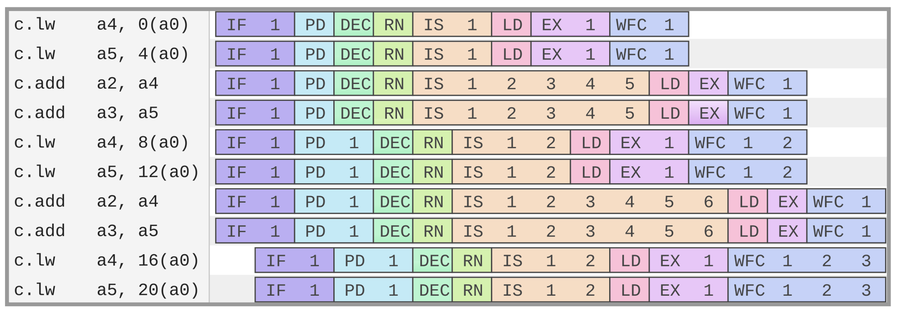
SoomRV: open source out-of-order RISC-V

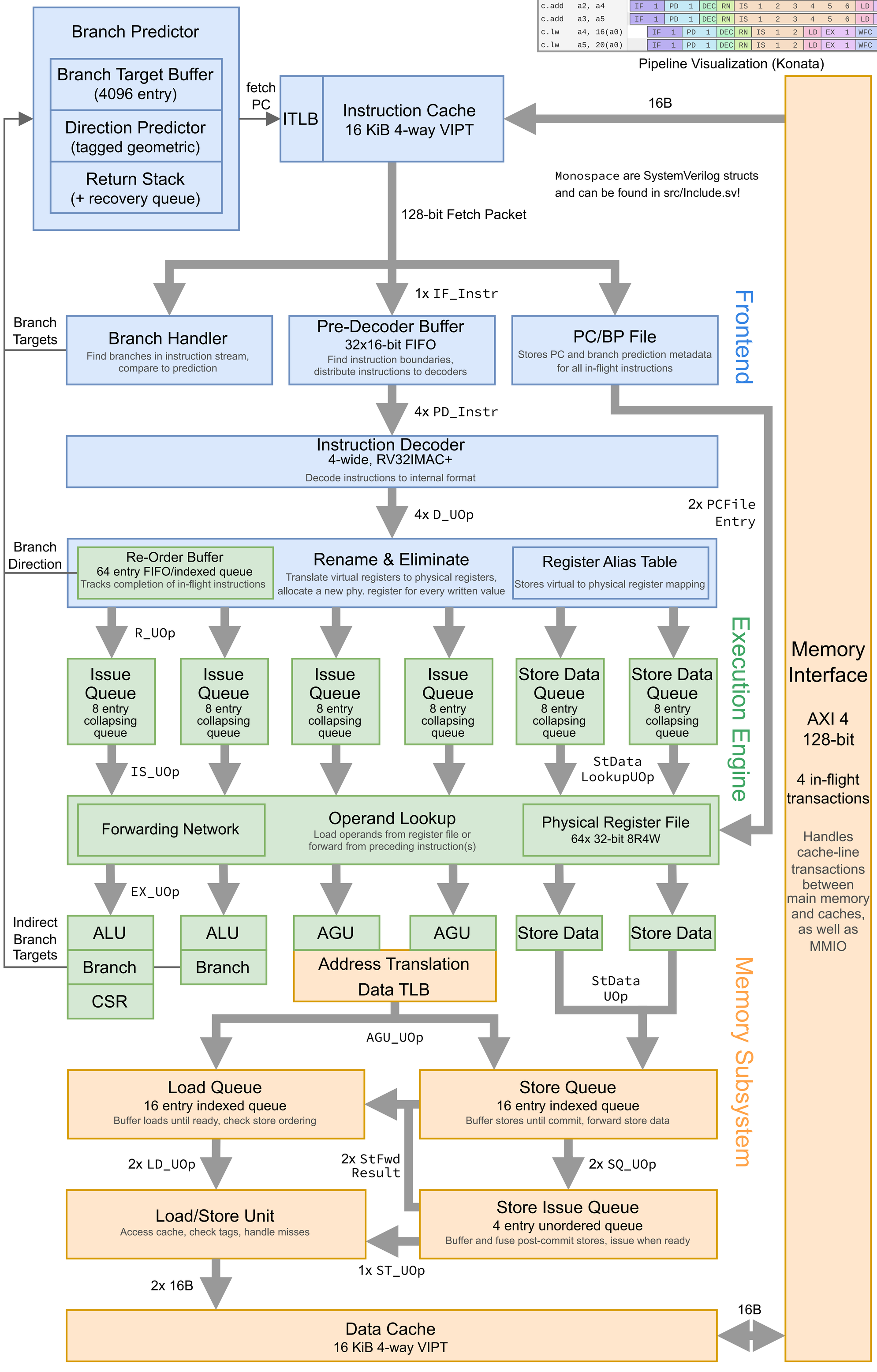
Mathis Salmen | Munich University of Applied Sciences (FK07) | github.com/mathis-s/SoomRV

4-wide superscalar out-of-order RV32IMAC core with focus on simplicity and ease of implementation.
Written in SystemVerilog, verified against Spike, tested on FPGA.

4.69 DMIPS/MHz @ 2.25 IPC (Dhrystone, O1, GCC 12.2.0)
4.05 Coremark/MHz @ 1.16 IPC (Coremark, O3, GCC 12.2.0)



Pipeline Visualization (Konata)



Features

- 10-stage out-of-order pipeline
- 4-wide superscalar
- ISA: rv32imac_zba_zbb
- Privileged Spec (MSU)
- Sv32 virtual memory
- Boots Linux

Frontend

- 128-bit IFetch from 4-way 16KiB VIPT ICache
- Support for 16 and 32-bit instructions
- 0-cycle latency for predicted branches
- 3-cycle latency for late (dir-only) predicted branches
- 4-wide instruction decode
- ~12-cycle mispredict penalty

Execution Engine

- 4-wide rename
- 4-wide retire
- 8 read/4 write register file
- Up to 2 ALU + 2 Memory instructions per cycle
- Small *load-immediate*s are eliminated

Memory Subsystem

- 2-port 16KiB VIPT write-back DCache
- 128-bit AXI4 to main memory and MMIO
- Non-blocking cache misses
- Store fusion
- Cache miss load data is forwarded from AXI
- Cache miss store data is fused into cache line load
- 4-cycle load to use
- ~12-cycle load to use on miss

Future Work

- Further Simplification
- 64-bit Support
- Speculative Issue on Loads
  reduce load latency
- Rename Checkpoints
  reduce mispredict penalty
- Rename Reference Counting
  for move elimination
- Vector/SIMD Instructions
  RISC-V Vector or other
- Area Optimization
  RF banking & port sharing, address deduplication