Implementing out-of-order issue in CVA6 for efficient support of long variable latency instructions

E. Guthmuller & T. Khandelwal
CEA List
eric.guthmuller@cea.fr

Motivation & Microarchitecture

Context:

- VRP [1] core developed in the context of European Processor Initiative (EPI) with support of floating point computing with up to 512 bits of mantissa
- Fork of CVA6 => in-order issue of instructions
- Long (>10 cycles) and variable instruction latency => code difficult to optimize, sometimes impossible due to conditional statements

Our proposal:

- The **VXP** core, as an evolution of the VRP, now supports **Out-of-Order (OoO) execution**
- It implements a variant of Tomasulo's algorithm:
  - One more pipeline stage, called "**dispatch**"
  - Register renaming for all register banks
  - Configurable bypass network
  - No speculated issue (branch, load) to avoid complex store queues in VPFPU
- Some generic CVA6 optimizations for better core scalability: simplified scoreboard, improved store queue, better instruction throughput, …

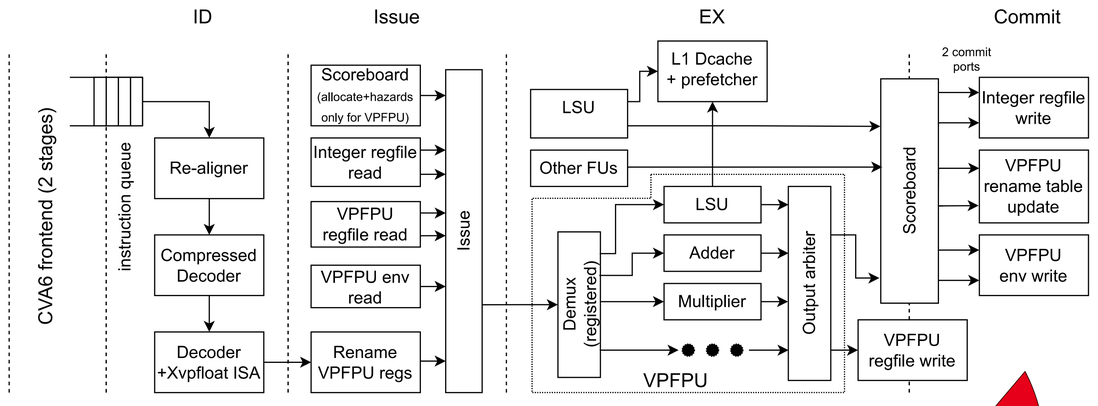

Fig 1. Starting point: VRP [1], CVA6-based in-order

Implementing OoO execution

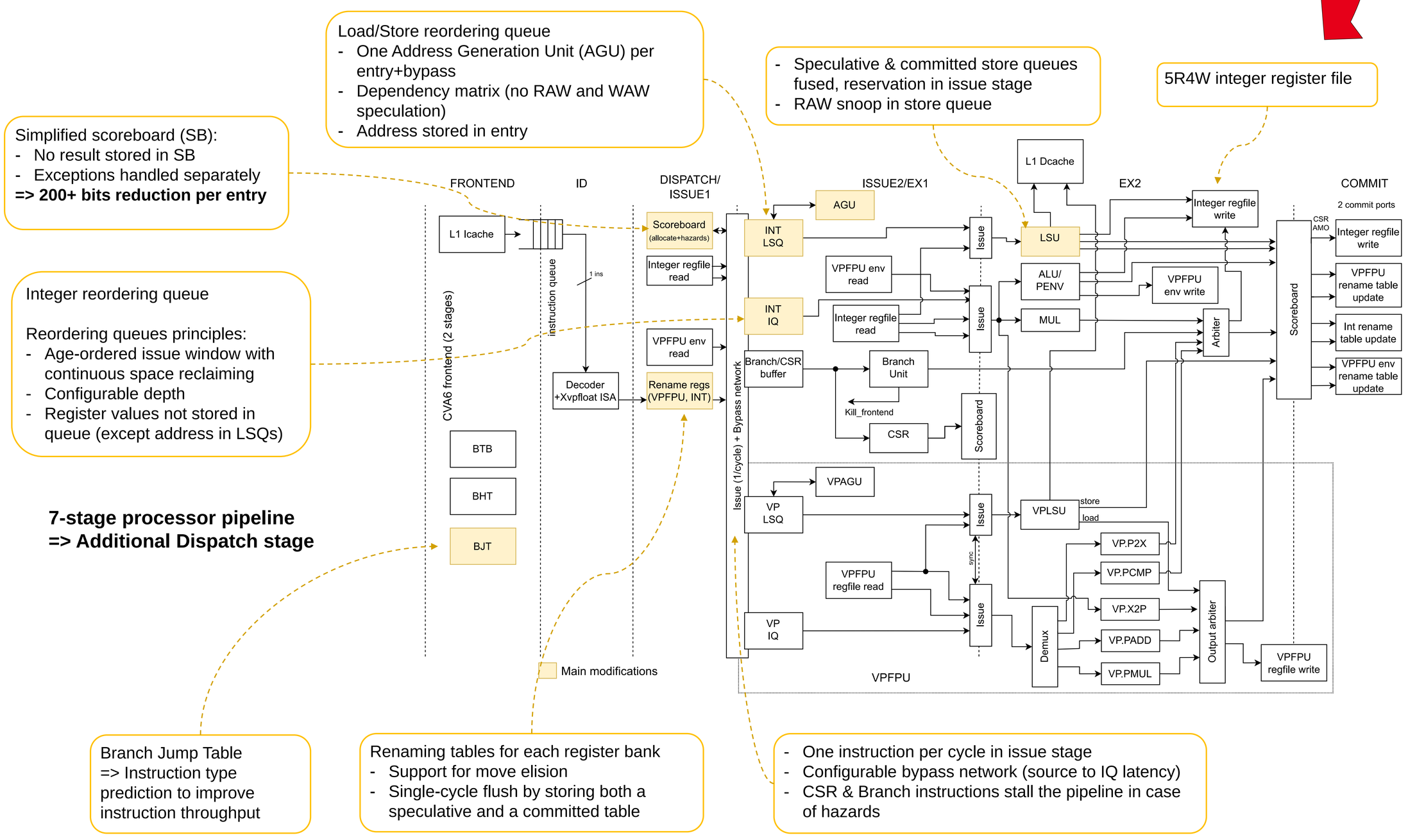

Fig 2. VXP, Variable eXtended Precision processor, with out-of-order execution (main modification of integer pipeline highlighted)

Results

Main design goals of OoO execution achieved:

- Performance of non-optimized (unrolled) BLAS kernels now on par with hand-tuned assembly code
- Sparse matrix-vector multiplication up to 75% faster depending on matrix structure and vector precision

Other improvements:

- Integer performance gains => 10% higher Coremark performance
- 10% frequency increase on FPGA of VXP over VRP

Future works:

- UVM verification using same approach as standard CVA6
- ASIC synthesis and Power Performance Area assessment

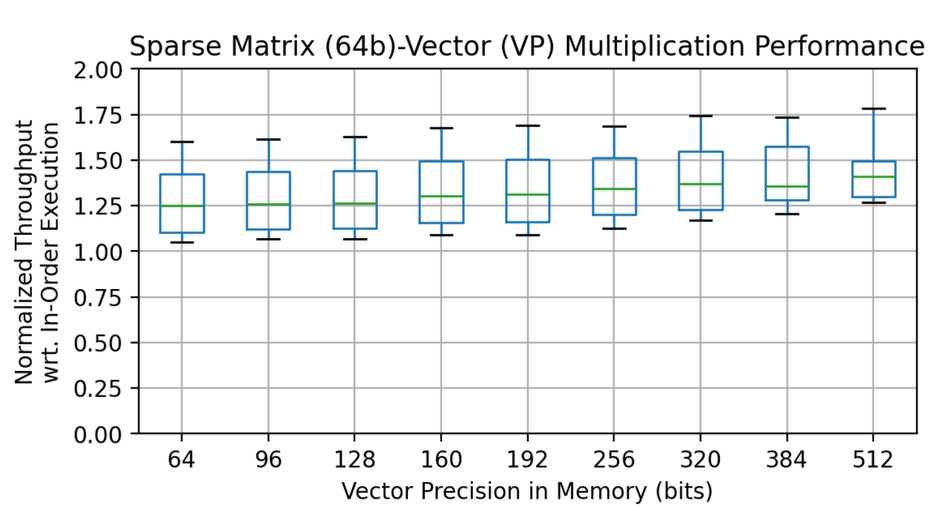

Fig 3. Sparse Matrix-Vector multiplication (SpMV) performance, measured on multiple sparse matrix structures and in function of vector precision (normalized with respect to in-order execution)

[1] E. Guthmuller, et al., "Xvpfloat: RISC-V ISA Extension for Variable Extended Precision Floating Point Computation", (2024) IEEE Transactions on Computers

E. Guthmuller[1], T. Khandelwal[1]
1 Univ. Grenoble Alpes, CEA, LIST, 38000 Grenoble

This work has been performed in the context of the EPI project. EPI has received funding from the European High Performance Computing Joint Undertaking (JU) under Framework Partnership Agreement No 800928 and Specific Grant Agreement No 101036168 (EPI SGA2).