ETH zürich

MemPool meets Systolic

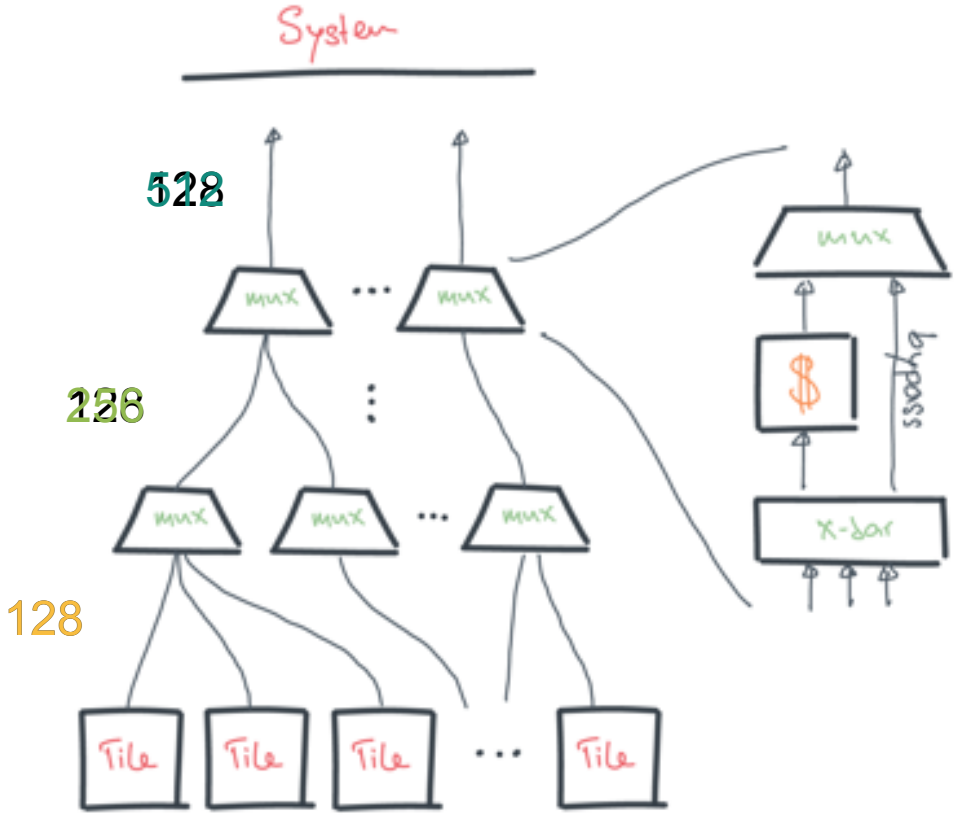
Samuel Riedel
Matheus Cavalcante
Prof. Luca Benini

MemPool Software

- Bare-metal kernels:
  - BLAS (sensible subset throughout all levels)
  - 2D convolution
  - DCT
  - Debayering
- OpenMP/Halide kernels
  - Histogram equalization
  - Image aligning (gaussian pyramid)
  - HDR pipeline (image merging)
  - NN (only one layer?)

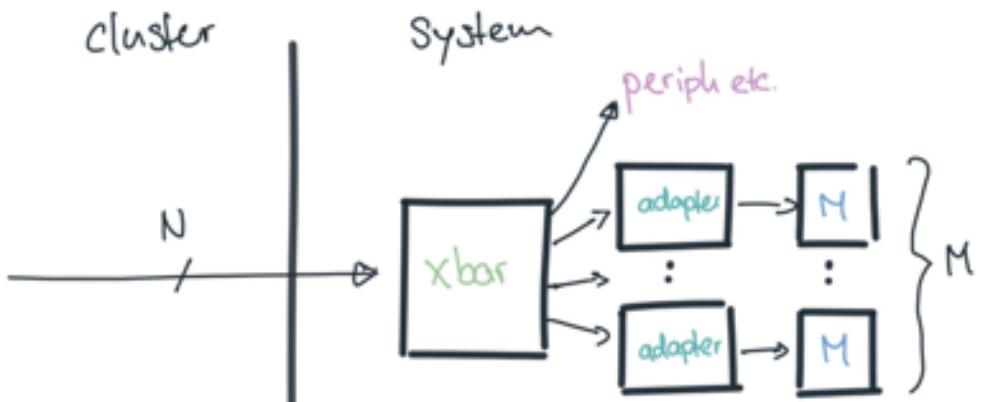
AXI Interconnect

- First version is merged to GitHub
  - Backend implementation running (complex hierarchy)
- The whole tree has a constant width
  - Reduced bandwidth at each mux
  - → Make the width a parameter at each stage
- L2 memory is currently a bottleneck
  - Implemented a multi-banked L2

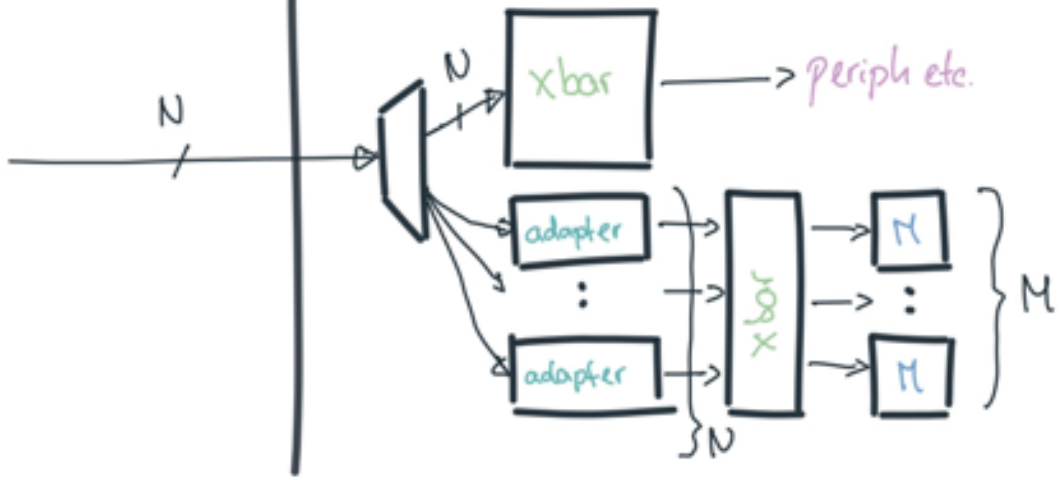


Multibanked L2 interface

- Now:
  - Single crossbar
  - Multiple separate `adapters` (axi to memory slaves)
  - Memories can easily be spread out
  - Hard to take advantage of in software

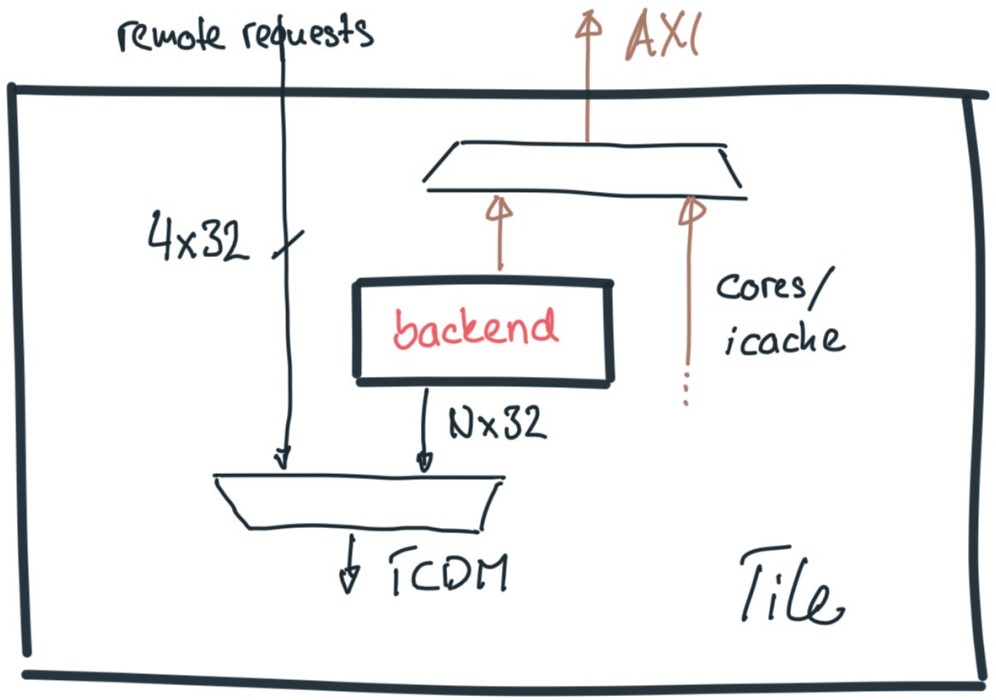


- Next:
  - Dedicated adapter for each group/axi channel
  - Crossbar after adapter allows interleaved memory
  - Distributing memory?
  - Easy to have parallel accesses



DMA

- MemPool is a single cluster: We only want one DMA
  - But: access to all tiles without adding another interconnect
- Idea:
  - Have one DMA frontend
  - Have one DMA backend **per tile**
- Each backend is responsible for its memory region
  - 64 backends will choke at the L2
  - Max TCDM ports might also be limiting at full AXI bandwidth
  - But we don’t need both interfaces to work at full speed
  - Can we reduce the number of backends?



DMA

- Have one DMA backend **per N tiles**
  - Similar interfaces
  - Balance the bandwidths