Spatzformer: Reconfigurable Dual-Core RVV Cluster for Mixed Scalar-Vector Workloads

Matteo Perotti[1], Michele Räber[1], Mattia Sinigaglia[2], Matheus Cavalcante[1], Davide Rossi[2], Luca Benini[1,2]
[1]ETH Zurich, Switzerland; [2]University of Bologna, Italy

The first two authors contributed equally to this work

1 Introduction

Hard to achieve high utilization on mixed scalar-vector workloads

The scalar sequential routines under-utilize the resources of flexible multi-core vector architectures targeting parallel workloads

RISC-V is an **open-source ISA** for general-purpose processors.

Its **vector extension** V helps speed up vector operations, especially on parallel regular workloads.

The **open-source** RISC-V V **dual-core Spatz** cluster[1] can flexibly **accelerate parallel computation** and execute **multiple tasks** at once.

How to **improve** its **performance** on **mixed scalar-vector** workloads?

2 Contributions

Spatzformer – Reconfigurable RISC-V V multi-core cluster

- Boost scalar-vector workloads performance with negligible area cost

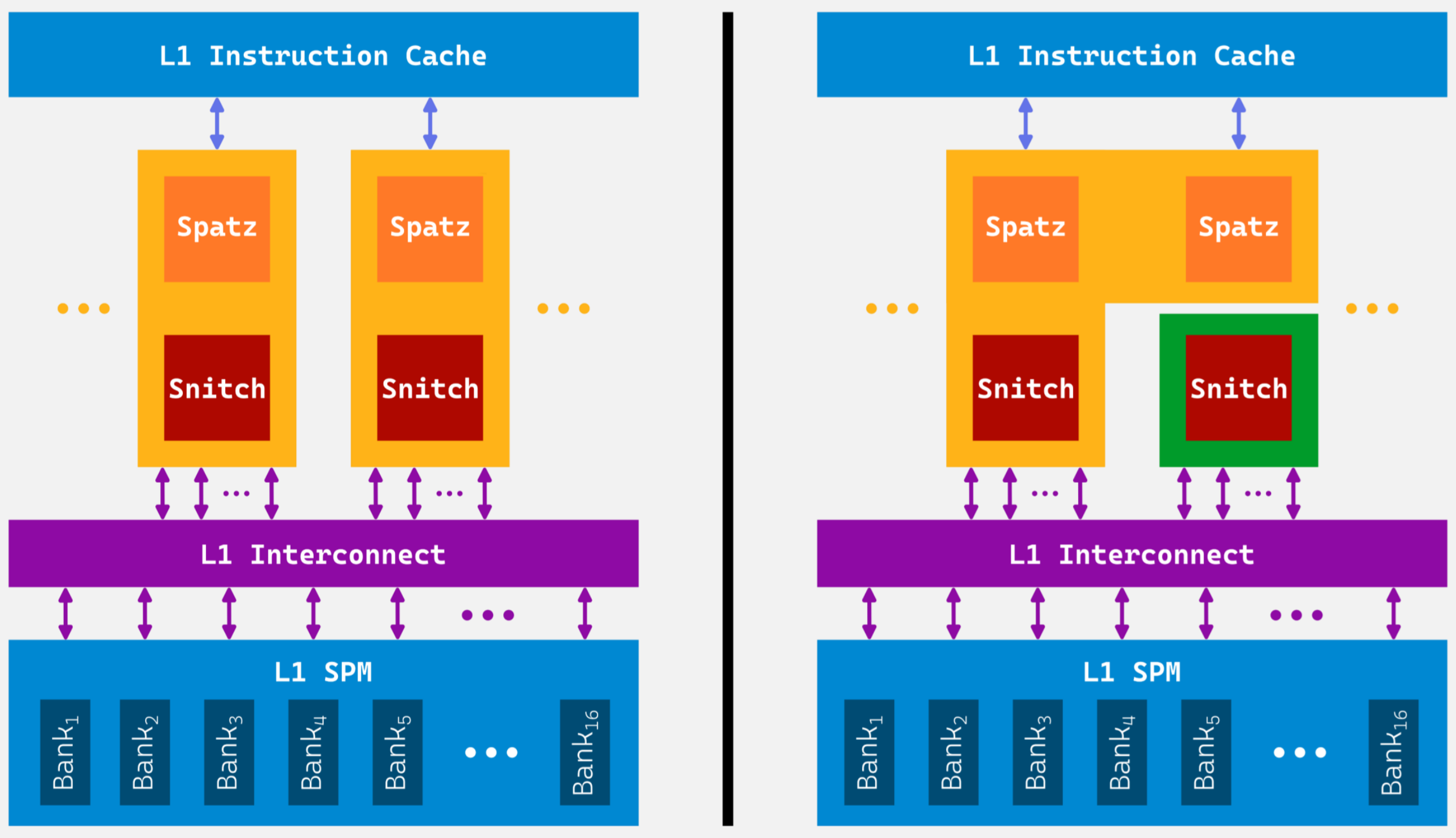
PPA analysis of the cost of the reconfigurability feature

- Implement Spatzformer vector architecture in 12-nm technology

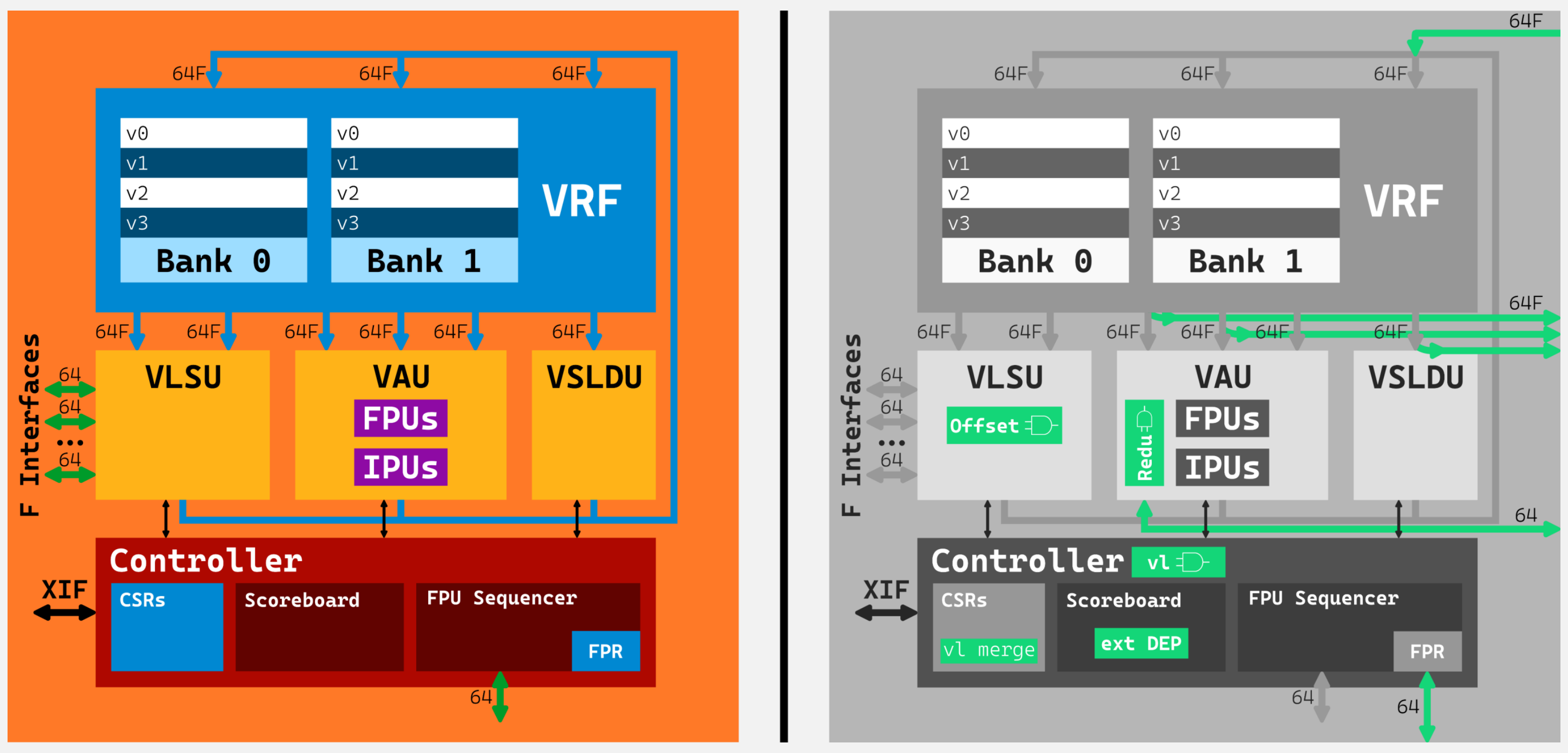
3 Implementation

Dual-core Spatzformer can be reconfigured at runtime in one of two modes

- **Split mode** – **each Snitch** scalar core **controls a Spatz** vector accelerator. The architecture works as a vector dual-core cluster
- **Merge mode** – **one Snitch controls both Spatz** vector accelerators. The **remaining Snitch** can **execute scalar tasks** independently



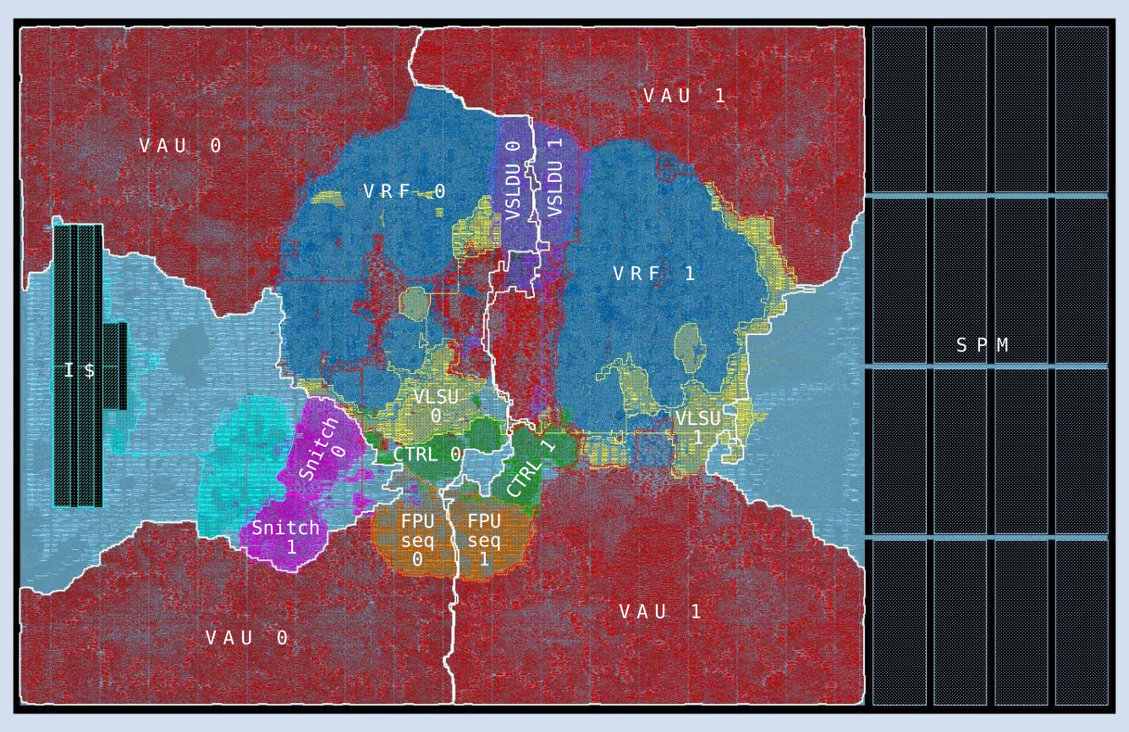
Hardware **modifications** to **Spatz** to **support reconfigurability**



4 Results and Discussion

Experiment Setup

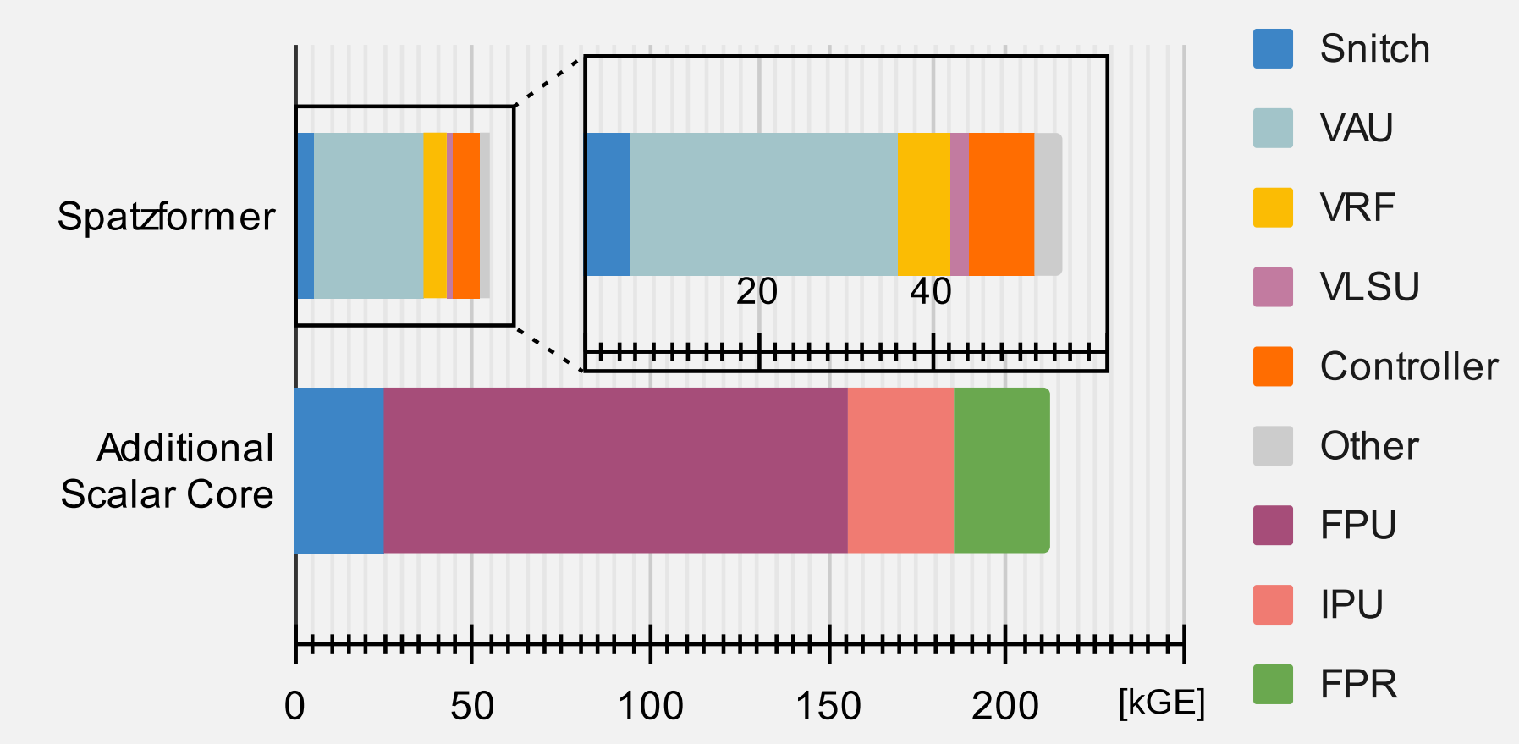
- Physical implementation of Spatzformer targeting 12-nm technology
- Comparison against Spatz cluster
- Simulation of multiple kernels for performance and energy-efficiency evaluation
- Performance evaluation of mixed vector (various kernels) and scalar (coremark) workloads



Minimal area cost

The **reconfigurability** feature incurs in **only +1.4% area** overhead.

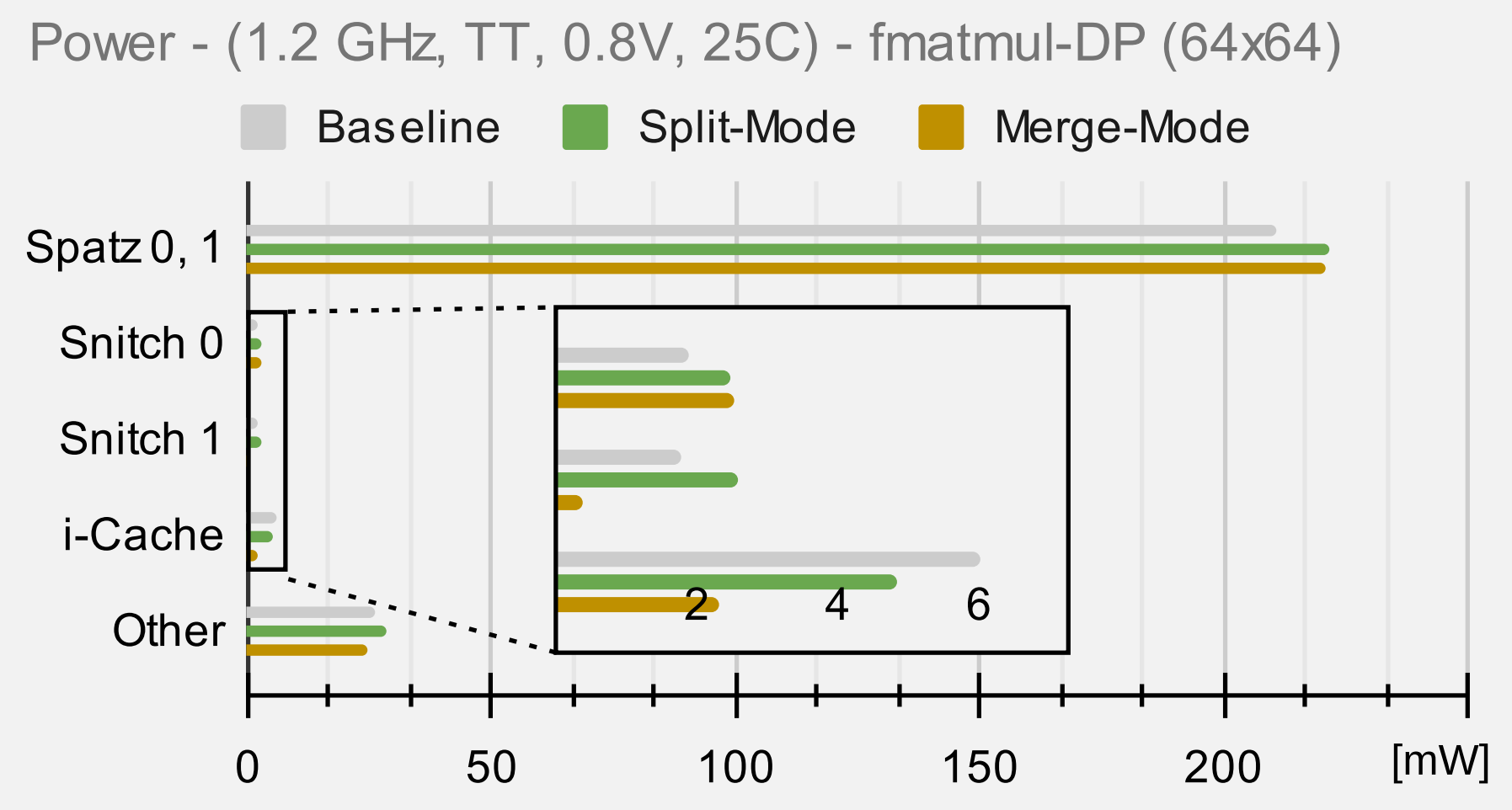
Instead, adding a dedicated additional scalar core would have costed 4x more area.



Lower scalar core power

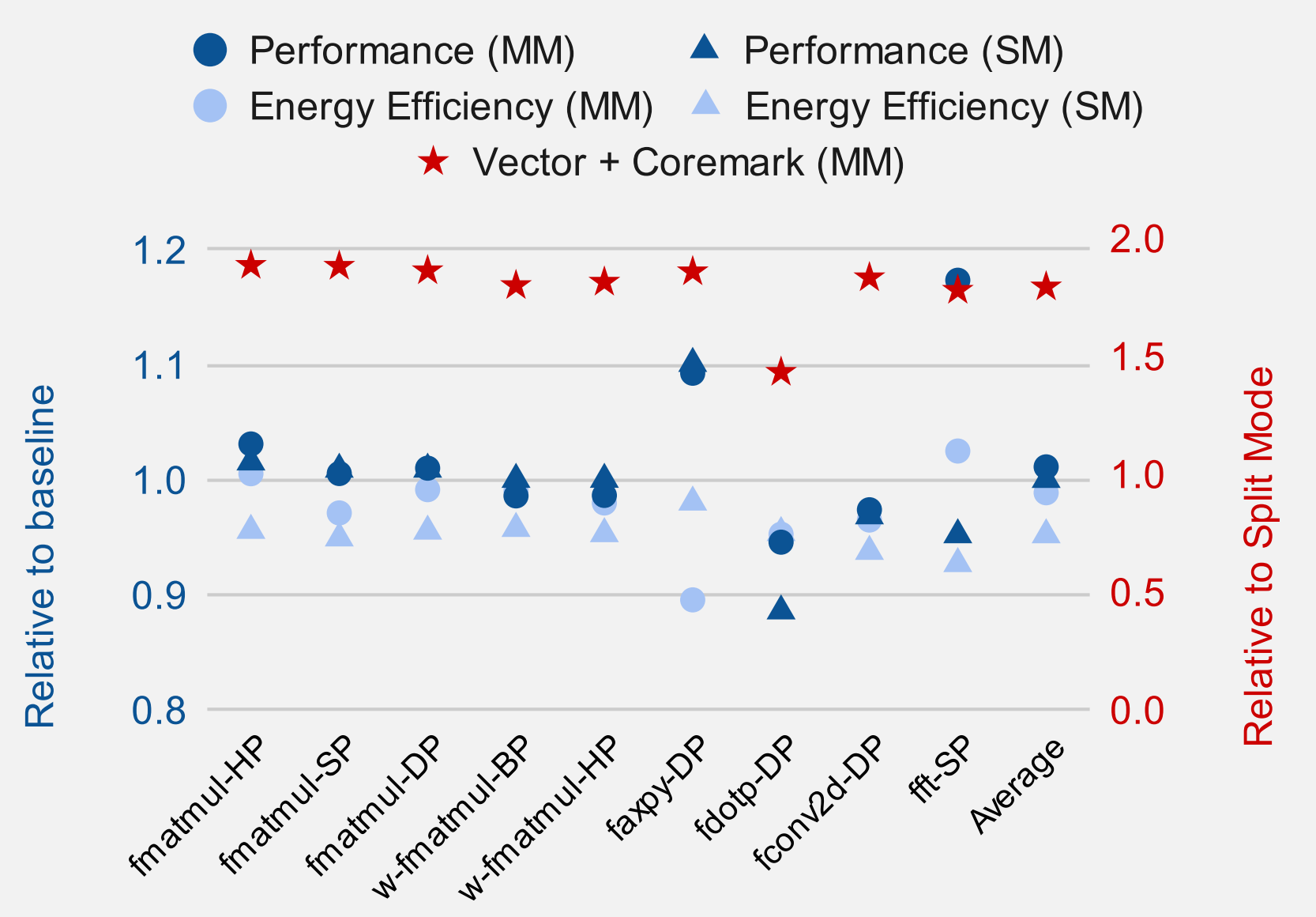
Spatzformer's merge mode power reduction:

- Independent scalar core
- i-Cache – with fewer instructions fetched thanks to longer vectors



Performance, Efficiency

- No maximum frequency degradation
- MM – **faster FFT** (avoid software sync)
- MM – **1.8x** average **speed-up** on **mixed vector-scalar** workloads



5 Conclusion

Spatzformer - Reconfigurable RISC-V V architecture

- **Change configuration** at **runtime** (**split** or **merge mode**)
- **Accelerate mixed** scalar-vector **workloads** by **1.8x**
- Speed up sw-synchronized kernels (**FFT**) by up to **20%**
- **No frequency drop** and **negligible area** (+1.4%) and **efficiency** (-5%) **cost**

[1] M. Cavalcante, M. Perotti, S. Riedel, and L. Benini, "Spatz: Clustering compact RISC-V-based vector units to maximize computing efficiency"