### Motivation

- Vector Processing Unit (VPU) - reduced instruction memory, reduced address translations and deeper pipelines - high compute performance

#### Challenges:
- Very long vectors? - limitation of register space on Vector Register Files (VRF) - energy constraints

### VecProM: Vector Processor, under MEEP Project

- Accelerated Compute and Memory Engine (ACME) - MEEP[1] project at BSC
- ACME accelerator core consists:
  - A RISC-V scalar core
  - A RISC-V VPU (VecProM)
  - Two systolic arrays
- VecProM is a modified version of a baseline RISC-V VPU [2] - support for very long vectors and sparse vectors

- Disaggregation of memory and arithmetic instruction execution - improvement in performance and energy savings
- A specialized high-bandwidth memory path - bypasses the cache hierarchy and connects VecProM to a scratchpad memory, termed **Long Vector Register File (LVRF)**
- LVRF - acts as virtual VRF for the physical VRF - maintains 32 vector registers as per the RISC-V ISA specifications.
- Loads and stores on LVRF - **Micro-engines (MEs)** - on-the-fly configuration

### VecProM operation

- Two operand-dependent modes: **Mode_SD**
  - Suitable for short dense vectors, i.e., VL fits on the VRF
  - Instructions, operands and results over OVI

### Micro-engines and Mode_LDSp

- Suitable for very long dense vectors and sparse vector operands, which do not benefit much from cache hierarchy
- Instructions over OVI, operand and results stored in LVRF, accessed through Micro-engines
- Hardware strip-mining
- MPCU [1] gathers non-zero elements of sparse data - packed as dense data - stored in LVRF

### References


---

**A RISC-V VPU for Very Long and Sparse Vectors**

Gopinath Mahale, Karim Charfi, Tejas Limbasiya, Teresa Cervero, John Davis
(gopinath.mahale, karim.charfi, tejas.limbasiya, teresa.cervero, john.davis)@bsc.es
Barcelona Supercomputing Center (BSC), Barcelona, Spain