ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

NextRAN-AI

ETH- Huawei Sweden

Marco Bertuletti	mbertuletti@iis.ee.ethz.ch
Yichao Zhang	yiczhang@iis.ee.ethz.ch
Mahdi Abdollahpour	mahdi.abdollahpout@unibo.it
Alessandro Vanelli-Coralli	avanelli@iis.ee.ethz.ch
Luca Benini	lbenini@iis.ee.ethz.ch

PULP Platform
Open Source Hardware, the way it should be!

@pulp_platform
pulp-platform.org
youtube.com/pulp_platform

We choosed to explore NeuralRX

Advantages of NeuralRX over other models

- **Flexible** = the same trained model supports different number of users, different number of subcarriers, different modulation schemes
- It generalizes well to many different channel models
- It is open-sourced and tested already on a real-time and standard compliant scenario (NeuralRX RT)

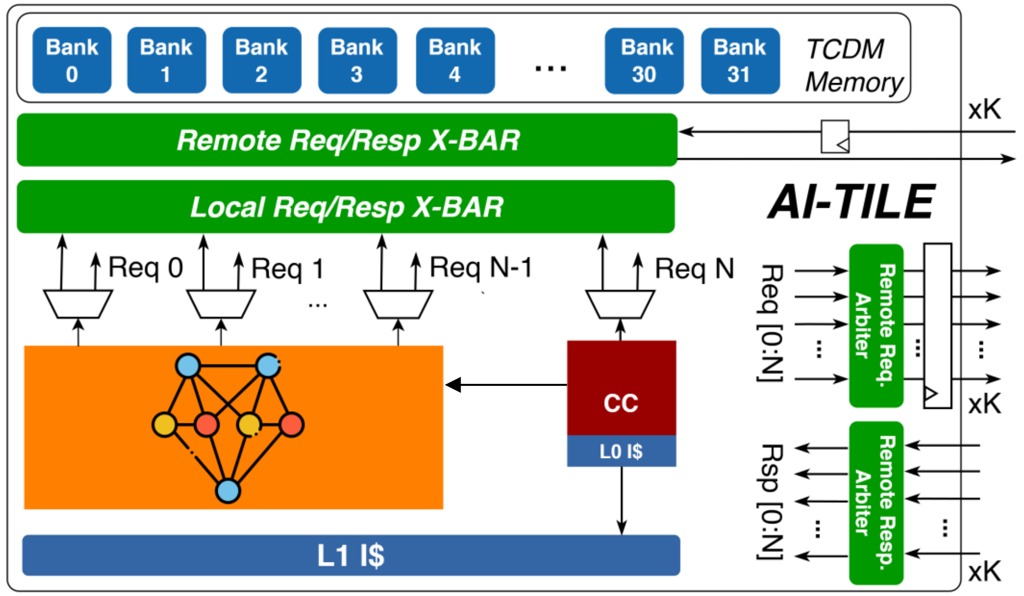
Next Steps:

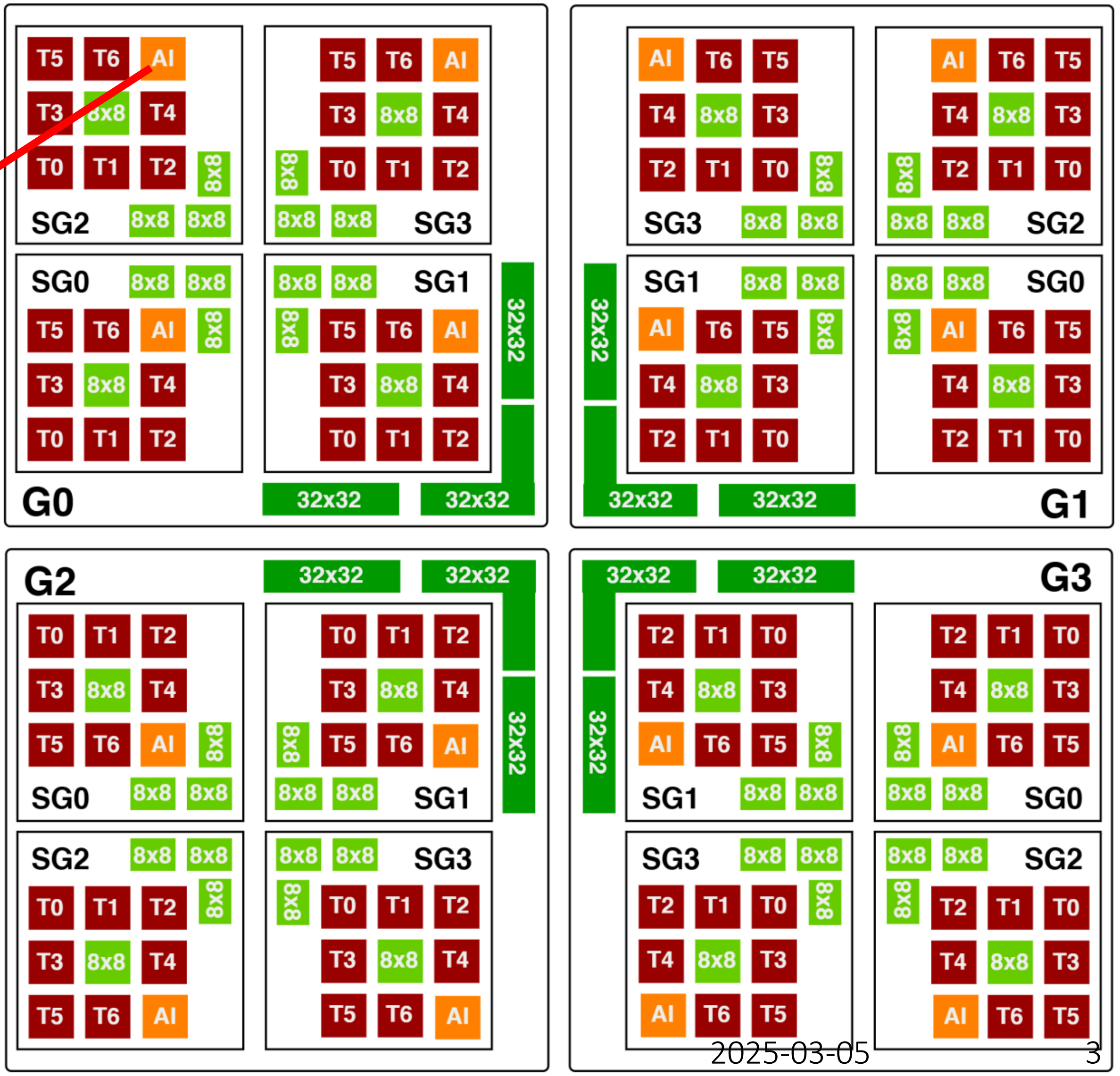
- Reduce model size and computational complexity for edge-deployment
- Possibly extend to more subcarriers, transceivers
- **Adequate TeraPool's computation per cycle**

We will present these results on next meeting

NextRAN-AI will increase TeraPool's compute capabilities

- Add AI-specialization (more compute), keeping configurability

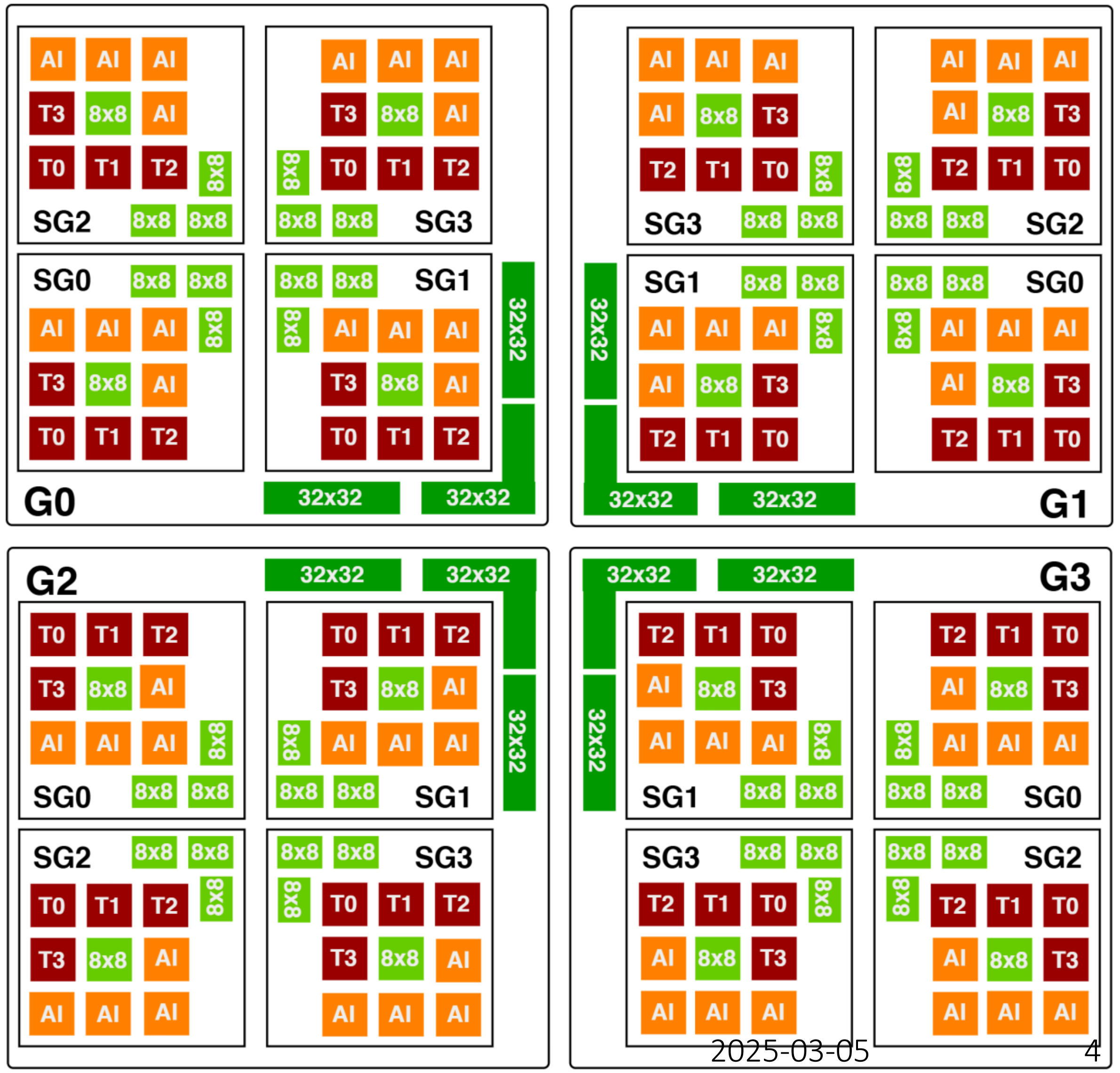




NextRAN-AI will increase TeraPool's compute capabilities

- Add AI-specialization (more compute), keeping configurability

How many accelerators?

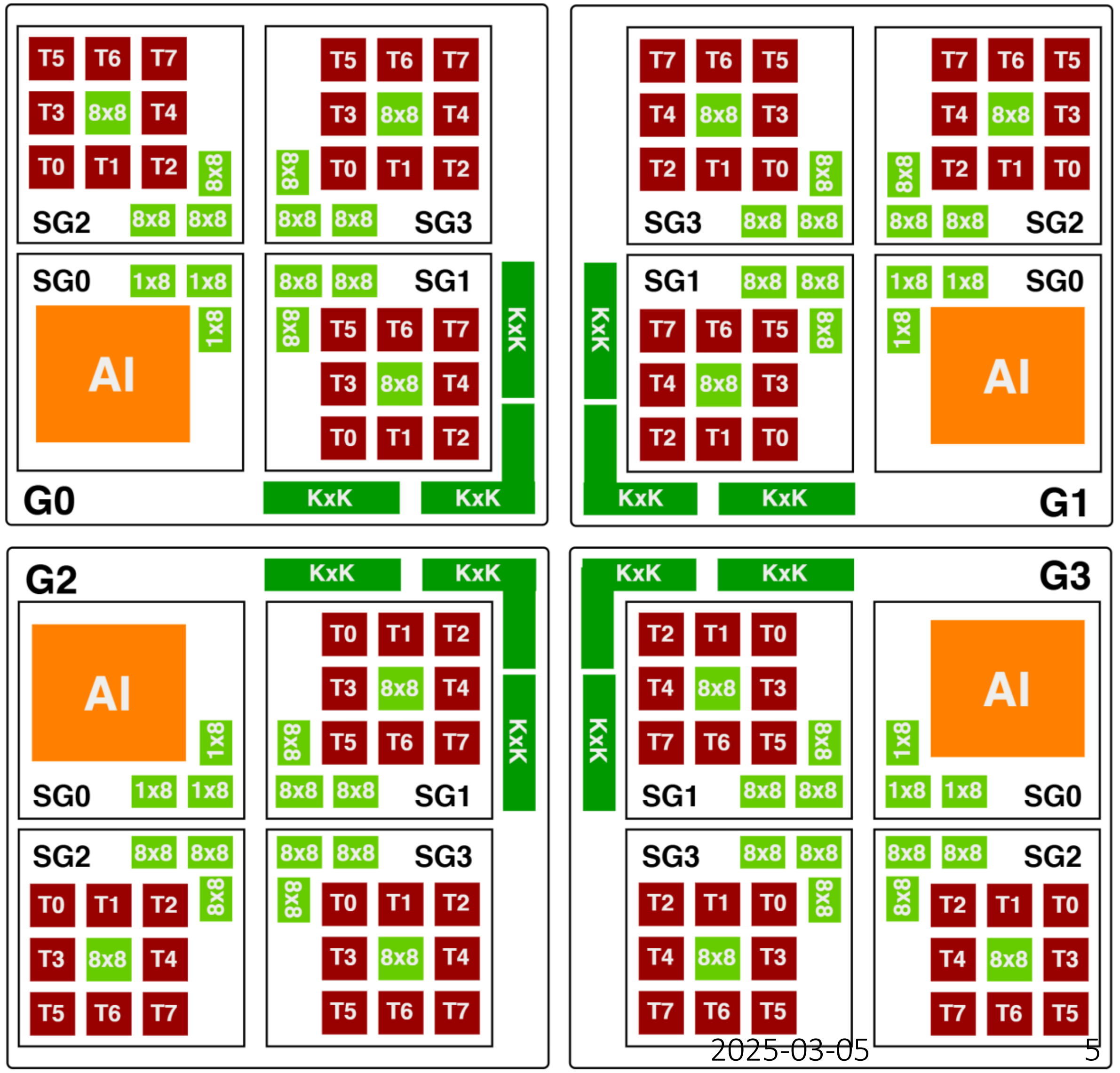


NextRAN-AI will increase TeraPool's compute capabilities

- Add AI-specialization (more compute), keeping configurability

How many accelerators?

How big?



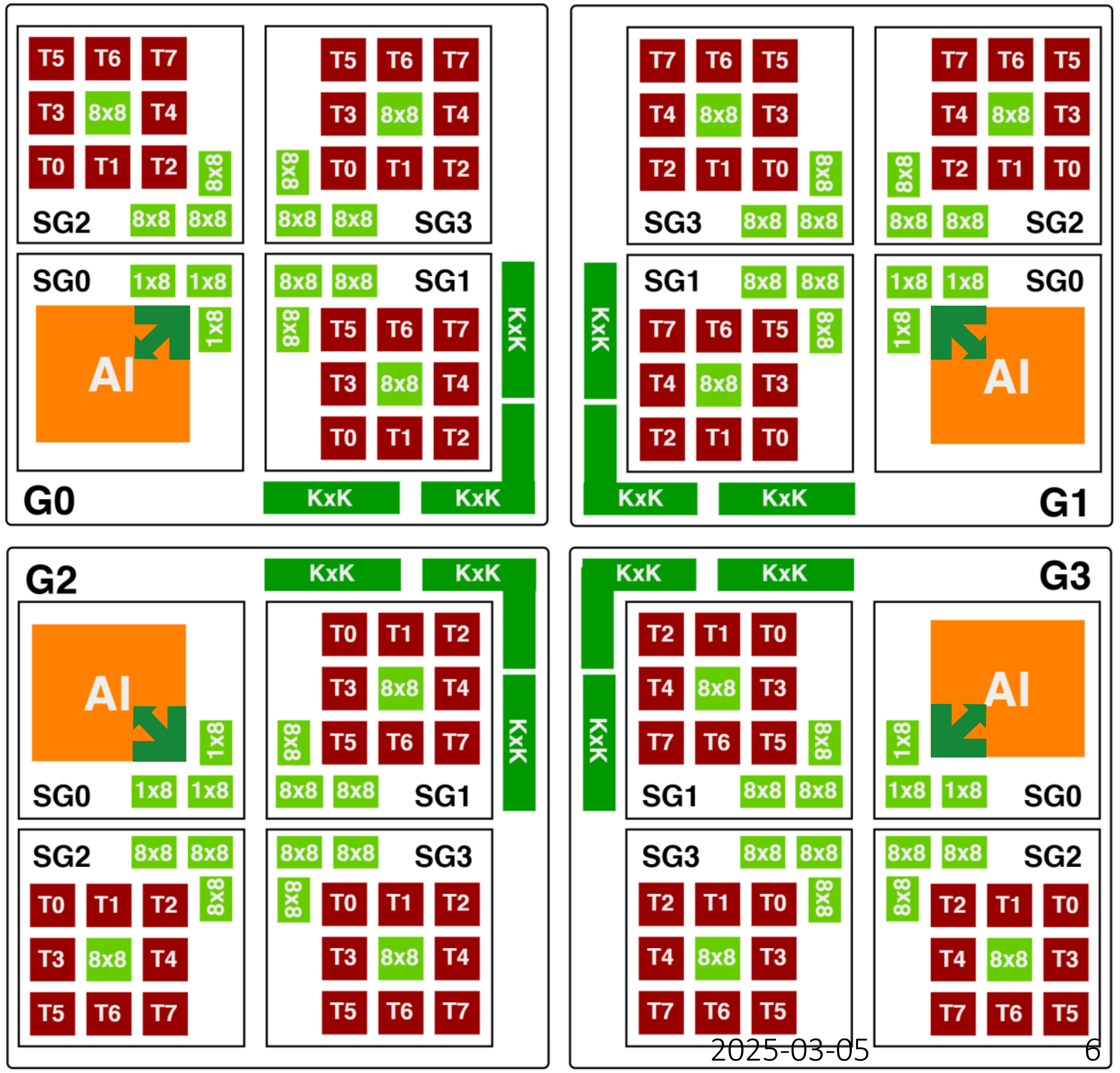
NextRAN-AI will increase TeraPool's compute capabilities

- Add AI-specialization (more compute), keeping configurability

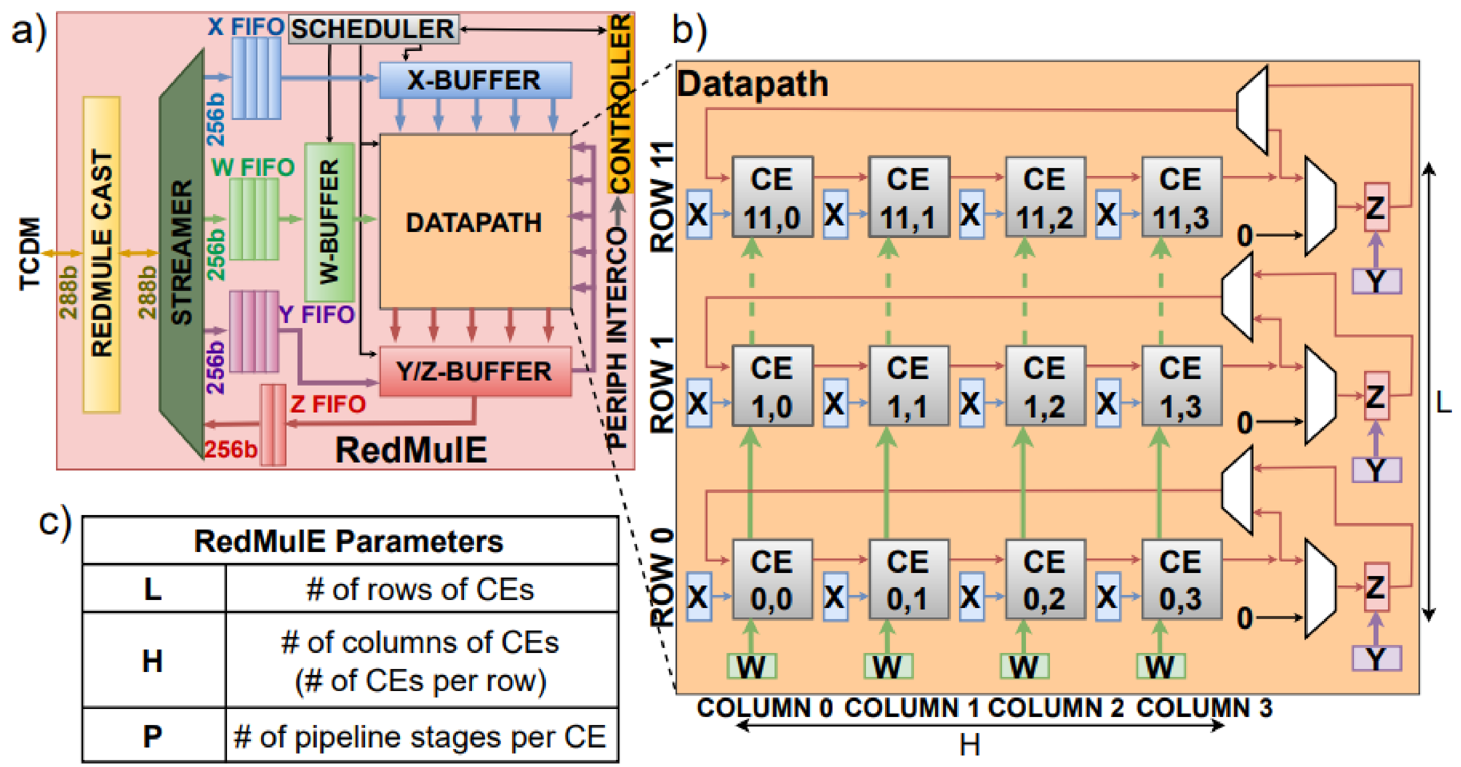
How many accelerators?

How big?

How connected?



RedMulE - GEMM Accelerator



c)

RedMulE Parameters	
L	# of rows of CEs
H	# of columns of CEs (# of CEs per row)
P	# of pipeline stages per CE

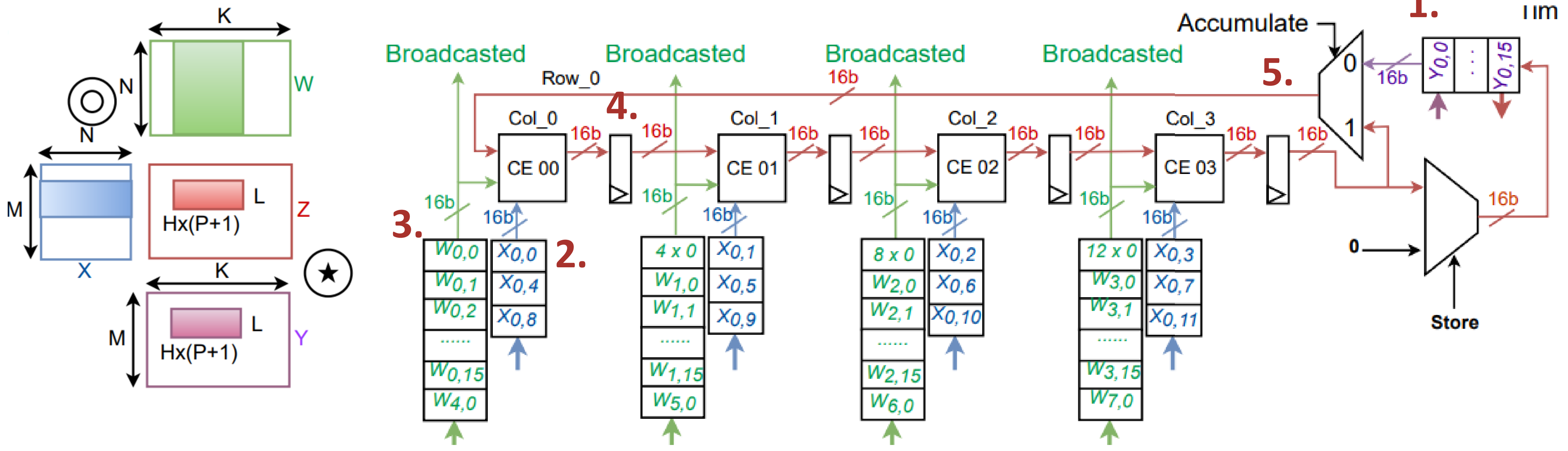
- **General Purpose →**
  - Attention/1D-Conv
  - Hermitian calculation
  - Beamforming
- **Open-Source**
- **Parametrizable**
- **TCDM-compatible**
- **Programmed via register interface**

github.com/pulp-platform/snitch

ETH zürich ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

RedMulE – Operation Z = Y ⍟ (X ⊙ W)

1. Preload the Z buffer with **L** rows (**H*(P+1)***16b elements each) from Y-matrix
2. Preload the X buffer with **L** rows (**H*(P+1)***16b elements each) from X-matrix
3. Load **H*(P+1)** 16b weights from W-matrix, broadcast to all CE on a column
4. After **(P+1)** cycles pass the result to next CE, load other H*(P+1) 16b weights
5. After **H*(P+1)** cycles **feedback**

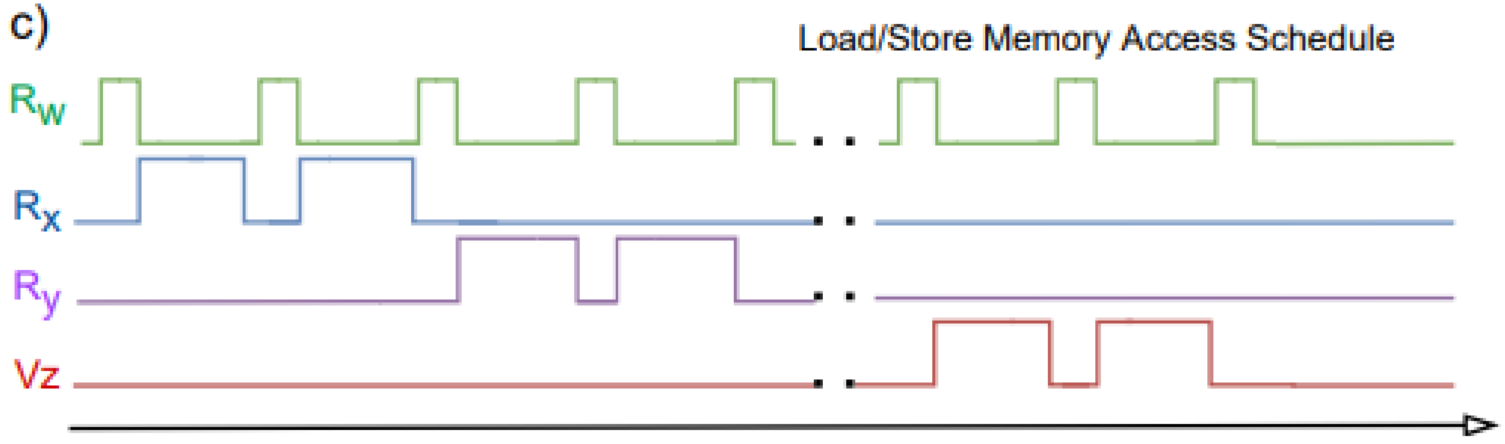


Designed for 1-cycle memory latency

- RedMulE has a 16b*H*(P+1) memory port to TCDM
- The memory requests are split by the TCDM protocol DataWdith (32b)
- If the memory has one cycle latency → Needs to be handled for TeraPool

Memory requests to different buffers are interleaved to access to 16b*H*(P+1) per cycle

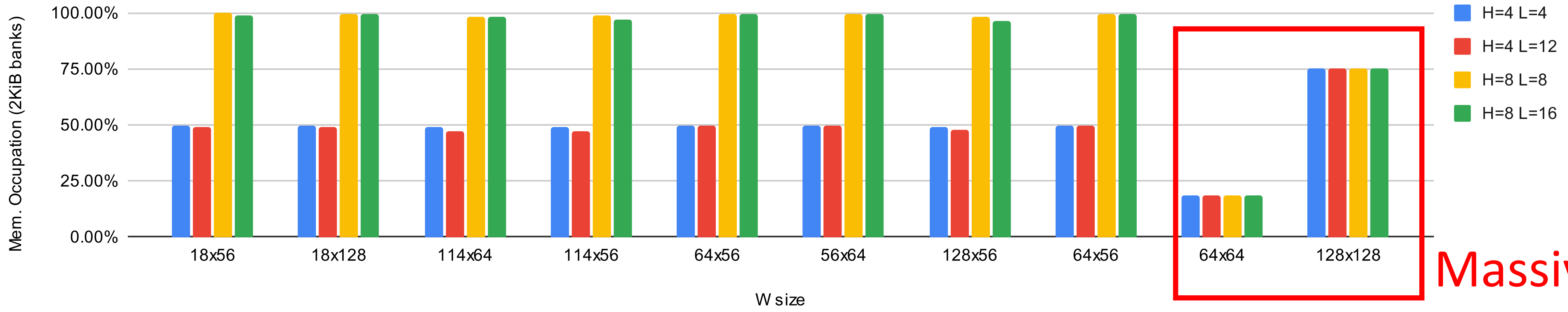
c)



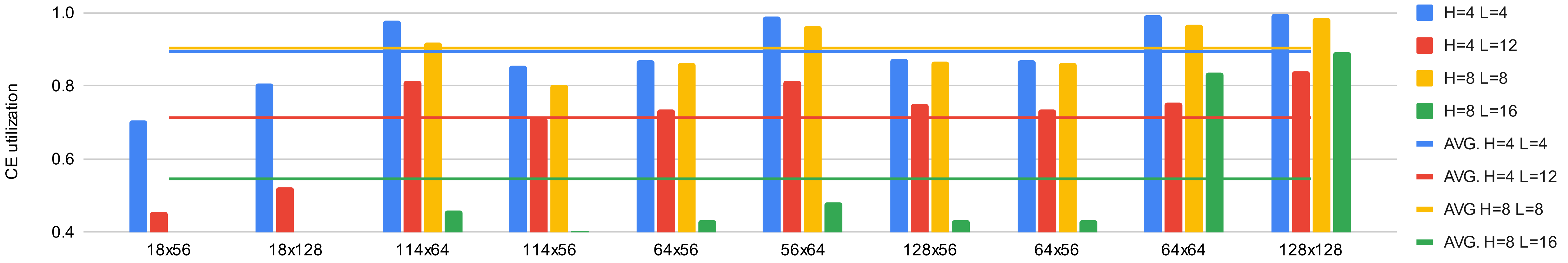
Parametrization – How big accelerator?

1. **Assumption:** 1-cycle access to memory
2. **Design-Choice:** 1 RedMulE per Tile
3. **Assumption:** 2KiB-bank, 32-banks/Tile (memory occupation 50% or 100%)

Double-buffering

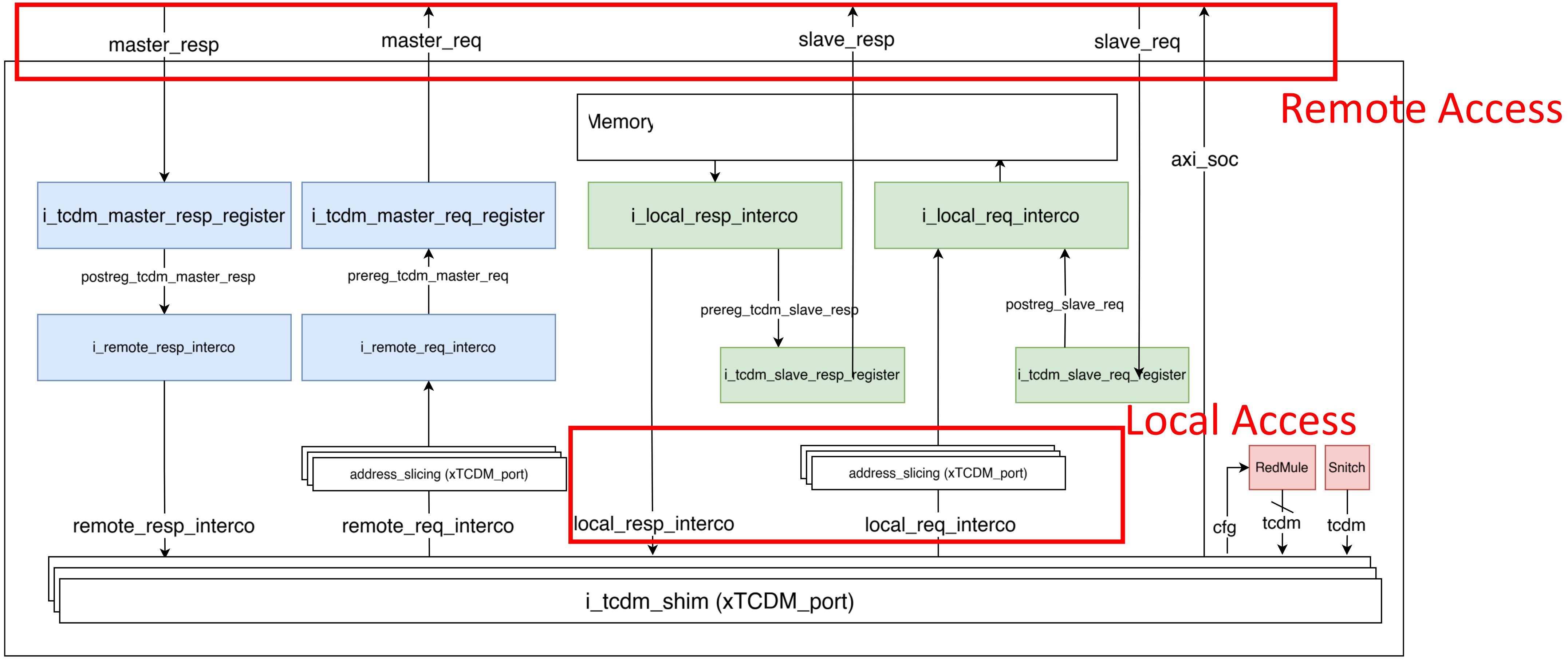


Massive MIMO



ETH zürich
ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Integration of RedMulE in the Tile

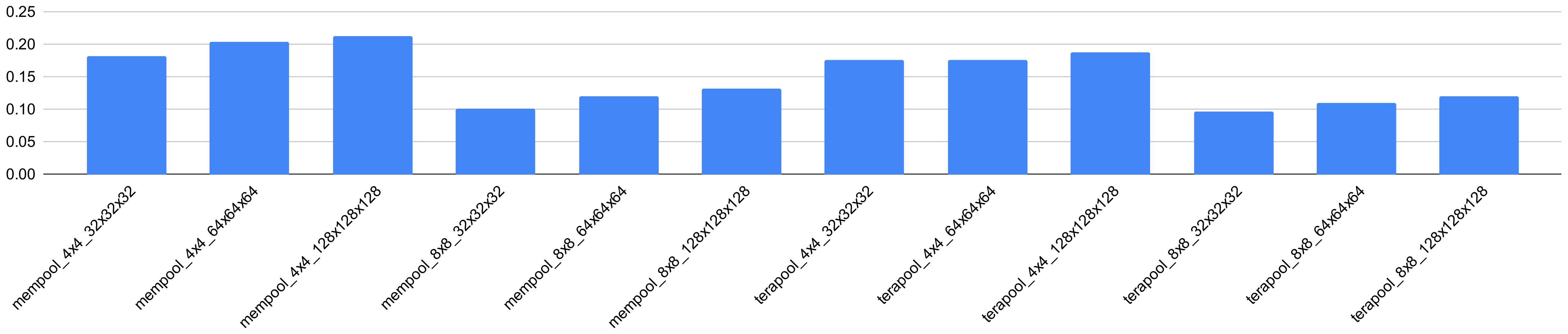


- RedMulE programmed by the Snitch core
- Parametrizable RedMulE accesses TCDM (32b at a time through remote ports)

Next steps

Low utilization of the CE because of NUMA latency

- Add functionalities to restrict weights/inputs allocation to the **local memory** (inside a Tile we have 1-cycle access)
- For **remote access** improve latency on large requests (e.g. bursts)


mempool_4x4_32x32x32
mempool_4x4_64x64x64
mempool_4x4_128x128x128
mempool_8x8_32x32x32
mempool_8x8_64x64x64
mempool_8x8_128x128x128
terapool_4x4_32x32x32
terapool_4x4_64x64x64
terapool_4x4_128x128x128
terapool_8x8_32x32x32
terapool_8x8_64x64x64
terapool_8x8_128x128x128