DATE 25
DESIGN, AUTOMATION AND TEST IN EUROPE
THE EUROPEAN EVENT FOR ELECTRONIC SYSTEM DESIGN & TEST
31 MARCH – 2 APRIL 2025
LYON, FRANCE
CENTRE DE CONGRÈS DE LYON

TCDM Burst Access: Breaking the Bandwidth Barrier in Shared-L1 RVV Clusters Beyond 1000 FPUs

Diyou Shen[1*], Yichao Zhang[1*], Marco Bertuletti[*], Luca Benini [* §]

[*]ETH Zurich, Switzerland *[§]University of Bologna, Italy*

[1] These two authors contributes equally to the paper

ETH zürich

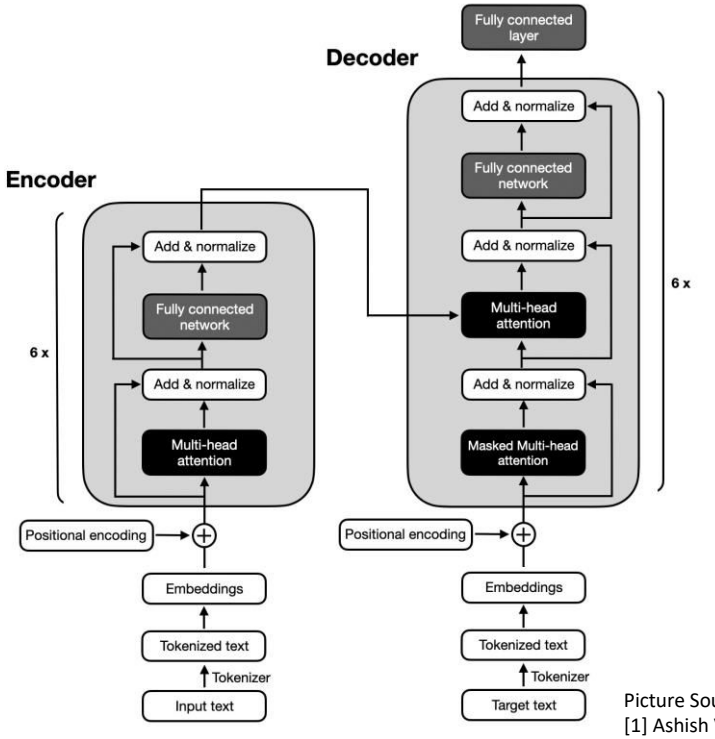
CORENEXT

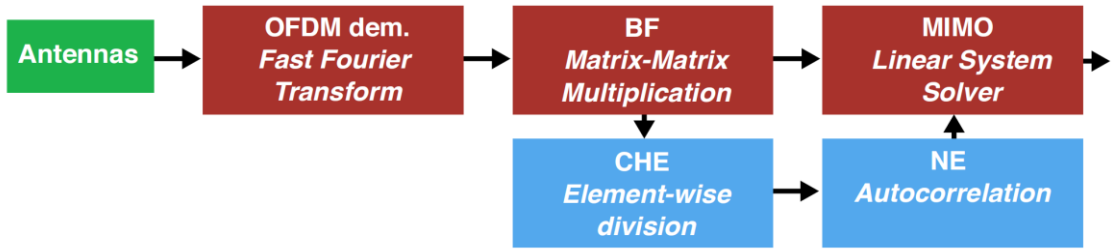
Many-Core Cluster is Rocking

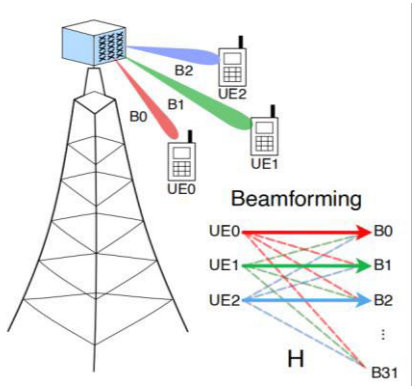
- **Emerging fields need more computation**
  - Attention-Based AI Models
  - B5G Communication

More Cores

More Data Movements







Picture Sources:
[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, & Illia Polosukhin. (2023). Attention Is All You Need.
[2] Marco Bertuletti, Yichao Zhang, Alessandro Vanelli-Coralli, & Luca Benini. (2022). Efficient Parallelization of 5G-PUSCH on a Scalable RISC-V Many-core Processor.

Many-Core Cluster is Rocking

- **Emerging fields need more computation**
  - Attention-Based AI Models
  - B5G Communication

→ More Cores
More Data Movements

- **How to keep efficient?**
  - Reduce number of instructions
  - Reduce data duplication & memory transfers

→ SIMD/SIMT
Shared L1 Cluster

Many-Core Cluster is Rocking

- **Emerging fields need more computation**
  - Attention-Based AI Models
  - B5G Communication

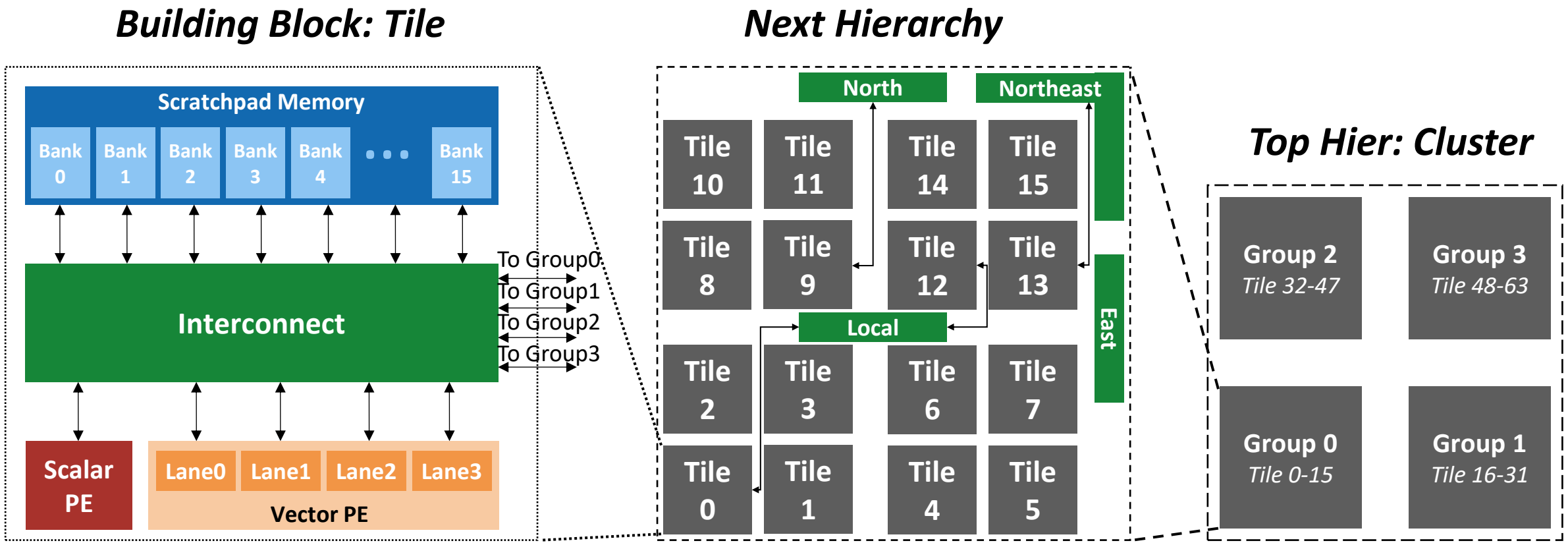
→ More Cores
More Data Movements

- **How to keep efficient?**
  - Reduce number of instructions
  - Reduce data duplication & memory transfers

→ SIMD/SIMT
Shared L1 Cluster

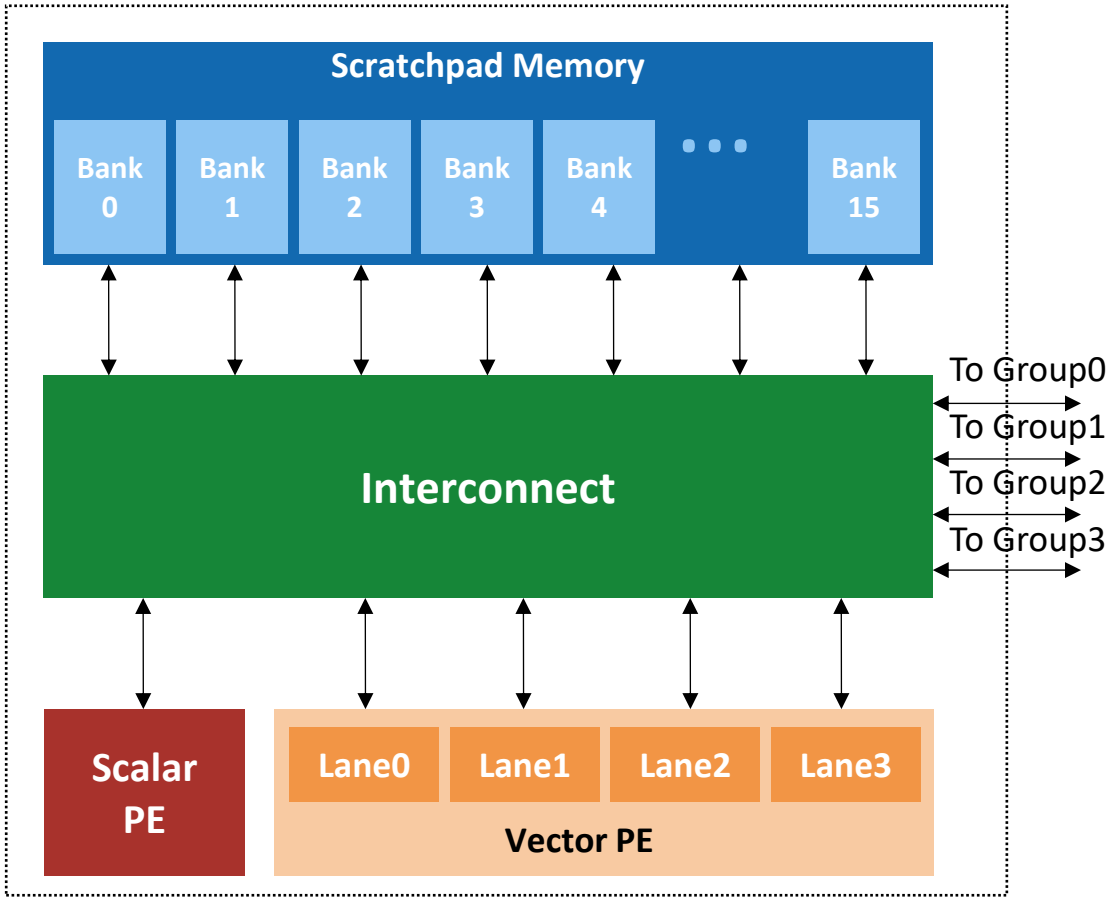
Many-Core Vector Shared-L1 Cluster

Hierarchical design is needed to route over **hundreds** cores



And what, Gul'Dan, must we give it return?

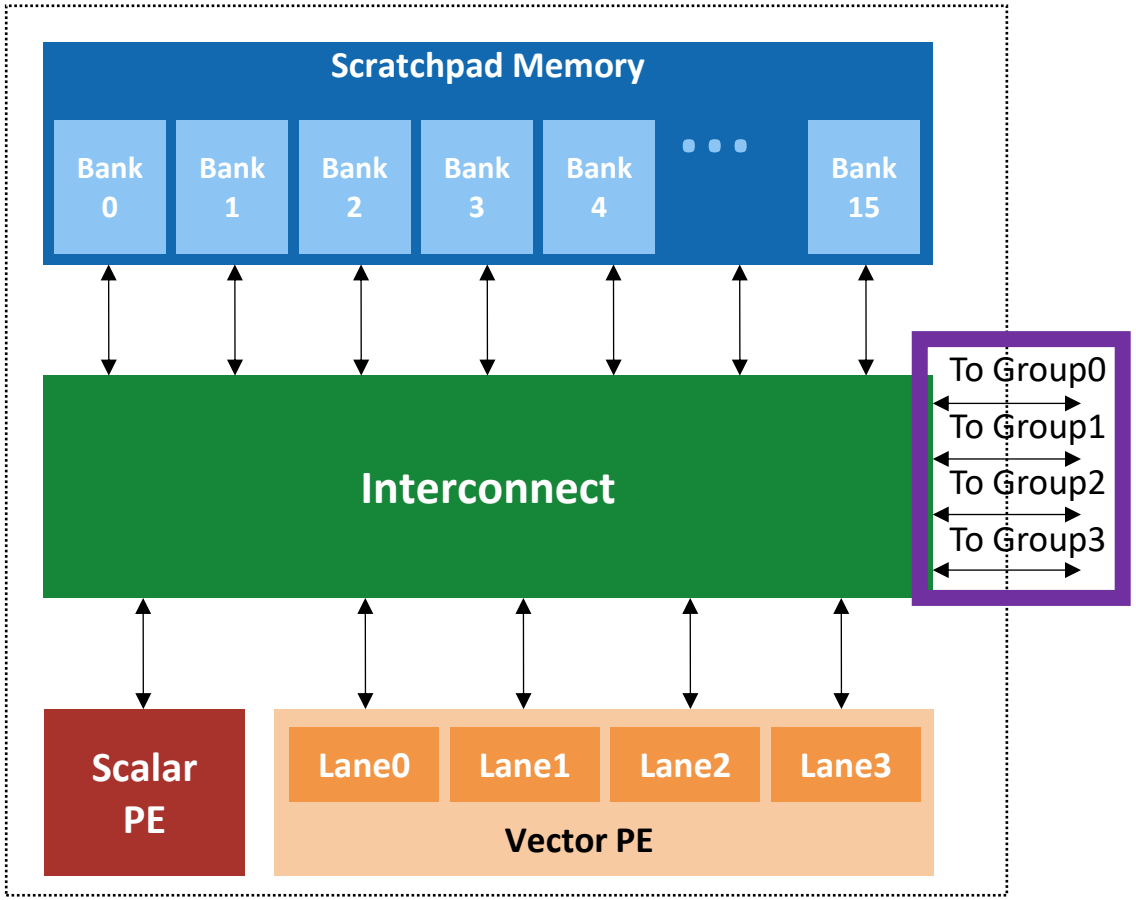
Building Block: Tile



- **Limited** routing resources for connecting hundreds of cores

And what, Gul'Dan, must we give it return?

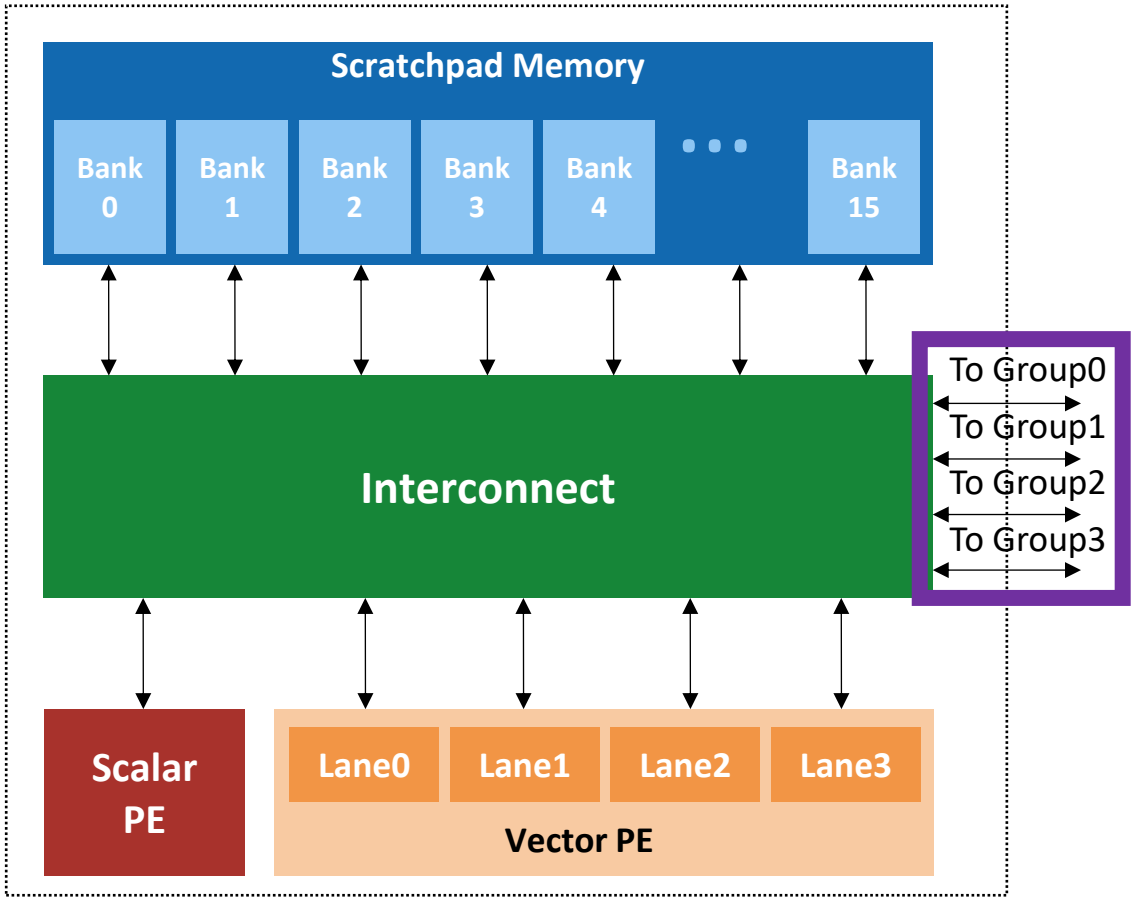
Building Block: Tile



- **Limited** routing resources for connecting hundreds of cores
- **Hierarchical** interco. designs
  - Physically feasible
  - Maintain low latency
  - But, limit the BW due to arbitration

And what, Gul'Dan, must we give it return?

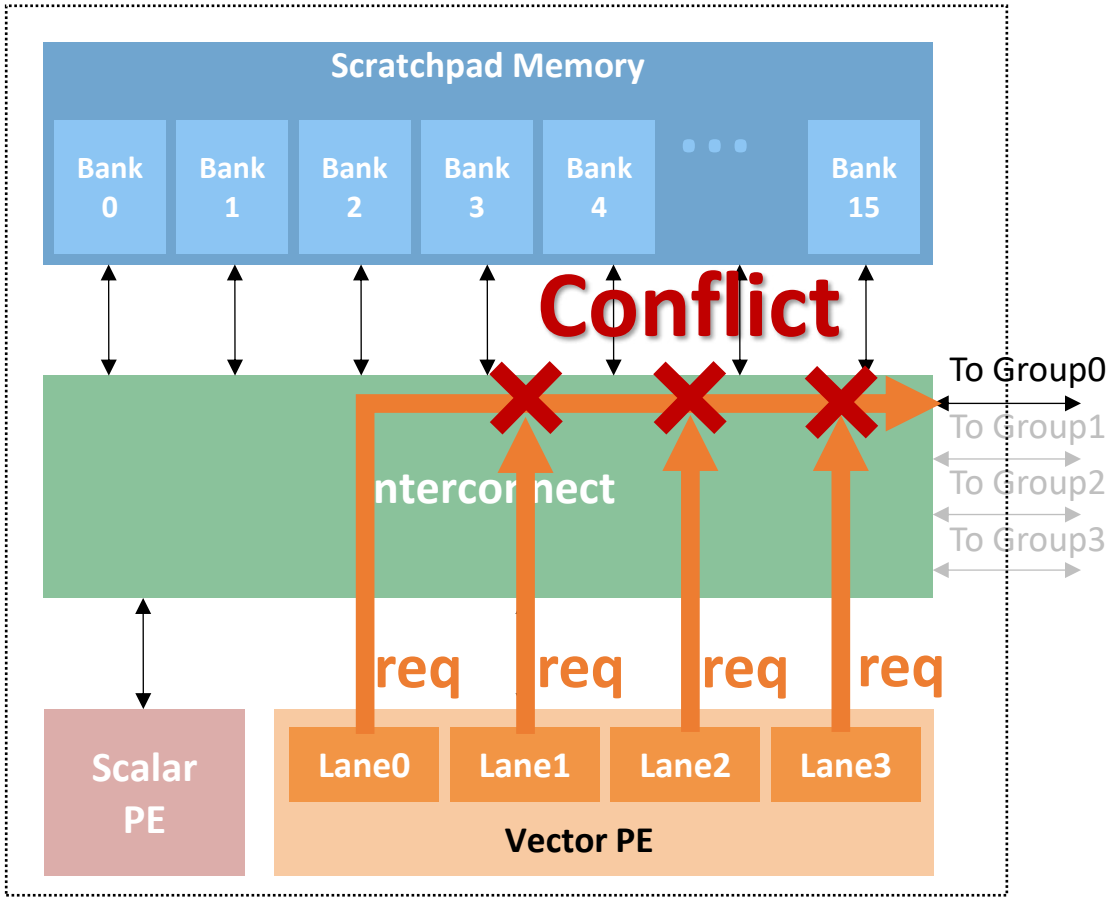
Building Block: Tile



- **Limited** routing resources for connecting hundreds of cores
- **Hierarchical** interco. designs
  - Physically feasible
  - Maintain low latency
  - But, limit the BW due to arbitration
  - More pronounced in Vector PEs due to **unit-stride** visiting patterns

And what, Gul'Dan, must we give it return?

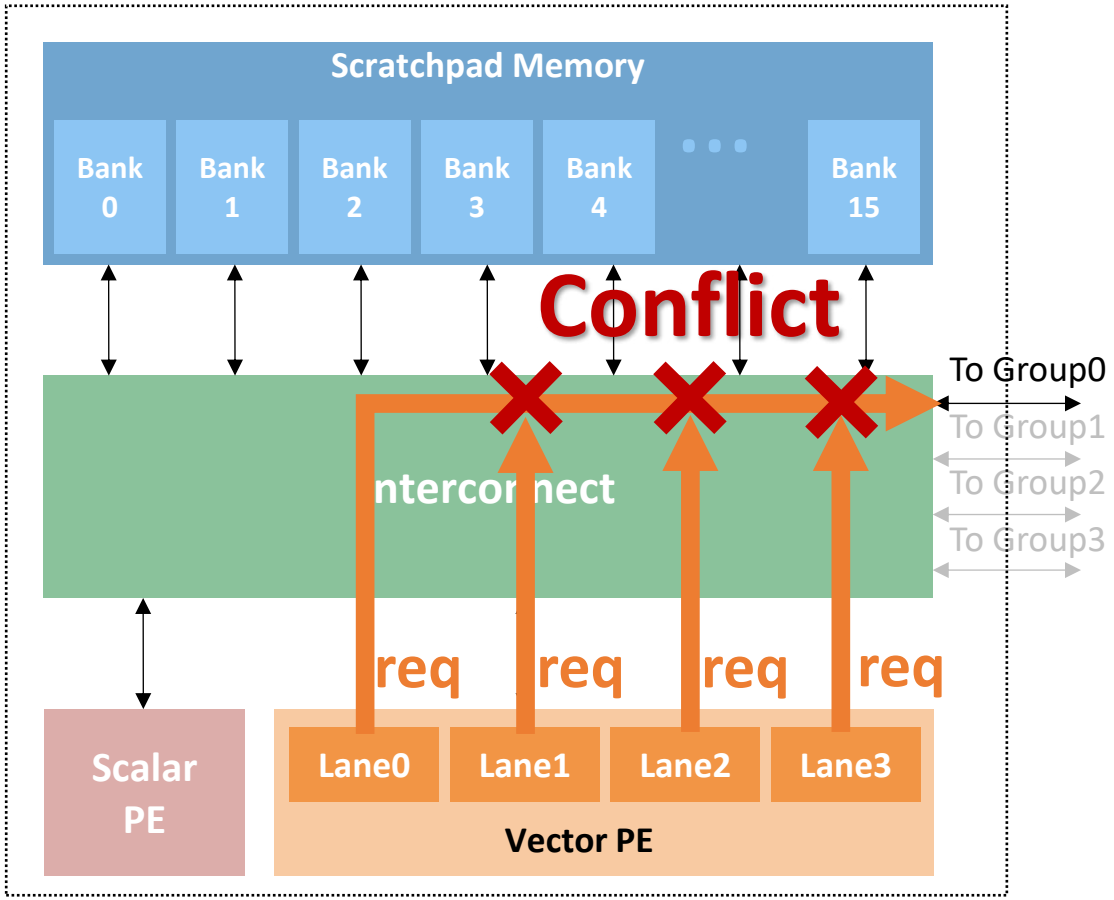
Building Block: Tile



- **Limited** routing resources for connecting hundreds of cores
- **Hierarchical** interco. designs
  - Physically feasible
  - Maintain low latency
  - But, limit the BW due to arbitration
  - More pronounced in Vector PEs due to **unit-stride** visiting patterns

And what, Gul'Dan, must we give it return?

Building Block: Tile



- **Limited** routing resources for connecting hundreds of cores
- **Hierarchical** interco. designs
  - Physically feasible
  - Maintain low latency
  - But, limit the BW due to arbitration
  - More pronounced in Vector PEs due to **unit-stride** visiting patterns

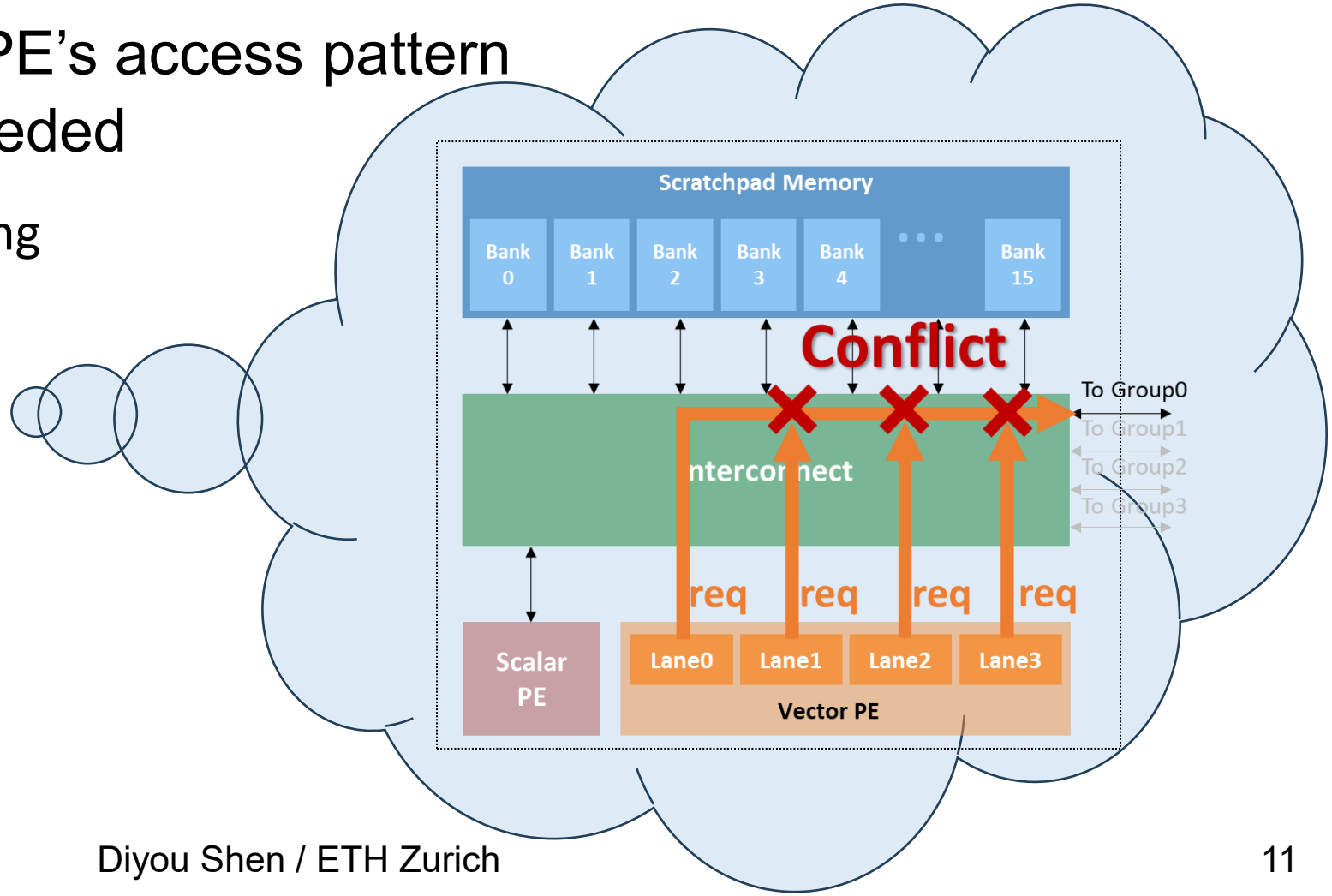
Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed

Baseline

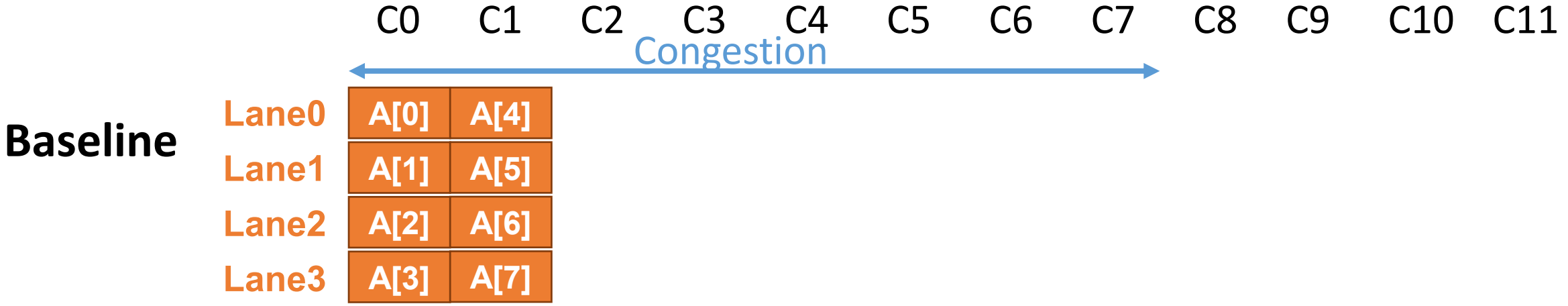
Vector Loading

Lane0	A[0]	A[4]
Lane1	A[1]	A[5]
Lane2	A[2]	A[6]
Lane3	A[3]	A[7]



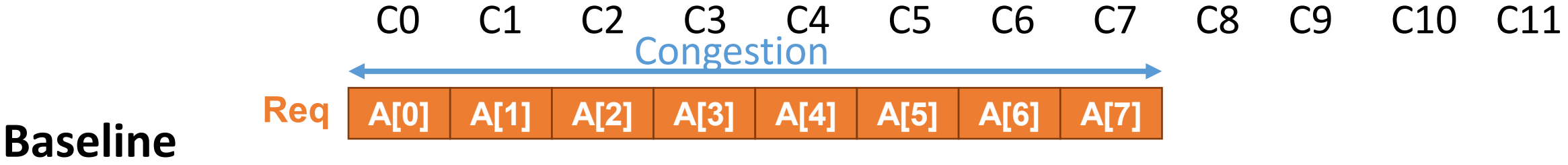
Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed



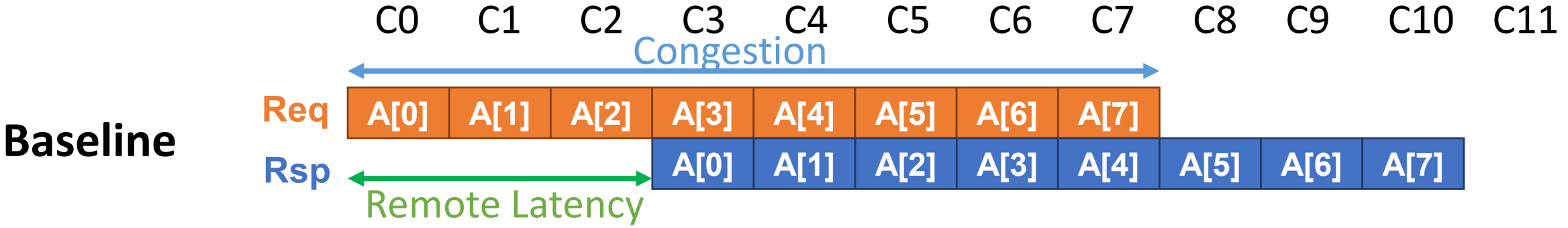
Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed



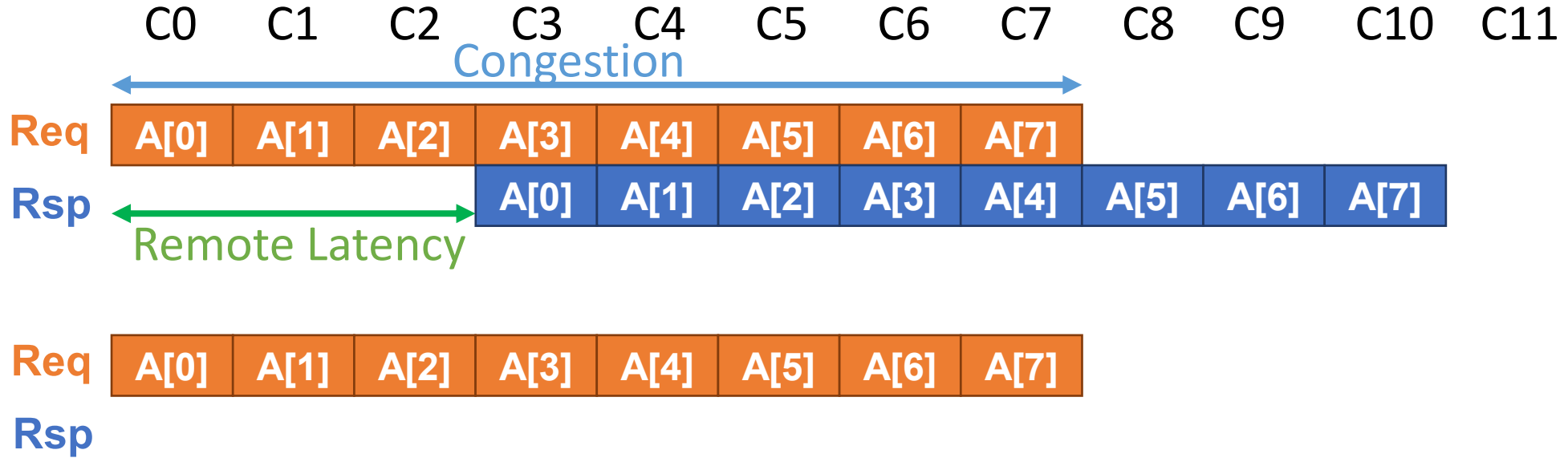
Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed



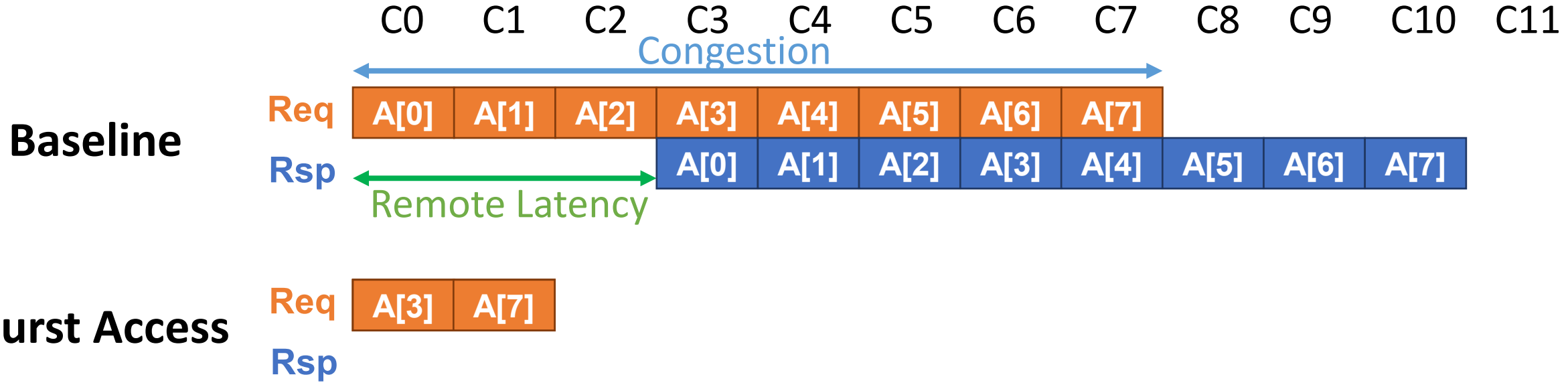
Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed



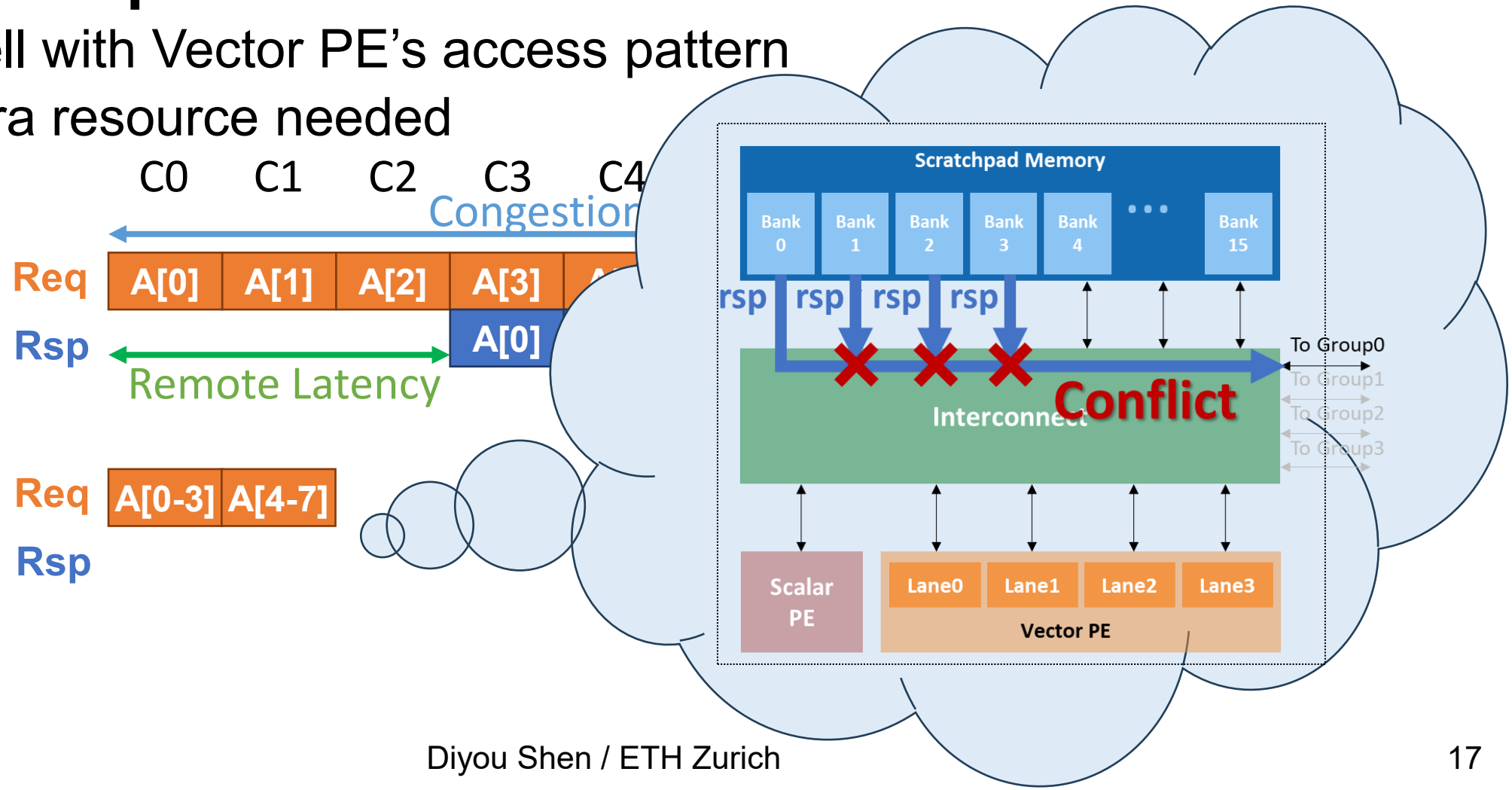
Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed



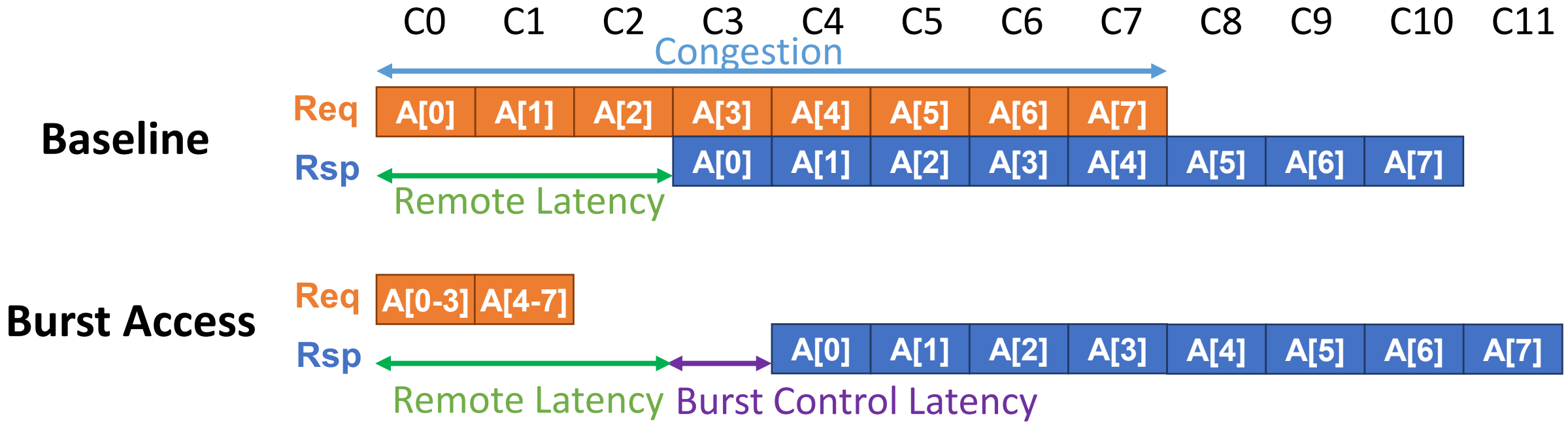
Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed



Our Solution: TCDM Burst Access

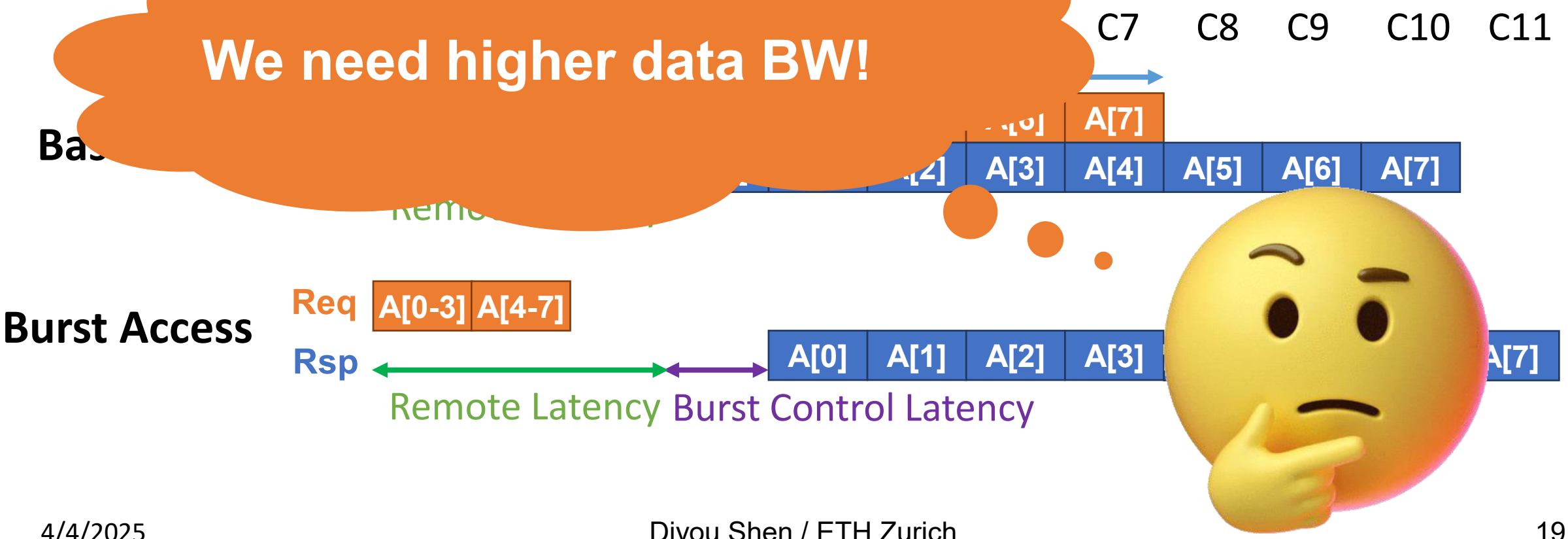
- **Pack the requests into a short burst**
  - Fits well with Vector PE's access pattern
  - No extra resource needed



Our Solution: TCDM Burst Access

- **Pack the requests into a short burst**
  - Fits well with Vector PE's
  - N

We need higher data BW!

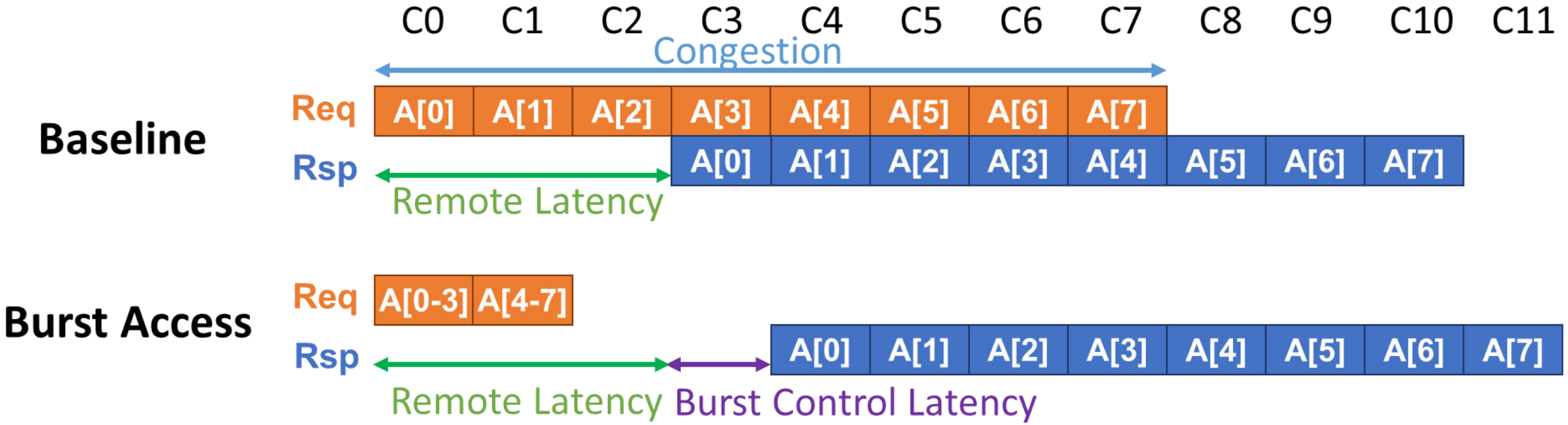


TCDM Burst Extension: Grouped Responses

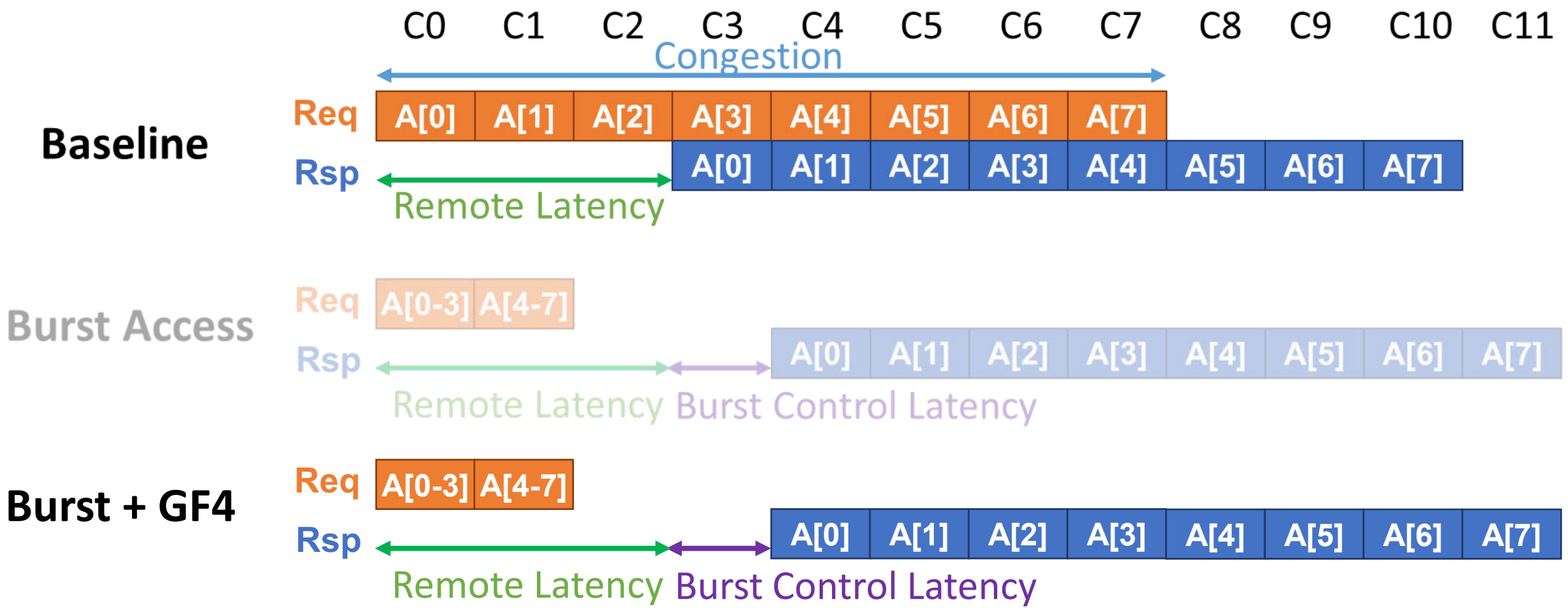
- **Group multiple data in a single transfer**
  - Have **higher data bandwidth** on response channel
  - **Minimize** the cost: only widen the response data field
  - **Configurable**: 2x data width (GF2), 4x data width (GF4), …

TCDM Burst Extension: Grouped Responses

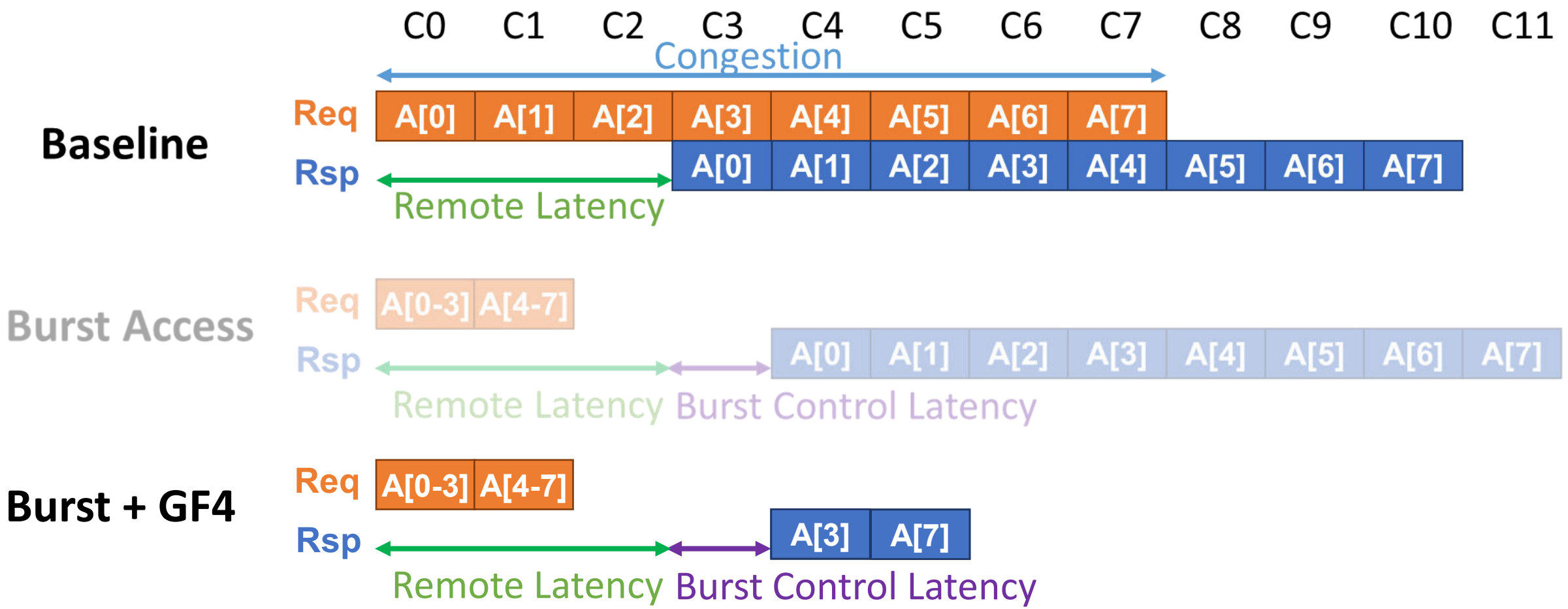
- **Group multiple data in a single transfer**
  - Have **higher data bandwidth** on response channel
  - **Minimize** the cost: only widen the response data field
  - **Configurable**: 2x data width (GF2), 4x data width (GF4), …



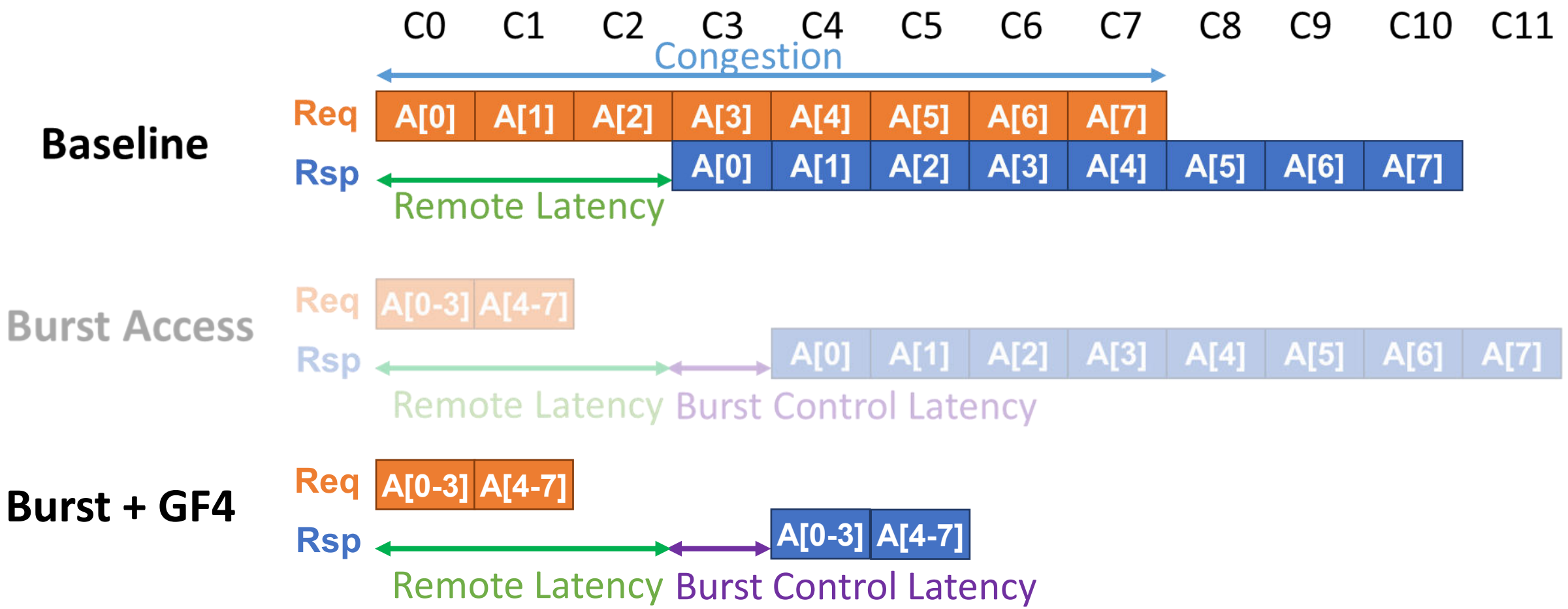
TCDM Burst Extension: Grouped Responses



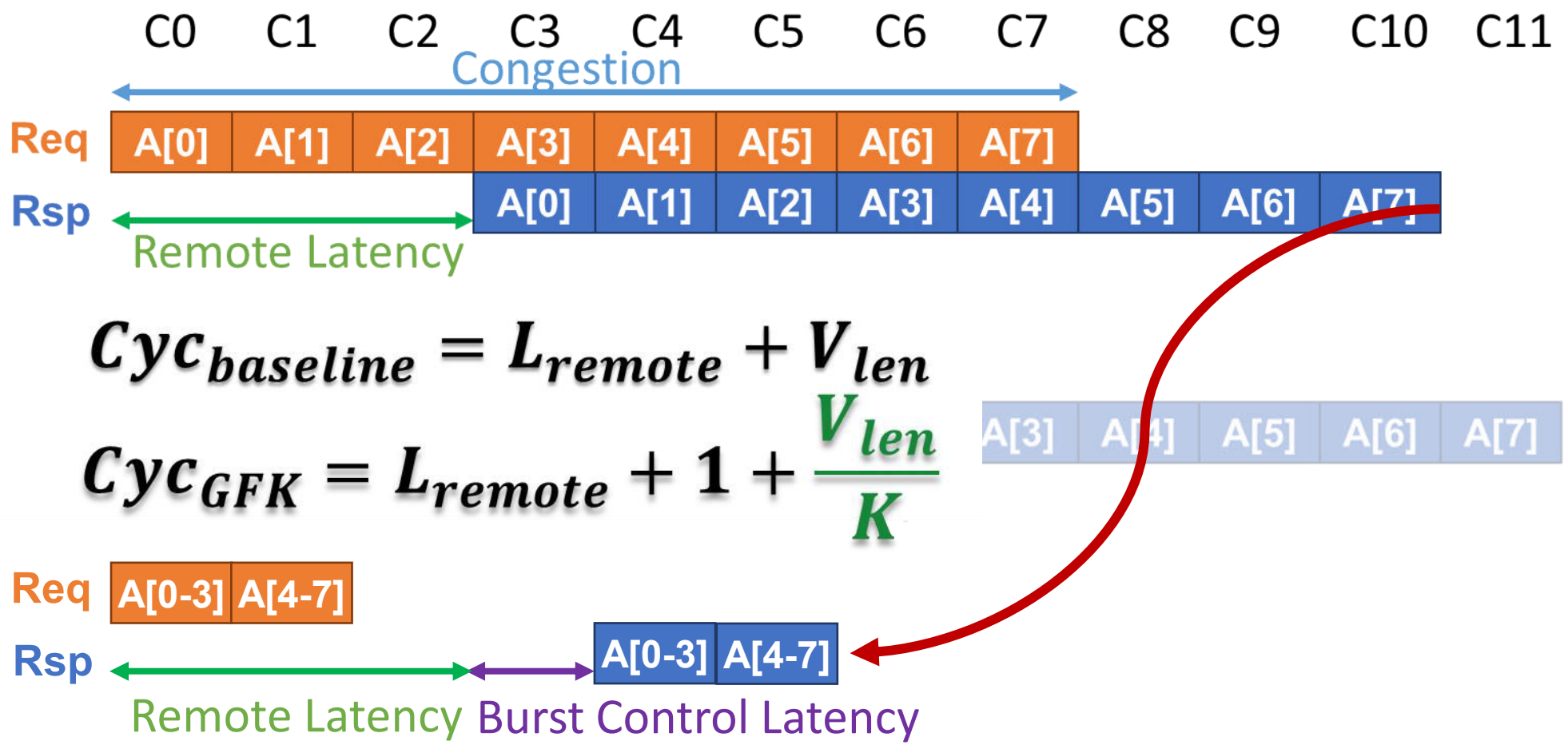
TCDM Burst Extension: Grouped Responses



TCDM Burst Extension: Grouped Responses



TCDM Burst Extension: Grouped Responses



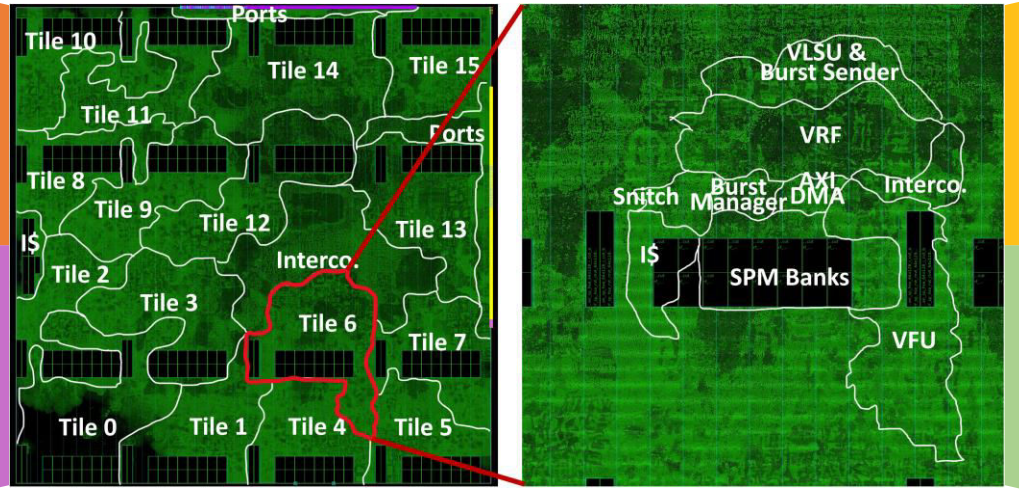
$Speedup_{GF4} \rightarrow 4$ for long vectors

Fully Routable and Highly Efficient!

Implement on $MemPool_{64}$-$Spatz_4$ Cluster in GF12 tech.

3.26x Effective BW

Up to **2.76x** Performance Improvement



770 MHz @WW
Less than **7.7%** Area

Up to **1.9x** Energy Efficiency

We design TCDM Burst Access

- A **physically feasible**, **high performance**, **low energy** interconnection extension for **large shared-L1 clusters**.

Come to our poster and discover more!

- Tested on different cluster sizes.
- Roofline models to evaluate the designs.
- Fully **open-sourced**!

`github.com/pulp-platform/mempool`

Diyou Shen dishen@iis.ee.ethz.ch
ETZ, Gloriastrasse 35, 8092 Zürich
@pulp_platform

ETH zürich CORENEXT