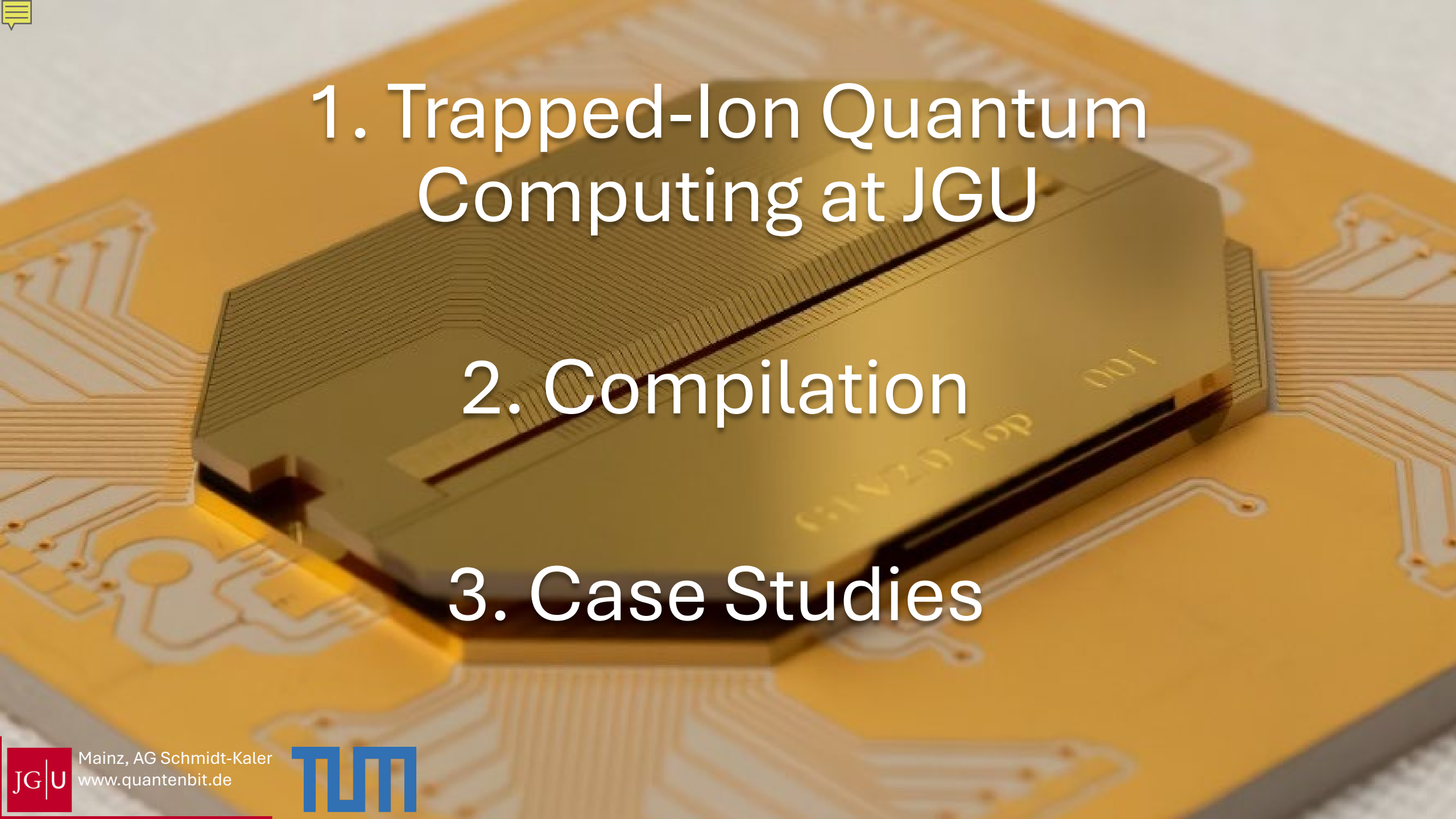
Computational Capabilities and Compilation Strategies for Trapped-Ion Quantum Computers

Jurek Eisinger, Ludwig Schmid, Daniel Schoenberger, Christian Marciniak, Janine Hilder, Ulrich Poschinger, Robert Wille, Ferdinand Schmidt-Kaler

Q-STAV Workshop, Bern

Mainz, AG Schmidt-Kaler
www.quantenbit.de



1. Trapped-Ion Quantum Computing at JGU

2. Compilation

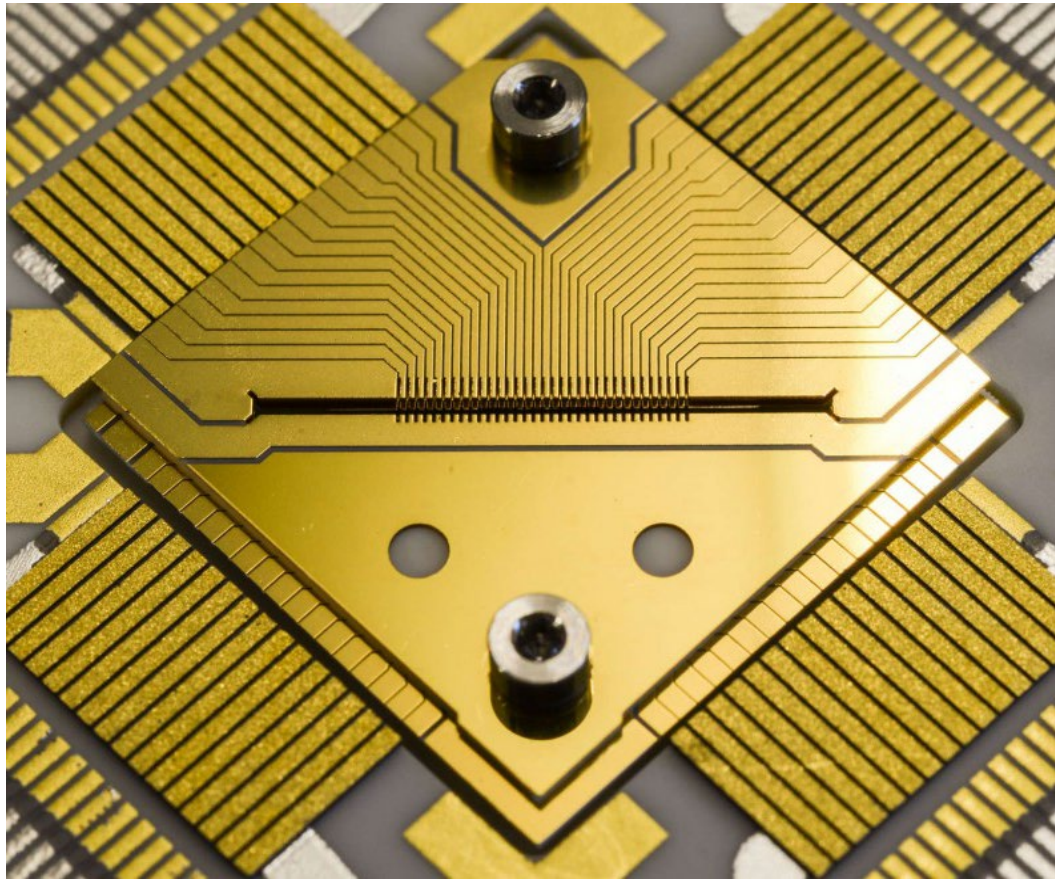
3. Case Studies

Trapped-Ion Quantum Computing at JGU

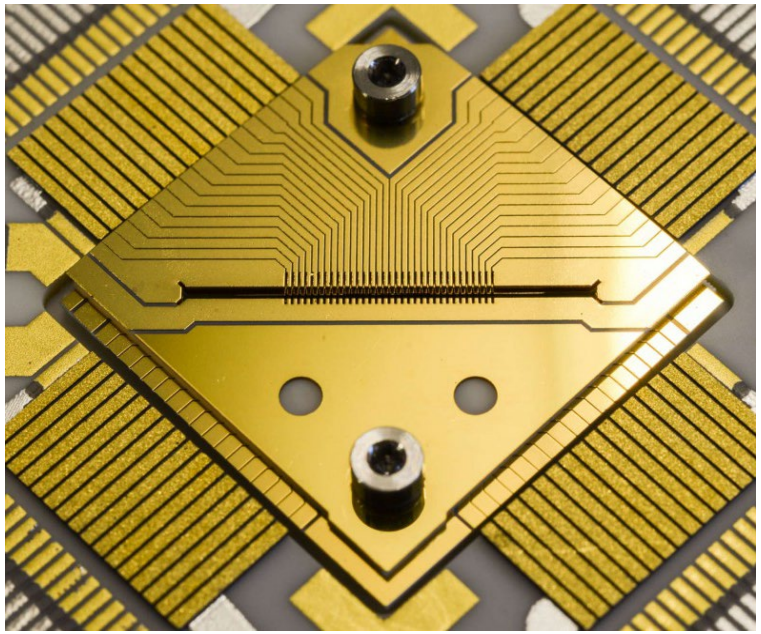
Mainz, AG Schmidt-Kaler
www.quantenbit.de

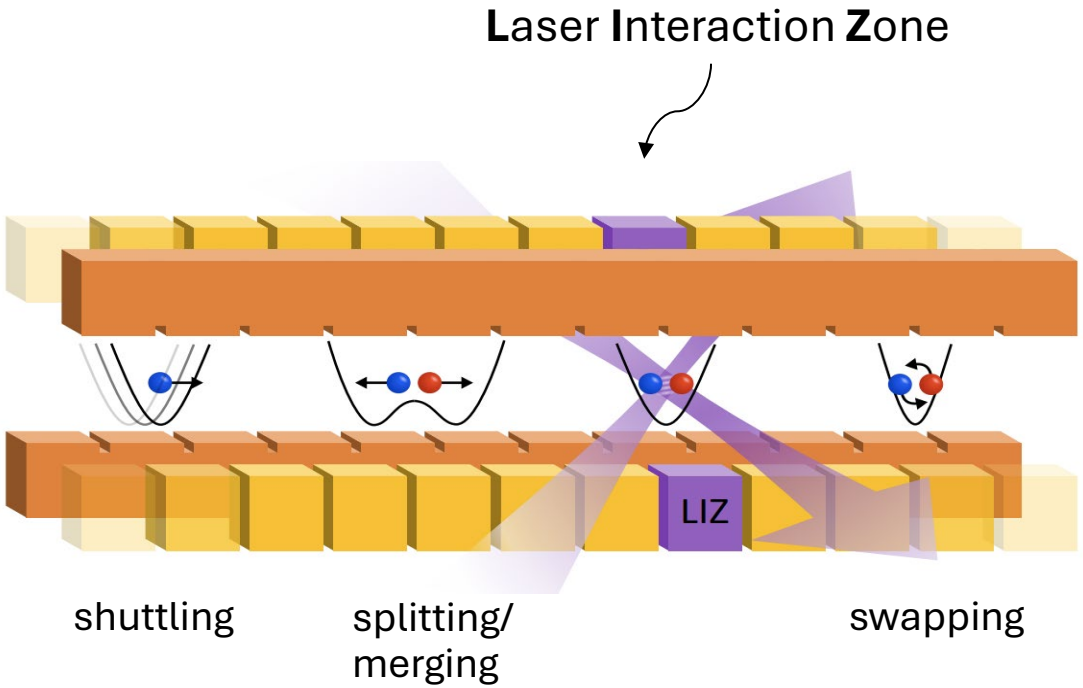
Quantum Computing in Mainz

linear
segmented
Paul trap

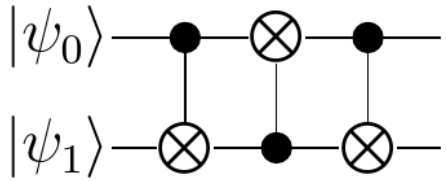


Quantum Computing in Mainz

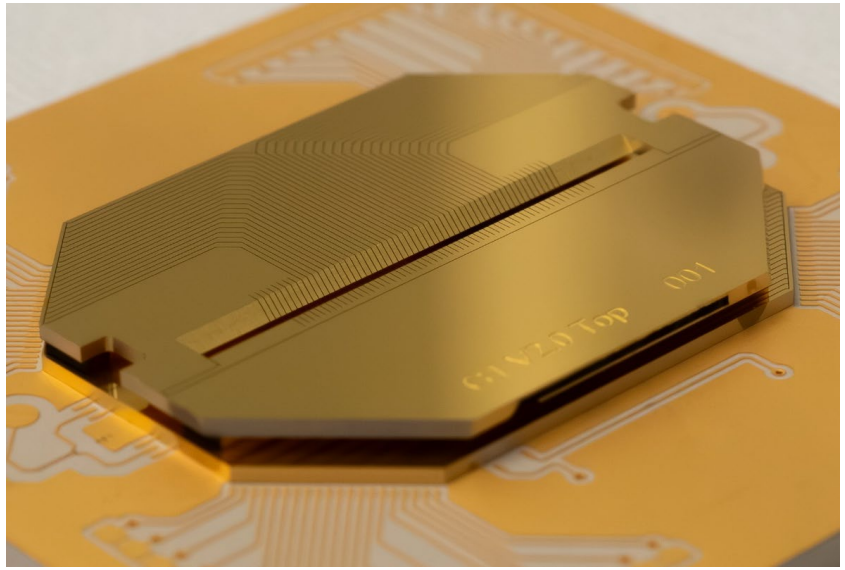


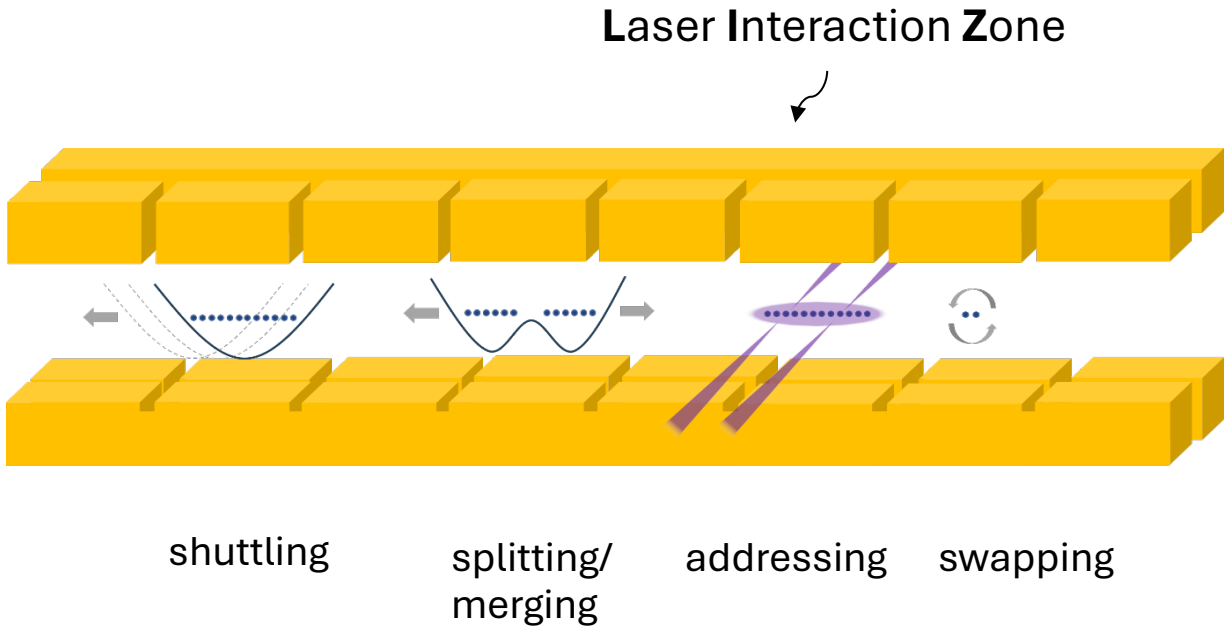


SWAP:



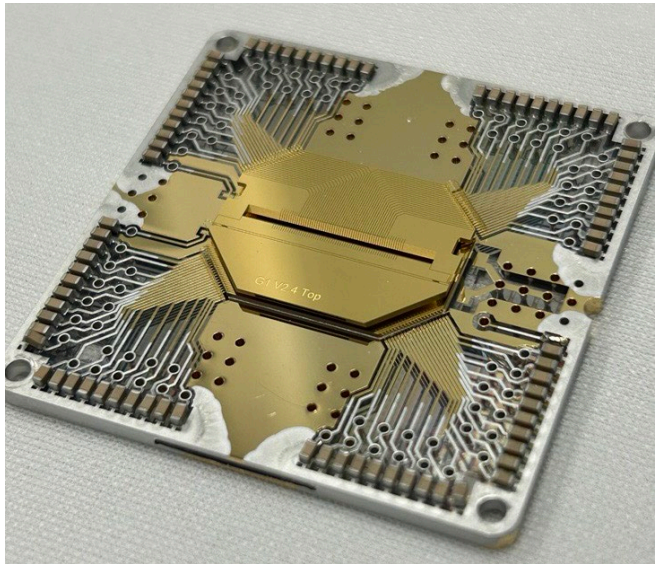
Quantum Computing in Mainz





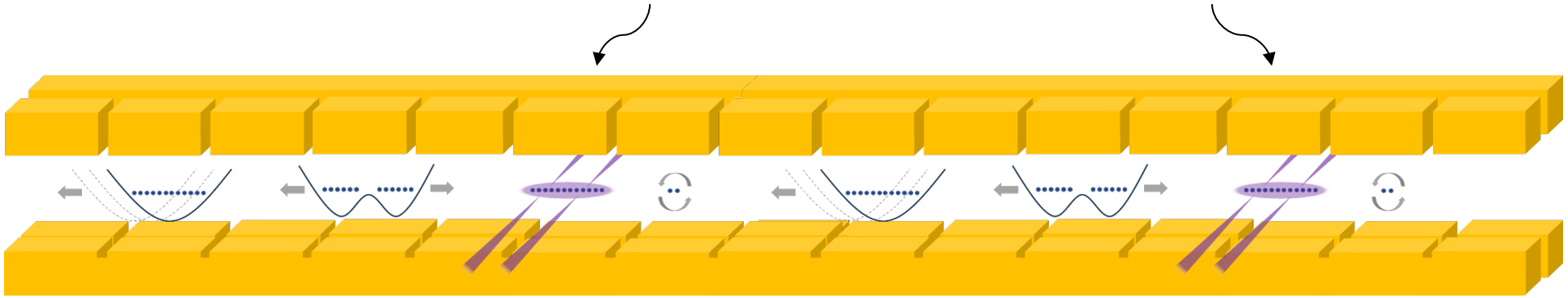
Under Construction!

Quantum Computing in Mainz



Under Construction!

Multiple
Laser **I**nteraction **Z**ones



Trapped-Ion Quantum Computing

Architecture choices

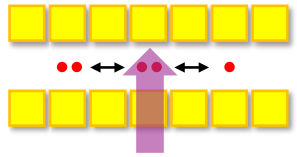
1. Only shuttling

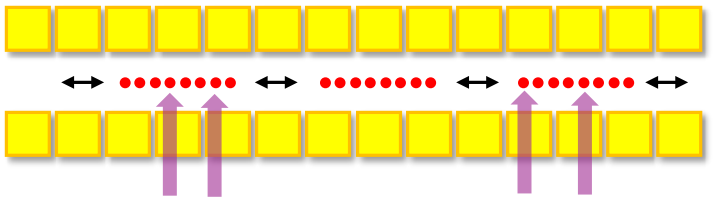
2. Addressing + shuttling

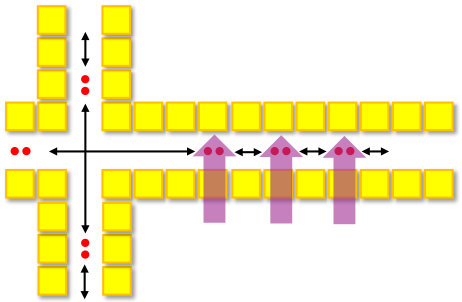
   - Number of processing zones

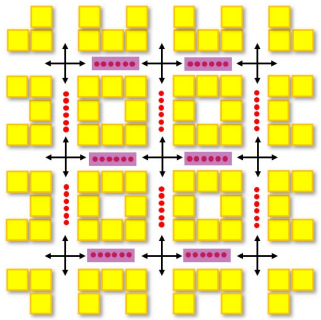
   - Number of qubits in processing zones

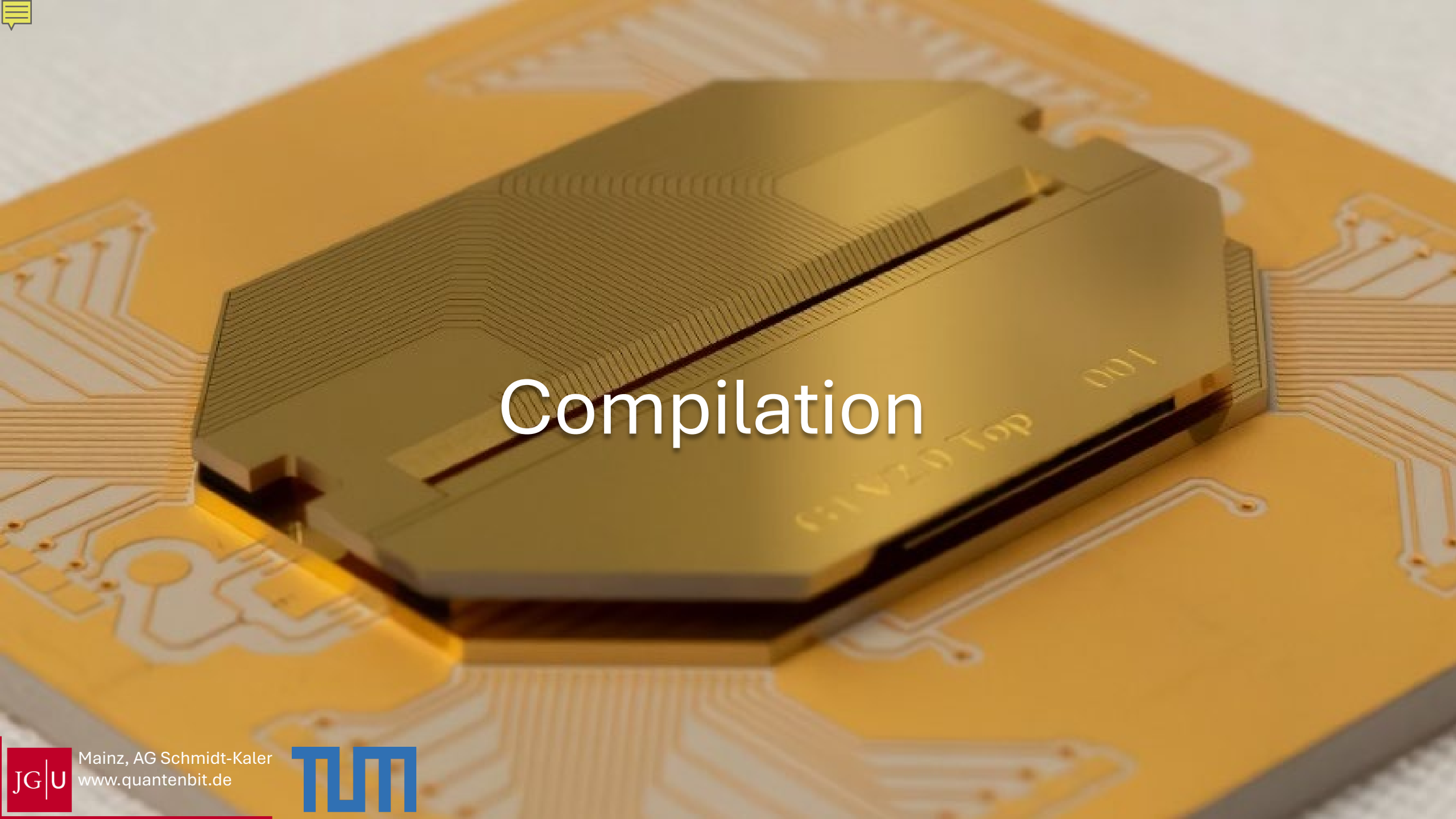
3. Dimensionality of the trap









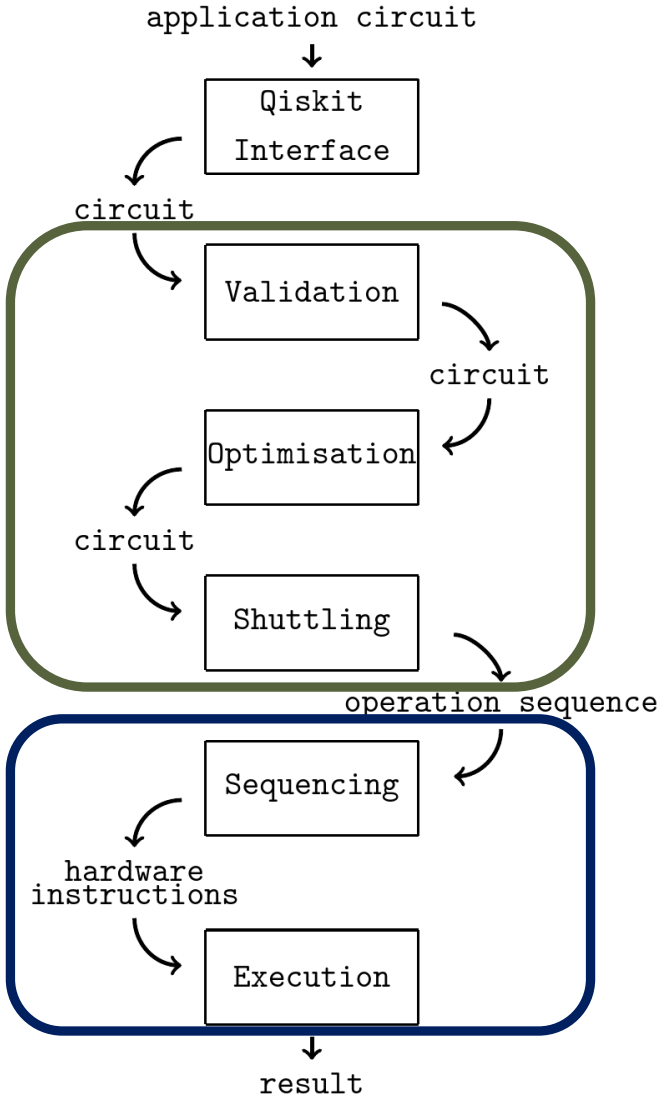


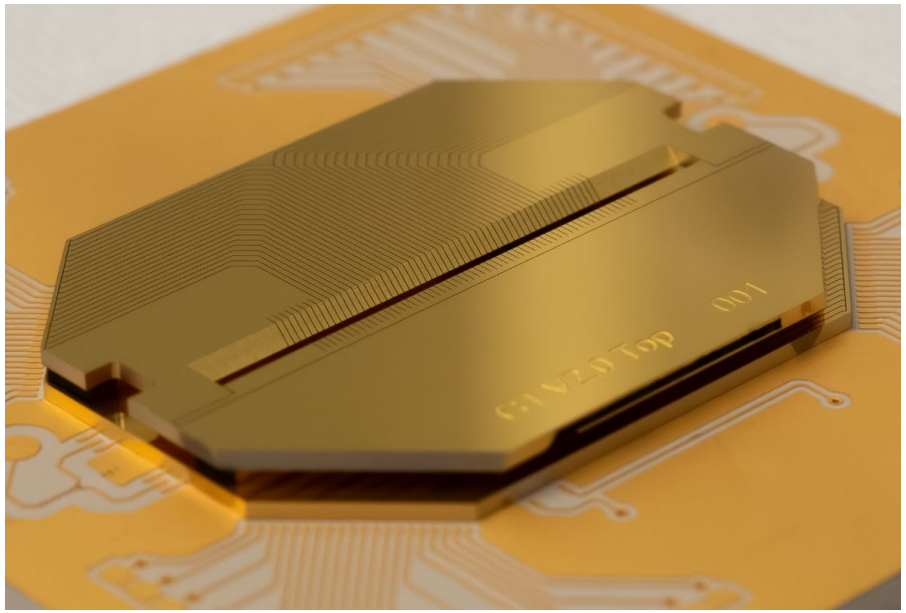
Compilation

Compilation - Software Stack

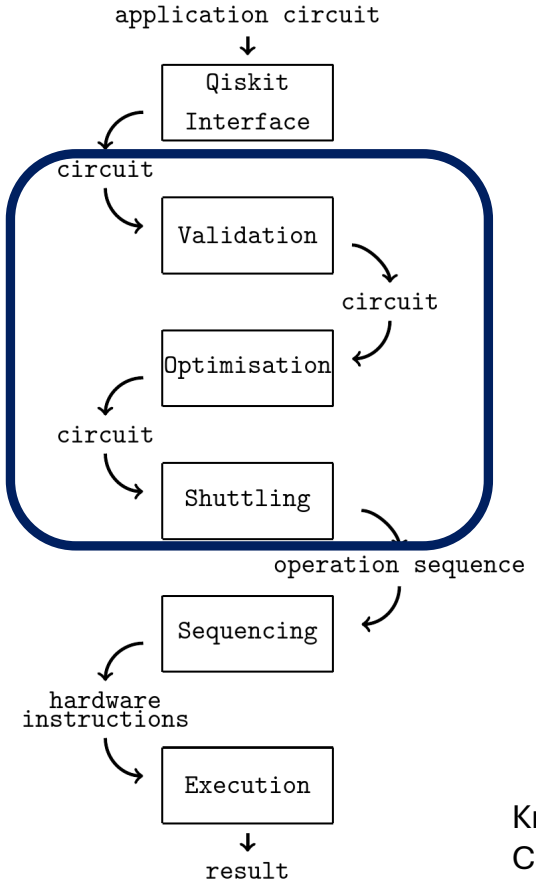
High Level Stack

Low Level Stack



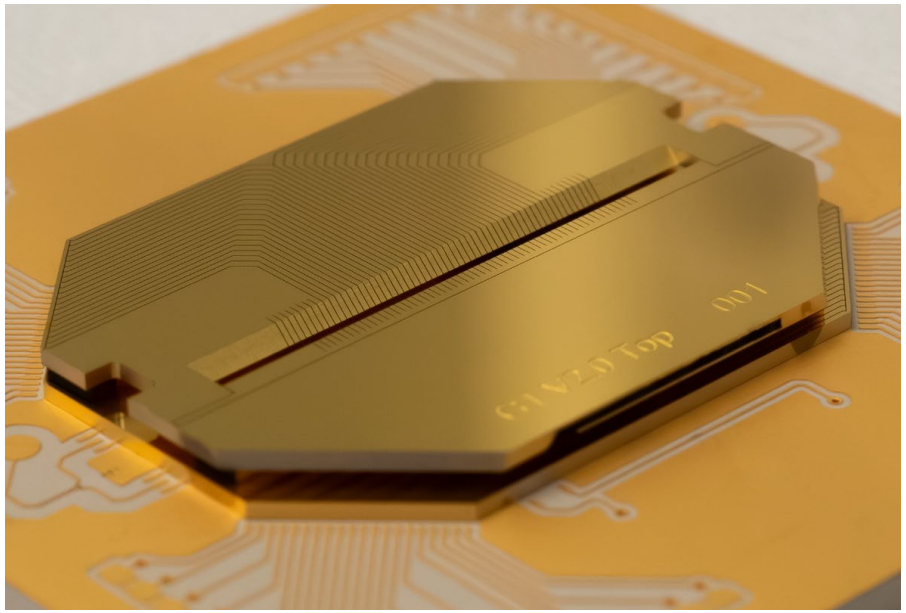


Compilation – Software Stack



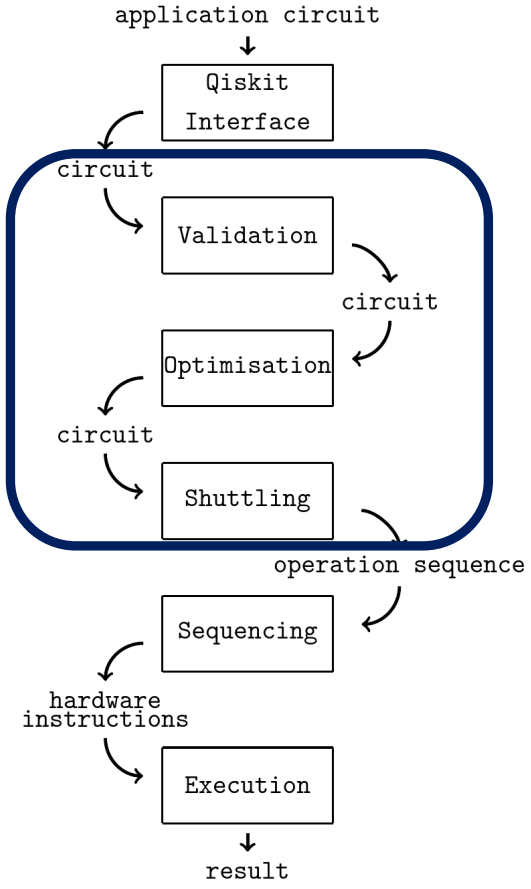
Goal:

Get computer scientists involved!



Kreppel *et al.*, "Quantum Circuit Compiler for a Shuttling-Based Trapped-Ion Quantum Computer," *Quantum*, vol. 7, p. 1176, 2023.

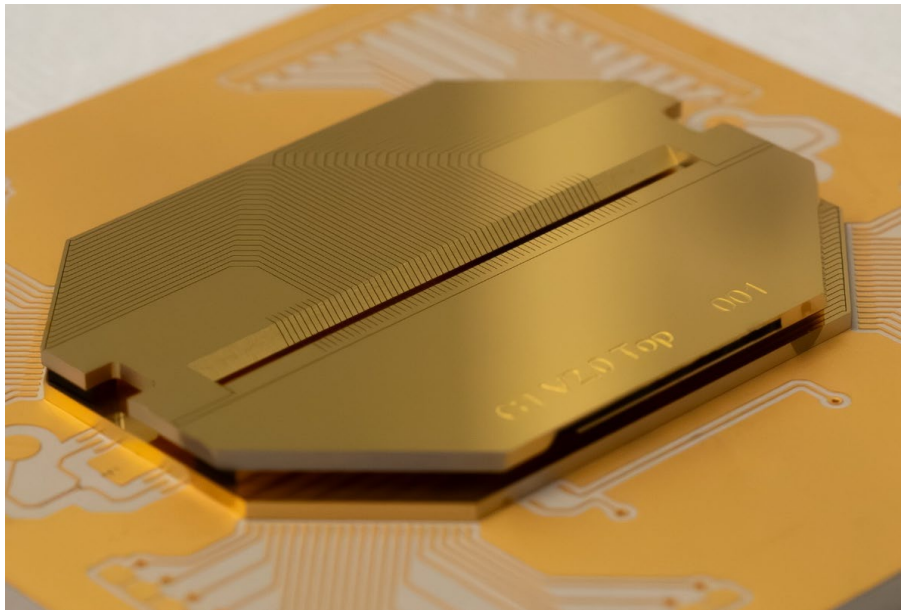
Compilation



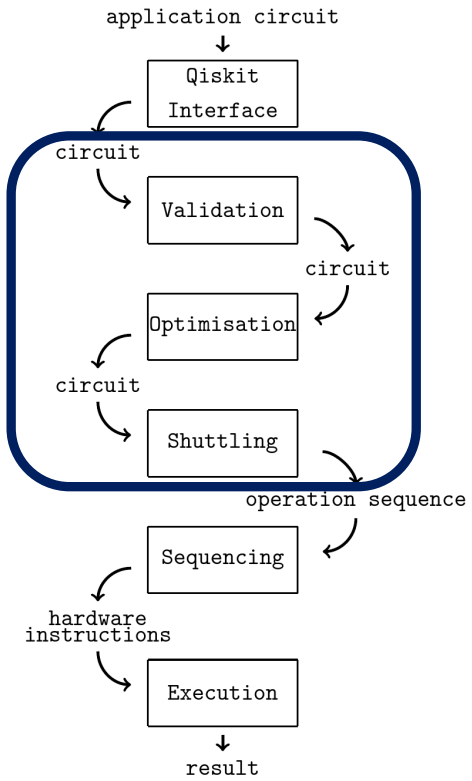
Technische Universität München

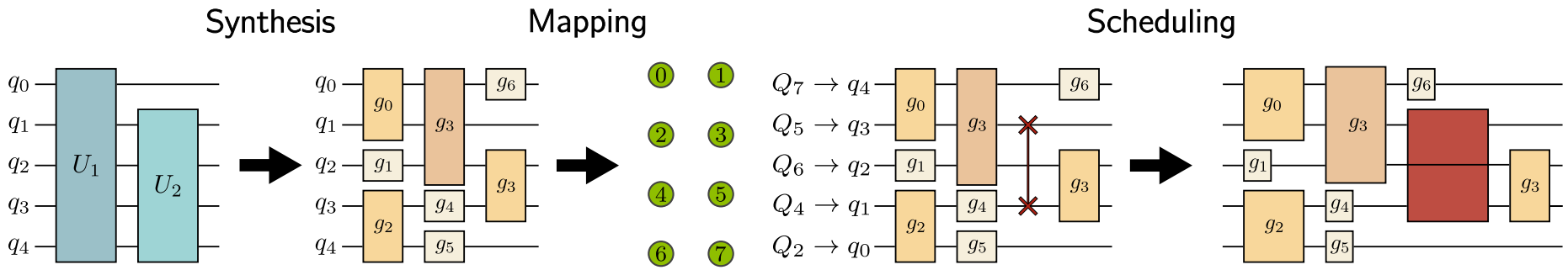
Goal:

Get computer scientists involved!

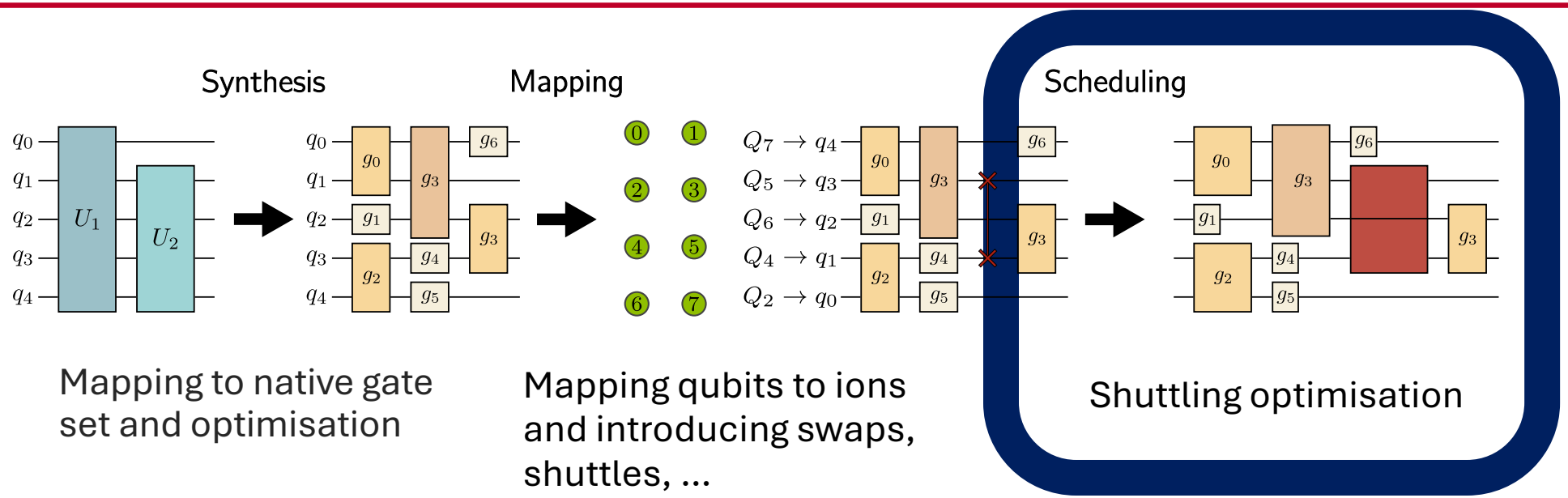


Compilation

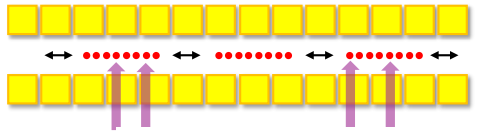




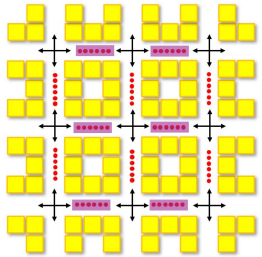
Compilation



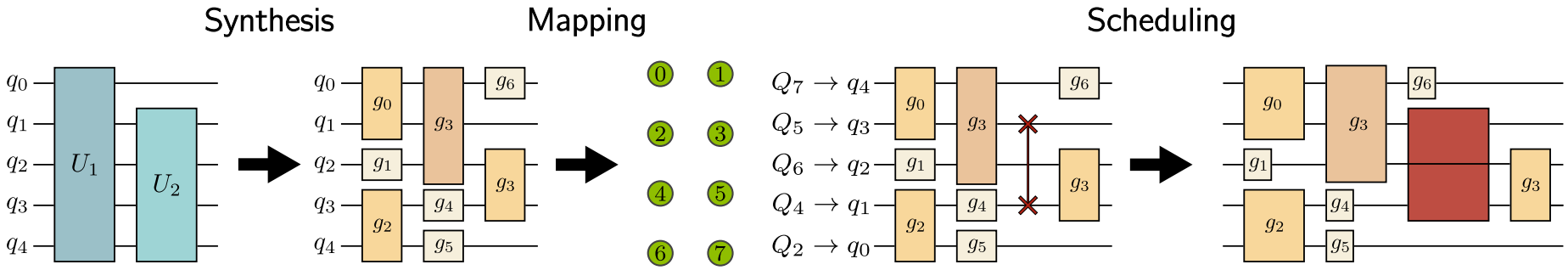
First: Linear setups (JGU)



Then: Two-dimensional setups (TUM)



Compilation



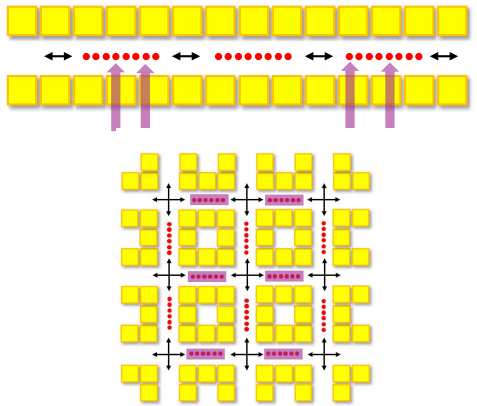
Mapping to native gate set and optimisation

Mapping qubits to ions and introducing swaps, shuttles, ...

Shuttling optimisation

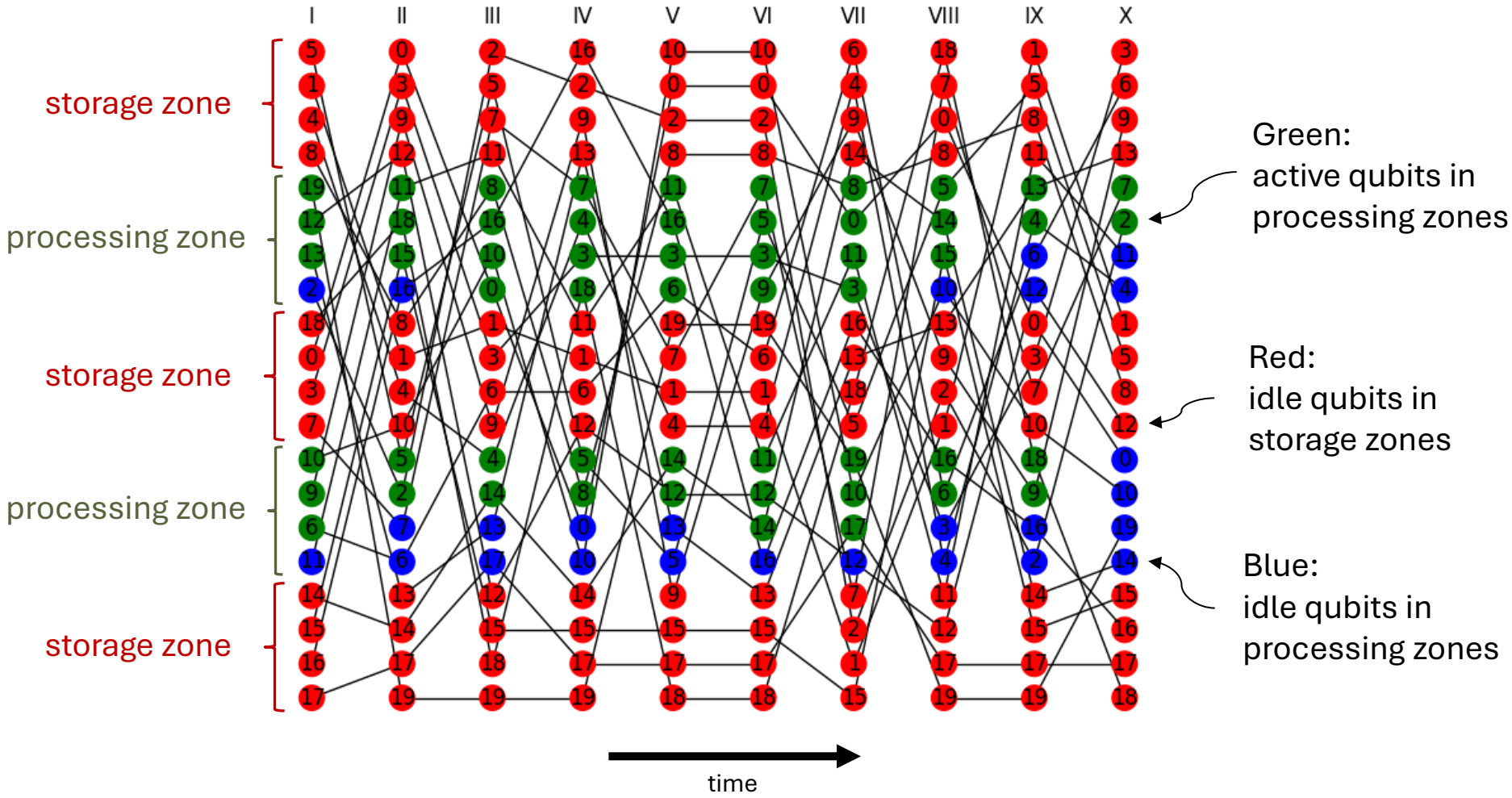
First: Linear Setups (JGU)

Then: Two-Dimensional Setups (TUM)

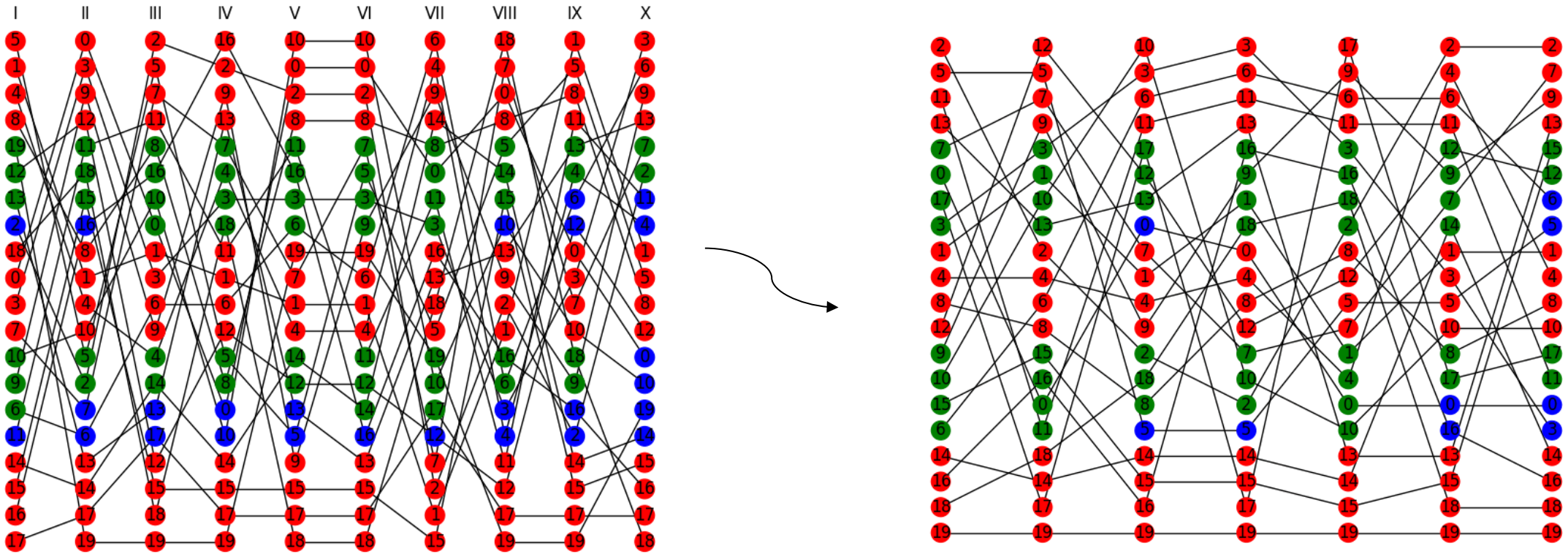


Goal: Benchmark Architectures for different Circuits!

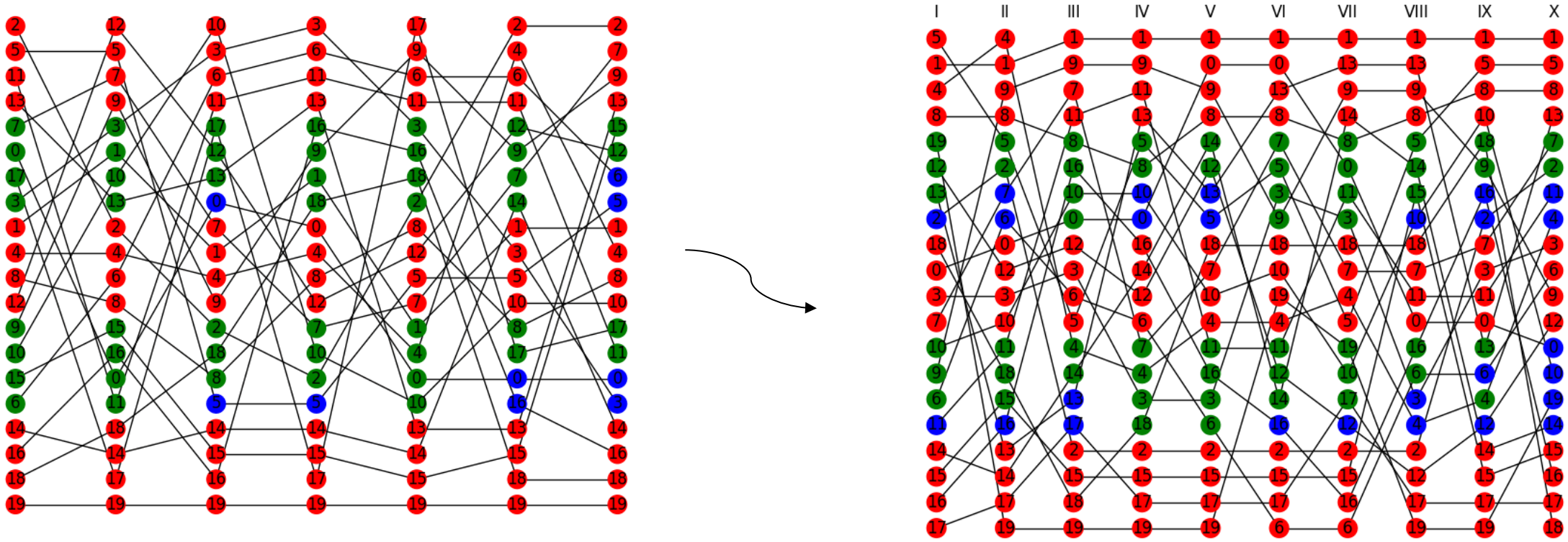
Linear Setups - Block Aggregation Algorithm



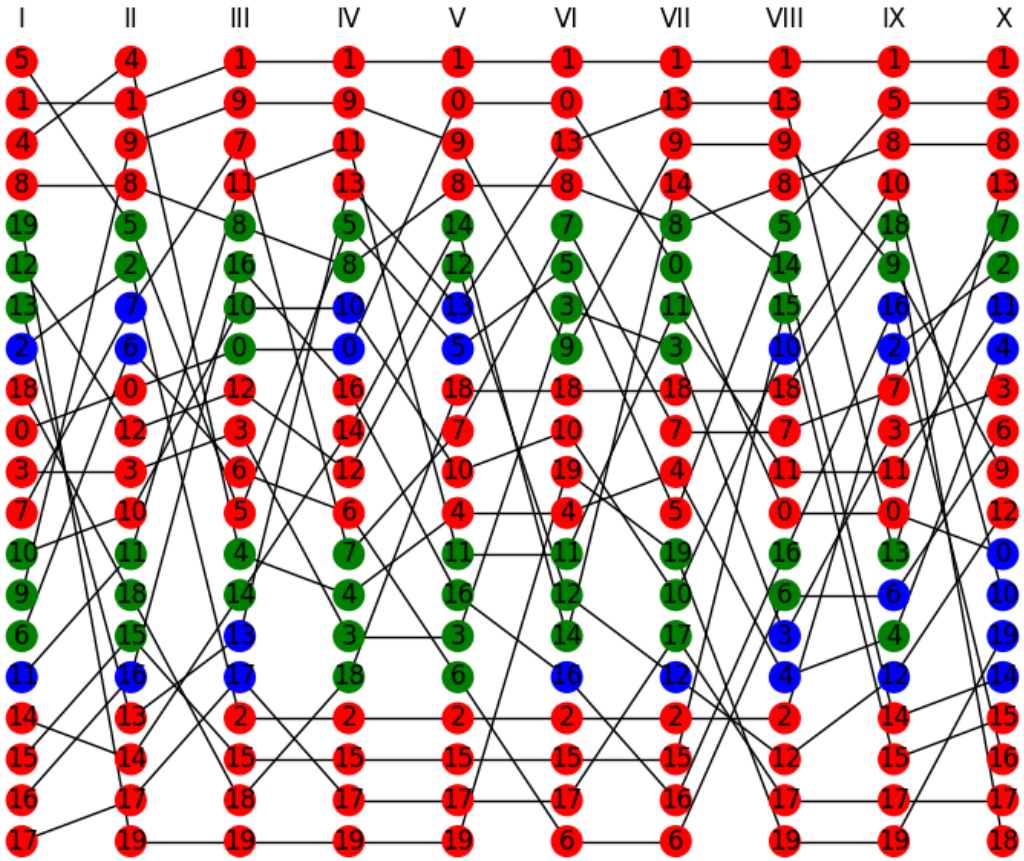
Linear Setups - Block Aggregation Algorithm



Linear Setups - Block Aggregation Algorithm



Linear Setups - Block Aggregation Algorithm



Measure of how well circuit can be executed on hardware

Linear Compiler – Metrics

Computational overhead: SWAPs

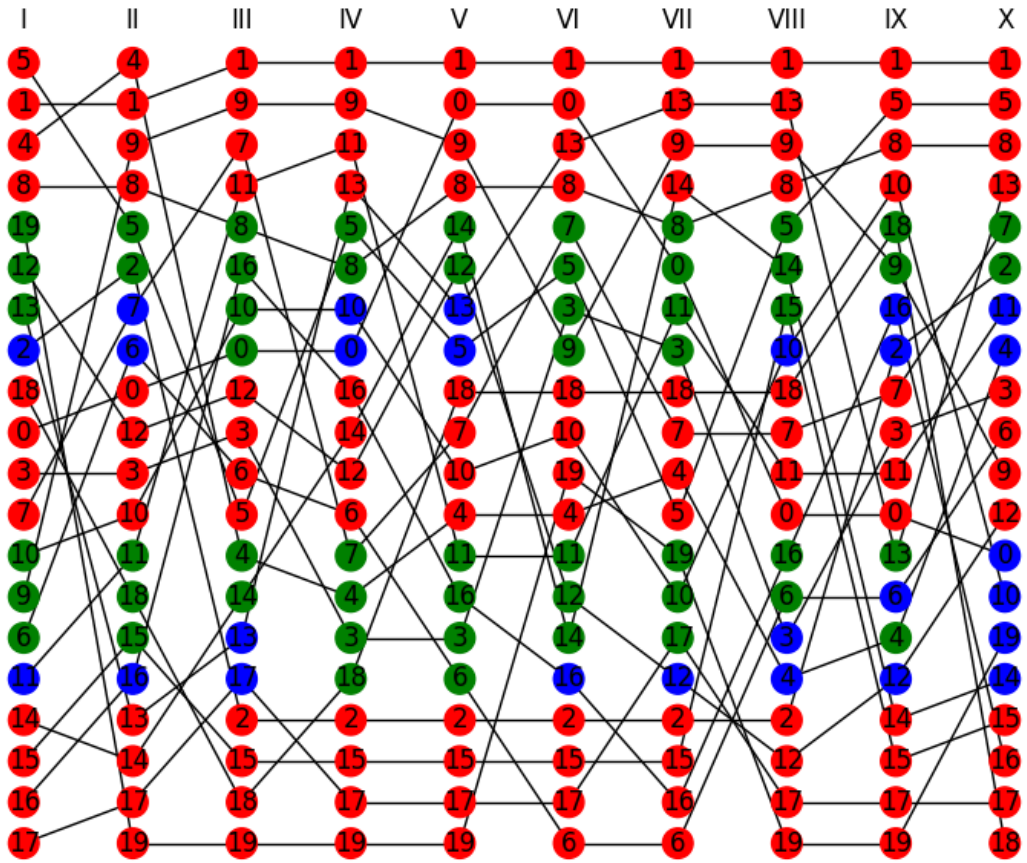
Performance Metrics:

1. Number of swaps:

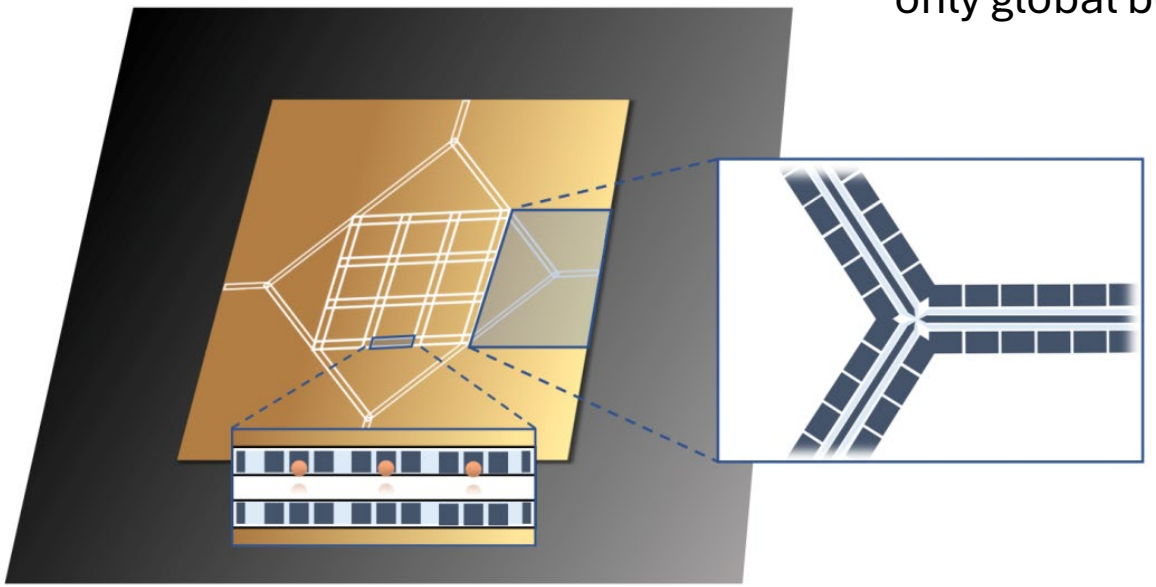
   Reduction of movement of each qubit

2. Number of timesteps that can be executed in parallel

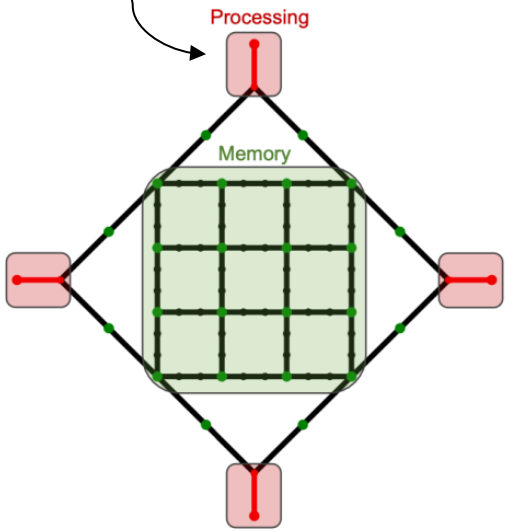
   Optimal movement for all qubits



Two-Dimensional Compiler



only global beams

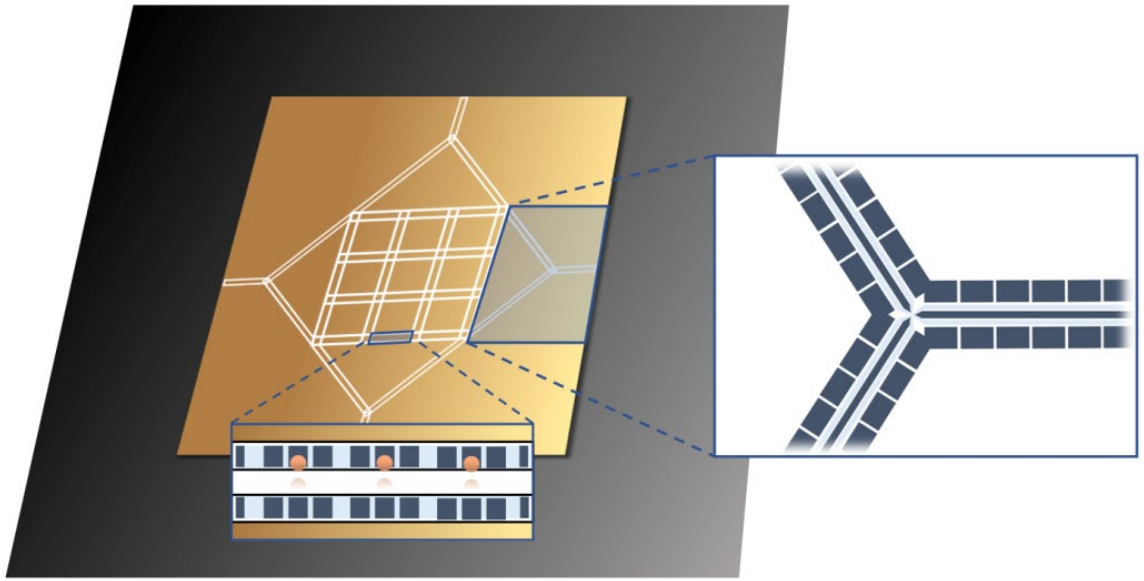


Computational overhead:

Junction shuttles

Schoenberger & Wille, *Orchestrating Multi-Zone Shuttling in Trapped-Ion Quantum Computers*, arXiv:2505.07928 (2025).

Two-Dimensional Compiler



Schoenberger & Wille, *Orchestrating Multi-Zone Shuttling in Trapped-Ion Quantum Computers*, arXiv:2505.07928 (2025).

Performance metrics:

1. Number of junction shuttles:

   Reduction of movement of each qubit

2. Number of timesteps that can be executed in parallel

   Optimal movement for all qubits

Challenge:

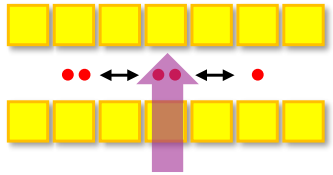
Compare 1D- to 2D metrics

Case Studies

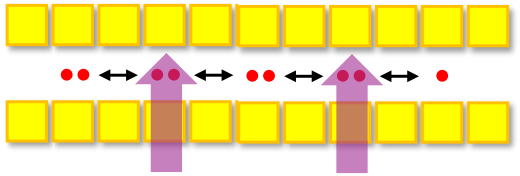
Case Studies

1. Scaling of computational overhead with system size:
   - Variation of #qubits in processing zones
   - Variation of #processing zones
   - Variation of trap dimension (1D vs. 2D)
   - Using random circuits

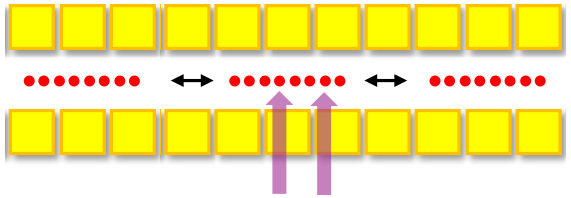
Setup 1



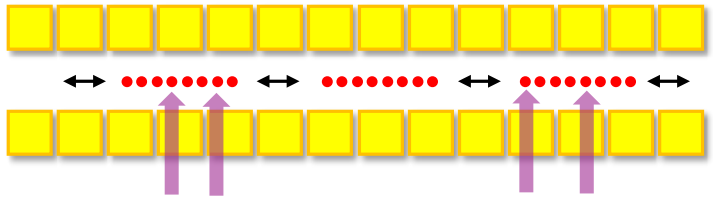
Setup 4

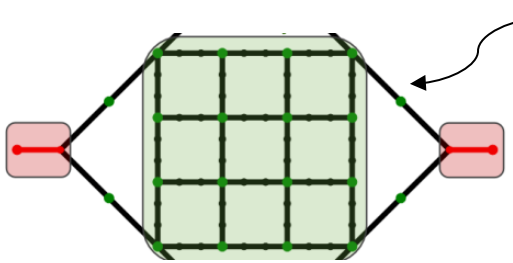


Setup 2



Setup 3



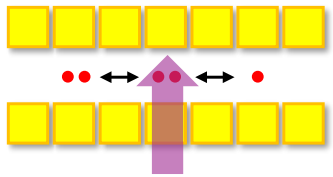


Case Studies

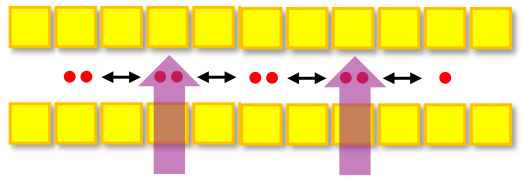
1. Scaling of computational overhead with system size:
   - Variation of #qubits in processing zones
   - Variation of #processing zones
   - Variation of trap dimension (1D vs. 2D)
   - Using random circuits

2. Comparison using logical teleportation circuit (structured circuit):
   - 1D:
     - Variation of #processing zones
     - Variation of #qubits in processing zones
   - 2D:
     - Variation of #processing zones

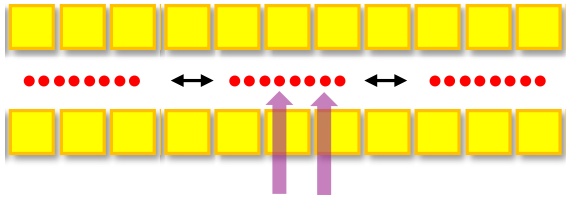
Setup 1



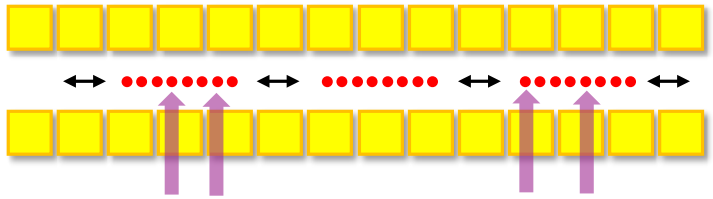
Setup 4

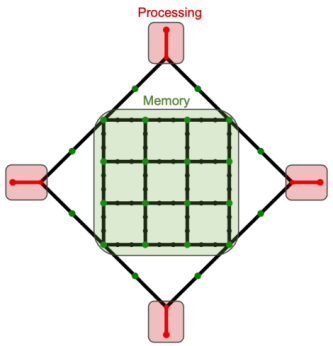


Setup 2

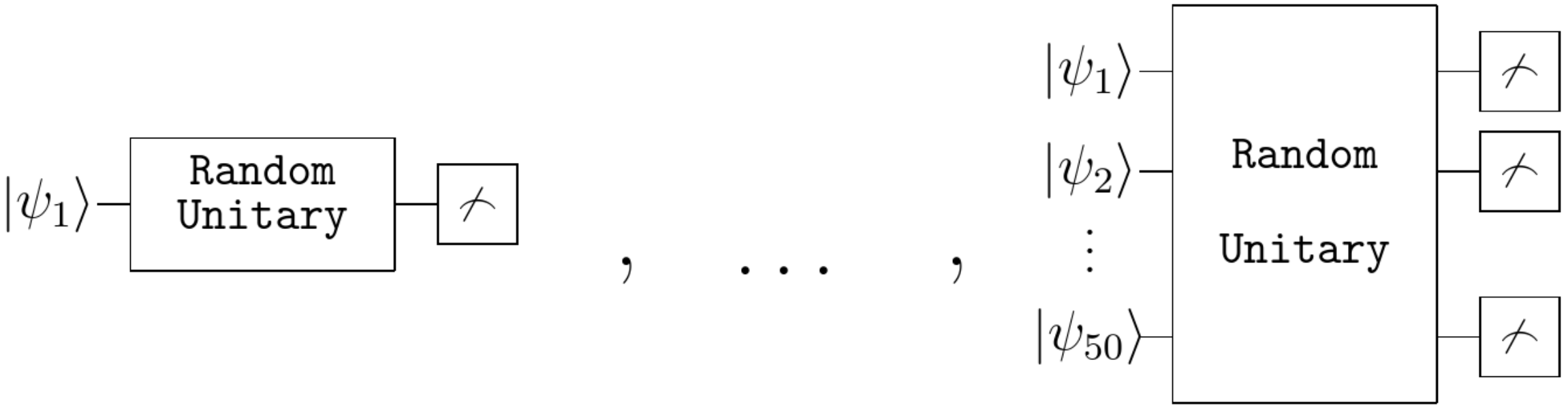


Setup 3

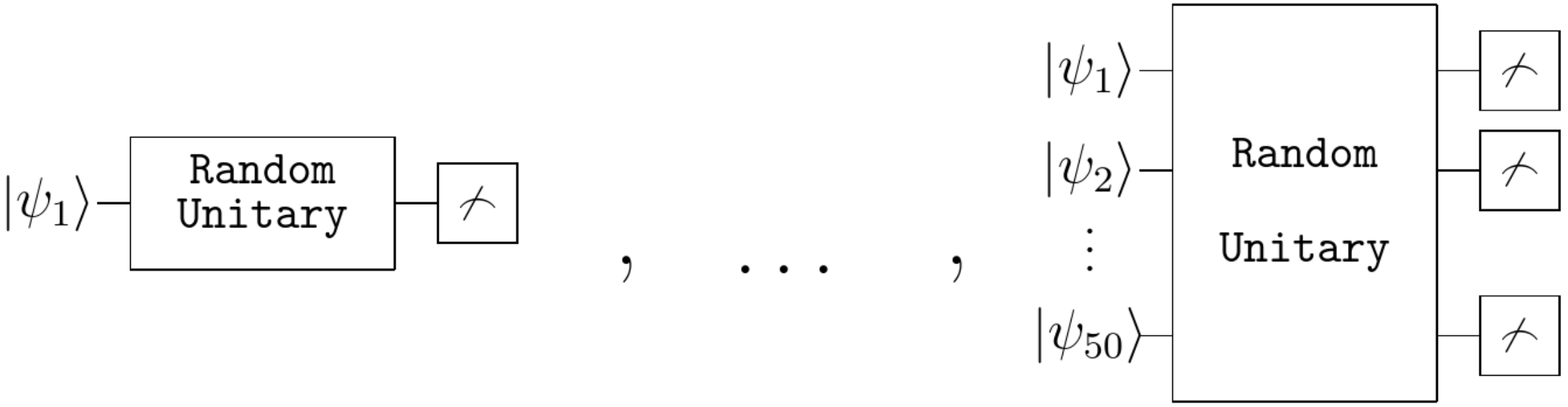




Random Circuits



Random Circuits

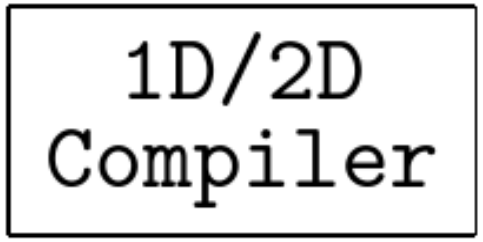


#processing zones

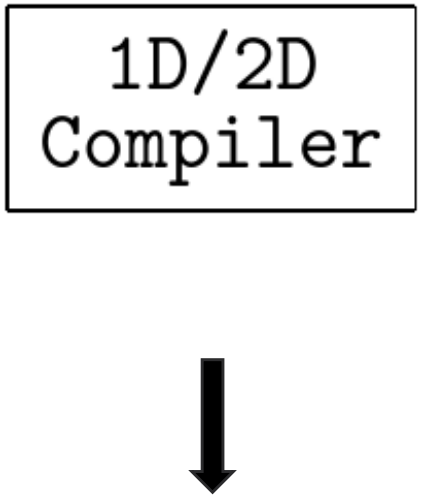
Capacity processing zones

1D/2D Compiler

Random Circuits

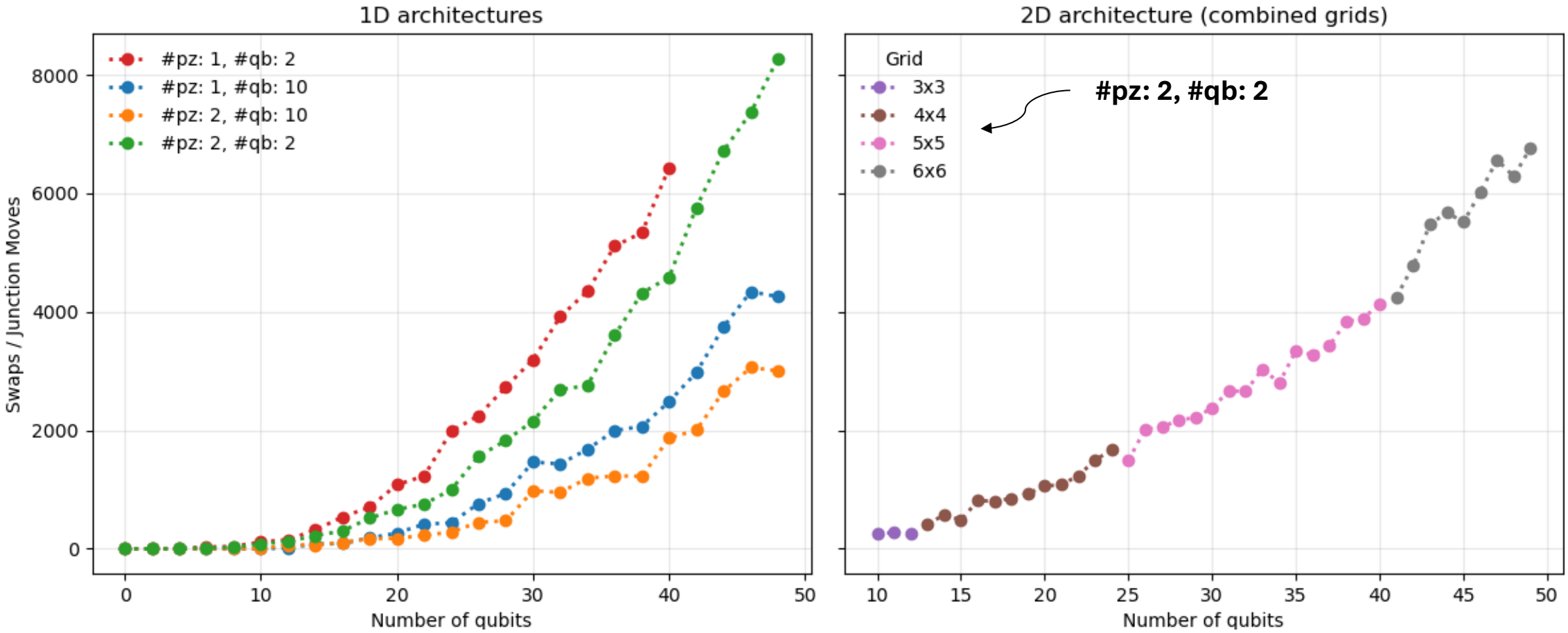


Random Circuits

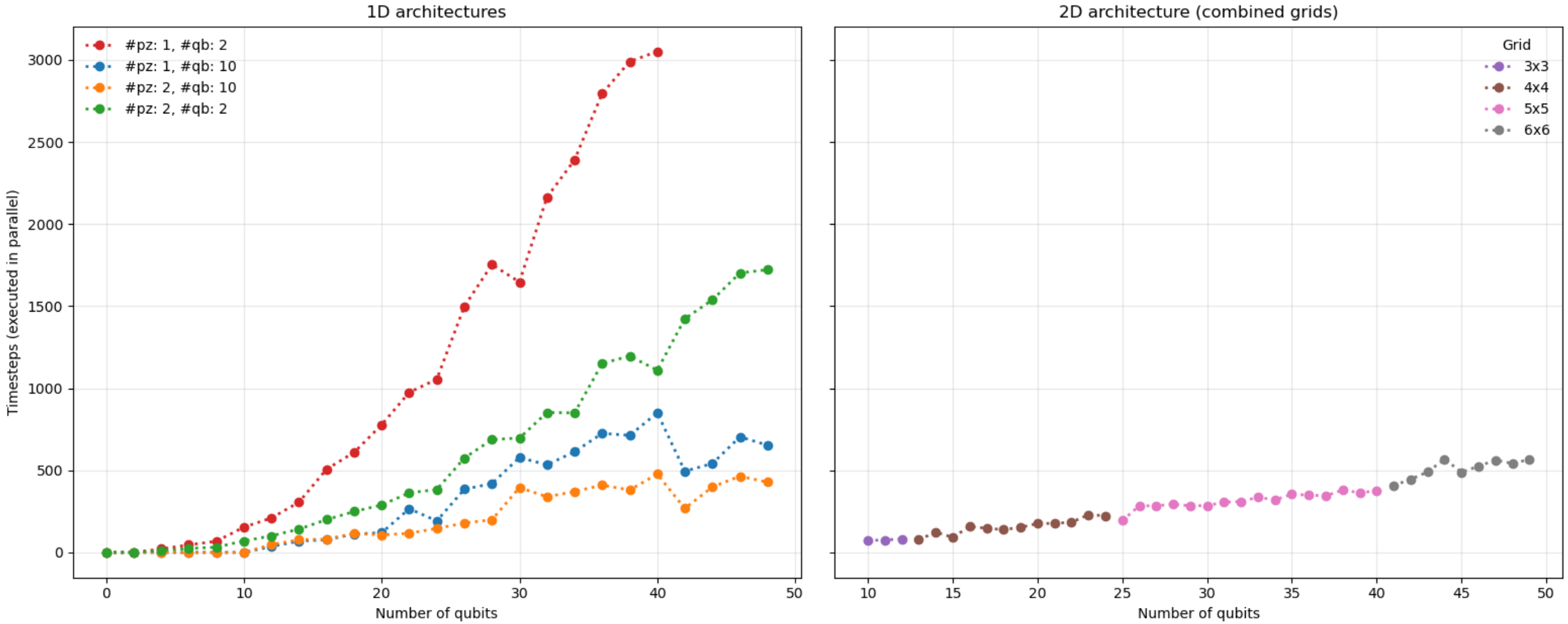


Performance metric

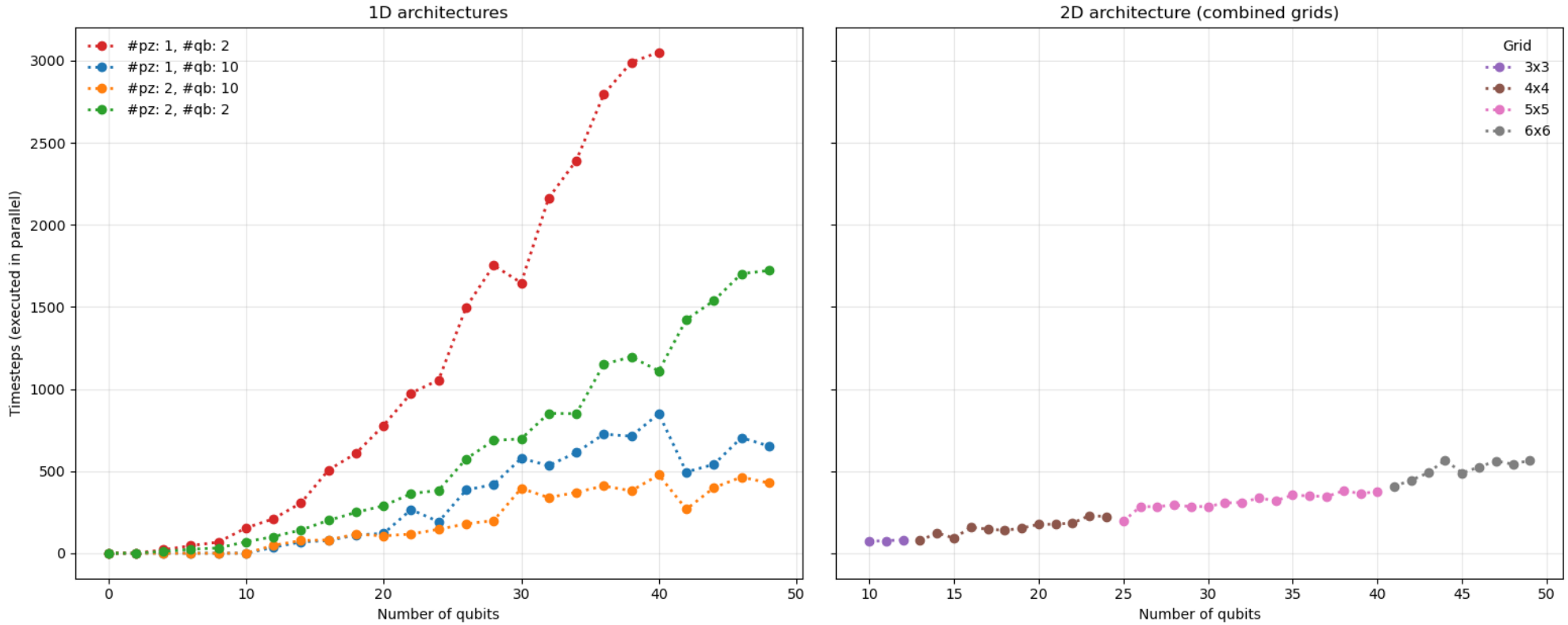
Random Circuits – Cost Scaling



Random Circuits – Time Scaling

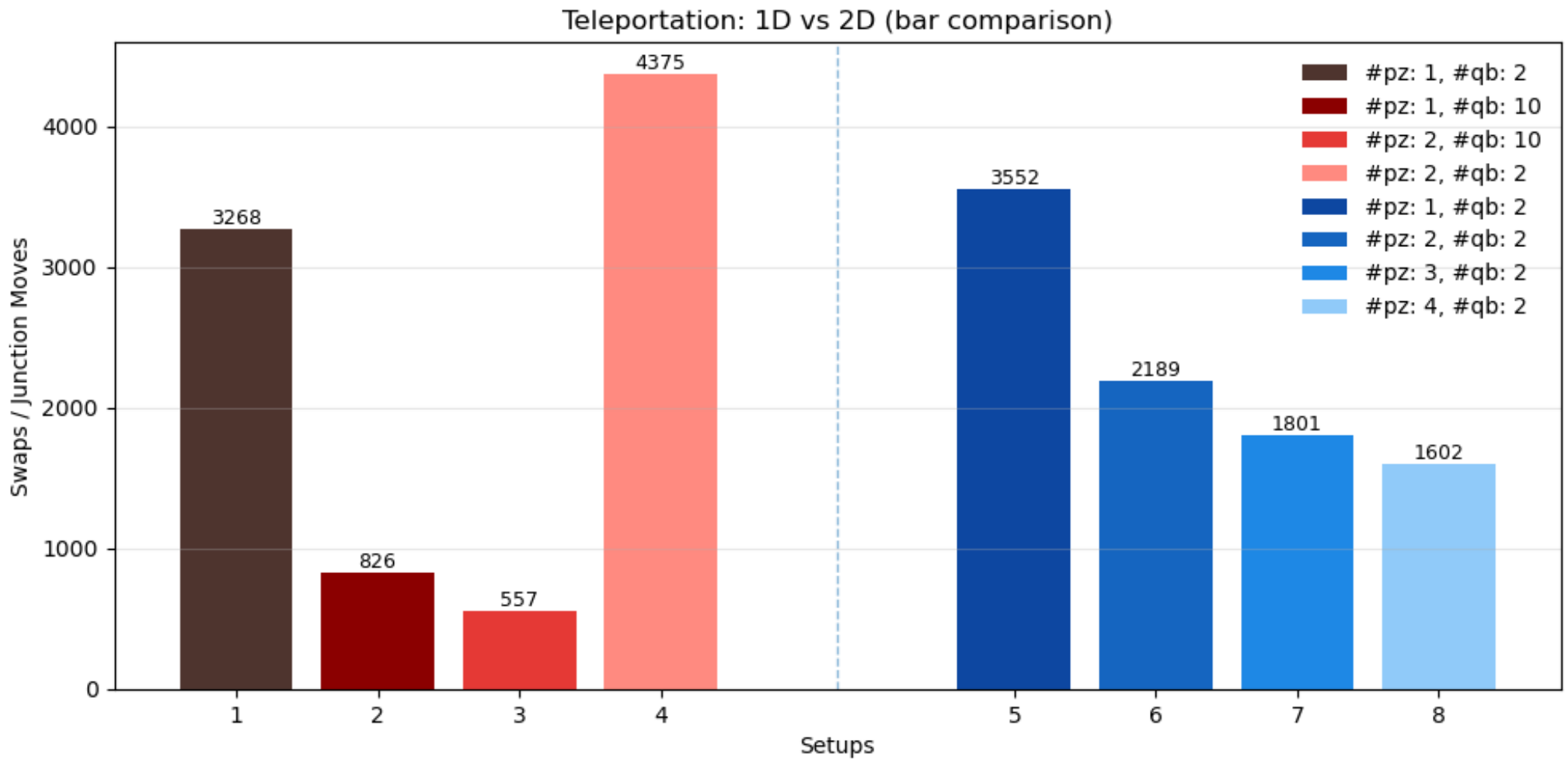


Random Circuits – Time Scaling

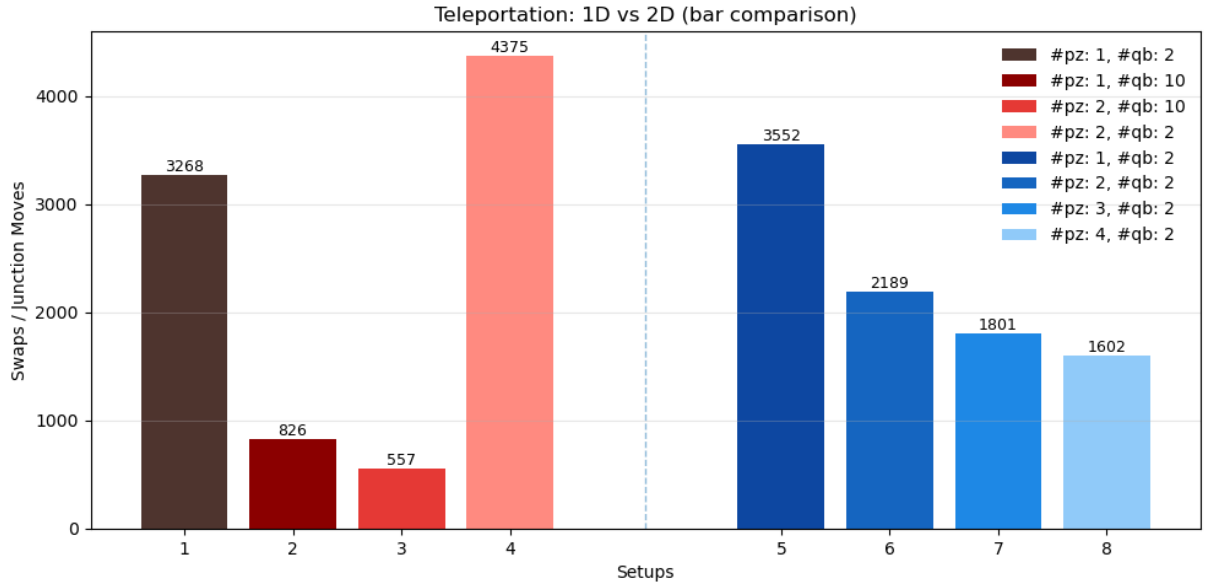


→ Two-Dimensional Setup allows for more parallelity

Logical Teleportation - Results



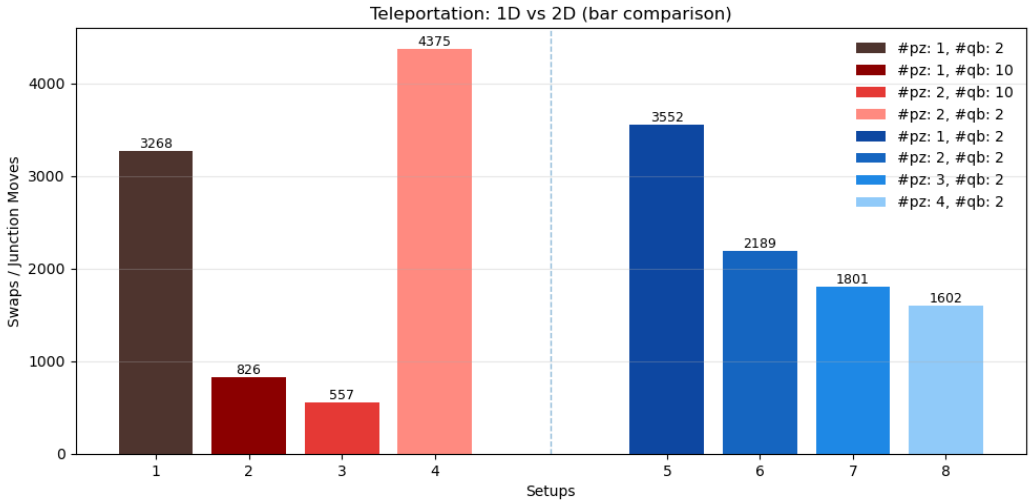
Logical Teleportation - Results

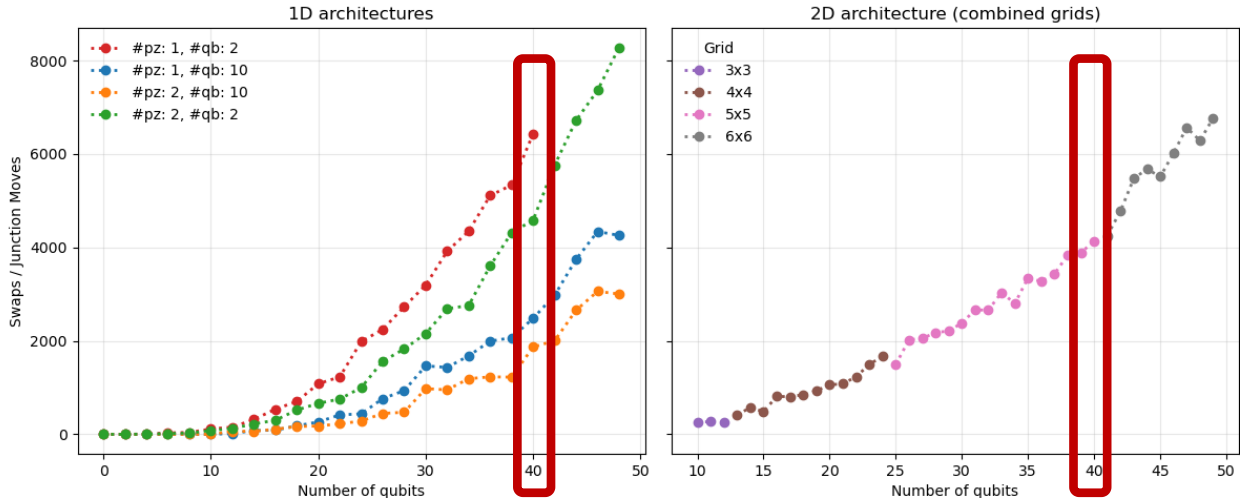


→ Comparable number of Junction Moves (2D) and SWAPs (1D)

→ Poor Comparability between metrics

Logical Teleportation - Results





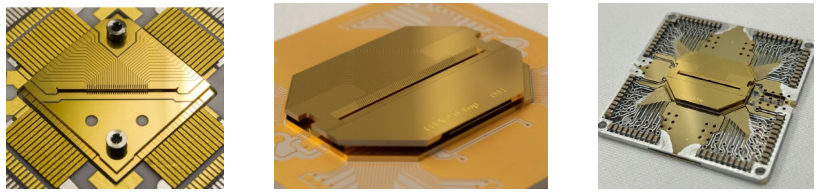
→ Comparable number of Junction Moves (2D) and SWAPs (1D)

→ Poor Comparability between metrics

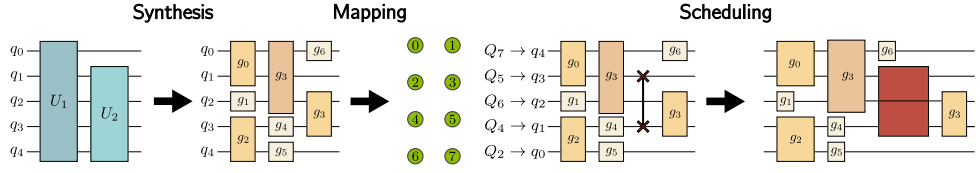
→ Structured Circuit yields lower cost for both 1D and 2D for most setups

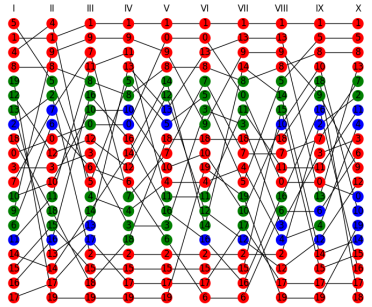
Summary

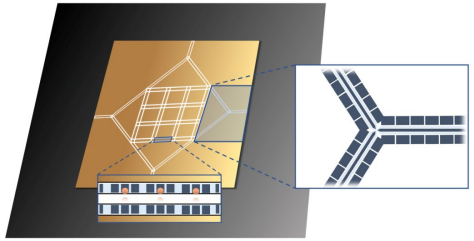
Goal: Connect computer scientists with trapped-ion quantum computing experts

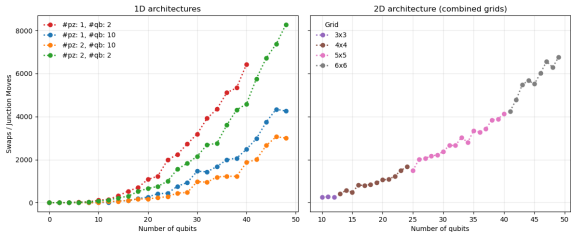


- Trapped-ion quantum computing introduction
- Compilation for trapped-ion quantum computing
- Shuttling optimisation
- Two compilers: One- and two-dimensional
- Scaling of cost metric for different architectures and different types of circuits
- Allows benchmarking architecture design without having to physically build them
  - Consider benchmarks in the building process



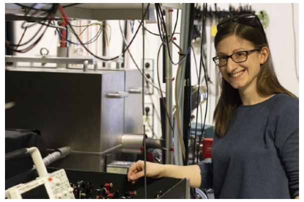


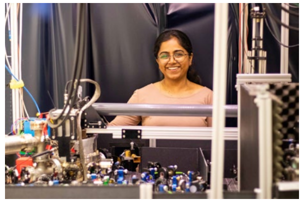
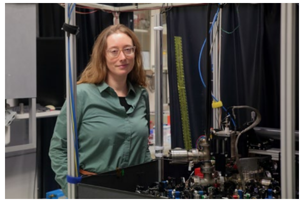
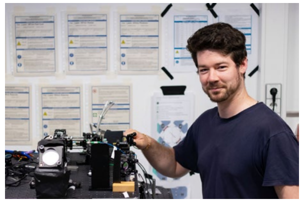
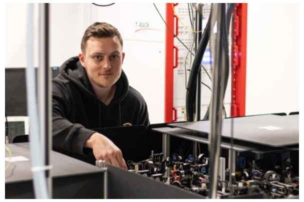


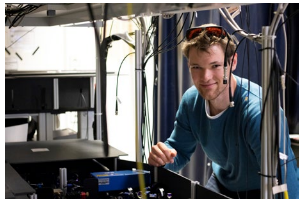
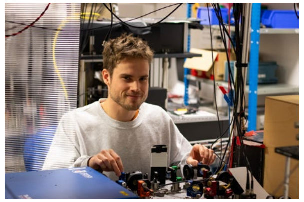


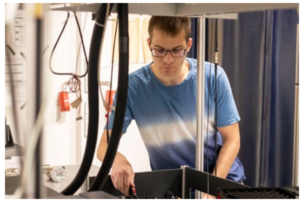
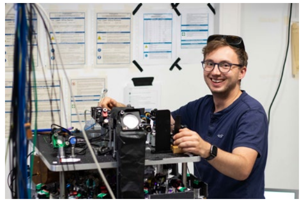
Quantum Computing Team @ JGU Mainz

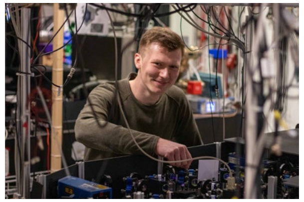


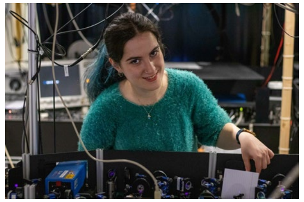
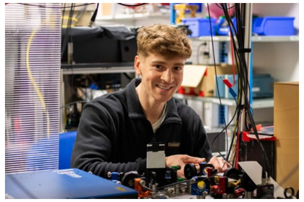
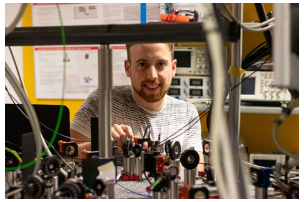


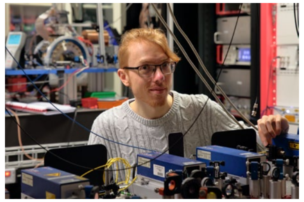
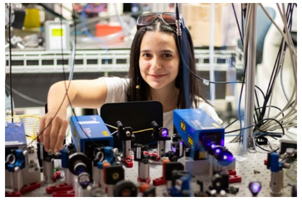





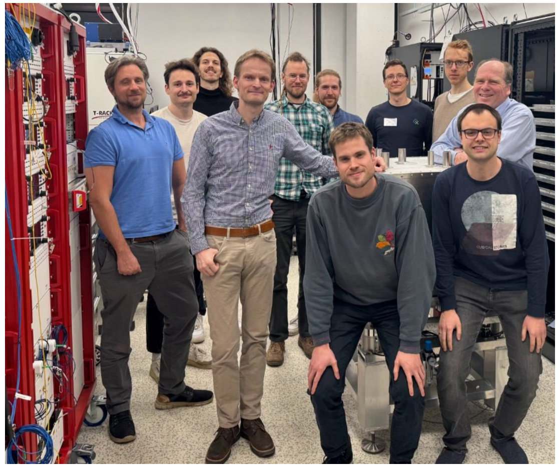
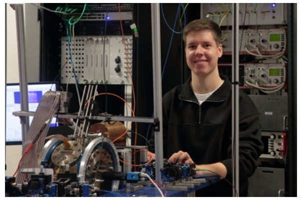
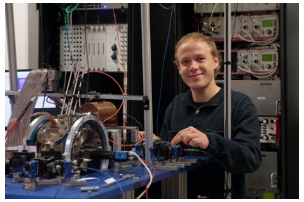
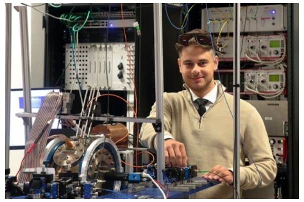
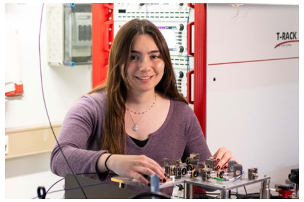





JG|U
JOHANNES GUTENBERG UNIVERSITÄT MAINZ

Technical University of Munich
TUM