Introduction to TKET

Callum Macpherson

QUANTINUUM

Quantum Software?

Qiskit

SILQ

- General purpose SDKs - qiskit, Cirq, pytket *
- Quantum Programming languages/high level languages - Q#, Silq, Quipper
- **Compiler - TKET,** qiskit, BSQKit
- Online services - AWS Braket, **Quantinuum Nexus**
- Quantum Error Correction/Mitigation- Qermit, others
- Application libraries - e.g. InQuanto, pennylane
- Simulators e.g. Qulacs, Stim

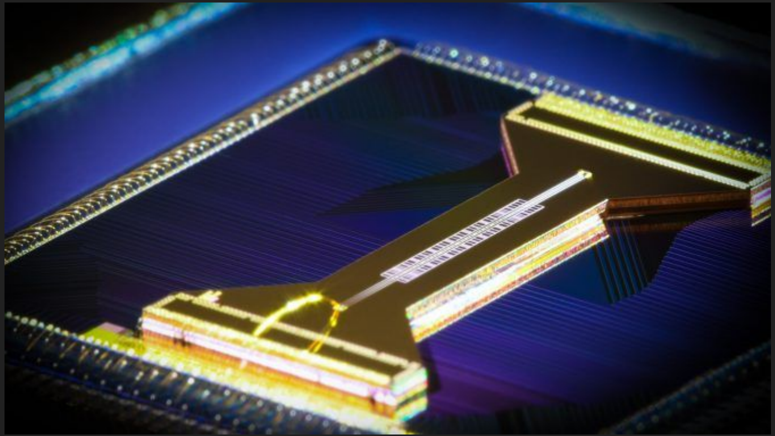
TKET

INQUANTO™ Computational Chemistry

PENNYLANE

Quantum Hardware?

- Trapped ions - **Quantinuum,** IONQ, AQT
- Superconductors - IBM, Google, Rigetti, IQM
- Photonics - PsiQuantum, Quandela…
- Neutral atoms - Pasqal, Infleqtion…
- Others - Semiconductors, topological qubits…



H-series Ion traps

Superconducting circuits - IBM

Some Challenges with Quantum computing

- Not enough qubits for many of the exciting applications
- The qubits we do have are subject to complex noise (hard to model)
- Quantum error correction at an early stage experimentally
- **Low-level details greatly influence performance -** gate count, connectivity

What is TKET?

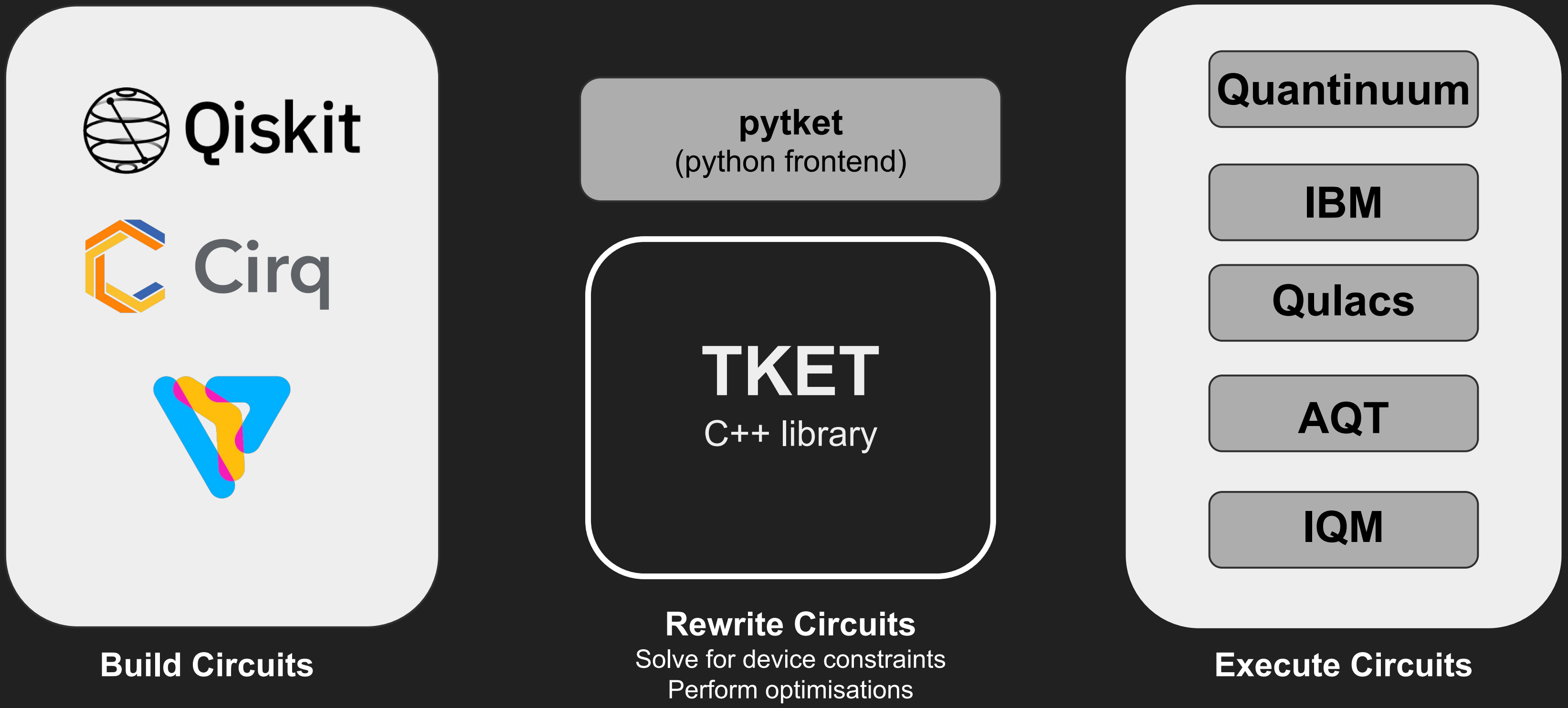
A quantum software library developed by Quantinuum

- A high performance quantum compiler
- Open source! https://github.com/CQCL/tket
- “Hardware agnostic” - Targets a range of devices and simulators
- Works with popular libraries - Qiskit, Cirq, Braket, pennylane + more

```
pip install pytket
```

TKET Architecture

Note: Cloud access through Azure and AWS Braket is also available



Qiskit

Cirq

Build Circuits

pytket
(python frontend)

TKET
C++ library

Rewrite Circuits
Solve for device constraints
Perform optimisations

Quantinuum

IBM

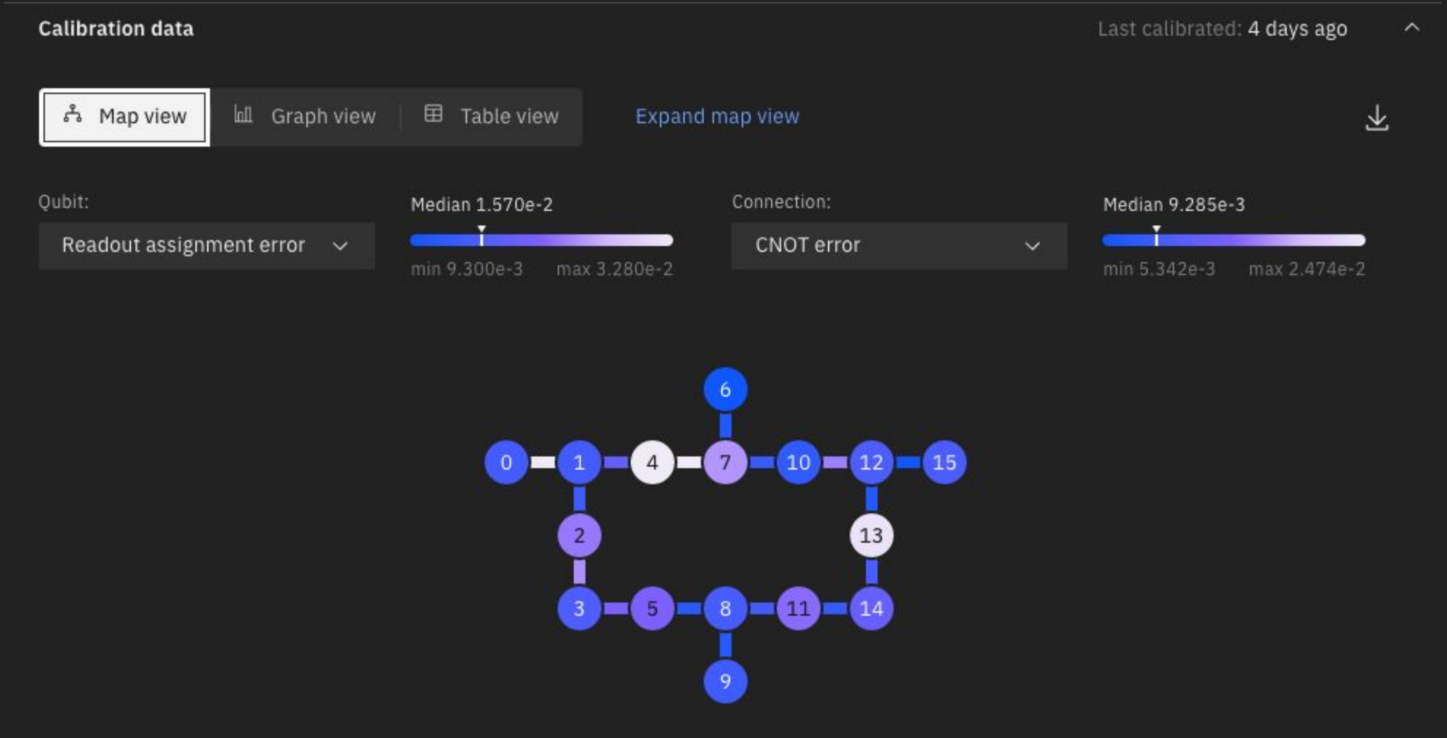
Qulacs

AQT

IQM

Execute Circuits

A Real Quantum device



Calibration data

Last calibrated: 4 days ago

Qubit: Readout assignment error

Median 1.570e-2

min 9.300e-3 max 3.280e-2

Connection: CNOT error

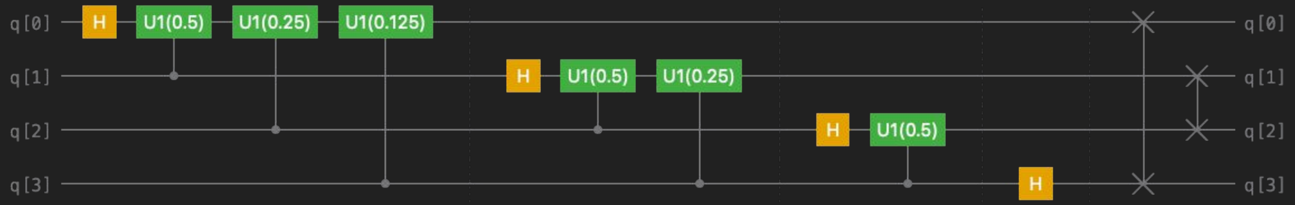
Median 9.285e-3

min 5.342e-3 max 2.474e-2

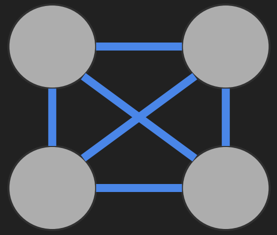
Source: IBM Quantum

Quantum compilation I

Example: Quantum Fourier Transform Circuit (Hardware-independent)

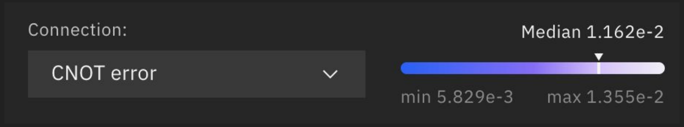


Complete connectivity graph



Target device: IBMQ Belem

- Nearest neighbour interaction only
- Limited gateset {X, SX, Rz, CX}
- CNOT error



Connection:

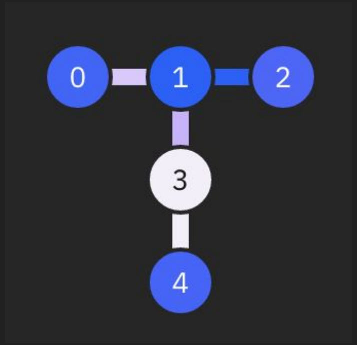
CNOT error

Median 1.162e-2

min 5.829e-3

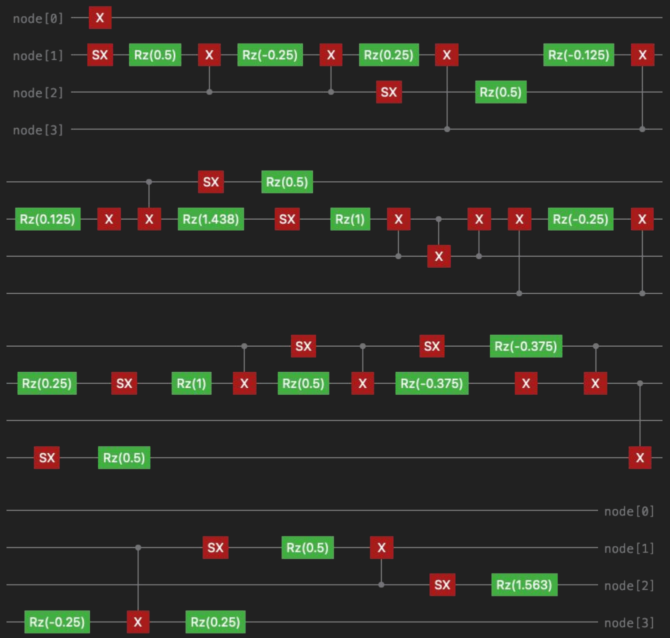
max 1.355e-2

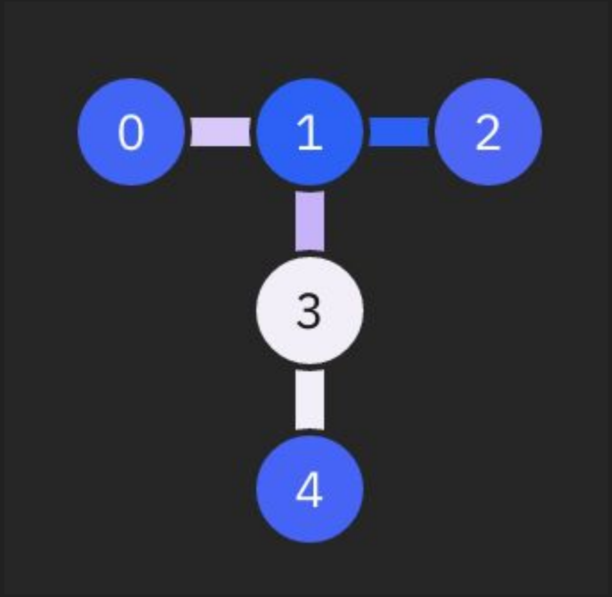
Belem qubit topology



Quantum compilation II (compiled QFT)

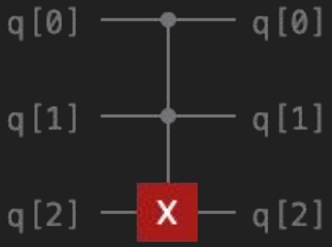
- Circuit is in IBM native gateset
- Each qubit is assigned to a physical node of the device

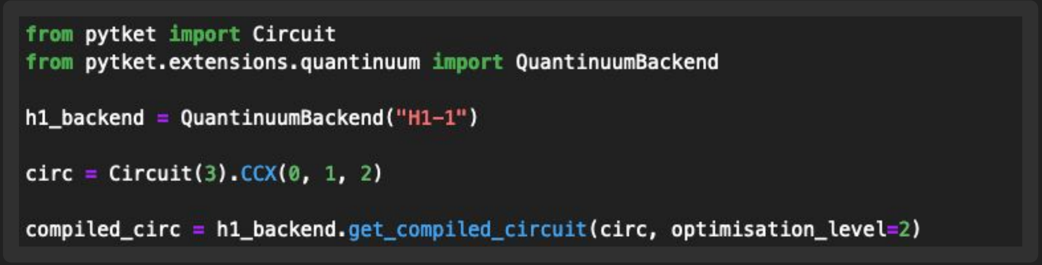




Quantum compilation III (CCX gate)

CCX gate





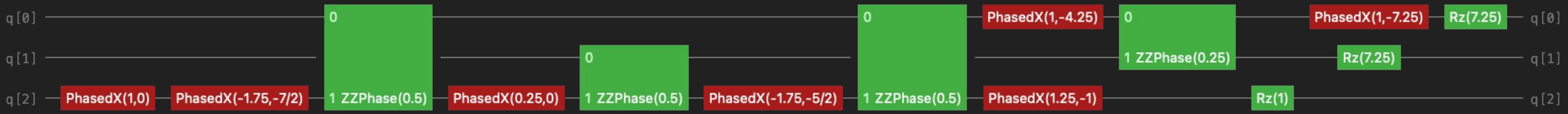
```
from pytket import Circuit
from pytket.extensions.quantinuum import QuantinuumBackend

h1_backend = QuantinuumBackend("H1-1")

circ = Circuit(3).CCX(0, 1, 2)

compiled_circ = h1_backend.get_compiled_circuit(circ, optimisation_level=2)
```

CCX gate (compiled to H-Series gateset)


q[0]
q[1]
q[2]
PhasedX(1,0)
PhasedX(-1.75,-7/2)
0
1 ZZPhase(0.5)
PhasedX(0.25,0)
0
1 ZZPhase(0.5)
PhasedX(-1.75,-5/2)
0
1 ZZPhase(0.5)
PhasedX(1,-4.25)
PhasedX(1.25,-1)
0
1 ZZPhase(0.25)
Rz(1)
PhasedX(1,-7.25)
Rz(7.25)
Rz(7.25)
q[0]
q[1]
q[2]