INTRODUCTION TO QUANTUM AND TKET

Dr. Kathrin Spendier
Quantum Evangelist
Quantinuum
October, 2023
Overview

- Introduction to Quantinuum
- Quantum Computing Hardware
- Quantum Computing Software
- Introduction to TKET
- TKET coding example
- Summary
Quantinuum at-a-glance

**WHO WE ARE**

350+
scientists and engineers
(of which > 200PhDs)

LARGEST PLAYER
in integrated quantum hardware and software

Global teams in largest quantum markets
(USA, UK, Europe, Japan)

**WHAT WE OFFER**

H-SERIES
World class QCCD, ion trap hardware

FULL STACK
InQuanto™ Quantum chemistry software. TKET™ open-source developer tool kit. LAMBEQ™ open-source natural language processing

**USER COMMUNITY**

>80
Scientific publications using H-Series hardware

>400
Global users of H-Series hardware

>1,000,000
Downloads of TKET™ open-source tool kit

Led by industry pioneers

- **Rajeeb Hazra**
  President & CEO

- **Tony Uttley**
  Founder & Chief Operating Officer

- **Ilyas Khan**
  Founder & Chief Product Officer
Platform Inclusive

INQUANTO™
Quantum Chemistry

Applied Quantum Algorithms (AQUA)
Machine Learning, Financial Modeling, Simulation, and Optimization

Quantum NLP

QUANTUM ORIGIN
Verifiably the world’s strongest encryption key generation

TKET

Microsoft
Google
OQC
IONQ

... and many more

H – Series Quantum Computers
Powered by Honeywell

© 2023 Quantinuum. All rights reserved.
QUANTUM COMPUTING HARDWARE
\[ |\psi\rangle = c_1 |0 \ldots 0\rangle + \cdots + c[2^N]|1 \ldots 1\rangle \]
What Does a Quantum Computer Look Like?

IBM

Quantinuum
Key Subsystems for H-Series Quantum Computers

- **Lasers & Optics**
  - Laser generation, timing, and frequency control
  - Precision delivery to ions

- **Control Electronics**
  - Scalable IO
  - Precision timing
  - Precision trap voltages

- **Cryogenics & Vacuum**
  - Ultra-high vacuum
  - Cryogenic temperatures (10-40K)
  - Stable magnetic fields
  - Laser access
  - Electronics pass-through

- **Control Room**
  - Operator interface
  - Real-time system info

- **Optical Imaging & Detection**
  - Imaging and Detection
  - Resolves individual ions
  - High photon sensitivity

- **Ion Trap Die**
  - Micro fabricated
  - Custom geometries
  - Tailored process
Quantinuum’s Trapped-Ion Quantum Computer

The $^2S_{1/2}$ hyperfine qubit

$|1\rangle$ F=1
$|0\rangle$ F=0

171$^{171}$Yb$^+$
138$^{138}$Ba$^+$

Gates
Cooling
Quantum Charge-Coupled Device proposal by NIST Ion Storage Group (1998)

Key technologies needed:
- **Segmented trap** supports multiple zones
- **Transport** of ions with low loss and low heating
- **Beam delivery** to multiple regions
- **Loading** large number of ions fast
- **Scalability** enabled by microfabricated traps

Additional reference:
Loading Ions

- Ytterbium atom is launched into the trap
- A laser ionizes the atom
- RF electric fields are used to create a potential well which holds the “trapped” ion
TRANSPORT PRIMITIVES

**Split and Combine**

Ion is transported into the same zone

Ions are combined into a single potential well and then re-separated

**Swap**

Ions are carefully manipulated to reorder positions
Physical Implementation

Quantum Circuit

$t=0$ Initialize qubits
$t=1$ 1-qubit gate
$t=2$ Move, join
$t=3$ 2-qubit gate
$t=4$ 2-qubit gate
$t=4$ Measure
Scaling High Quality Qubits

- **Industry Leading Performance**
  - **32 Qubits** – globally entangled at high fidelity
  - 1Q Fidelity 99.998%
  - 2Q Fidelity 99.8%
  - SPAM 99.8%
  - Memory Fidelity 99.98%
  - Measurement cross-talk error 0.0005%
  - All-to-all connectivity
  - Mid-circuit measurement with conditional logic
  - Qubit reuse
  - Parametrized angle 2Q gate included in native set

H2 ion trap

32 qubits transporting around
Roadmap for Quantum Hardware Improvement

Quantinuum's quantum computer has achieved a Quantum Volume (QV) of $2^{19}$, equivalent to 524,288.

The next closest reported QV is 512.

**Quantinuum has increased the QV by a factor of 10 every 12 months** and believes they can continue with this for the next several years.
QUANTUM COMPUTING SOFTWARE
Noisy intermediate scale quantum (NISQ)

**CONSTRATNITS:**

- 50 – 100 qubits
- High error rates
- Low coherence time
- Limited error correction
- Connectivity
- Basis gate set

**NISQ ALGORITHMS:**

Hybrid quantum + classical

Small circuit depth
Typical Quantum Algorithm Workflow on a Gate-model Quantum Computer

- **Use Case**
- **Application Software**
- **Quantum Circuit**
  - All-to-all qubit connectivity
  - Universal gateset
- **Quantum Circuit Compiler**
  - Native gateset
  - Qubit connectivity
  - Qubit error
  - Gate errors
  - More...
- **Machine Code**
- **Quantum Simulator**
- **Quantum Processor**

Quantum Computing Stack
TWO-QUBIT GATE ERRORS

\[ % \text{Error} = (100\%)(1 - \text{Fidelity}^N) \]
Example circuit optimization

For more technical detail see arXiv:2003.10611
Example routing problem

**IBM MELBOURNE**

4 qubit coupling max

**QUANTUM CIRCUIT**

5 qubit coupling
Routing

**PROBLEM**
At each time step a pair of qubits need to interact but are not adjacent.

**SOLUTION**
Insert SWAP operations to move them closer together.
Very expensive! Need to do this optimally.
Difficult combinatorial problem — NP-hard.
INTRODUCTION TO TKET
• Use gate-level quantum computers to solve my problems
• Program a quantum computer in my favorite language
• Run my quantum algorithms efficiently
• Get the most accurate result while focusing on solving my problems
TKET as a universal SDK

TKET optimizes quantum circuits, reducing the number of required operations – essential for NISQ devices.

> 1,000,000 downloads
Get started with TKET, a universal quantum SDK

TKET is available for free on GitHub (https://github.com/CQCL/pytket) and is installed by

```
pip install pytket
```

The extension module interfacing pytket with the H-series is installed by

```
pip install pytket-quantinuum
```

and for interfacing with Qiskit and IBM Quantum install

```
pip install pytket-qiskit
```
INTEGRATING OTHER QUANTUM TOOLKITS IS EASY

$qiskit\_to\_tk(circuit)$

$qiskit$ $\iff$ $TKET$

$tk\_to\_qiskit(circuit)$

Other bidirectional conversions exist: Cirq, Bracket etc.
TKET has a default pass manager for each backend

```python
getCompiledCircuit(circuit, optimization_level)
```

<table>
<thead>
<tr>
<th>Level 0</th>
<th>Level 1</th>
<th>Level 2 (default)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solves the device</td>
<td>Additionally performs some light optimizations.</td>
<td>Adds more intensive optimizations that can increase compilation time for large circuits.</td>
</tr>
<tr>
<td>constraints without</td>
<td></td>
<td></td>
</tr>
<tr>
<td>optimizing.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
TARGET A DIFFERENT Backend EASILY

```python
from pyket.extensions.qiskit import IBMQEmulatorBackend
backend = IBMQEmulatorBackend('ibmq_belem', hub='partner-cqc', group='internal', project='default')
```

```python
from pyket.extensions.quantinuum import QuantinuumBackend
backend = QuantinuumBackend('H1-1E') # emulator
```
TKET CODING EXAMPLE: GOOGLE COLAB

https://tinyurl.com/dtdrhvab
SUMMARY
# QCCD Architecture Differentiating Features

## All-to-All Connectivity

<table>
<thead>
<tr>
<th>Nearest Neighbor</th>
<th>All-to-All</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Nearest Neighbor Connectivity" /></td>
<td><img src="image2.png" alt="All-to-All Connectivity" /></td>
</tr>
</tbody>
</table>

## High-Fidelity Gates

- **Isolated Gates = Scalable Gates**

## Qubit Measurement and Reuse

**Measurement and reuse**

\[
|\psi_2\rangle \quad \text{Quantum state remains intact} \\
|\psi_1\rangle \quad \text{Measure} \\
|0\rangle \quad \text{Reinitialize} \\
c
\]

**Conditional logic**

- If \(c=1\), perform gate
- If \(c=0\), do not

\[
|\psi\rangle \quad \text{u}(\pi/2) \\
c \quad == 1
\]

### Error Rates

- 1Q infidelity \(< 3 \times 10^{-5}\)
- 2Q infidelity \(< 2 \times 10^{-3}\)
- State preparation and measurement error \(< 2 \times 10^{-3}\)
- Measurement cross-talk error \(< 5 \times 10^{-5}\)
- Memory error per qubit at average depth-1 circuit \(< 6 \times 10^{-4}\)
USE TKET!

pip install pytket

- State-of-the-art circuit compilation
- Retargetable
- Works with other libraries
- Application-specific tools
- Other software is built on top of TKET
Growing Quantum Ecosystem with Quantinuum

quantinuum.com
Quantinuum Trapped-Ion Architecture

Quantinuum’s quantum charge coupled device (QCCD) architecture

**QCCD Requirements**
1. Trap a large # of small ion crystals
2. Fast transport operations
3. Clock synchronization across trapping regions
4. Ability to trap 2 different ion species (1 for qubit, 1 for cooling)
5. Parallelization of transport and quantum operations

**Architecture Features**
- Perfect qubits (Yb+ ions)
- Dedicated interaction zones (gating, storage & auxiliary)
- Short ion chains
- High fidelity quantum gates
- Ion transport within & across zones
- Split, combine & swap primitives
- Real-time control system

**Quantum bits (qubits) are stored in the electronic states of Yb+ ions**

- $^{171}\text{Yb}^+$
  - $|1\rangle$ Hyperfine qubit
- $^{138}\text{Ba}^+$
  - $|0\rangle$ Cooling ion

© 2023 Quantinuum. All rights reserved.
HIGH FIDELITY STATE PREPARATION AND MEASUREMENT

State Preparation:

(a) Microwave-assisted optical pumping (MAOP)

1762 nm narrow-band optical pumping (NBOP)

Detection:

(c) Cabinet shelving

Trap Scaling Roadmap

2020

H1
LINEAR

H2
PARALLEL GATE ZONES

H3
GRID

H4
INTEGRATED OPTICS

H5
LARGE SCALE

2030+

Grid Trap Prototype

Photonic Waveguides & Couplers

Modular, Tiled Traps

Akhtar et al. (Sussex/Universal Quantum)
arxiv.org/pdf/2203.14062.pdf
Understanding Quantum Volume

The volume of a cube is determined by its qubit count and error rate. The greater the volume, the more powerful the machine.

The Quantum Volume Matrix

Error Rate (Y-AXIS)
An expression of how well the device can implement arbitrary pairwise interactions between qubits

Qubits (X-AXIS)
The number of qubits active in the system

Improving the error rate can increase quantum volume even when the number of qubits remains the same. For example:

<table>
<thead>
<tr>
<th>Qubits Exisitng</th>
<th>Qubits Added</th>
<th>Error Rate Decrease</th>
<th>Quantum Volume Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 Q</td>
<td>0 Q</td>
<td>10x</td>
<td>24x</td>
</tr>
</tbody>
</table>

If the error rate is high, merely adding qubits will not increase quantum volume.

<table>
<thead>
<tr>
<th>Qubits Existing</th>
<th>Qubits Added</th>
<th>Error Rate Decrease</th>
<th>Quantum Volume Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 Q</td>
<td>100 Q</td>
<td>0x</td>
<td>0x</td>
</tr>
</tbody>
</table>