Trapped-Ion Quantum Computer

Seminar: Your passion for physics

Ilja Zebergs
Heidelberg University

November 2, 2021

Historical context

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

Historical context

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

- 1981 R. Feynman: Simulating physics with computers

  "Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical ..."

Historical context

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

- 1981 R. Feynman: Simulating physics with computers
  "Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical ..."
- 1982 P. Benioff: Quantum mechanical Hamiltonian models of Turing machines

Historical context

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

- 1981 R. Feynman: Simulating physics with computers
  "Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical ..."
- 1982 P. Benioff: Quantum mechanical Hamiltonian models of Turing machines
- 1985 D. Deutsch: description of a universal quantum computer

Historical context

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

- 1981 R. Feynman: Simulating physics with computers
"Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical ..."
- 1982 P. Benioff: Quantum mechanical Hamiltonian models of Turing machines
- 1985 D. Deutsch: description of a universal quantum computer
- 1992 D. Deutsch, R. Josza: Deutsch-Josza algorithm (deterministic on QC, exponential speed-up)

Historical context

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

- 1981 R. Feynman: Simulating physics with computers
  "Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical ..."
- 1982 P. Benioff: Quantum mechanical Hamiltonian models of Turing machines
- 1985 D. Deutsch: description of a universal quantum computer
- 1992 D. Deutsch, R. Josza: Deutsch-Josza algorithm (deterministic on QC, exponential speed-up)
- 1994 P. Shor: factoring algorithm

Historical context

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

- 1981 R. Feynman: Simulating physics with computers
  "Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical ..."
- 1982 P. Benioff: Quantum mechanical Hamiltonian models of Turing machines
- 1985 D. Deutsch: description of a universal quantum computer
- 1992 D. Deutsch, R. Josza: Deutsch-Josza algorithm (deterministic on QC, exponential speed-up)
- 1994 P. Shor: factoring algorithm
- 1995 I. Cirac and P. Zoller: trapped-ion quantum computer blueprint

Goal: solve problems using a quantum mechanical system more efficiently than on a classical computer

- 1981 R. Feynman: Simulating physics with computers
  "Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical ..."
- 1982 P. Benioff: Quantum mechanical Hamiltonian models of Turing machines
- 1985 D. Deutsch: description of a universal quantum computer
- 1992 D. Deutsch, R. Josza: Deutsch-Josza algorithm (deterministic on QC, exponential speed-up)
- 1994 P. Shor: factoring algorithm
- 1995 I. Cirac and P. Zoller: trapped-ion quantum computer blueprint
- 1996 J. A. Jones and M. Mosca: first physical implementation of a quantum algorithm (NMR, 2 qbits)

Quantum computing basics

Qubits and Quantum Circuits [1]

Qubits and Quantum Circuits [1]

- q(u)bit: QM 2-level system (2D complex Hilbert space $\mathcal{H}_1$)
  general state:
  $$\alpha\,|0\rangle + \beta\,|1\rangle = \begin{bmatrix}\alpha\\ \beta\end{bmatrix} \qquad \alpha, \beta \in \mathbb{C}, \quad |\alpha|^2 + |\beta|^2 = 1$$

Qubits and Quantum Circuits [1]

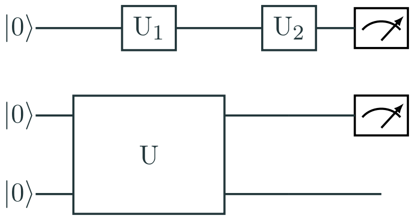
- q(u)bit: QM 2-level system (2D complex Hilbert space $\mathcal{H}_1$)
  general state:
  $$\alpha |0\rangle + \beta |1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \qquad \alpha, \beta \in \mathbb{C}, \quad |\alpha|^2 + |\beta|^2 = 1$$
- n-qbit register: $\mathcal{H}_\mathrm{n} = \mathcal{H}_1^{\otimes \mathrm{n}}$

Qubits and Quantum Circuits [1]

- q(u)bit: QM 2-level system (2D complex Hilbert space $\mathcal{H}_1$)
  general state:
  $$\alpha |0\rangle + \beta |1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \qquad \alpha, \beta \in \mathbb{C}, \quad |\alpha|^2 + |\beta|^2 = 1$$
- n-qbit register: $\mathcal{H}_\mathrm{n} = \mathcal{H}_1^{\otimes \mathrm{n}}$
- quantum gate: unitary operator on the register Hilbert space
  surjective: $\mathcal{U} |\Psi\rangle \in \mathcal{H}$
  preserves inner product: $\mathcal{U}^\dagger \mathcal{U} = 1$
  (reversible)

Qubits and Quantum Circuits [1]

- q(u)bit: QM 2-level system (2D complex Hilbert space $\mathcal{H}_1$)
  general state:
  $$\alpha\,|0\rangle + \beta\,|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \qquad \alpha, \beta \in \mathbb{C}, \qquad |\alpha|^2 + |\beta|^2 = 1$$
- n-qbit register: $\mathcal{H}_{\mathrm{n}} = \mathcal{H}_1^{\otimes \mathrm{n}}$
- quantum gate: unitary operator on the register Hilbert space
  surjective: $\mathcal{U}\,|\Psi\rangle \in \mathcal{H}$
  preserves inner product: $\mathcal{U}^\dagger \mathcal{U} = 1$
  (reversible)
- quantum circuit example



Example: Deutsch-Josza algorithm [2],[3]

- Black box function $\mathrm{f} : \{0, 1\}^{\otimes \mathrm{n}}$ (or $[0, 2^{\mathrm{n}} - 1]$) $\rightarrow \{0, 1\}$ constant or balanced (half mapped to 0 and half mapped to 1)

Example: Deutsch-Josza algorithm [2],[3]

- Black box function $\mathrm{f} : \{0,1\}^{\otimes \mathrm{n}}$ (or $[0, 2^{\mathrm{n}} - 1]$) $\rightarrow \{0,1\}$ constant or balanced (half mapped to 0 and half mapped to 1)
- Problem: determine whether f constant or balanced

Example: Deutsch-Josza algorithm [2],[3]

- Black box function $\mathrm{f} : \{0,1\}^{\otimes \mathrm{n}}$ (or $[0, 2^{\mathrm{n}} - 1]$) $\rightarrow \{0,1\}$ constant or balanced (half mapped to 0 and half mapped to 1)
- Problem: determine whether f constant or balanced
- classically: 2 to $2^{\mathrm{n}-1} + 1$ queries

Example: Deutsch-Josza algorithm [2],[3]

- Black box function $f : \{0,1\}^{\otimes n}$ (or $[0, 2^n - 1]$) $\rightarrow \{0,1\}$ constant or balanced (half mapped to 0 and half mapped to 1)
- Problem: determine whether f constant or balanced
- classically: 2 to $2^{n-1} + 1$ queries
- quantum mechanically: 1 query (exponential speed-up!)

Example: Deutsch-Joszsa algorithm [2],[3]

- Hadamard gate: $\mathrm{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

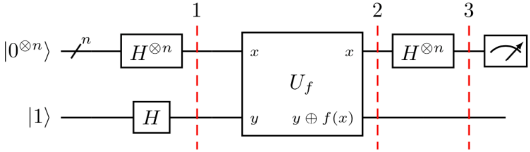
  $\mathrm{H}\left|0\right\rangle = \frac{\left|0\right\rangle + \left|1\right\rangle}{\sqrt{2}}$
  $\mathrm{H}\left|1\right\rangle = \frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}$

- Hadamard gate: $\mathrm{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

  $\mathrm{H}\left|0\right\rangle = \frac{\left|0\right\rangle + \left|1\right\rangle}{\sqrt{2}}$
  $\mathrm{H}\left|1\right\rangle = \frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}$

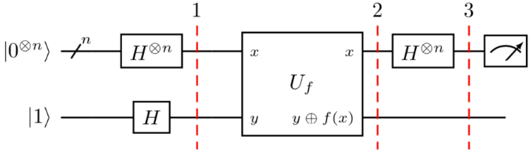
- (n+1)-qubit gate $\mathrm{U_f} : \left|\mathrm{x}, \mathrm{y}\right\rangle \mapsto \left|\mathrm{x}, \mathrm{y} \oplus \mathrm{f}(\mathrm{x})\right\rangle$
  $\mathrm{x} \in [0, 2^{\mathrm{n}} - 1]$, $\mathrm{y} \in \{0, 1\}$

Example: Deutsch-Joszsa algorithm [2],[3]



Source: [2]

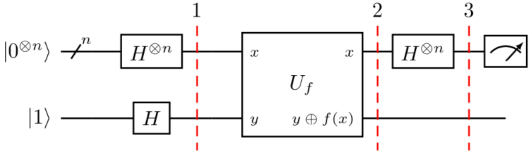
Example: Deutsch-Joszsa algorithm [2],[3]



Source: [2]

1. $\frac{1}{\sqrt{2^{\mathrm{n}+1}}} \sum_{\mathrm{x}=0}^{2^{\mathrm{n}}-1} |\mathrm{x}\rangle \left(|0\rangle - |1\rangle\right)$

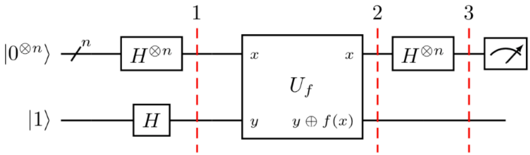
Example: Deutsch-Joszsa algorithm [2],[3]



Source: [2]

1. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle \left(|0\rangle - |1\rangle\right)$
2. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle \left(|f(x)\rangle - |1 \oplus f(x)\rangle\right) =$
   $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle \left(|0\rangle - |1\rangle\right)$

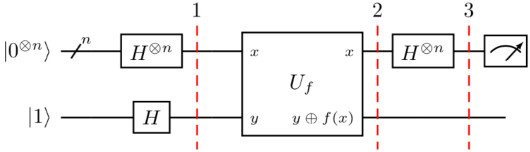
Example: Deutsch-Joszsa algorithm [2],[3]



Source: [2]

1. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle \left(|0\rangle - |1\rangle\right)$
2. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle \left(|f(x)\rangle - |1 \oplus f(x)\rangle\right) =$
   $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle \left(|0\rangle - |1\rangle\right)$
3. $\frac{1}{\sqrt{2^{n+1}}} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x \cdot y} |y\rangle \left(|0\rangle - |1\rangle\right)$
   with $x \cdot y = x_1 y_1 \oplus x_2 y_2 \oplus \cdots \oplus x_n y_n$

Example: Deutsch-Joszsa algorithm [2],[3]

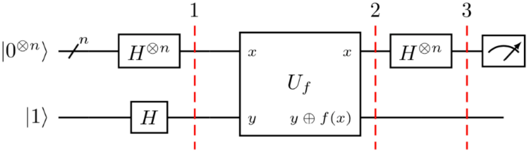


Source: [2]

1. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle (|0\rangle - |1\rangle)$
2. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle (|f(x)\rangle - |1 \oplus f(x)\rangle) =$
   $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle (|0\rangle - |1\rangle)$
3. $\frac{1}{\sqrt{2^{n+1}}} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x \cdot y} |y\rangle (|0\rangle - |1\rangle)$
   with $x \cdot y = x_1 y_1 \oplus x_2 y_2 \oplus \cdots \oplus x_n y_n$

   $P(|0\rangle^{\otimes n}) = [\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}]^2$

Example: Deutsch-Joszsa algorithm [2],[3]



Source: [2]

1. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle (|0\rangle - |1\rangle)$
2. $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle (|f(x)\rangle - |1 \oplus f(x)\rangle) =$
   $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle (|0\rangle - |1\rangle)$
3. $\frac{1}{\sqrt{2^{n+1}}} \sum_{y=0}^{2^n-1} \sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x \cdot y} |y\rangle (|0\rangle - |1\rangle)$
   with $x \cdot y = x_1 y_1 \oplus x_2 y_2 \oplus \cdots \oplus x_n y_n$

   $P(|0\rangle^{\otimes n}) = [\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}]^2$

   f constant $\implies P(|0\rangle^{\otimes n}) = 1$ f balanced $\implies P(|0\rangle^{\otimes n}) = 0$

Other algorithms [1]

- Shor's algorithm: factoring (RSA) (exponential speed-up)

Other algorithms [1]

- Shor's algorithm: factoring (RSA) (exponential speed-up)
- Grover's search: unstructured database search (quadratic speedup)

Other algorithms [1]

- Shor's algorithm: factoring (RSA) (exponential speed-up)
- Grover's search: unstructured database search (quadratic speedup)
- Quantum simulations

Other algorithms [1]

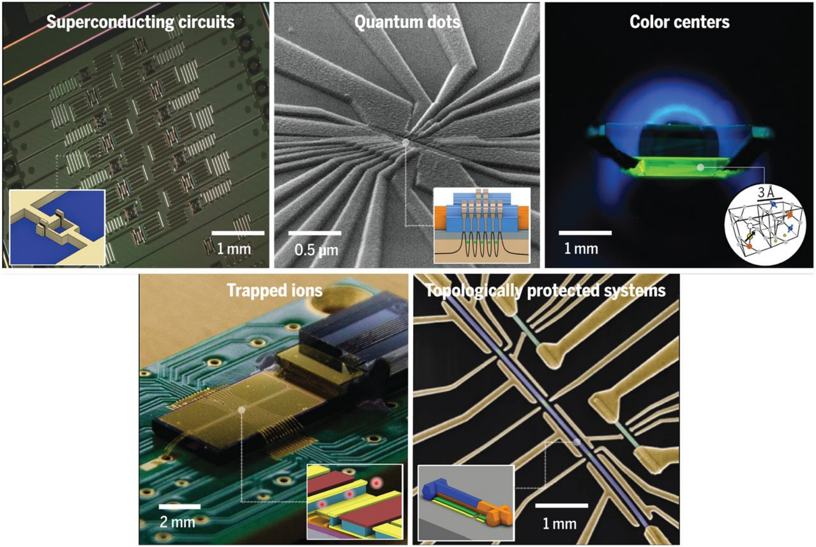
- Shor's algorithm: factoring (RSA) (exponential speed-up)
- Grover's search: unstructured database search (quadratic speedup)
- Quantum simulations
- Hybrid algorithms

Other algorithms [1]

- Shor's algorithm: factoring (RSA) (exponential speed-up)
- Grover's search: unstructured database search (quadratic speedup)
- Quantum simulations
- Hybrid algorithms
- Error correction

Trapped-Ion Quantum Computer

Different existing platforms



Source: DOI: 10.1126/science.abb2823

What should hardware be capable of?
DiVincenzo criteria [4]

What should hardware be capable of?
DiVincenzo criteria [4]

1. A scalable physical system with well characterized qubits

What should hardware be capable of? DiVincenzo criteria [4]

1. A scalable physical system with well characterized qubits
2. The ability to initialize the state of the qubits to a simple fiducial state, such as $|00\ldots0\rangle$

What should hardware be capable of?
DiVincenzo criteria [4]

1. A scalable physical system with well characterized qubits
2. The ability to initialize the state of the qubits to a simple fiducial state, such as $|00\ldots0\rangle$
3. Long relevant decoherence times, much longer than the gate operation time

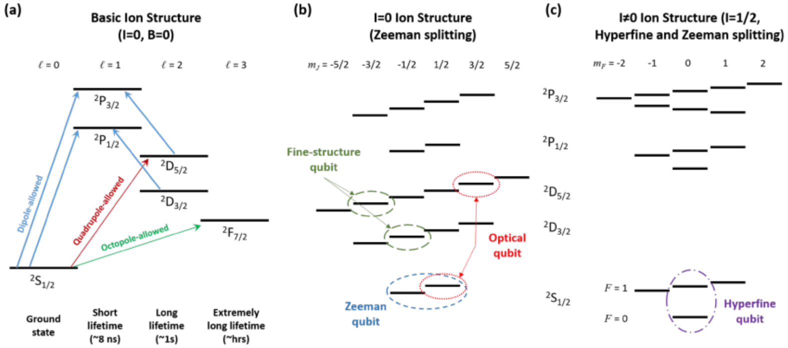
What should hardware be capable of?
DiVincenzo criteria [4]

1. A scalable physical system with well characterized qubits
2. The ability to initialize the state of the qubits to a simple fiducial state, such as $|00\ldots0\rangle$
3. Long relevant decoherence times, much longer than the gate operation time
4. A "universal" set of quantum gates

What should hardware be capable of?
DiVincenzo criteria [4]

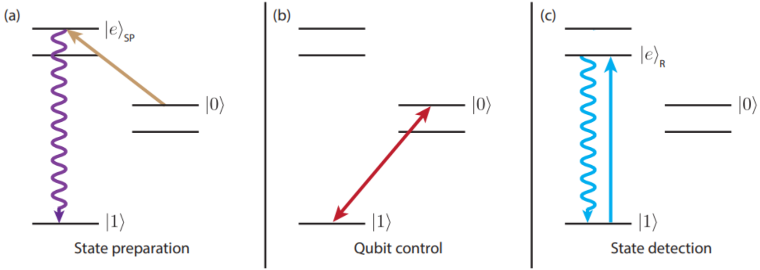
1. A scalable physical system with well characterized qubits
2. The ability to initialize the state of the qubits to a simple fiducial state, such as $|00\ldots0\rangle$
3. Long relevant decoherence times, much longer than the gate operation time
4. A "universal" set of quantum gates
5. A qubit-specific measurement capability

Physical qubits in ions [5]



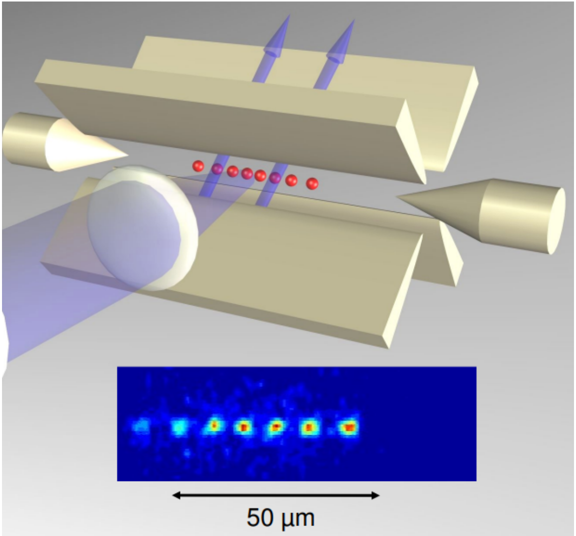
Level structure in singly-ionized earth-alkali and typical qubit choices. Source: [5]

Preparation, Control and Read-out



Source: [5]

Trapping ions



Linear Paul trap. Source: [6]

Coupling different states: typical realizations [5]

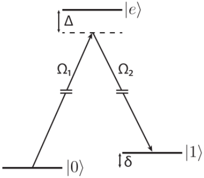
- Optical gates: resonant coupling with monochromatic laser (e.g. electric quadrupole coupling)

Coupling different states: typical realizations [5]

- Optical gates: resonant coupling with monochromatic laser (e.g. electric quadrupole coupling)
- Microwave gates: use antennae to generate EM field

Coupling different states: typical realizations [5]

- Optical gates: resonant coupling with monochromatic laser (e.g. electric quadrupole coupling)
- Microwave gates: use antennae to generate EM field
- Raman gates: stimulated Raman coupling using 2 lasers



Source: [5]

Single-qubit gates: Rabi oscillation [1],[6]

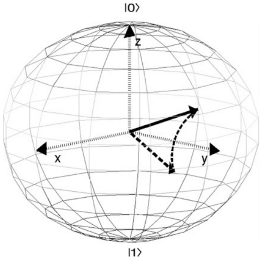
$$\mathrm{R}(\theta,\phi) = \exp(\mathrm{i}\tfrac{\theta}{2}(\mathrm{e}^{\mathrm{i}\phi}\sigma_+ + \mathrm{e}^{-\mathrm{i}\phi}\sigma_-)) = \begin{bmatrix} \cos\frac{\theta}{2} & \mathrm{i}\mathrm{e}^{\mathrm{i}\phi}\sin\frac{\theta}{2} \\ \mathrm{i}\mathrm{e}^{-\mathrm{i}\phi}\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}$$

$\theta = \Omega_{\mathrm{Rabi}}\tau$ - pulse area, ϕ - laser field phase

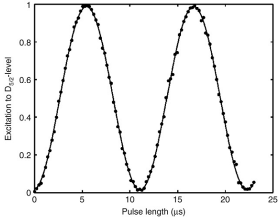
Single-qubit gates: Rabi oscillation [1],[6]

$$\mathrm{R}(\theta,\phi) = \exp(\mathrm{i}\tfrac{\theta}{2}(\mathrm{e}^{\mathrm{i}\phi}\sigma_+ + \mathrm{e}^{-\mathrm{i}\phi}\sigma_-)) = \begin{bmatrix} \cos\frac{\theta}{2} & \mathrm{i}\mathrm{e}^{\mathrm{i}\phi}\sin\frac{\theta}{2} \\ \mathrm{i}\mathrm{e}^{-\mathrm{i}\phi}\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}$$

$\theta = \Omega_{\mathrm{Rabi}}\tau$ - pulse area, ϕ - laser field phase



(c) Rotation around y-axis on Bloch sphere



(d) Rabi oscillation in Ca^+ ion (1000 measurements per point)

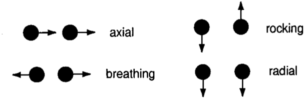
Source: [6]

Entangling gate: Cirac-Zoller gate [5],[7]

- Use Coulomb interaction to create entanglement

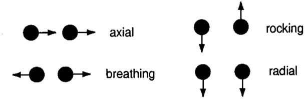
Entangling gate: Cirac-Zoller gate [5],[7]

- Use Coulomb interaction to create entanglement
- Normal modes of longitudinal motion:
  centre-of-mass (CM), breathing etc.



Entangling gate: Cirac-Zoller gate [5],[7]

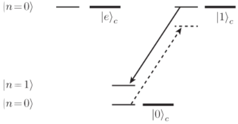
- Use Coulomb interaction to create entanglement
- Normal modes of longitudinal motion:
  centre-of-mass (CM), breathing etc.



- Preparation: cool the CM mode to the ground state

Entangling gate: Cirac-Zoller gate [5],[7]

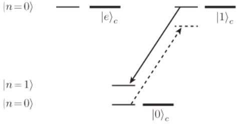
1. Encode first qubit to CM motional state



Source:[5]

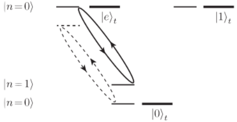
Entangling gate: Cirac-Zoller gate [5],[7]

1. Encode first qubit to CM motional state

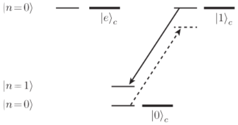


Source:[5]

2. Add phase to the second qubit conditioned on motional state

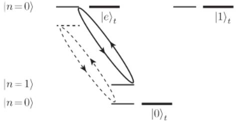


1. Encode first qubit to CM motional state

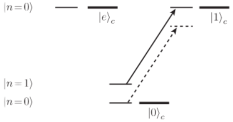


Source:[5]

2. Add phase to the second qubit conditioned on motional state

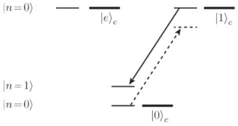


3. Decode CM motional state to first qubit state



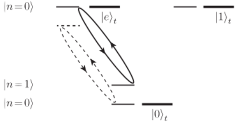
Entangling gate: Cirac-Zoller gate [5],[7]

1. Encode first qubit to CM motional state

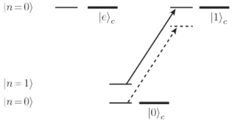


Source:[5]

2. Add phase to the second qubit conditioned on motional state



3. Decode CM motional state to first qubit state



Overall: $|11\rangle \mapsto -|11\rangle$; $\quad |00\rangle, |01\rangle, |10\rangle$ unaltered

Example: IonQ 11 qubit QC [8]

- Qubits: $|0\rangle = |\mathrm{F} = 0\rangle$ and $|1\rangle = |\mathrm{F} = 1\rangle$ of ${}^2\mathrm{S}_{1/2}$ in ${}^{171}\mathrm{Yb}^+$

Example: IonQ 11 qubit QC [8]

- Qubits: $|0\rangle = |\mathrm{F} = 0\rangle$ and $|1\rangle = |\mathrm{F} = 1\rangle$ of ${}^2\mathrm{S}_{1/2}$ in ${}^{171}\mathrm{Yb}^+$
- Linear surface RF-trap

Example: IonQ 11 qubit QC [8]

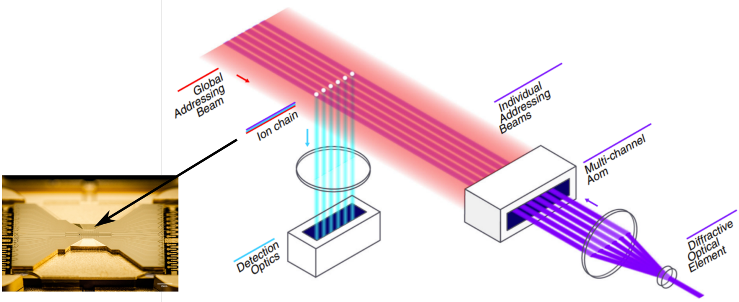
- Qubits: $|0\rangle = |\mathrm{F} = 0\rangle$ and $|1\rangle = |\mathrm{F} = 1\rangle$ of ${}^2\mathrm{S}_{1/2}$ in ${}^{171}\mathrm{Yb}^+$
- Linear surface RF-trap
- Raman (2-photon) coupling

Example: IonQ 11 qubit QC [8]

- Qubits: $|0\rangle = |\mathrm{F} = 0\rangle$ and $|1\rangle = |\mathrm{F} = 1\rangle$ of $^2\mathrm{S}_{1/2}$ in $^{171}\mathrm{Yb}^+$
- Linear surface RF-trap
- Raman (2-photon) coupling
- Entangling mediated via transverse motion

Example: IonQ 11 qubit QC [8]

- Qubits: $|0\rangle = |\mathrm{F} = 0\rangle$ and $|1\rangle = |\mathrm{F} = 1\rangle$ of $^2\mathrm{S}_{1/2}$ in $^{171}\mathrm{Yb}^+$
- Linear surface RF-trap
- Raman (2-photon) coupling
- Entangling mediated via transverse motion



Source: [8], IonQ website

Technical progress [5]

- Fully connected register size up to 20 qubits (2018)
- Coherence time: up to several minutes
- Single-qubit gate time: $\sim 1\ \mu$s
- Single-qubit gate fidelity: $\sim 0.999 \ldots 0.999999$
- Two-qubit gate time: $\sim 10\ \mu$s
- Two-qubit gate fidelity: $\sim 0.98 \ldots 0.998$

Recap: Recipe for quantum supremacy

1. Find a hard problem for a classical computer and
   a quantum algorithm giving you a relevant speed-up
2. Construct a quantum system with well-characterized qubit states
   (trap and cool ions, choose qubit states)
3. Implement control: initialization, 1-qubit gates, 2-qubit entangling gate
   (laser or microwave pulses)
4. Implement a quantum mechanical observer/measurement
   (state-dependent fluorescence)
5. Optimize and hope for the best

Literature

[1] Nielsen, M., Chuang, I. (2010). Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511976667

[2] Qiskit textbook. https://qiskit.org/textbook/preface.html

[3] Deutsch, D., Jozsa, R. (1992) Rapid solution of problems by quantum computation. Proc. R. Soc. Lond. A439553558. doi:10.1098/rspa.1992.0167

[4] DiVincenzo, D. P. (2000) The Physical Implementation of Quantum Computation. Fortschr. Phys. 48 9-11. doi: 10.1002/1521-3978(200009)48:9/11<771::AID-PROP771>3.0.CO;2-E

[5] Bruzewicz et al. (2019) Trapped-ion quantum computing: Progress and challenges. Appl. Phys. Rev. 6, 021314. doi: 10.1063/1.5088164

[6] Häffner, H. et al. (2008) Quantum computing with trapped ions. J. Phys. Rep. 469 155-203. doi:10.1016/j.physrep.2008.09.003

[7] Cirac, J.I., Zoller, P. (1994) Quantum computations with cold trapped ions. Phys. Rev. Lett. 74, 4091. doi:10.1103/PhysRevLett.74.4091

[8] Wright, K. et al. (2019) Benchmarking an 11-qubit quantum computer. Nature Communications. doi:10.1038/s41467-019-13534-2