



UNIVERSITÀ  
DI CAMERINO



Consiglio Nazionale  
delle Ricerche



INO-CNR  
ISTITUTO  
NAZIONALE DI  
OTTICA

# Quantum computing

David Vitali

Physics Division, School of Science and  
Technology

University of Camerino



Hybrid Optomechanical  
Technologies



- **Informazione classica**: digitalizzata in **bit**, sistemi fisici che possono assumere solo due stati, 0 e 1
- $n$  bit  $\Rightarrow 2^n$  stati per immagazzinare ed elaborare dati
- **Informazione quantistica**: digitalizzata in **qubit**, sistemi fisici che possono assumere infiniti stati,  $a|0\rangle + b|1\rangle$ ,  $|a|^2 + |b|^2 = 1$  (grazie al principio di sovrapposizione)
- $n$  qubit  $\Rightarrow$  **infinità di stati** per immagazzinare ed elaborare dati  
 $\Rightarrow$  **calcolo infinitamente parallelo**

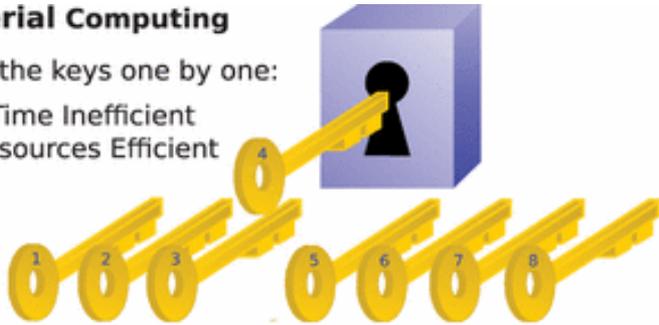
**Computer quantistici (Benioff 1980, Feynman 1982, Deutsch 1985)**: insiemi di qubit che evolvono in maniera quantistica, **con** trascurabile influenza esterna.

**Evoluzione = calcolo = realizzazione di un algoritmo quantistico**

### Serial Computing

Try all the keys one by one:

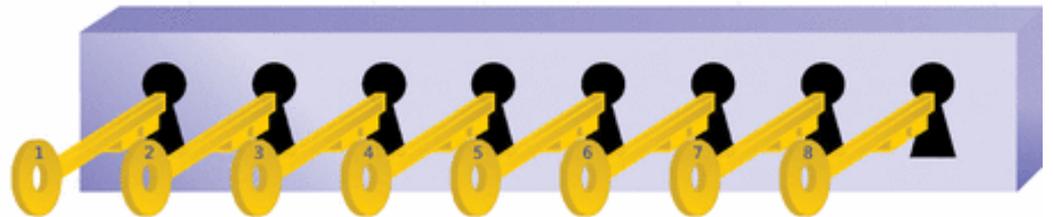
Time Inefficient  
Resources Efficient



### Quantum Computing

Try all the keys in parallel to a single box:

Time Efficient  
Resources Efficient



### Parallel Computing

Create as many boxes as the keys and try all the keys in parallel:

Time Efficient  
Resources Inefficient



# Proprietà fondamentali di un QC

- Risolve gli stessi problemi di un computer classico, e soddisfa l'ipotesi di Church-Turing
- Però svolge **alcuni algoritmi** (**fattorizzazione di Shor 1994, simulazione di un sistema fisico, sistema di equazioni lineari (HHL)**) molto più velocemente di qualsiasi computer classico (**speed-up** esponenziale)
- Per fattorizzare un numero di 300 cifre un computer classico impiega miliardi di anni, mentre un computer quantistico impiegherebbe circa un mese
- **ricerca in una base non strutturata di N dati**: Computer classico:  $N/2$  passi (in media); computer quantistico:  $\sqrt{N}$  passi con l'algoritmo di Grover (1996)

# Principale nemico: Errori causati da effetti esterni (“decoerenza”) = distruzione della coerenza del calcolo quantistico

- Eliminarla completamente è impossibile; è caratterizzata da una scala temporale  $t_{dec}$ : l'importante è che  $t_{dec} \gg t_{gate}$
- Vari metodi per controllarla: il principale è la **quantum error correction**: **codifica** di qubit in spazi a dimensione maggiore; 1 qubit -> più qubit

$$|0\rangle \rightarrow |0\rangle|0\rangle|0\rangle \quad |1\rangle \rightarrow |1\rangle|1\rangle|1\rangle$$

# Error correction e fault-tolerance

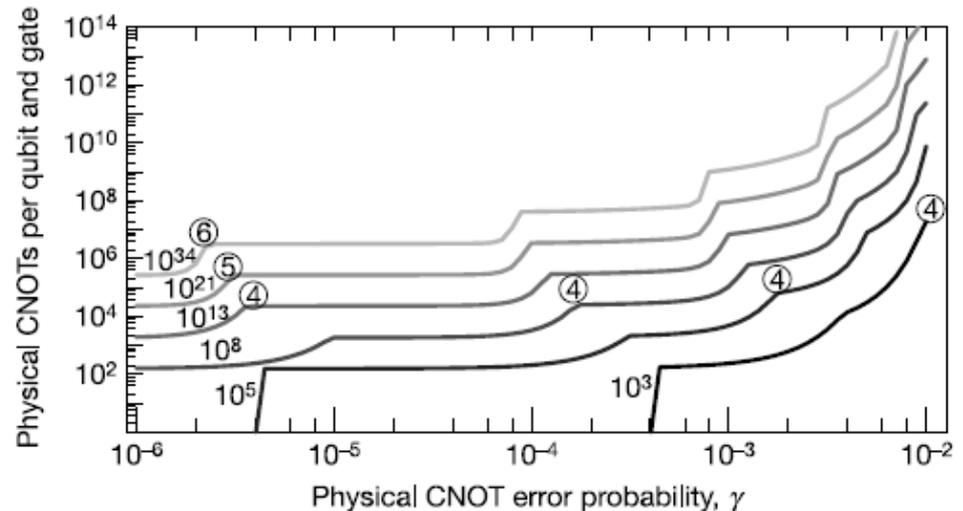
- Quantum error correction (QEC) simile a quella classica: usa la ridondanza (**bit**: bit-flip unico tipo di errore; **qubit**: bit-flip + phase-flip): codifica minima 1 qubit  $\rightarrow$  5 qubit
- QEC permette **fault-tolerant quantum computation** mediante **encoding concatenati**

$$|0\rangle_L = |0\rangle|0\rangle|0\rangle \quad |0\rangle_L^2 = |0\rangle_L|0\rangle_L|0\rangle_L$$

# Teorema della soglia

- Esiste una soglia di errore per gate (EPG) al di sotto della quale **è possibile avere un QC funzionante anche in presenza di errori** (decoerenza) (**Fault-tolerant QC**, Preskill, Aharonov, Knill...)

**Soglia:  $EPG \sim 10^{-2}$**  (Knill 2005,,  
ma con livelli di  
concatenazione = numero di  
qubit esponenziale con EPG



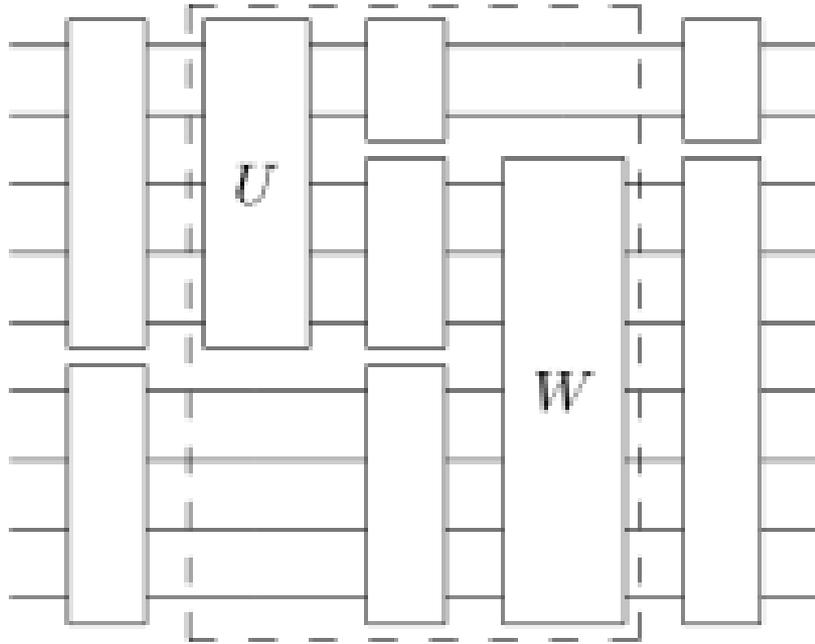
Fowler 2009, surface codes hanno un overhead di risorse più piccolo. **Sotto soglia un QC corregge errori più velocemente di quanto ne genera.**

Analogo al teorema di von Neumann del caso classico

# Modelli di quantum computation

- **Rete di porte logiche** a pochi qubit
- **One-way quantum computer** (calcolo = misure su qubit su uno stato iniziale opportuno = cluster state)
- **Adiabatic quantum computer** or computer basato su “Quantum annealing” (calcolo = continua trasformazione di un sistema in modo che cambi in modo continuo il suo stato fondamentale)
- **Topological quantum computer** (calcolo = braiding di anyons in un sistema bidimensionale)

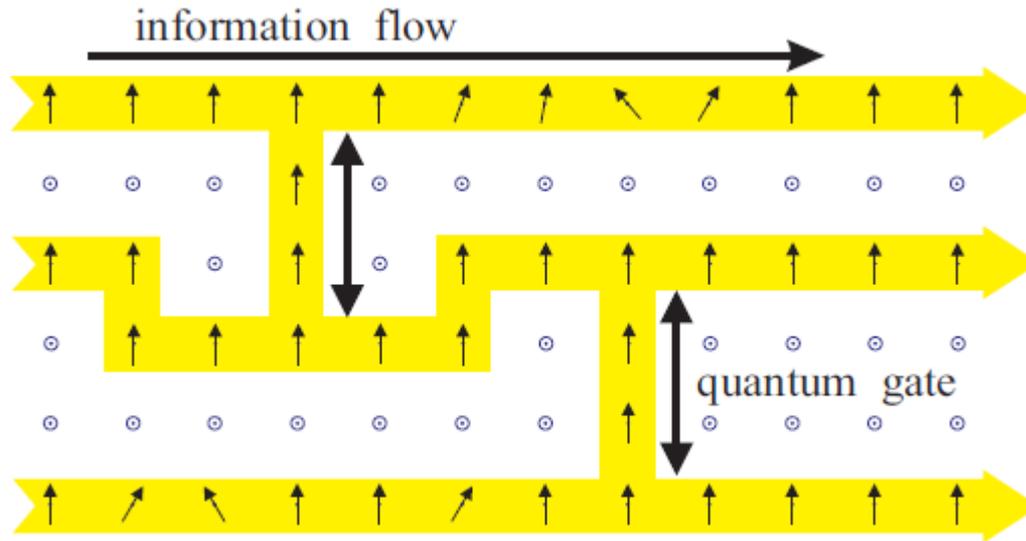
# Modelli di QC: array di gates (porte)



Tradizionale array di gate a due o più qubits

Un circuito arbitrario può essere sempre scomposto in modo efficiente e con precisione arbitraria **con un numero finito di elementi di un set universale** (es. una sola porta a due qubit universale (CNOT) e tutti i gate a singolo qubit)

# One-way QC (o measurement-based QC)



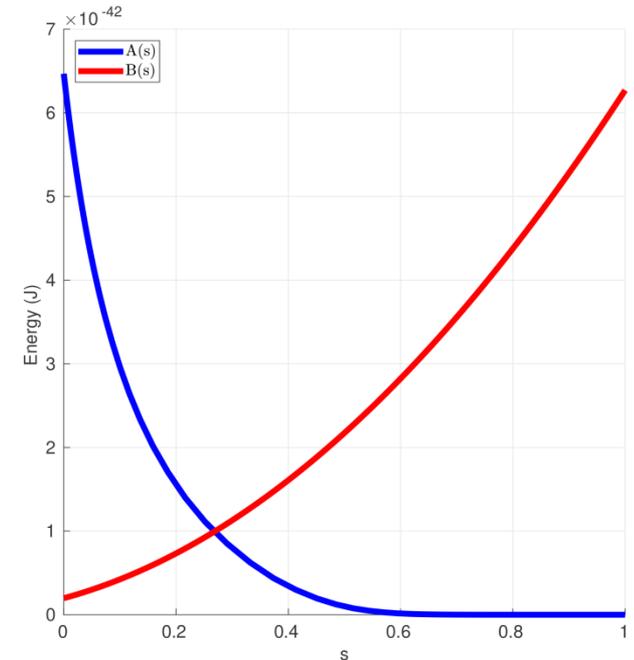
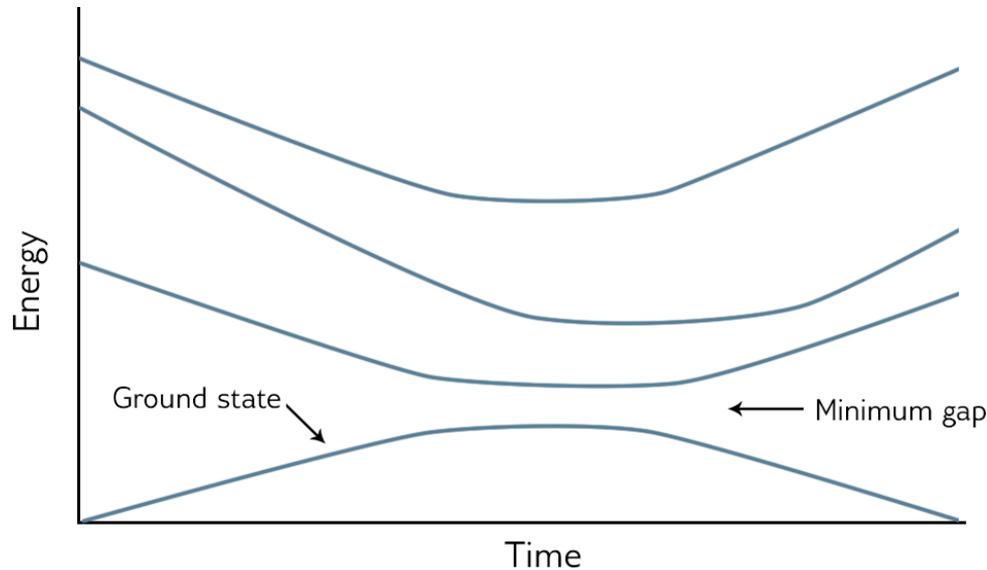
Stati particolari di un reticolo (Graph states o cluster states)  
**soggetti a sequenze di misure a singolo qubit** (in generale condizionate dal risultato delle misure precedenti)

Il programma è la sequenza di misure: la “risorsa” è il graph state iniziale preparato altamente “entangled”

# Adiabatic quantum computation/Quantum annealing

- Calcolo = evoluzione da stato iniziale a stato finale desiderato, realizzato come **evoluzione adiabatica sempre nello stato di equilibrio ad energia minima del processore (teorema adiabatico)** (Fahri-Goldston, 2000)
- In principio immune a decoerenza perché nello stato fondamentale, ma con tanti qubit gli stati eccitati possono essere esponenzialmente vicini

# Adiabatic QC



$$\mathcal{H}_{ising} = \underbrace{-\frac{A(s)}{2} \left( \sum_i \hat{\sigma}_x^{(i)} \right)}_{\text{Initial Hamiltonian}} + \underbrace{\frac{B(s)}{2} \left( \sum_i h_i \hat{\sigma}_z^{(i)} + \sum_{i>j} J_{i,j} \hat{\sigma}_z^{(i)} \hat{\sigma}_z^{(j)} \right)}_{\text{Final Hamiltonian}}$$

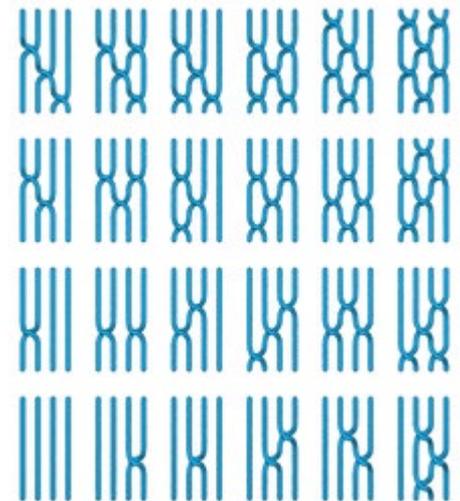
Caso specifico del QC Dwave, quantum annealer a disposizione del CINECA

# Topological quantum computer

- Gate = “treccia” (braid) tra due particelle “anyons” in un sistema 2D
- Intrinsecamente robusto rispetto alla decoerenza perché il gate dipende solo proprietà topologiche della treccia e non dalle perturbazioni

Realizzabile in semiconduttori GaAs 2D  
in campo magnetico.

Schema perseguito da Microsoft (Majorana Fermions) ma  
tuttora poco chiara la realizzabilità

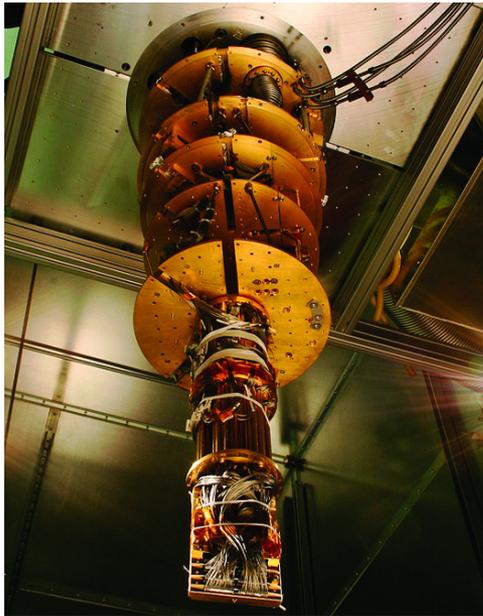


# Principali implementazioni/prototipi

1. Dispositivi superconduttori (superconducting QC): IBM, Google, Rigetti, Amazon (?), D-Wave\*\* (annealer)
2. Trapped ion quantum computer (Ion-Q- Honeywell, AQT Innsbuck)
3. Linear optics quantum computer su chip al silicio (Psi-Quantum, Xanadu)
4. Atomi neutri (Rydberg) in trappole ottiche (Pasqal)

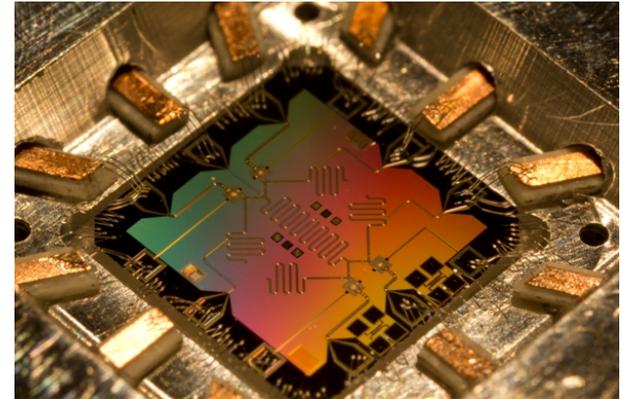
# Superconducting Quantum computer

- Usa dispositivi su chip superconduttori a basse T (0.01 K)
- Gate con controlli elettrici, magnetici e microonde
- E' scalabile, ma non funziona a T ambiente

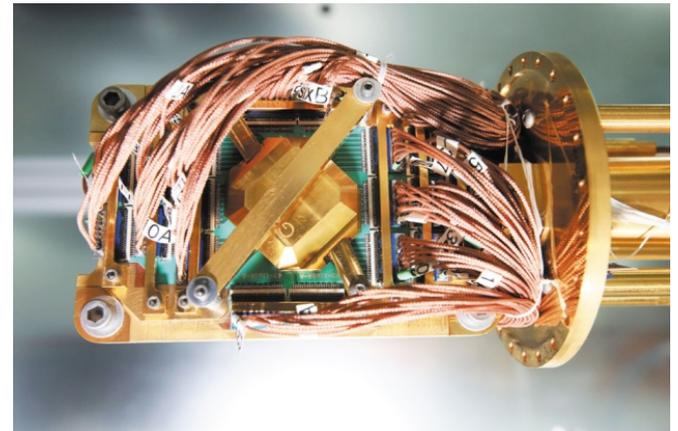


IBM. Google, Rigetti: circuito a porte logiche

I qubit sono “macroscopici”, coinvolgono molti atomi che si comportano come un singolo atomo



Chip superconduttore



IBM Q-Experience: Tanti QC su cloud (da 5 qubit a 21 qubit)

Facilmente usabile e programmabile

IBM Q Experience 

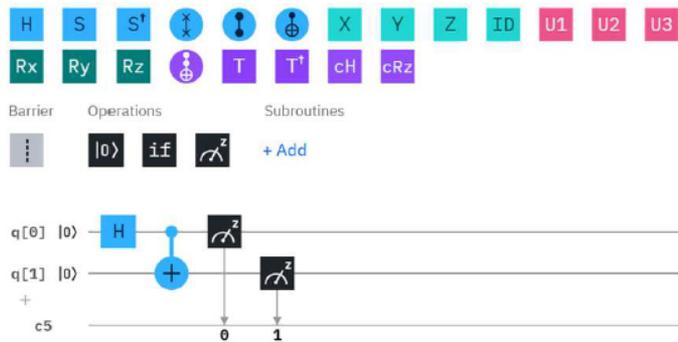
PROGRAMMAZIONE

- QISKit (Quantum Information Science Kit)



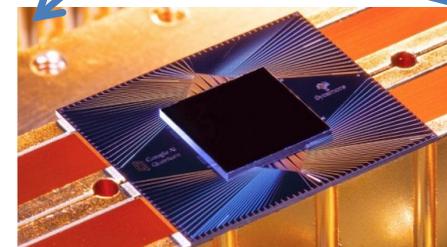
Super-conducting qubit technology

- Quantum Composer



<https://quantum-computing.ibm.com/>

Google: Stessa tecnologia:  
Supremazia quantistica con  
prototipo a 53 qubit

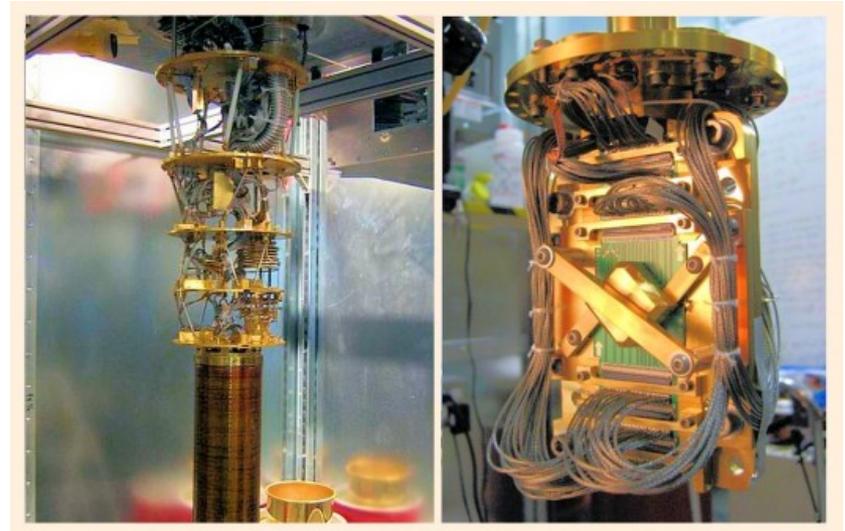


Google Sycamore – Quantum supremacy (Nature 2019)

<https://research.google/teams/applied-science/quantum/>

# D-WAVE: quantum annealer (ma stessa tecnologia a chip superconduttore)

- Pegasus P16: - 5640 qubits
- Simulazione di sistemi quantistici: può essere utilizzati per ottimizzare grafi e problemi riconducibili (maximum independent set)



# QC a ioni intrappolati

- Prima proposta completa (Cirac-Zoller 1995)

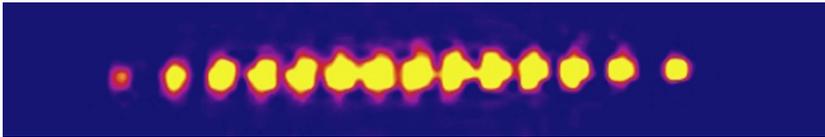
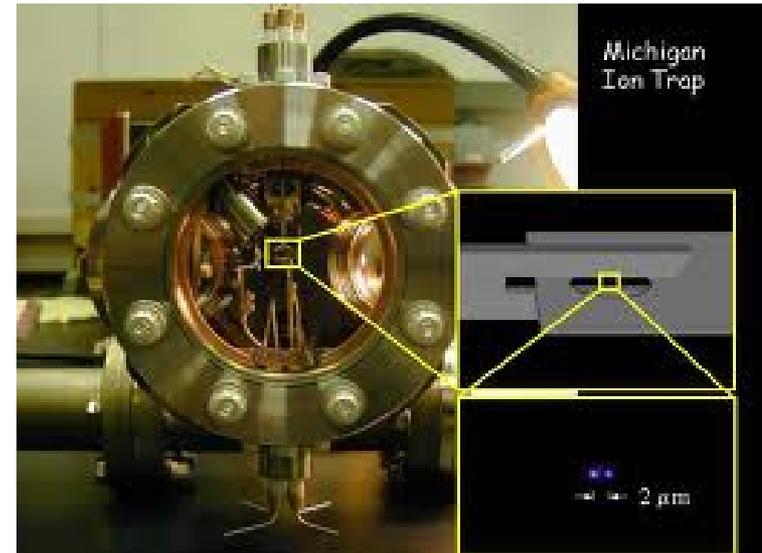


Immagine su ccd di ioni

Qubit = due livelli interni dello ione (spin nucleare)

1-qubit gate = impulsi laser

2 qubit gate = impulsi laser che usano un "quantum bus vibrazionale"



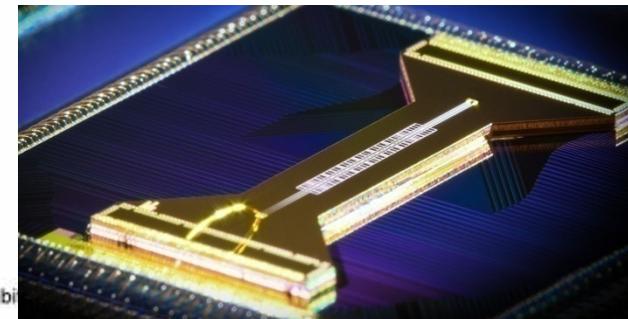
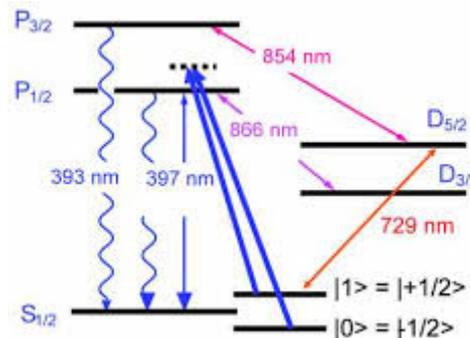
Esempio di trappola per ioni

**AQT (Innsbruck), IonQ (Honeywell)**

Trapped-ions quantum computer

Funziona a temperatura ambiente

32 qubit



# Dopo IBM-Q, vari servizi su cloud

<https://aws.amazon.com/it/braket/>

**Amazon braket: cloud service:** si possono mandare programmi a pagamento su Ion-Q, Rigetti..

## Partner Clouds

Whether you're on AWS, Google or Azure, hardware access is just a few clicks away.



[Go to Amazon Braket](#) ↗



[Go to Google Cloud](#) ↗



[Go to Azure Quantum](#) ↗

## Direct Access

IonQ partners enjoy direct access to our latest quantum hardware.

<https://ionq.com/>

# Linear optics quantum computer

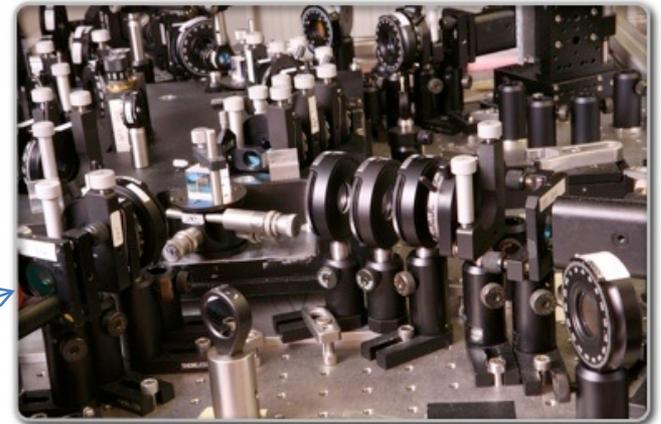
- Knill-Laflamme-Milburn (KLM) - 2001

- Pros: qubit semplici = singoli fotoni che hanno decoerenza trascurabile

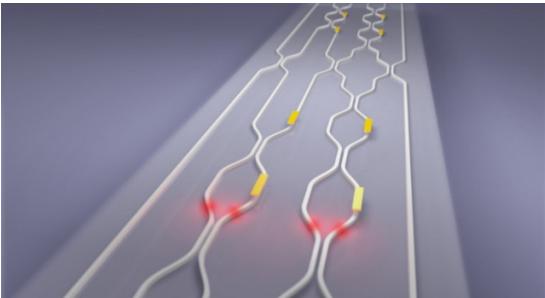
- Gate semplici (elementi ottici semplici)

- Cons: scalabilità non banale (anche se polinomiale)

(meglio con i chip fotonici)

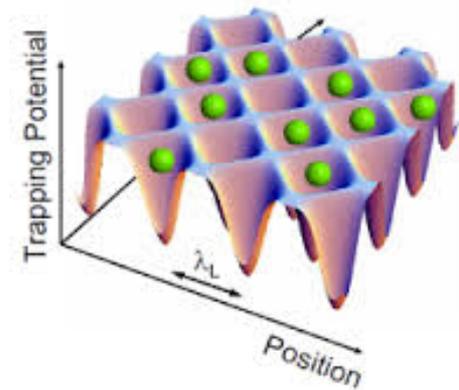
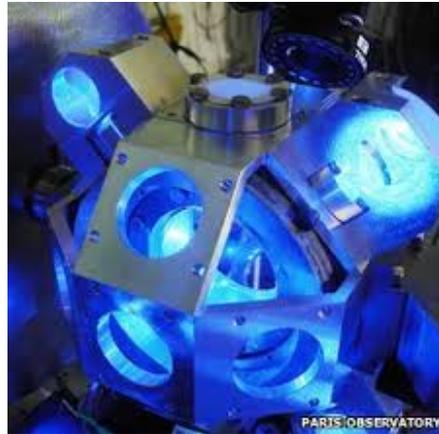
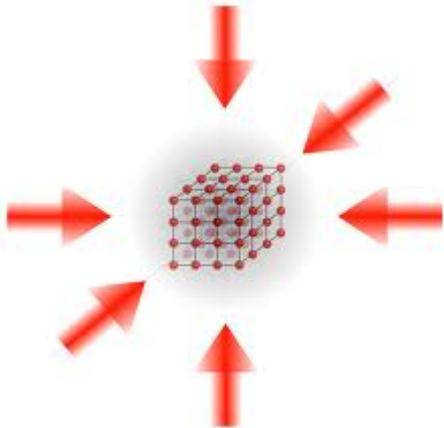


E' uno schema **One-way**  
**(basato sulla misura di fotoni e feedforward del risultato)**



# Atomi in reticoli ottici

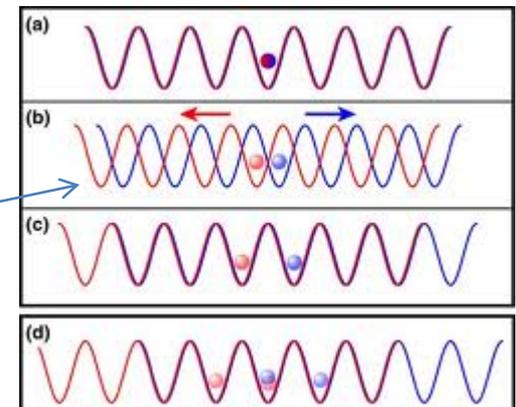
- Singoli atomi neutri intrappolati da vari fasci laser intensi e “pinzette ottiche”



1 qubit gates = impulsi laser sui singoli atomi

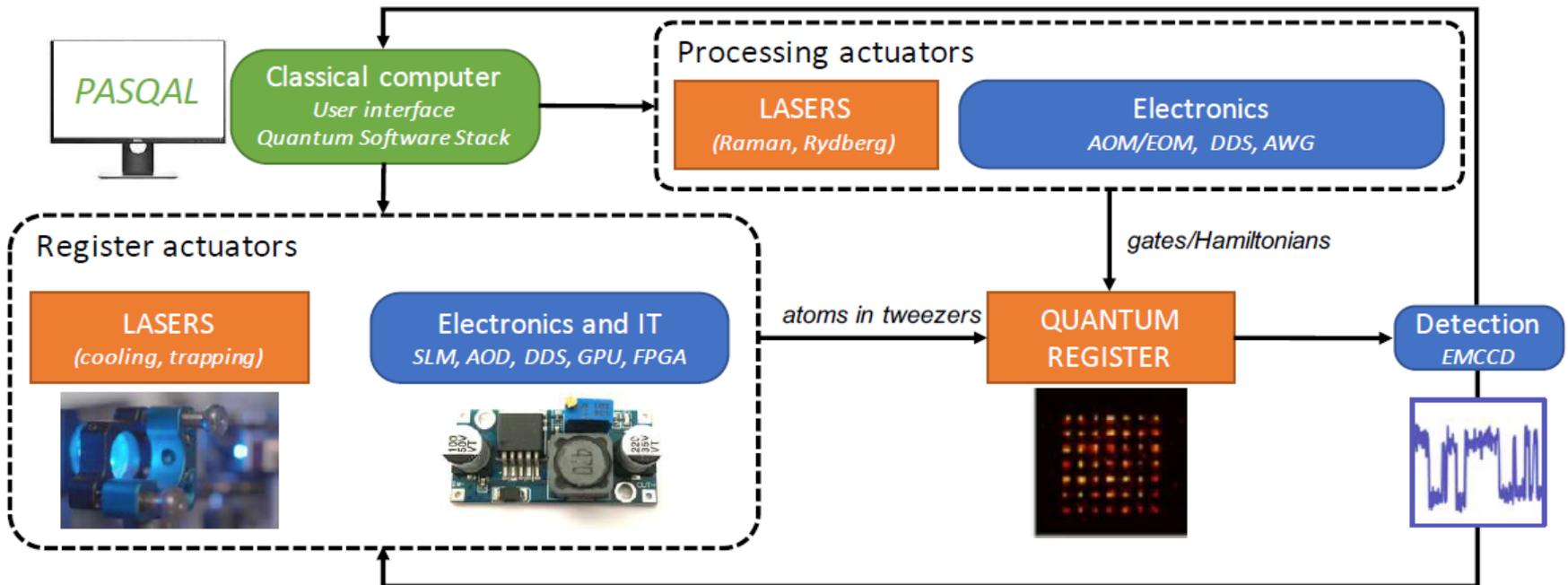
2-qubit gates = collisioni con shift di potenziali e impulsi laser

**Usati principalmente per simulazioni di sistemi quantistici**



# PASQAL

(reticoli di atomi di Rydberg (eccitati) intrappolati otticamente, Palaiseau, Francia)



Anche in questo caso, simulazione di sistemi quantistici e ottimizzazione di problemi su grafi

# NISQ (noisy intermediate scale quantum) technology era:

- QC con 50-100 qubit (quantum circuits) capaci di risolvere specifici problemi meglio di computer classici, ma con task limitati (a causa del noise che si accumula)
- Non particolarmente utili, ma uno step necessario
- Universalità e utilità pratica richiedono fault-tolerance
- “Complexity/entanglement frontier”