

# Rompere RSA con un computer quantistico

Un approccio pratico

Matteo Bonacini, 12 giugno 2023

# Rompere RSA con un computer quantistico

- Cos'è RSA?
- Cos'è un computer quantistico?
- Quantum Fourier Transform e Quantum Factorization
- Dimostrazione pratica
- Qualche osservazione

**RSA è un algoritmo di crittografia asimmetrica.**

# RSA è un algoritmo di crittografia asimmetrica.



# Le chiavi sono dei numeri *molto grandi*.



pubblica = prodotto di due primi **grandi**  
 $N = pq$



privata = I due primi  
( $p, q$ )

# Rompere RSA equivale a fattorizzare N...

$$\log_2 p \sim \log_2 q \sim 2048$$

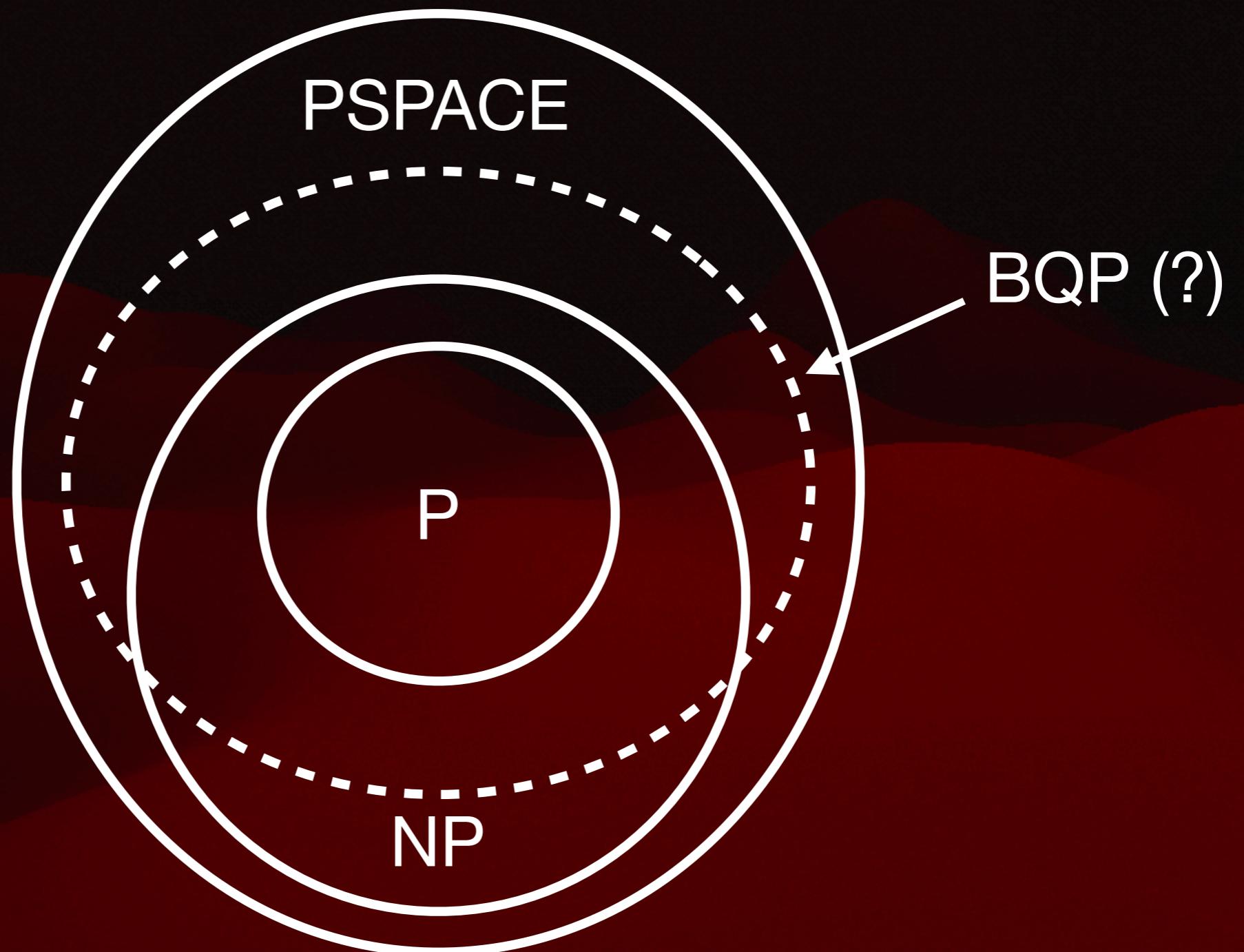
Miglior algoritmo classico (NFS):

$$\mathcal{O} \left[ \exp \sqrt[3]{N (\log N)^2} \right]$$

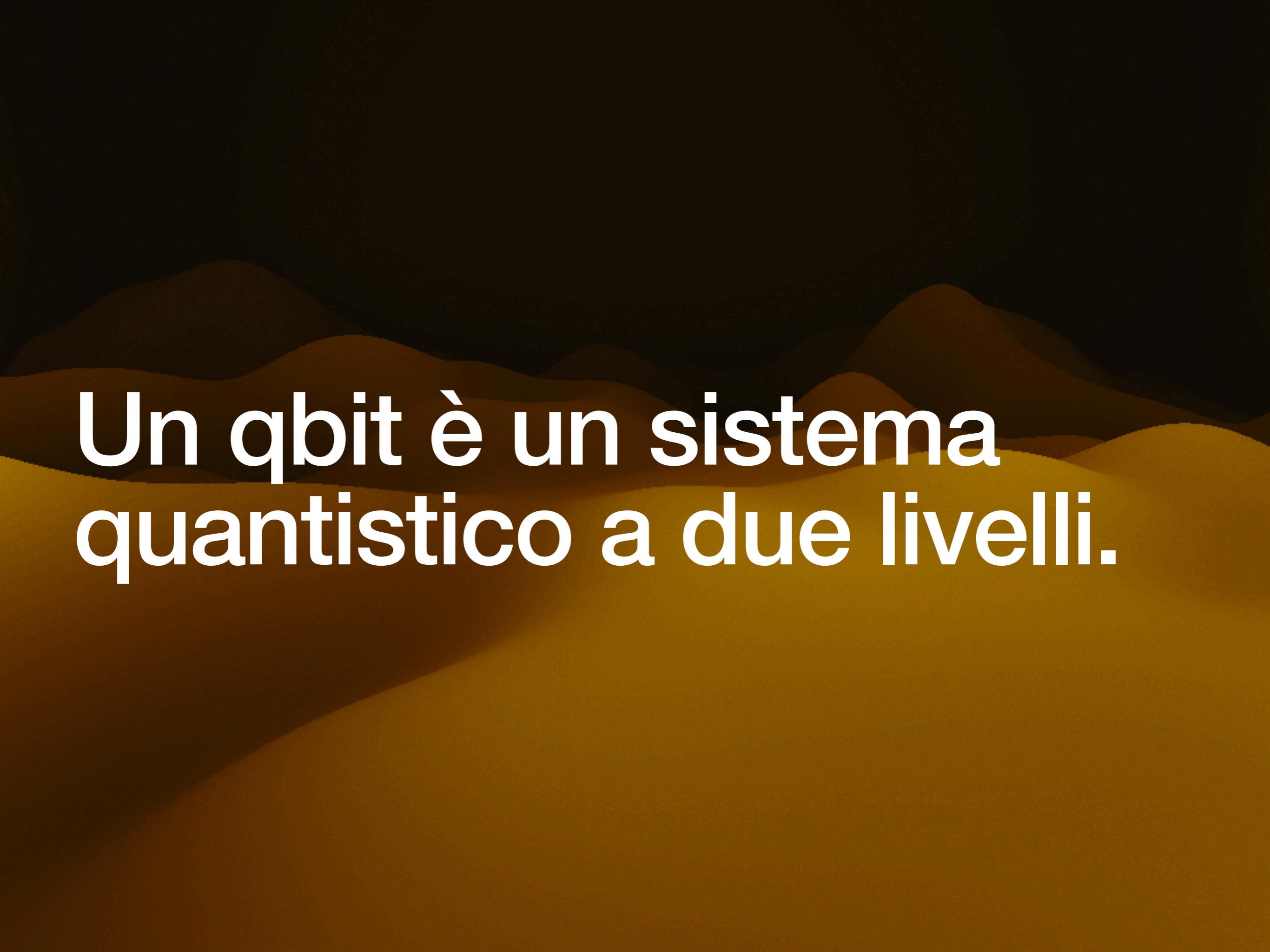
Miglior algoritmo quantistico (Shor):

$$\mathcal{O} \left[ (\log N)^2 \log \log N \right]$$

...e non è facile...



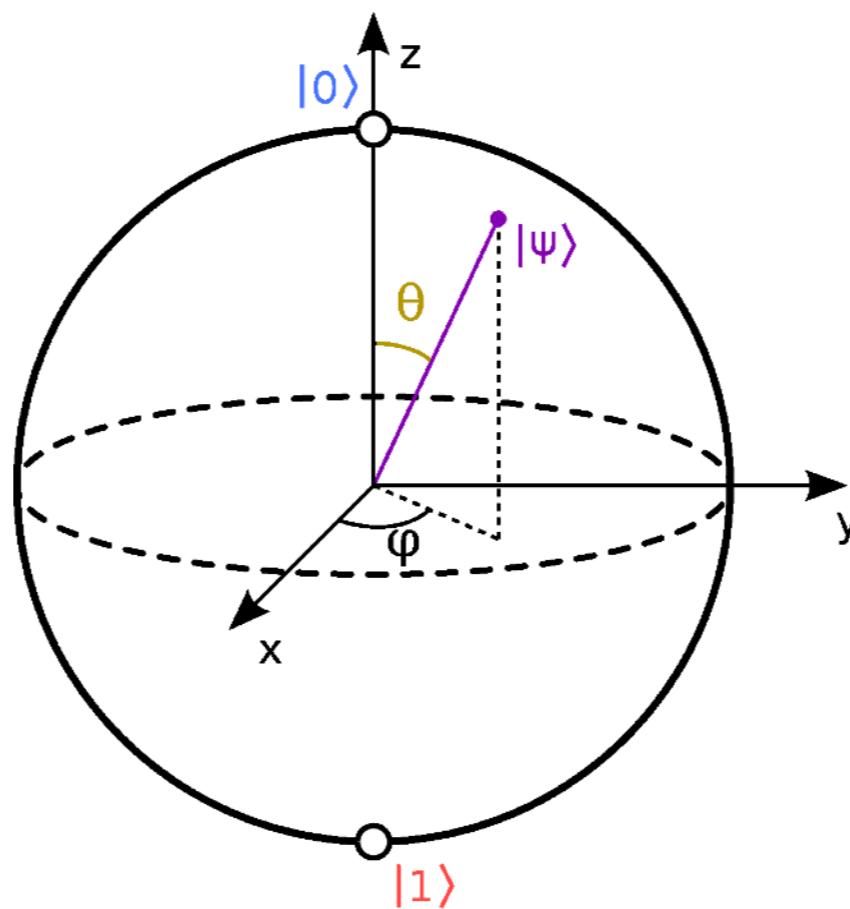
...senza un computer quantistico.



Un qbit è un sistema quantistico a due livelli.

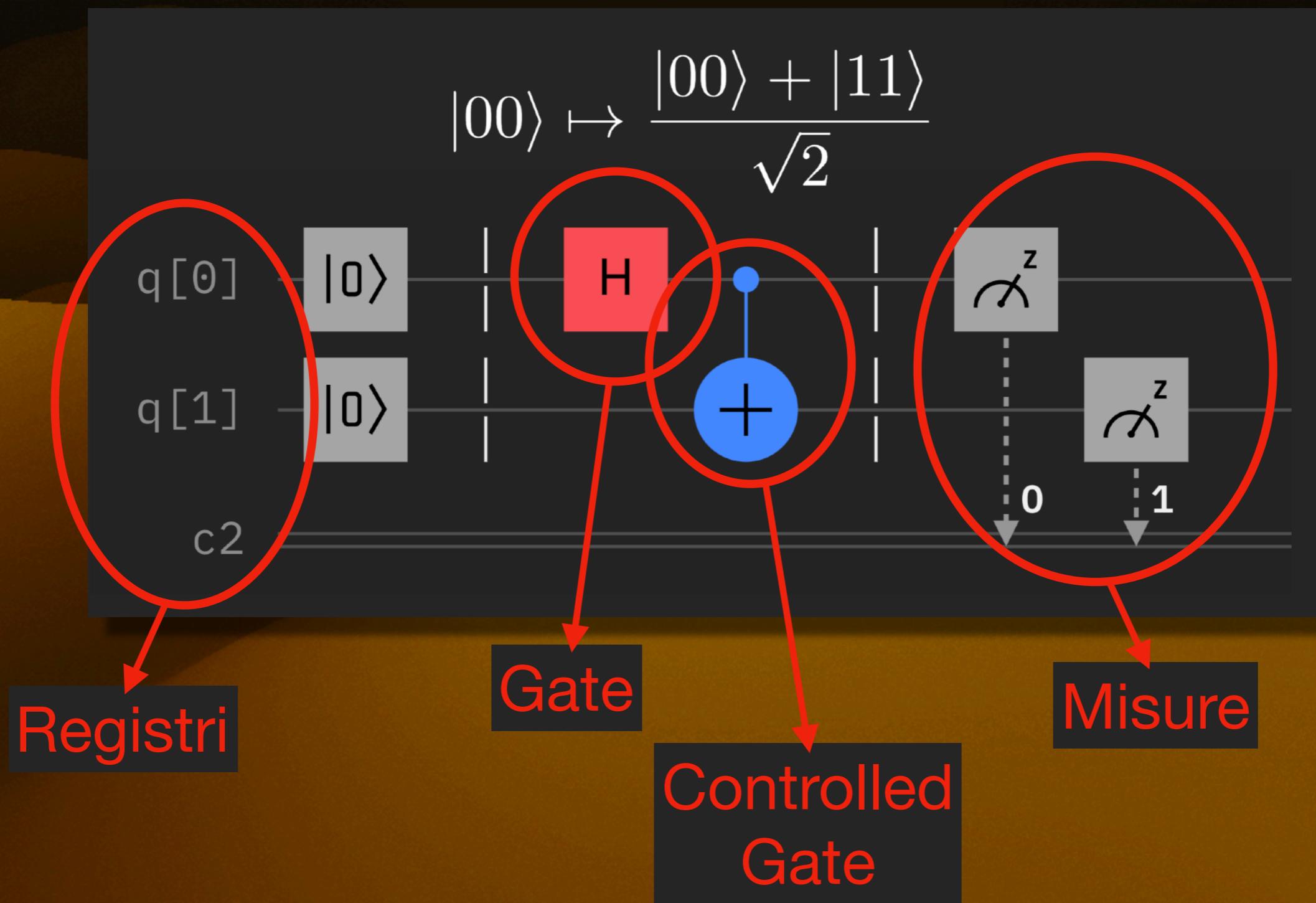
# Un qbit è un sistema quantistico a due livelli.

Sfera di Bloch



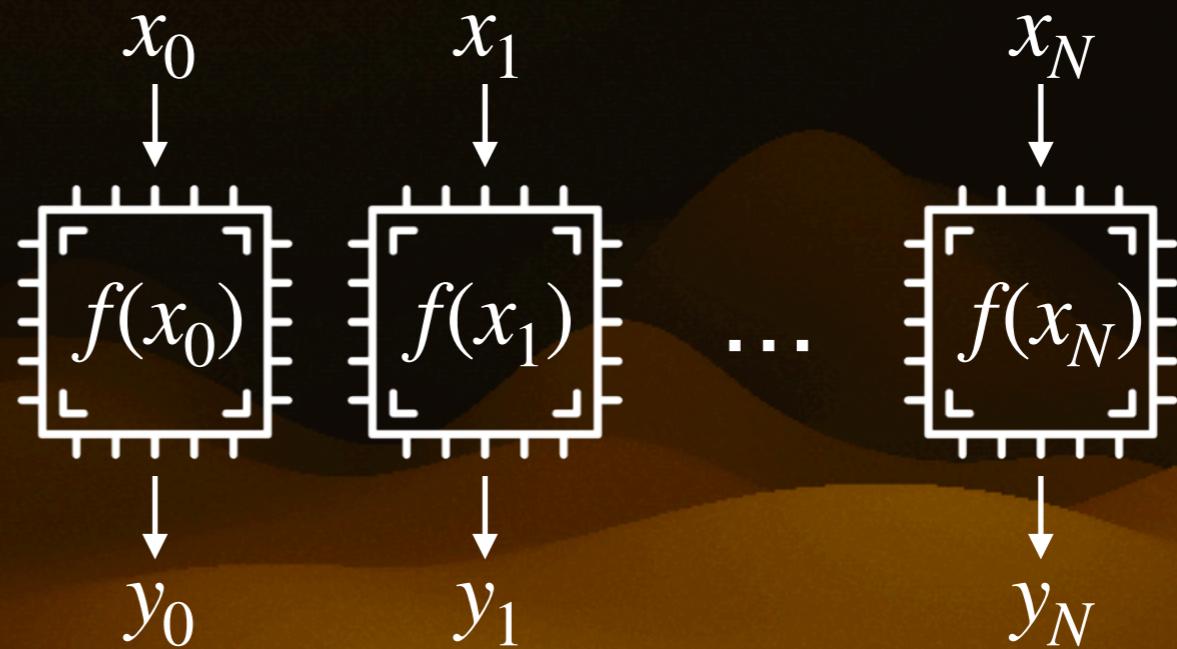
$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle$$

# Un computer quantistico esegue operazioni sui qbit.



# Il parallelismo quantistico è diverso dal parallelismo classico.

Parallelismo  
classico:



Parallelismo  
quantistico:

$$|x_0\rangle + |x_1\rangle + \dots + |x_N\rangle$$


$$\sum_{i=0}^{N-1} |f(x_i)\rangle$$

$$|y_0\rangle + |y_1\rangle + \dots + |y_N\rangle$$

Il parallelismo quantistico  
rende facile la  
fattorizzazione.

# Il parallelismo quantistico rende facile la fattorizzazione.

Problema  
Equivalente:

$$F_N(x) \equiv a^x \pmod{N}$$

? = Periodo di  $F$

Strategia  
(Algoritmo di Shor):

$$\text{Atom icon} \quad |F_N(0)\rangle + |F_N(1)\rangle + \dots$$

$$\downarrow \\ \text{Atom icon} \quad \text{QFT}^\dagger$$

$$\downarrow \\ \text{Processor icon} \quad p, q$$

# Introduciamo la Quantum Fourier Transform:

$$\{x_0, x_1, \dots, x_N\} \mapsto \{y_0, y_1, \dots, y_N\}$$

Definizione:

$$y_k \equiv \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi i j k / N}$$

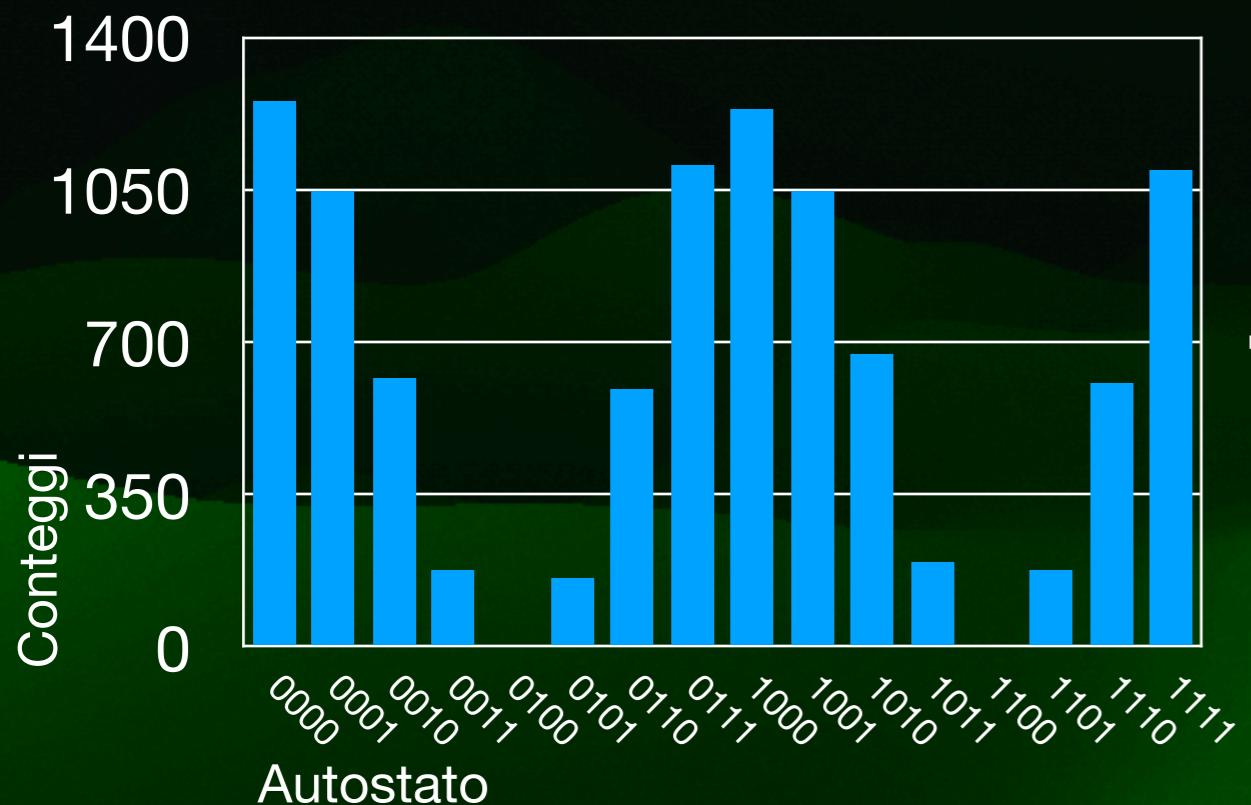
In notazione  
bra-ket:

$$\sum_{j=0}^{N-1} x_j |j\rangle \longrightarrow \sum_{k=0}^{N-1} y_k |k\rangle$$

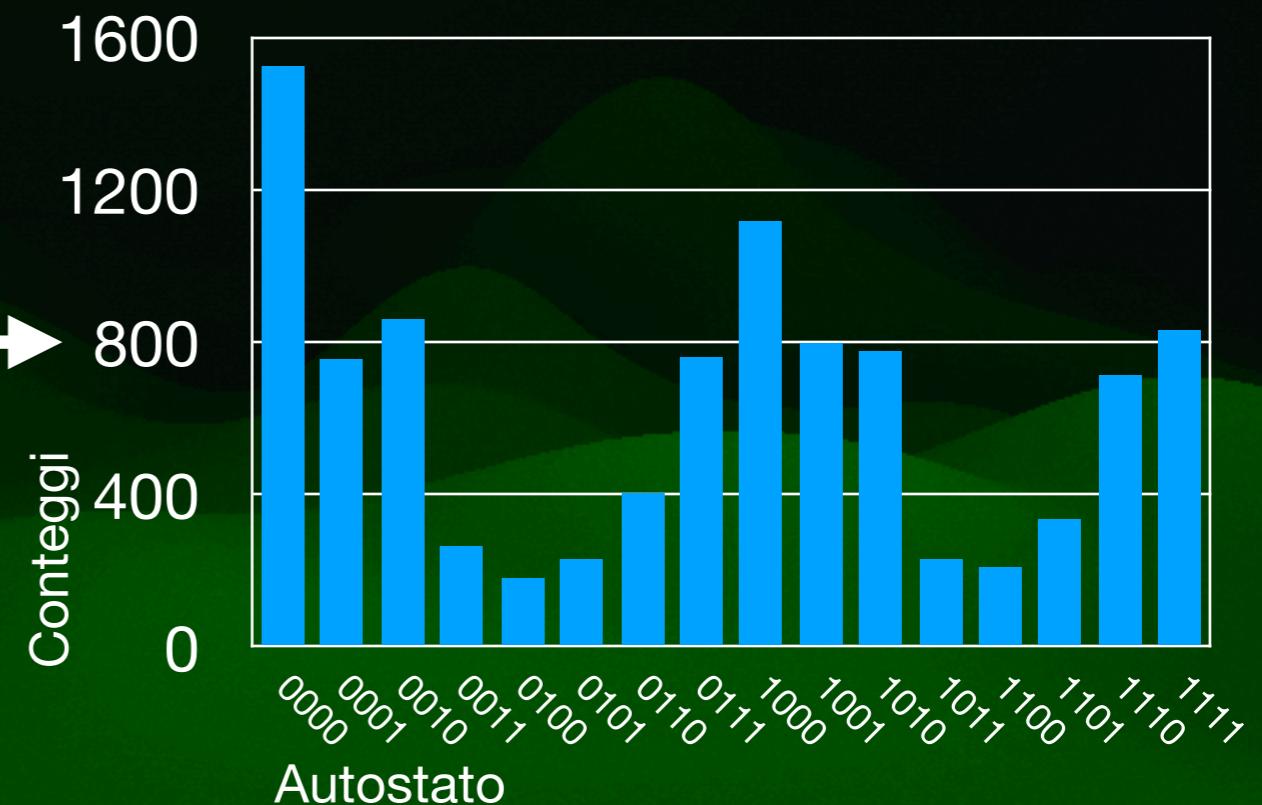
Vediamo un esempio  
pratico...

# Il circuito funziona, ma è inutile.

Simulazione



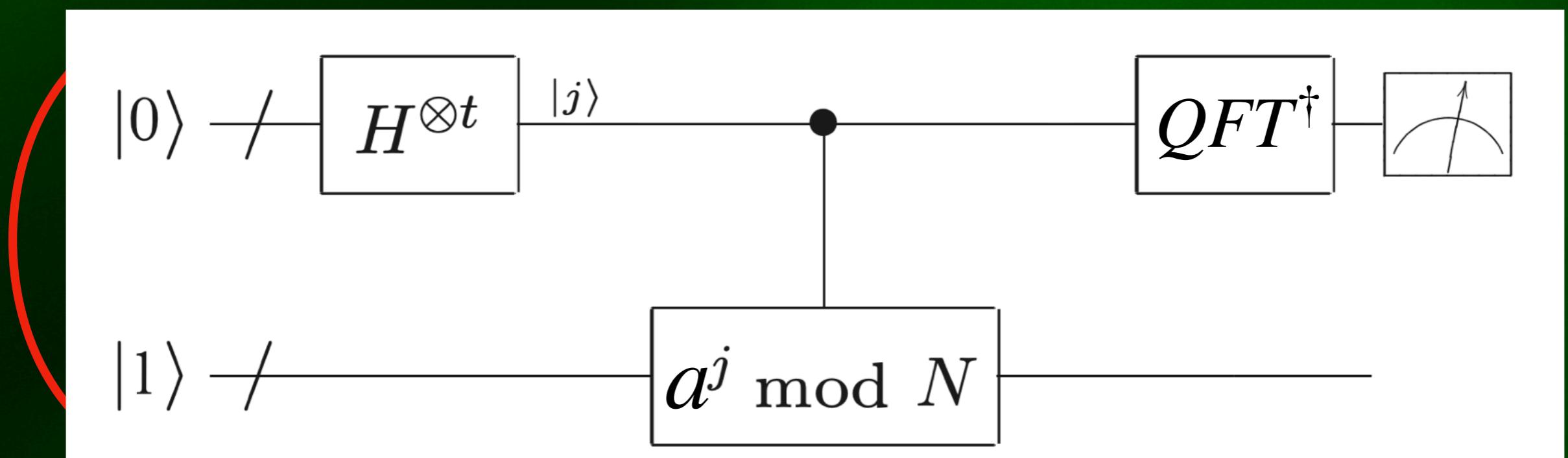
Risultato



Non possiamo conoscere le ampiezze complesse!

# QFT eccelle se usata come subroutine di un altro circuito.

$$|0\rangle \otimes |1\rangle \xrightarrow{\frac{1}{\sqrt{2^t}} \left( \sum_{j=0}^{2^t-1} |j\rangle \right) \otimes |1\rangle} \frac{1}{\sqrt{2^t}} \left( \sum_{j=0}^{2^t-1} |j\rangle \right) \otimes |a^j \bmod N\rangle$$



# QFT eccelle se usata come subroutine di un altro circuito.

$$\frac{1}{\sqrt{2^t}} \left( \sum_{j=0}^{2^t-1} |j\rangle \right) \otimes |a^j \bmod N\rangle$$

$$\frac{1}{\sqrt{2^t}} \frac{1}{\sqrt{r}} \left[ \sum_{s=0}^{r-1} \sum_{j=0}^{2^t-1} \exp \left( 2\pi i j \frac{s}{r} \right) |j\rangle \right] \otimes |u_s\rangle$$

