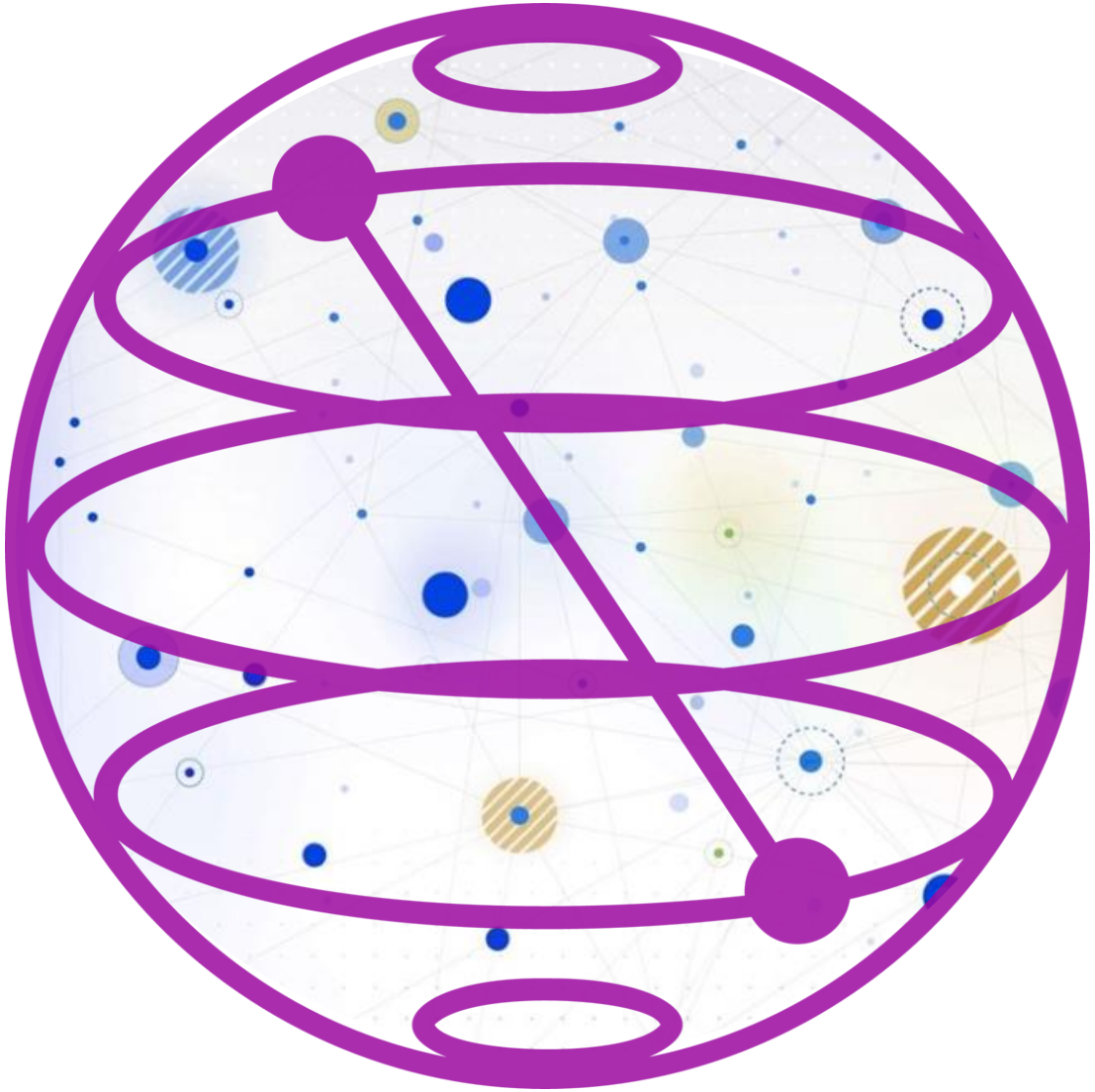
Corso di Quantum Computing – Giorno 2

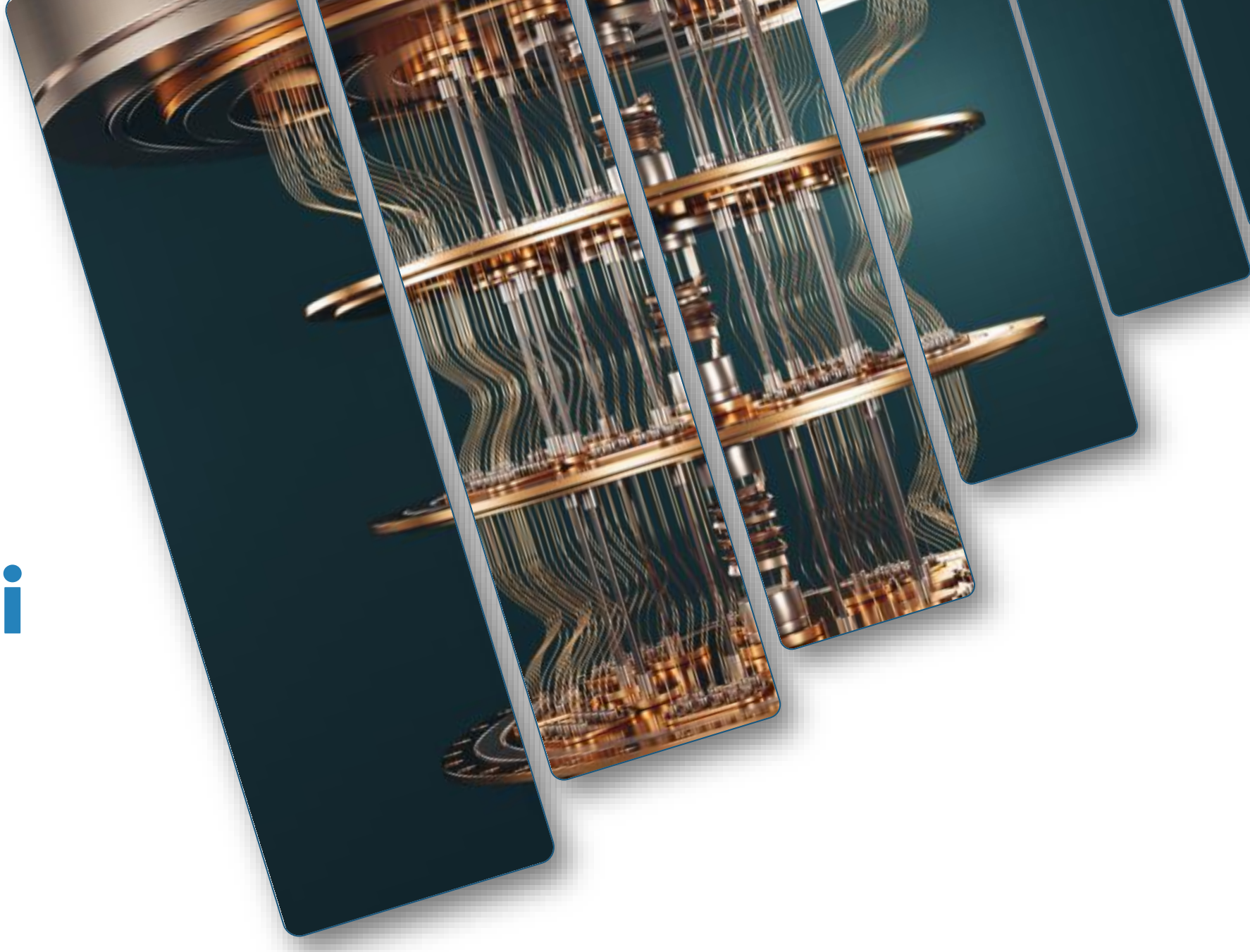


Corso per Epigenesys s.r.l.

Docenti: Sara Galatro e Lorenzo Gasparini

Supervisore: Prof. Marco Pedicini

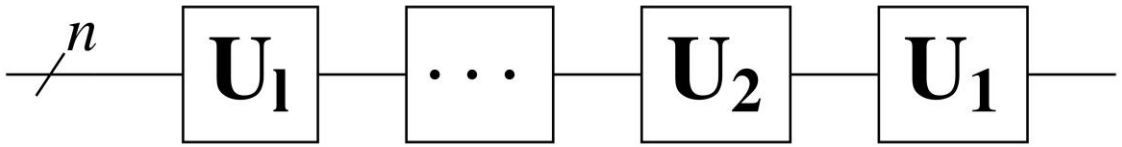
Circuiti quantistici



Definizione di circuiti quantistici

- Un circuito quantistico, nella sua forma più semplice, è la connessione di più gate quantistici elementari.

- Formalmente, un **circuito quantistico** che agisce su n qubit, è la composizione di $l > 0$ operatori $U_1, \dots, U_l$ (ognuno dei quali agisce su n qubit). Identificheremo un circuito tramite un unico operatore $U = U_1 \cdots U_l$.

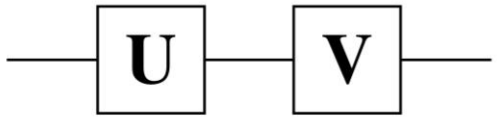


- Eseguire un circuito quantistico con uno stato in input $|\psi\rangle$, vuol dire applicare l'operatore U allo stato $|\psi\rangle$. Abbiamo dunque, $U|\psi\rangle = U_1 \cdots U_l|\psi\rangle = |\psi'\rangle$, una moltiplicazione tra matrice e vettore.

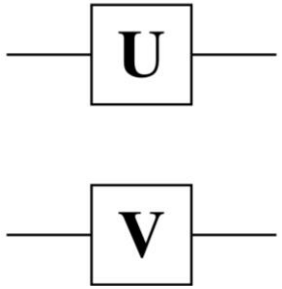
Regole di composizione

- Specifichiamo ora in che modo possiamo costruire un gate quantistico a partire da altri. Ci sono due modi in cui possiamo comporre i gate tra di loro.

- **Composizione sequenziale**: due operatori vengono eseguiti uno dopo l'altro e applicheremo tra essi il **prodotto matriciale** $V \cdot U$.



- **Composizione in parallelo**: due operatori vengono eseguiti nello stesso momento a due parti differenti del registro e applicheremo tra essi il **prodotto tensoriale** $U \otimes V$.

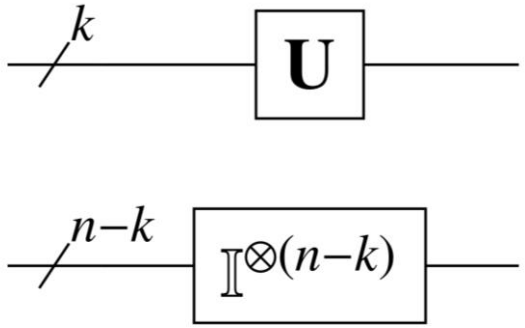


Azioni parziali

- Un'ulteriore possibilità la abbiamo quando viene applicato un gate solo su una parte del registro, mentre sulla restante parte non viene applicata alcuna trasformazione.

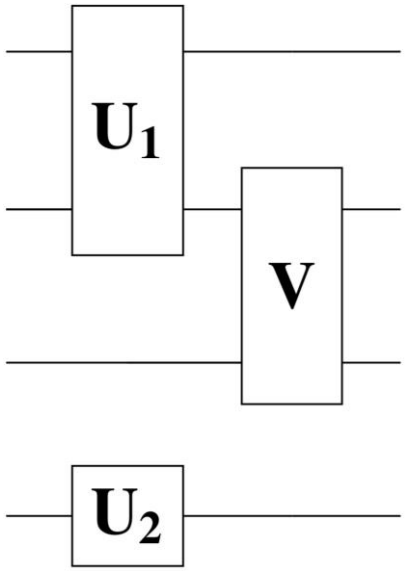
- Consideriamo un circuito ad n qubit e prendiamo U come operatore a $k < n$ qubit. Ci possiamo immaginare che sulla restante parte dei qubit $(n - k)$ venga applicato il gate identità. U **agisce parzialmente sul circuito**.

- Applichiamo quindi in parallelo il gate U e il gate identità



Esempio circuito

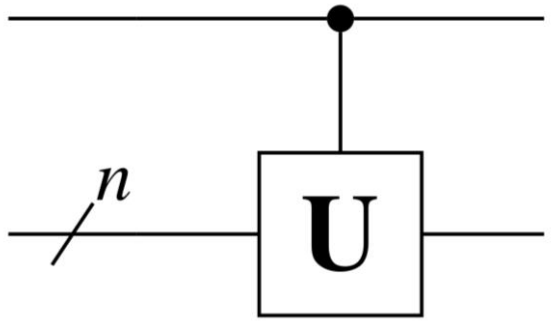
- Facciamo un esempio su come ricondurci ad un unico operatore U che descriva l'azione globale di un circuito.
- Consideriamo il seguente circuito



- Ricordando come funzionano le regole di composizione, l'operatore U che descrive il circuito è dato da $U = (I \otimes V \otimes I) \cdot (U_1 \otimes I \otimes U_2)$.

Controlled-Gate

- Ricordiamo che un generico controlled-gate è definito come $CU = |0\rangle\langle 0| \otimes I_R + |1\rangle\langle 1| \otimes U$.
- Rappresenteremo circuitalmente un'**operazione controllata** nel seguente modo.



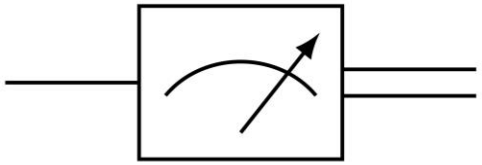
- In generale tracceremo una linea verticale per connettere sistema target e qubit target. Nel sistema target rappresenteremo l'applicazione del gate U, mentre nel qubit di controllo rappresenteremo un pallino pieno.

Misurazione e output

- Per ottenere un risultato reale è necessario interagire con il sistema ed eseguire una **misurazione**. La misurazione, come sappiamo, comporta un collasso del sistema su uno degli stati della base standard.

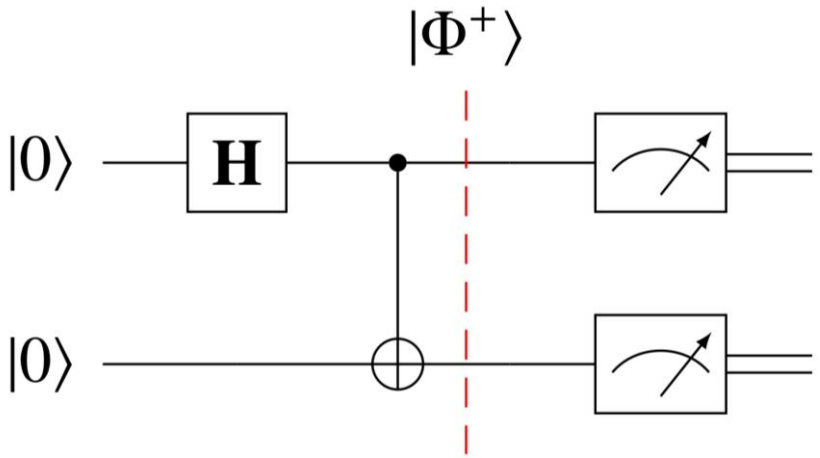
- Per come abbiamo visto i circuiti nella loro definizione semplificata, consistono nell'applicazione sequenziale di operatori unitari, la quale genera uno stato quantistico $|\psi'\rangle$. Per ottenere informazioni su tale stato possiamo solo misurarlo, ottenendo così una **stringa binaria in output**.

- Rappresenteremo l'azione di misurazione nei circuiti nel seguente modo



Circuito per lo stato di Bell

- Riportiamo il circuito che permette di costruire lo **stato di Bell** in entanglement $|\Phi^+\rangle$



- Otteniamo lo stato di Bell $|\Phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$ applicando all'input $|0\rangle|0\rangle$ l'operatore $U = CNOT(H \otimes I)$.

- Misurando tutto il circuito otteniamo con uguale probabilità la stringa 00 oppure 11.

Esercizio

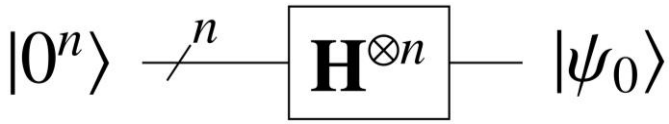
- Tramite un circuito a due qubit, costruire un $CNOT$ gate, utilizzando due Hadamard gate H e un gate controllato CZ.

Preparazione input

- Nella maggior parte degli algoritmi fondamentali che vedremo è utile scegliere come stato quantistico in input una **sovrapposizione equiprobabile** di tutti gli stati della base standard.

- Tale costruzione è sempre inclusa nei circuiti che richiedano di lavorare con una sovrapposizione. In generale, quindi, il circuito presenta come stato in input standard lo stato $|0^n\rangle$.

- Possiamo ottenere una sovrapposizione uniforme partendo da $|0^n\rangle$ applicando Hadamard in parallelo su tutto il circuito



- Applicando un gate H ad ogni qubit del circuito, otteniamo lo stato $|\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle$.

Esempio sovrapposizione uniforme

- Facciamo un esempio prendendo $n = 2$ e calcoliamo $H^{\otimes 2}|00\rangle$. Procediamo con il calcolo come segue:

$$H^{\otimes 2}(|0\rangle \otimes |0\rangle) = H|0\rangle \otimes H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

- Riscrivendo tramite vettori otteniamo

$$\frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} \otimes \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} = \frac{1}{2}\begin{pmatrix}1\\1\\1\\1\end{pmatrix} = |\psi_0\rangle$$

- Abbiamo come risultato quindi $|\psi_0\rangle = \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle + \frac{1}{2}|2\rangle + \frac{1}{2}|3\rangle$, sovrapposizione equiprobabile degli stati della base standard.

Implementazioni funzioni booleane

- Poter implementare quantisticamente funzioni booleane generiche $f:\{0,1\}^n \to \{0,1\}^m$ risulta cruciale per la costruzione di algoritmi quantistici.

- Per tale costruzione, faremo uso dell'operatore U_f, definito su un registro di $n+m$ qubit, la cui azione sugli stati della base è la seguente

$$|x\rangle \otimes |y\rangle \longmapsto |x\rangle \otimes |y \oplus f(x)\rangle.$$

- Applicare questo operatore U_f ad uno stato $|x\rangle \otimes |0^m\rangle$ permette di ottenere $|x\rangle \otimes |f(x)\rangle$, facendo così in modo di avere sul secondo registro la valutazione di f sull'input x. Ecco perché U_f **implementa la funzione** f.

- Tramite questa costruzione possiamo implementare qualsiasi funzione booleana classica **anche non invertibile** con l'utilizzo di $n+m$ qubit.

- Si può dimostrare facilmente che U_f **è unitaria**, quindi è un'operazione valida.

Esempio U_f

- Implementiamo per esempio la seguente funzione booleana $f(x) = (x, x \oplus 1)$. Faremo uso 3 qubit. Riportiamo di seguito l'azione di U_f sui vettori della base standard e dunque la matrice associata:

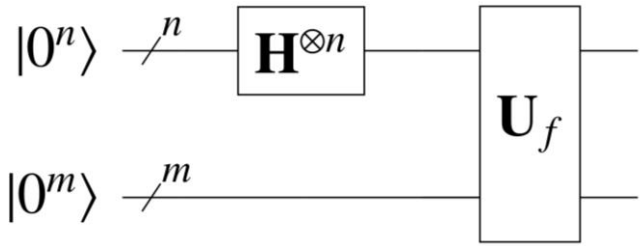
$$U_f |000\rangle = |001\rangle$$
$$U_f |001\rangle = |000\rangle$$
$$U_f |010\rangle = |011\rangle$$
$$U_f |011\rangle = |010\rangle$$
$$U_f |100\rangle = |110\rangle$$
$$U_f |101\rangle = |111\rangle$$
$$U_f |110\rangle = |100\rangle$$
$$U_f |111\rangle = |101\rangle$$

$$U_f := \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & 0 & \mathbb{I} \\ 0 & 0 & \mathbb{I} & 0 \end{bmatrix}$$

Quantum Parallelism

- Una caratteristica unica del calcolo quantistico è il **quantum parallelism**, proprietà che nasce proprio dalla combinazione di sovrapposizioni equiprobabili e l'implementazione di qualsiasi funzione booleana f.

- Comprendiamo tale proprietà, partendo dal circuito quantistico che la descrive:

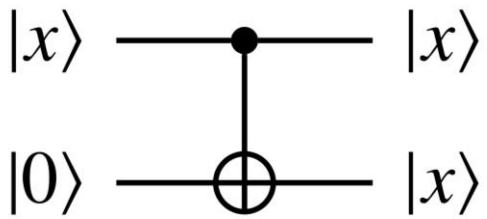


- Applicando tale circuito otteniamo il vettore $s_f = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|f(x)\rangle$. Tale vettore rappresenta, sul secondo registro, una sovrapposizione equiprobabile di tutti i possibili output di f.

- L'importanza di questa proprietà sta nel fatto che con una sola applicazione di U_f abbiamo calcolato simultaneamente f su tutti i possibili valori del dominio.

No-cloning Theorem

- Il quantum computing, nonostante molte potenzialità fino ad ora enunciate, ha anche delle limitazioni, di cui la principale prende il nome di **no-cloning theorem**.
- Il teorema afferma che **non è possibile clonare uno stato quantistico arbitrario**.
- Formalmente: non esiste alcun operatore U tale che $U(|\varphi\rangle \otimes |\psi\rangle) = |\varphi\rangle \otimes |\varphi\rangle$ per ogni stato $|\varphi\rangle$.
- Il teorema però non mette alcun limite sulla possibilità di clonare uno stato fissato. Per esempio, possiamo clonare un semplice qubit tramite un $CNOT$ e inizializzando $|\psi\rangle = |0\rangle$.



Dimostrazione

- Consideriamo due stati distinti $|x\rangle$ e $|y\rangle$. Supponiamo per assurdo che esista operatore U che permetta di clonare qualsiasi stato a partire da uno stato $|\psi\rangle$. In particolare abbiamo:

$$U(|x\rangle \otimes |\psi\rangle) = |x\rangle \otimes |x\rangle \quad \text{e} \quad U(|y\rangle \otimes |\psi\rangle) = |y\rangle \otimes |y\rangle$$

- Vogliamo calcolare la seguente trasformazione: $U\left(\left(\frac{1}{\sqrt{2}}|x\rangle + \frac{1}{\sqrt{2}}|y\rangle\right) \otimes |\psi\rangle\right)$

- Per linearità di U e poi per definizione di U abbiamo:

$$\frac{1}{\sqrt{2}}U(|x\rangle \otimes |\psi\rangle) + \frac{1}{\sqrt{2}}U(|y\rangle \otimes |\psi\rangle) = \frac{1}{\sqrt{2}}|x\rangle \otimes |x\rangle + \frac{1}{\sqrt{2}}|y\rangle \otimes |y\rangle$$

- Prima applichiamo la definizione di U e poi continuiamo per linearità:

$$\left(\frac{1}{\sqrt{2}}|x\rangle + \frac{1}{\sqrt{2}}|y\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}|x\rangle + \frac{1}{\sqrt{2}}|y\rangle\right)$$

Applicazioni dell'entanglement

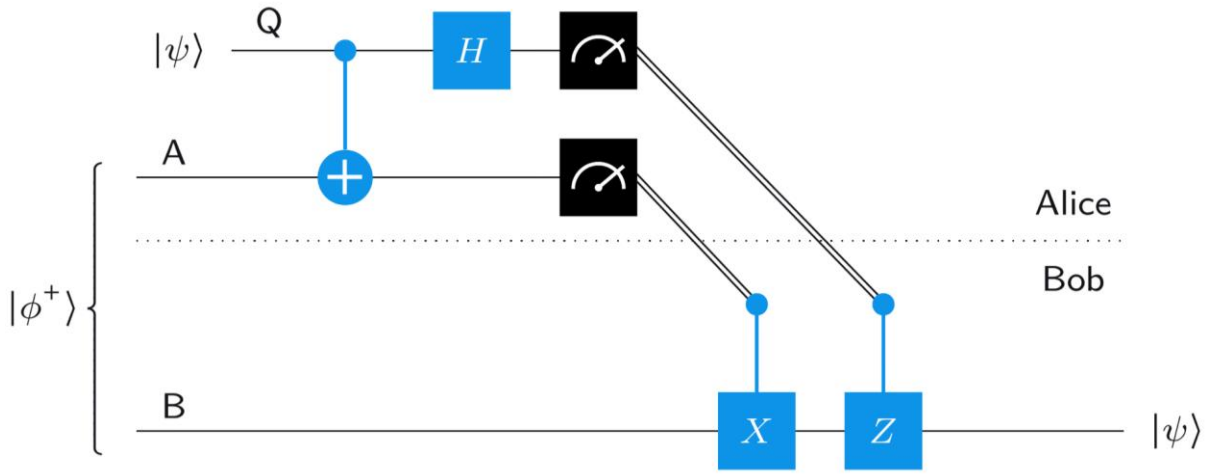
- Abbiamo visto come costruire lo stato di Bell con un semplice circuito. Mostriamo subito le potenzialità di uno stato in entanglement, descrivendo due protocollo quantistici molto potenti.

- In entrambi i protocolli che vedremo, supporremo di avere due soggetti, Alice e Bob, che condividono uno stato in entanglement. Ci immaginiamo che Alice sia in possesso di un qubit A e Bob uno nello stato B tale che lo stato congiunto (A, B) sia rappresentato da $|\Phi^+\rangle$.

- Questo è il setting generale. Nei protocolli l'idea sarà quella in cui **Alice e Bob devono scambiarsi delle informazioni**.

Teletrasporto

- Nel protocollo del **teletrasporto**, Alice vuole trasmettere un qubit Q a Bob, facendo uso della stato in entanglement condiviso e con l'ausilio di due bit di comunicazione classica.
- Trasmettere il qubit Q, significa che Bob alla fine del protocollo possiederà un qubit nello stesso stato di Q, comprese le eventuali correlazioni che aveva con altri sistemi.
- Riportiamo il circuito che descrive suddetto protocollo.



Teletrasporto

- Supponiamo che il qubit Q sia della forma generica $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Analizziamo il protocollo per step.

1) Il sistema è inizializzato nello stato $(\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.

2) Alice applica un $CNOT$ gate, con qubit di controllo Q e qubit target A, ottenendo:

$$\frac{\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle}{\sqrt{2}}$$

3) Alice applica a questo punto la porta di Hadamard sul qubit Q così da ottenere:

$$\frac{1}{2}|00\rangle(\alpha|0\rangle+\beta|1\rangle)+\frac{1}{2}|01\rangle(\alpha|1\rangle+\beta|0\rangle)+\frac{1}{2}|10\rangle(\alpha|0\rangle-\beta|1\rangle)+\frac{1}{2}|11\rangle(\alpha|1\rangle-\beta|0\rangle)$$

Teletrasporto

4) Dobbiamo ora distinguere i quattro casi a seconda di cosa misura Alice sui suoi qubit.

- Se $xy = 00$ Bob rimane nello stato $\alpha|0\rangle + \beta|1\rangle$. Non vengono applicate ulteriori trasformazioni e Bob si ritrova con un qubit nello stesso stato di Q.
- Se $xy = 01$ Bob rimane nello stato $\alpha|1\rangle + \beta|0\rangle$. Bob applica un gate X così da ottenere:

$$X(\alpha|1\rangle + \beta|0\rangle) = \alpha X|1\rangle + \beta X|0\rangle = \alpha|0\rangle + \beta|1\rangle = Q$$

- Se $xy = 10$ Bob rimane nello stato $\alpha|0\rangle - \beta|1\rangle$. Bob applica un gate Z per ottenere:

$$Z(\alpha|0\rangle - \beta|1\rangle) = \alpha Z|0\rangle - \beta Z|1\rangle = \alpha|0\rangle + \beta|1\rangle = Q$$

- Se $xy = 11$ Bob rimane nello stato $\alpha|1\rangle - \beta|0\rangle$. Bob applica prima un gate X poi un gate Z:

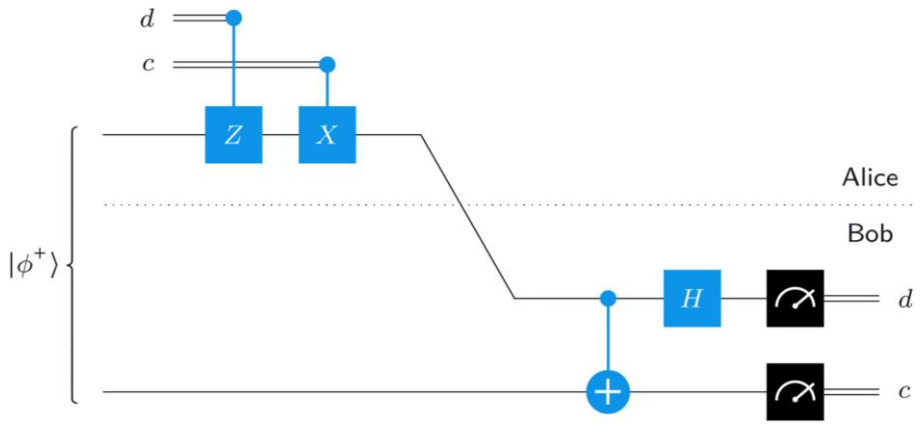
$$ZX(\alpha|1\rangle - \beta|0\rangle) = Z(\alpha|0\rangle - \beta|1\rangle) = \alpha|0\rangle + \beta|1\rangle = Q$$

Superdense Coding

- Il **superdense coding** è un protocollo inverso rispetto al teletrasporto. L'idea è la seguente: Alice vuole trasmettere 2 bit classici tramite l'invio di un qubit e con lo stato in entanglement condiviso con Bob.

- L'importanza di questo protocollo è messo in evidenza soprattutto dal teorema di Holevo, il quale afferma che non è possibile inviare più di un bit classico, con l'invio di un solo qubit e senza uso di stato entangled.

- Riportiamo di seguito il circuito del superdense coding.



Superdense Coding

- Per capire al meglio il perché tale protocollo funzioni, facciamo due semplici osservazioni: ognuno dei quattro stati di Bell si può ottenere da uno dei quattro stati della base computazionale e tutti e quattro stati di Bell si ottengono tutti da $|\Phi^+\rangle$ applicando X e/o Z.

$$|\Phi^+\rangle = CNOT\left(H \otimes \mathbb{I}\right)|00\rangle \qquad |\Phi^+\rangle = \left(\mathbb{I} \otimes \mathbb{I}\right)|\Phi^+\rangle$$

$$|\Psi^+\rangle = CNOT\left(H \otimes \mathbb{I}\right)|01\rangle \qquad |\Psi^+\rangle = \left(X \otimes \mathbb{I}\right)|\Phi^+\rangle$$

$$|\Phi^-\rangle = CNOT\left(H \otimes \mathbb{I}\right)|10\rangle \qquad |\Phi^-\rangle = \left(Z \otimes \mathbb{I}\right)|\Phi^+\rangle$$

$$|\Psi^-\rangle = CNOT\left(H \otimes \mathbb{I}\right)|11\rangle \qquad |\Psi^-\rangle = \left(ZX \otimes \mathbb{I}\right)|\Phi^+\rangle$$

- Bob si ritroverà quindi con uno degli stati di Bell, che chiameremo $|b\rangle$. Applicando dunque il circuito al contrario ottiene:

$$\left(H \otimes \mathbb{I}\right)CNOT\,|b\rangle = \left(H \otimes \mathbb{I}\right)CNOT \cdot CNOT\left(H \otimes \mathbb{I}\right)|xy\rangle = |xy\rangle$$

Query model computation

- L'idea naturale che abbiamo di un processo computazionale, è quella in cui alcune informazioni vengono fornire in input e poi processate per restituire un output.

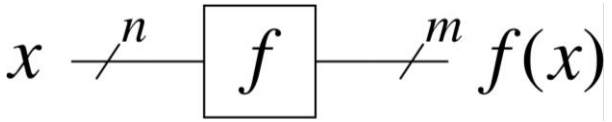
- Nel **modello query**, l'input viene fornito sotto forma di funzione, detta **oracolo** o **black box**.

- La funzione f in input va immaginata come una funzione di cui non conosciamo il funzionamento ma di cui possiamo conoscerne le valutazioni su generici input x del dominio di definizione tramite **query**.

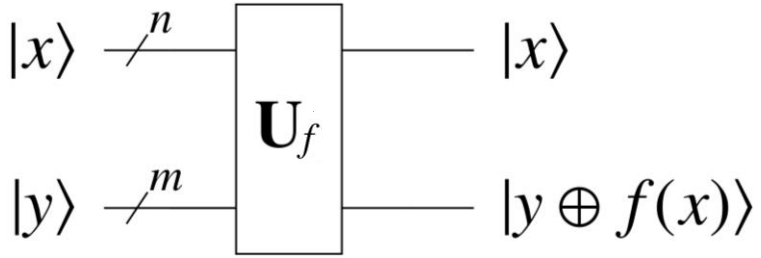
- Una query viene dunque eseguita da un processo computazionale se quest'ultimo seleziona una stringa x e la stringa $f(x)$ viene resa disponibile per la computazione.

Query gate

- Lavorando con i circuiti, le query vengono rappresentate tramite gate.
- Nel caso di un circuito booleano classico abbiamo una semplice trasformazione portata avanti dalla stessa funzione booleana f presa in considerazione



- Nel caso quantistico non possiamo fare la stessa cosa, dal momento che i possibili operatori che otterremmo per f non sarebbero di certo unitari.
- Nel caso quantistico, il **quantum query gate** viene implementato tramite l'operatore U_f già definito.



Quantum query model

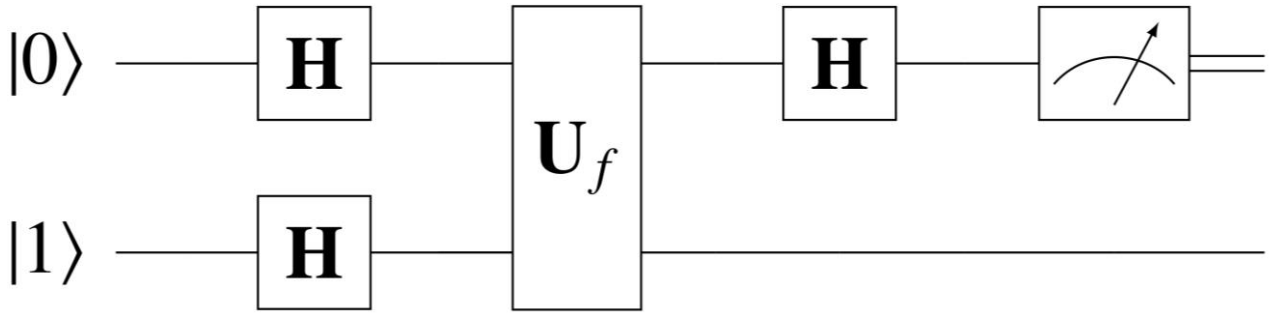
- L'operatore unitario associato ad f quindi, nel contesto del **quantum query model**, ha il ruolo della singola query presente nel circuito.
- Misureremo l'efficienza computazionale, per i prossimi algoritmi, nel contesto del modello query quantistico, contando il numero di quantum query presenti nel circuito.
- Rimarchiamo che in tale modello stiamo ignorando l'effettiva difficoltà nel costruire il quantum query gate U_f.
- Grazie al modello query metteremo in evidenza i possibili vantaggi dei computer quantistici rispetto a quelli classici, lasciando da parte, per ora, il calcolo specifico del costo computazionale di un algoritmo.

Algoritmo di Deutsch

- L'**algoritmo di Deutsch** è il primo esempio di algoritmo quantistico che vedremo. Permetterà di mettere in luce i vantaggi del calcolo quantistico.

- L'algoritmo nasce come soluzione al **problema di Deutsch** definito come segue: data una funzione $f: \{0,1\} \to \{0,1\}$ stabilire se essa è costante o bilanciata.

- Partiamo innanzitutto mostrando il circuito dell'algoritmo



- Se misurando il primo qubit otteniamo 0 allora concludiamo che f è costante altrimenti, se misuriamo 1 concludiamo che f è bilanciata.

Algoritmo di Deutsch

- Analizziamo passo passo gli step dell'algoritmo.

1) Inizializziamo l'input del circuito nello stato $|0\rangle \otimes |1\rangle$.

2) Applichiamo $H \otimes H$ al nostro stato in input. L'azione prodotto si distribuisce sul nostro stato prodotto, ottenendo così lo stato $|+\rangle \otimes |-\rangle$.

3) Applichiamo l'oracolo U_f:

$$U_f\left(\frac{1}{\sqrt{2}}|0\rangle|-\rangle + \frac{1}{\sqrt{2}}|1\rangle|-\rangle\right) = \frac{1}{\sqrt{2}}U_f|0\rangle|-\rangle + \frac{1}{\sqrt{2}}U_f|1\rangle|-\rangle$$

$$\frac{1}{\sqrt{2}}|0\rangle \otimes (-1)^{f(0)}|-\rangle + \frac{1}{\sqrt{2}}|1\rangle \otimes (-1)^{f(1)}|-\rangle = \left(\frac{(-1)^{f(0)}}{\sqrt{2}}|0\rangle + \frac{(-1)^{f(1)}}{\sqrt{2}}|1\rangle\right) \otimes |-\rangle$$

Algoritmo di Deutsch

4) Applichiamo la porta di Hadamard al primo qubit. Non eseguendo trasformazioni sul secondo qubit, applichiamo l'operatore $H \otimes I$ su tutto il registro

$$\left(\frac{(-1)^{f(0)}}{\sqrt{2}} H|0\rangle + \frac{(-1)^{f(1)}}{\sqrt{2}} H|1\rangle\right) \otimes |-\rangle = \left(\frac{(-1)^{f(0)}}{2}\left(|0\rangle + |1\rangle\right) + \frac{(-1)^{f(1)}}{2}\left(|0\rangle - |1\rangle\right)\right) \otimes |-\rangle$$

$$\left(\frac{(-1)^{f(0)} + (-1)^{f(1)}}{2}|0\rangle + \frac{(-1)^{f(0)} - (-1)^{f(1)}}{2}|1\rangle\right) \otimes |-\rangle$$

5) Ignoreremo il secondo qubit nello stato $|-\rangle$ e ci soffermeremo sul primo qubit, che chiameremo $|\psi\rangle$. Analizzando tale stato, dividendo nei due casi f costante e f bilanciata, risulta facile verificare che $|\psi\rangle$ si troverà, rispettivamente, nello stato $|0\rangle$ e nello stato $|1\rangle$. La misurazione di $|\psi\rangle$ dunque ci permette di ottenere il risultato desiderato.

Algoritmo di Deutsch

- L'algoritmo di Deutsch permette di risolvere il problema omonimo tramite l'utilizzo di **una singola query quantistica**.
- Qui si può percepire almeno in parte il vantaggio del calcolo quantistico rispetto a quello classico
- Se volessimo trovare la soluzione al problema di Deutsch con un computer classico, non basterebbe una singola query. Conoscendo il valore della funzione f sul bit x, non possiamo concludere nulla sul valore di f sulla negazione di x. Sono perciò necessarie **due query classiche** per ottenere la soluzione.
- Vedremo più approfonditamente questo vantaggio con una generalizzazione dell'algoritmo appena vista.

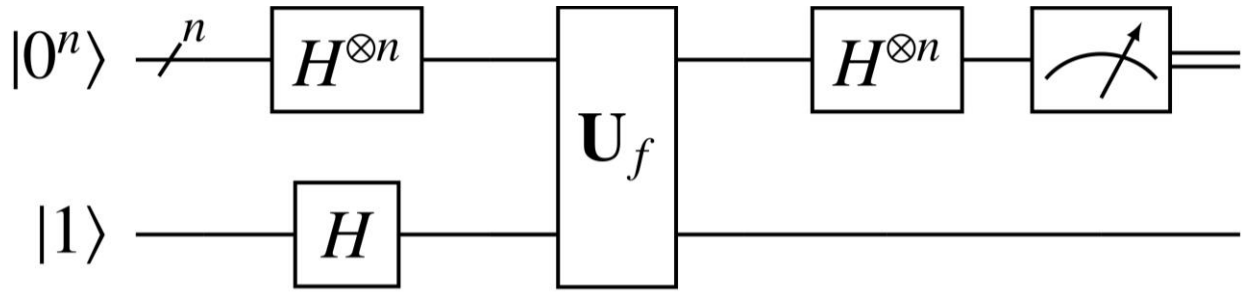
Esercizio

- Cosa succederebbe se nell'algoritmo di Deutsch inizializzassimo il circuito nello stato $|00\rangle$ invece che nello stato $|01\rangle$?

Algoritmo di Deutsch-Josza

- Generalizziamo l'algoritmo di Deutsch partendo da un problema più generale, chiamato **problema di Deutsch-Josza**: data una funzione $f: \{0,1\}^n \to \{0,1\}$, stabilire se f è costante o bilanciata (garantito il fatto che ci troviamo in uno di questi due casi).

- L'**algoritmo di Deutsch-Josza** porta alla soluzione del problema di Deutsch-Josza, implementando il seguente circuito



- Misureremo i primi n qubit: se otteniamo la stringa 0^n, f è costante altrimenti, se otteniamo una qualsiasi altra stringa concludiamo che f è bilanciata.

Algoritmo di Deutsch-Josza

- Analizziamo passo passo il circuito.

1) Inizializziamo l'input nello stato $|0^n\rangle \otimes |1\rangle = |0^n 1\rangle$.

2) Applichiamo $H^{\otimes n} \otimes H$ su tutto il registro:

$$\left(H^{\otimes n} \otimes H\right)\left(|0^n\rangle \otimes |1\rangle\right) = \left(H^{\otimes n}|0^n\rangle\right) \otimes \left(H|1\rangle\right) = \left(H^{\otimes n}|0^n\rangle\right) \otimes |-\rangle$$

$$H^{\otimes n}|0^n\rangle \otimes |-\rangle = \left(\frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} |y\rangle\right) \otimes |-\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{y\in\{0,1\}^n} \left(|y\rangle \otimes \left(|0\rangle - |1\rangle\right)\right)$$

Algoritmo di Deutsch-Josza

3) Applichiamo U_f ed otteniamo:

$$U_f\left(\frac{1}{\sqrt{2^{n+1}}}\sum_{y\in\{0,1\}^n}\left(|y\rangle\otimes(|0\rangle-|1\rangle)\right)\right)=\frac{1}{\sqrt{2^{n+1}}}\sum_{y\in\{0,1\}^n}U_f\left(|y\rangle\otimes(|0\rangle-|1\rangle)\right)$$

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{y\in\{0,1\}^n}\left(|y\rangle\otimes(|f(y)\rangle-|1\oplus f(y)\rangle)\right)$$

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{y\in\{0,1\}^n}\left(|y\rangle\otimes(-1)^{f(y)}(|0\rangle-|1\rangle)\right)=\left(\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{f(y)}|y\rangle\right)\otimes|-\rangle$$

Algoritmo di Deutsch-Josza

4) Infine l'ultima trasformazione da effettuare è $H^{\otimes n} \otimes I$:

$$H^{\otimes n}\left(\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{f(y)}|y\rangle\right)\otimes \mathbb{I}|-\rangle = \left(\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{f(y)}H^{\otimes n}|y\rangle\right)\otimes|-\rangle$$

$$\left(\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{f(y)}\left(\frac{1}{\sqrt{2^n}}\sum_{z\in\{0,1\}^n}(-1)^{y\bullet z}|z\rangle\right)\right)\otimes|-\rangle$$

$$\left(\frac{1}{2^n}\sum_{y}\sum_{z}(-1)^{f(y)+y\bullet z}|z\rangle\right)\otimes|-\rangle =: |\varphi_0\rangle\otimes|-\rangle$$

Algoritmo di Deutsch-Josza

5) Come ultimo step, misuriamo i primi n qubit. Non è così banale del perché si arrivi ad una soluzione. L'idea è che lo stato $|\varphi\rangle$ sul primo registro, nel caso in cui f è costante, non è una sovrapposizione. A questo scopo possiamo calcolare la probabilità di ottenere 0^n come stringa in output della misurazione:

$$\left|\frac{1}{2^n}\sum_y (-1)^{f(y)+0^n \bullet y}\right|^2 = \left|\frac{1}{2^n}\sum_y (-1)^{f(y)}\right|^2 = \begin{cases} 1 & \text{se } f \text{ è costante} \\ 0 & \text{se } f \text{ è bilanciata} \end{cases}$$

Algoritmo di Deutsch-Josza

- L'algoritmo descritto trova la soluzione al problema tramite l'implementazione di **una singola quantum query**.

- Paragonando con un'eventuale algoritmo classico, si nota la vera e proprio differenza di calcolo. Si ottiene qui uno **speed-up quantistico esponenziale**.

- Un computer classico necessiterebbe nel caso peggiore di $2^{n-1} + 1$ **query**. Dovremmo valutare f almeno su metà degli elementi del dominio, ma per distinguere con certezza tra costante e bilanciata è necessaria una valutazione in più.

- Naturalmente un modo più efficiente per risolvere il problema di Deutsch-Josza con un computer classico è quello di usare un algoritmo probabilistico invece di uno deterministico, permettendo dunque una certa soglia di errore probabilistica.