CPEN 400Q / EECE 571Q Lecture 05
Complexity, speedups, and working with oracles

Tuesday 25 January 2022

Announcements

- Assignment 0 and quiz 1 grades have been uploaded
- Assignment 1 due Thursday 23:59
- Quiz 2 at the end of class today
- TA problem-solving session **tomorrow** at 15:00 (look for link on Canvas/Piazza)

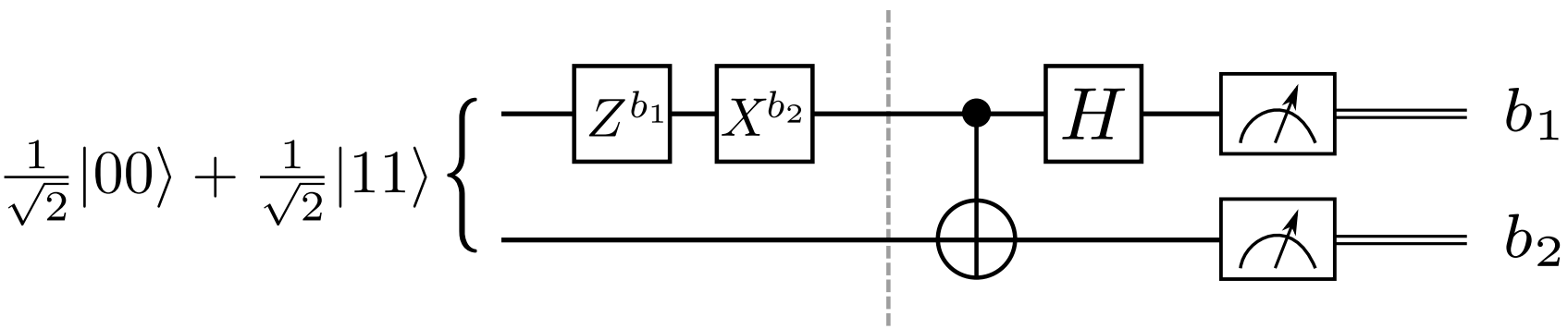
Last time

We learned how to take measurements of multi-qubit systems:

```
@qml.qnode(dev)
def circuit(x):
    qml.Hadamard(wires=0)
    qml.CRX(x, wires=[0, 1])
    return qml.expval(qml.PauliZ(0) @ qml.PauliZ(1))
```

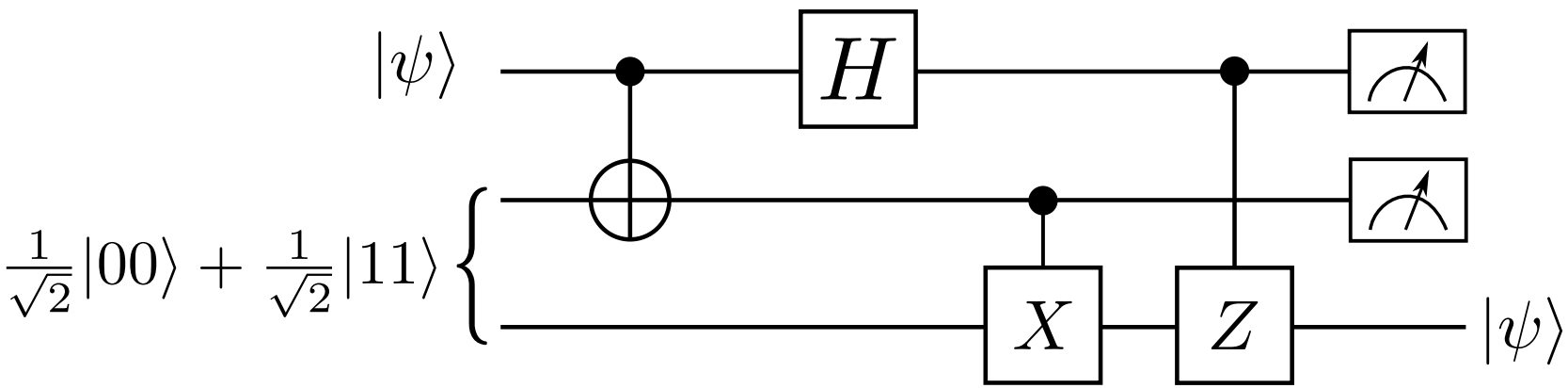
Last time

We used superdense coding to send two classical bits by sending only a single qubit.



Last time

We proved that we can't clone arbitrary quantum states, so started to explore quantum teleportation.



Learning outcomes

- Implement quantum teleportation
- Define and compute the query complexity of a quantum algorithm
- Apply the phase kickback technique
- Implement Deutsch's algorithm in PennyLane

Superdense coding & teleportation: take 2

The Bell basis

This entangled state,

$$|\Psi_{00}\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right),$$

has 3 siblings:

$$\begin{aligned}
|\Psi_{01}\rangle &= \frac{1}{\sqrt{2}}\left(|10\rangle + |01\rangle\right) \\
|\Psi_{10}\rangle &= \frac{1}{\sqrt{2}}\left(|00\rangle - |11\rangle\right) \\
|\Psi_{11}\rangle &= \frac{1}{\sqrt{2}}\left(-|10\rangle + |01\rangle\right)
\end{aligned}$$

The Bell basis

These 4 entangled states form an *orthonormal basis* for 2 qubits.

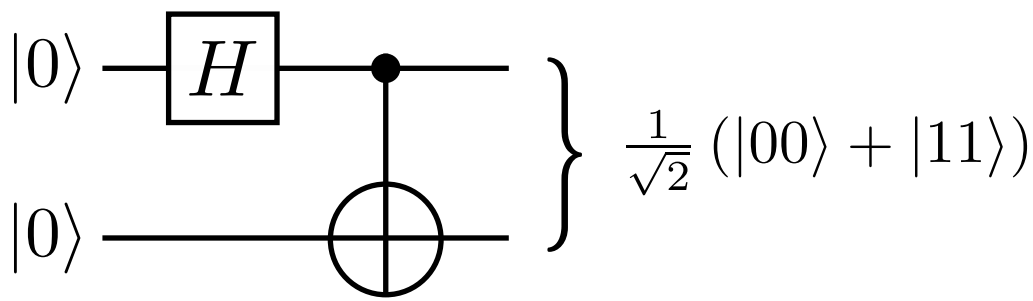
$$
\begin{aligned}
|\Psi_{00}\rangle &= \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \\
|\Psi_{01}\rangle &= \frac{1}{\sqrt{2}}\left(|10\rangle + |01\rangle\right) \\
|\Psi_{10}\rangle &= \frac{1}{\sqrt{2}}\left(|00\rangle - |11\rangle\right) \\
|\Psi_{11}\rangle &= \frac{1}{\sqrt{2}}\left(-|10\rangle + |01\rangle\right)
\end{aligned}
$$

The Bell basis

Remember how we created

$$|\Psi_{00}\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right),$$

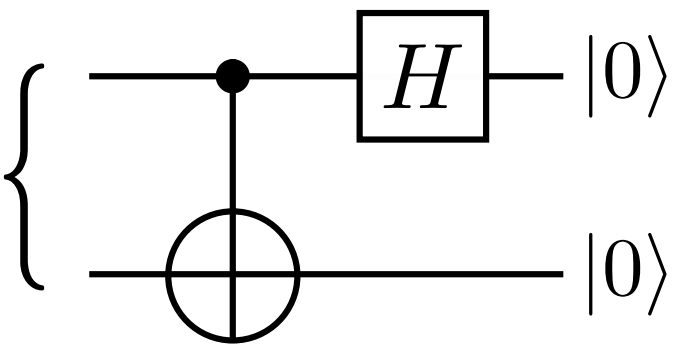
from the $|00\rangle$ state:



The Bell basis

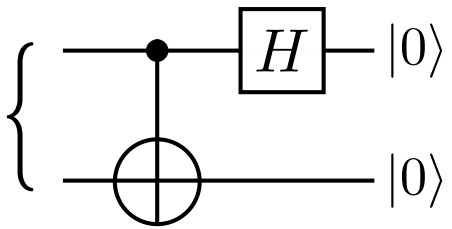
We can undo this by applying the operations in reverse:

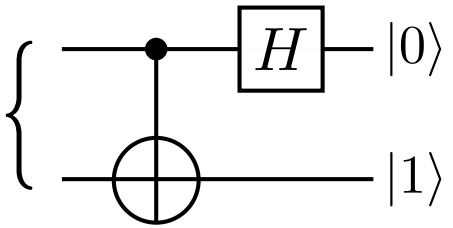
$$\frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

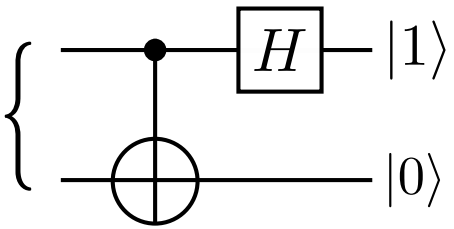


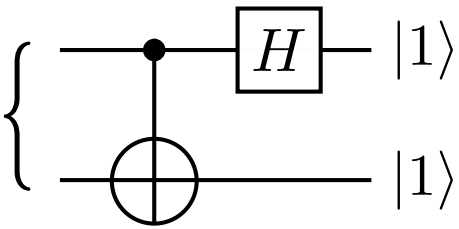
This sequence of operations actually corresponds to a basis rotation from the Bell basis to the computational basis...

The Bell basis

$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ 

$\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$ 

$\frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$ 

$\frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$ 

Both superdense coding and teleportation work by performing a measurement in the Bell basis (or, performing the above basis rotation, and measuring in the computational basis).

Superdense coding

Alice can send 2 classical bits to Bob by sending him just a single qubit, *if her and Bob share a pair of entangled qubits*:

$$|\Phi\rangle_{AB} = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

Alice will perform some operations on her qubit depending on which bit she wants to send. Then she will send her qubit to Bob.

Once Bob receives Alice's qubit, he measure both qubits to determine the bits she wanted to send him.

Superdense coding

Alice and Bob start the protocol with this entangled state:

$$|\Phi\rangle_{AB} = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

Next, depending on her bits, Alice performs one of the following operations on her qubit:

$$\begin{aligned} 00 &\rightarrow I \\ 01 &\rightarrow X \\ 10 &\rightarrow Z \\ 11 &\rightarrow ZX \end{aligned}$$

Superdense coding

What happened to the entangled state?

$$|\Phi\rangle_{AB} = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

It will transform to:

$$00 \quad \rightarrow_I$$

$$01 \quad \rightarrow_X$$

$$10 \quad \rightarrow_Z$$

$$11 \quad \rightarrow_{ZX}$$

Superdense coding

These are precisely the *Bell basis* states!

$$
\begin{aligned}
|\Psi_{00}\rangle &= \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \\
|\Psi_{01}\rangle &= \frac{1}{\sqrt{2}}\left(|10\rangle + |01\rangle\right) \\
|\Psi_{10}\rangle &= \frac{1}{\sqrt{2}}\left(|00\rangle - |11\rangle\right) \\
|\Psi_{11}\rangle &= \frac{1}{\sqrt{2}}\left(-|10\rangle + |01\rangle\right)
\end{aligned}
$$

Bob can perform a measurement in it to determine with certainty which state he has, and correspondingly which bits Alice sent him.

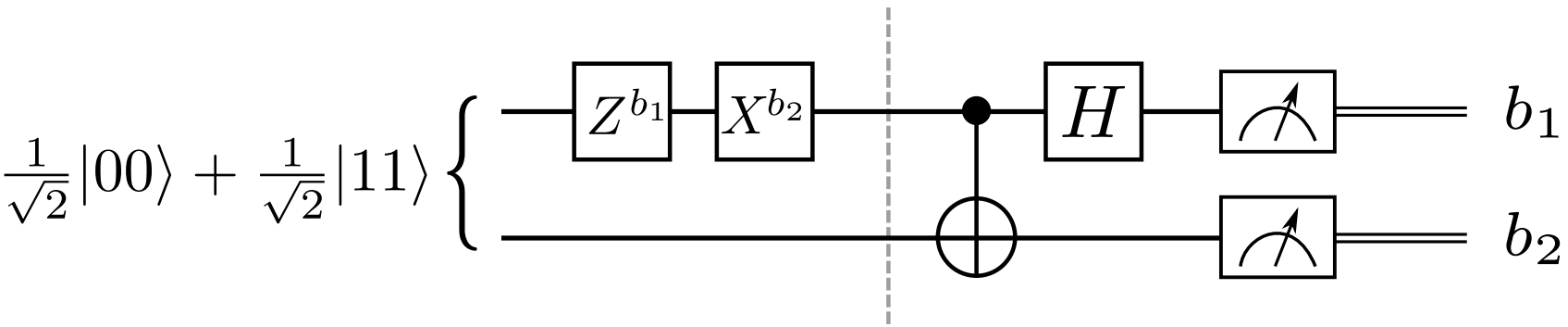
Superdense coding

Bob can measure directly in the Bell basis, or rotate back to the computational basis:

$$
\begin{aligned}
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) &= |00\rangle \\
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle) &= |01\rangle \\
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) &= |10\rangle \\
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) &= |11\rangle
\end{aligned}
$$

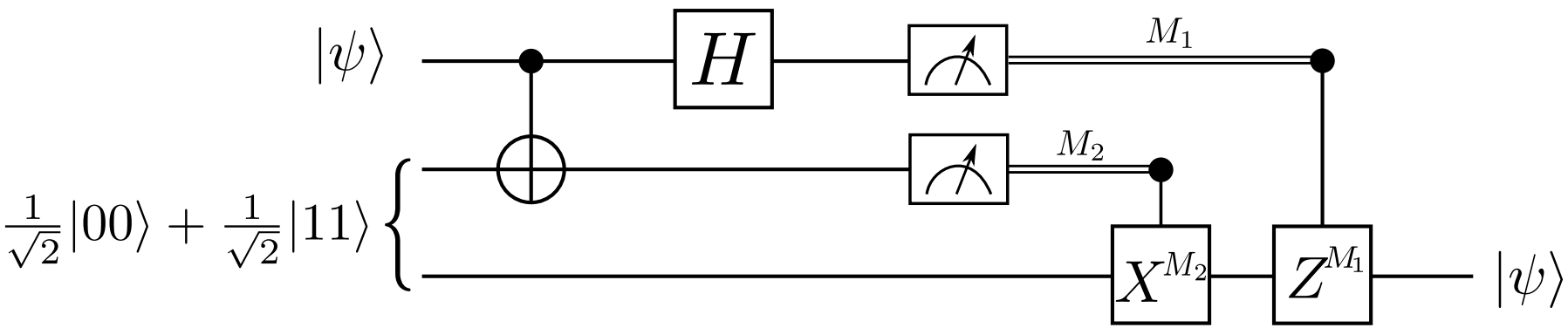
If he measures in computational basis, he obtains the bits directly.

Superdense coding: implementation



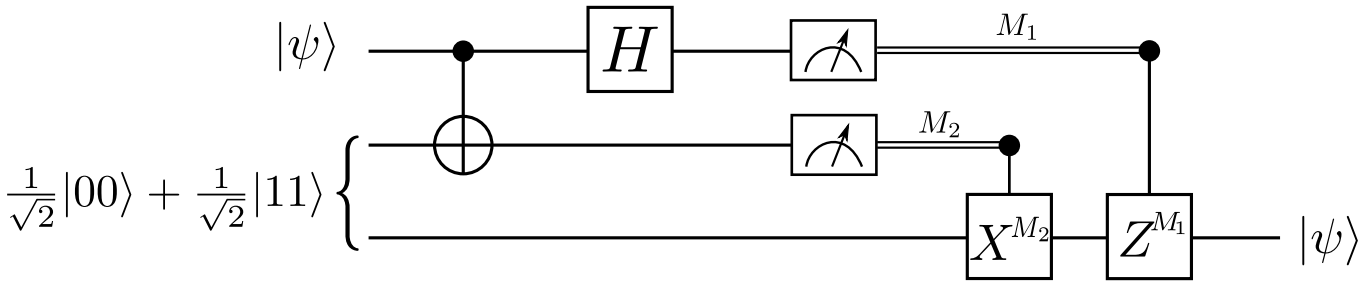
Teleportation

We cannot clone arbitrary qubit states, but we *can* teleport them!



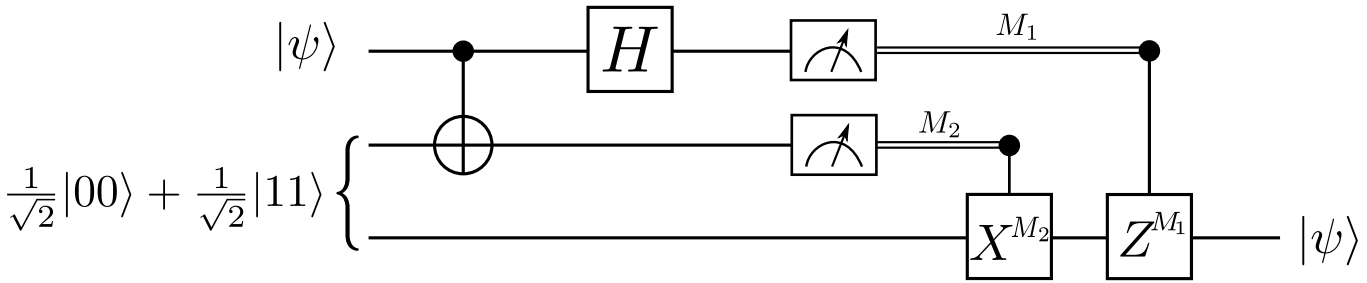
Quantum teleportation: the details

Let's go one gate at a time.



Quantum teleportation: the details

Let's go one gate at a time.



Quantum teleportation: the details

Before measurements, the combined state of the system is (removing the $\frac{1}{2}$ for readability):

$$\begin{aligned}
|00\rangle &\otimes (\alpha|0\rangle + \beta|1\rangle) + \\
|01\rangle &\otimes (\alpha|1\rangle + \beta|0\rangle) + \\
|10\rangle &\otimes (\alpha|0\rangle - \beta|1\rangle) + \\
|11\rangle &\otimes (\alpha|1\rangle - \beta|0\rangle)
\end{aligned}$$

This is a *uniform* superposition of 4 distinct terms. If we measure the first two qubits in the computational basis, we are equally likely to obtain each of the four outcomes.

Quantum teleportation: the details

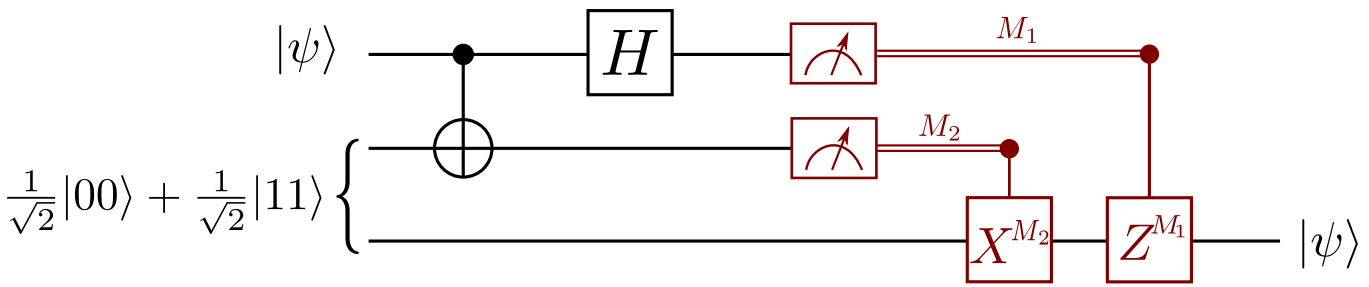
You can see that Bob's state is always some variation on the original state of Alice:

$$
\begin{aligned}
|00\rangle &\otimes (\alpha|0\rangle + \beta|1\rangle) + \\
|01\rangle &\otimes (\alpha|1\rangle + \beta|0\rangle) + \\
|10\rangle &\otimes (\alpha|0\rangle - \beta|1\rangle) + \\
|11\rangle &\otimes (\alpha|1\rangle - \beta|0\rangle)
\end{aligned}
$$

Quantum teleportation: the details

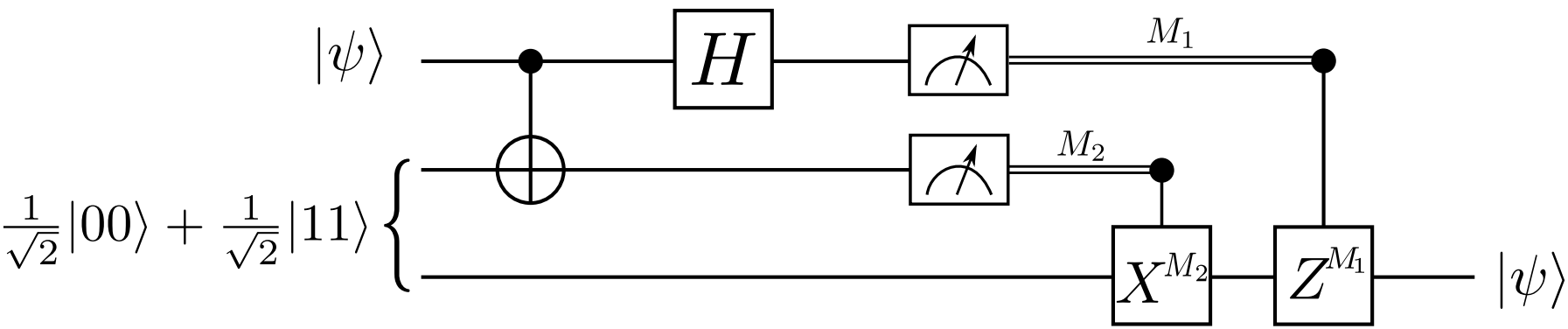
Alice measures in the computational basis and sends her results to Bob. Once Bob knows the results, he knows exactly what term of the superposition they had, and can adjust his state accordingly.

$$00 : I(\alpha|0\rangle + \beta|1\rangle) = (\alpha|0\rangle + \beta|1\rangle)$$
$$01 : X(\alpha|1\rangle + \beta|0\rangle) = (\alpha|0\rangle + \beta|1\rangle)$$
$$10 : Z(\alpha|0\rangle - \beta|1\rangle) = (\alpha|0\rangle + \beta|1\rangle)$$
$$11 : ZX(\alpha|1\rangle - \beta|0\rangle) = (\alpha|0\rangle + \beta|1\rangle)$$



Hands-on: teleportation

Need to deal with the fact that we cannot perform classically controlled operations in PennyLane...

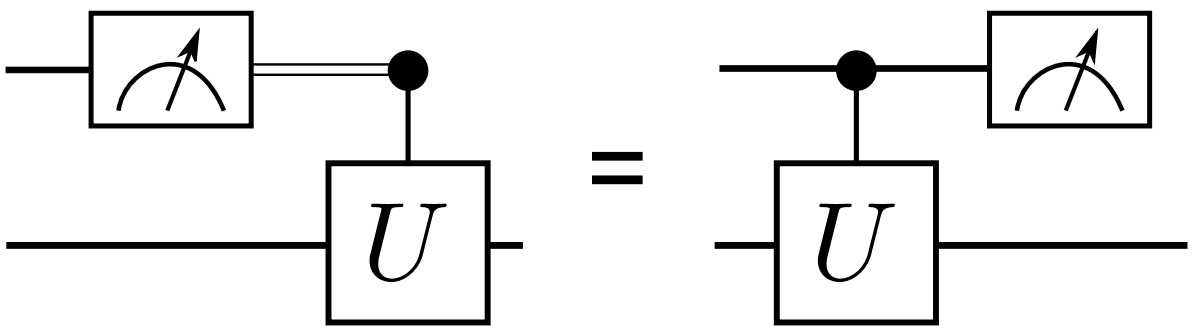


Principle of deferred measurement

Nielsen & Chuang:

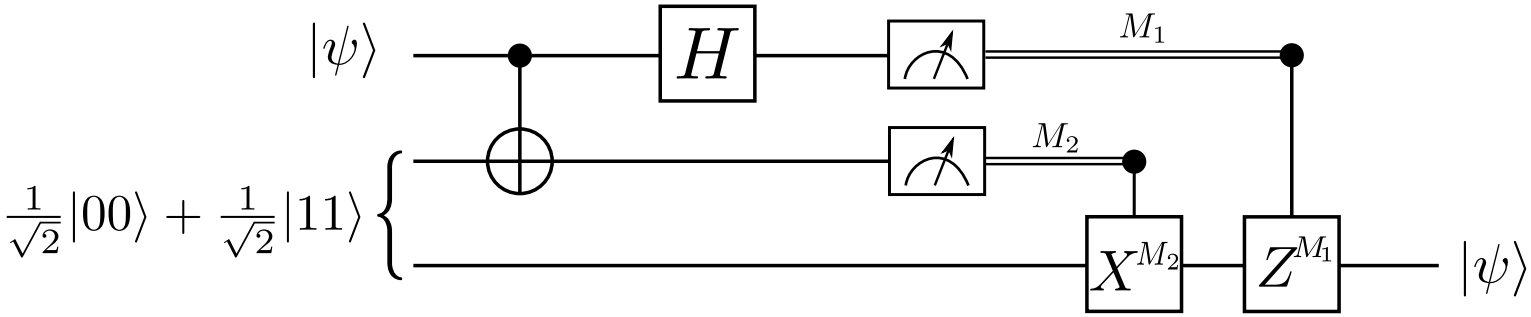
> *"Measurements can always be moved from an intermediate stage of a quantum circuit to the end of the circuit; if the measurement results are used at any stage of the circuit then the classically controlled operations can be replaced by conditional quantum operations."*

Basically,

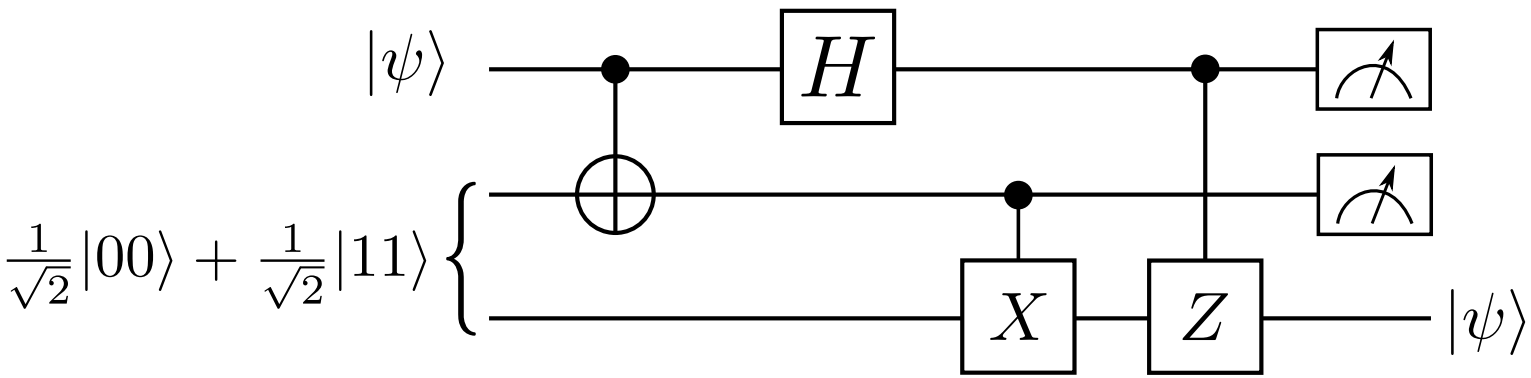


Principle of deferred measurement

Means we can replace our original circuit:



with

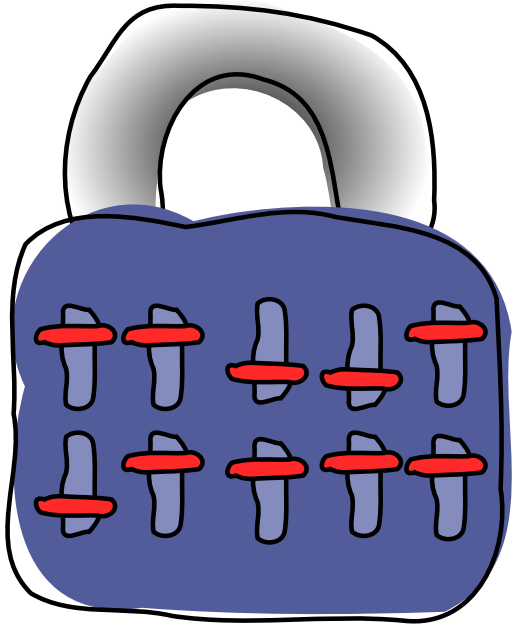


Let's implement it!

Oracles, queries, and Deutsch's algorithm

Motivating problem

Suppose we would like to find the combination for a "binary" lock:

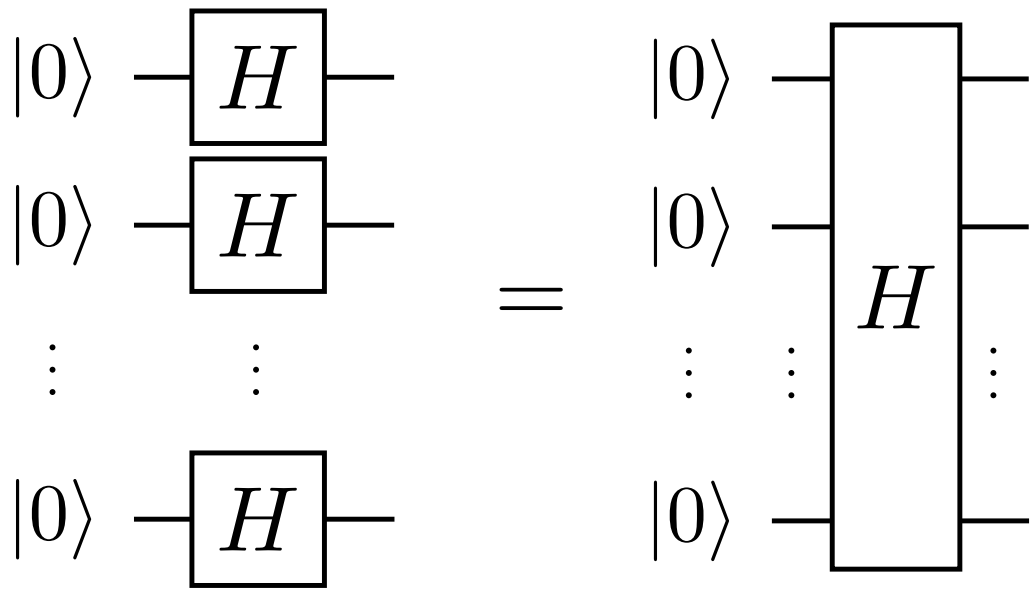


Classically, we would have to try every possible combination. If there are n bits, that's 2^n attempts in the worst case. Can we do better with a quantum computer?

Image credit: Codebook node A.1

Idea: use superposition

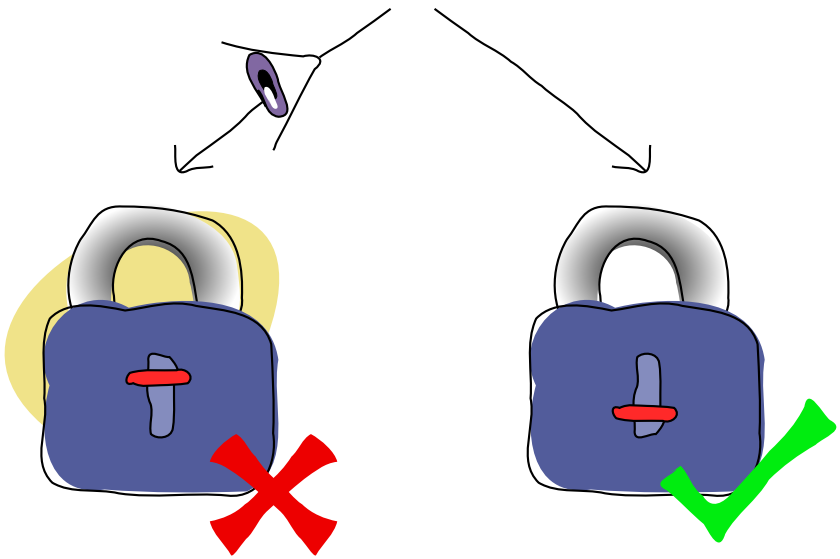
What if we take n qubits and put them in a superposition with all possible combinations?



Often called the *Hadamard transform*. Let's check that this works...

Idea: use superposition

Measurements are probabilistic - just because we put things into a uniform superposition of states, and our solution is "in" there, doesn't mean we are any closer to solving our problem.



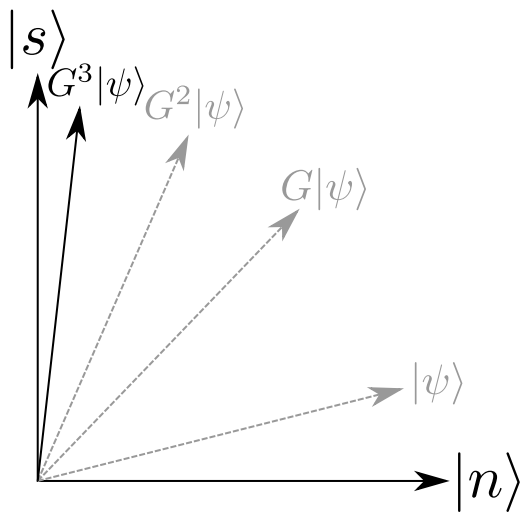
Quantum computers are **NOT** faster because they can "compute everything at the same time."

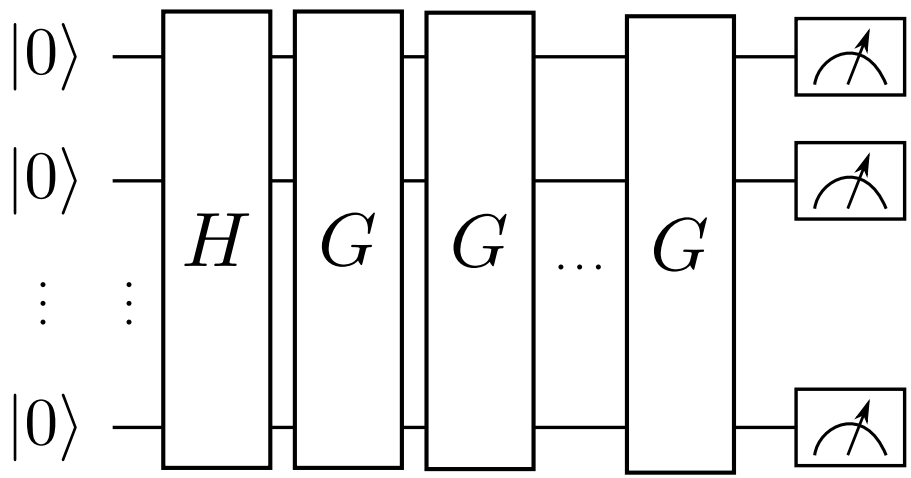
Image credit: Codebook node A.1

Solving problems with quantum computers

Can we solve this problem better with a quantum computer?

Yes: **amplitude amplification**, and **Grover's algorithm**. We will build up to this over the course of today and Thursday.





Today we will see some of the algorithmic primitives that are involved, and explore some smaller use cases where we can do better with quantum computing.

Oracles

Motivating problem

Suppose we would like to find combination for a "binary" lock:

Classically, we would have to try every possible combination. If there are n bits, that's 2^n possible tries. Can we do better with a quantum computer?

Image credit: Codebook node A.1

6 / 26

What is a "try"?

Oracles

We often express these tries as evaluations of a function that tells us whether we have found the correct answer.

Let

- $\mathbf{x}$ be an n-bit string that represents an input to the lock
- $\mathbf{s}$ be the solution to the problem (i.e., the correct combination)

We can represent trying a lock combination as a function:

$$f(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} = \mathbf{s} \\ 0 & \text{otherwise.} \end{cases}$$

Oracles

We don't necessarily care *how* this function gets evaluated, only that it gives us an answer (more specifically, a yes/no answer).

$$f(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} = \mathbf{s} \\ 0 & \text{otherwise.} \end{cases}$$

We consider this function as a black box, or an **oracle**.

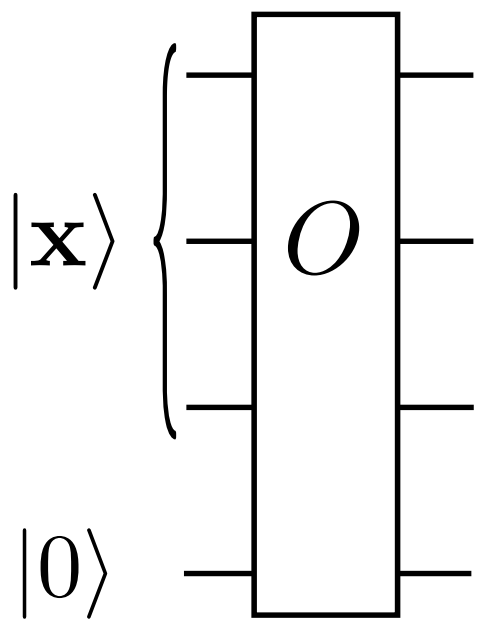
Every time we try a lock combination, we are **querying the oracle**. The amount of queries we make is the **query complexity**.

Quantum oracles

To solve this problem using quantum computing, we need some circuit that plays the role of the oracle.

Idea 1: encode the result in the state of an additional qubit.

$$O|\mathbf{x}\rangle|y\rangle = |\mathbf{x}\rangle|y \oplus f(\mathbf{x})\rangle$$

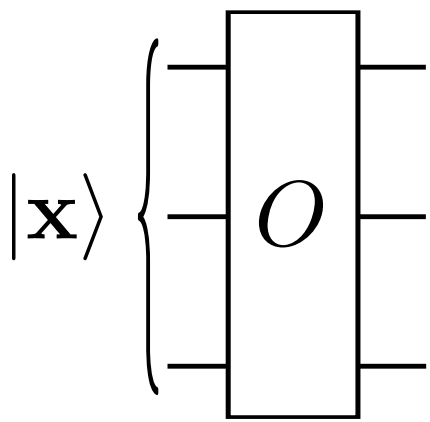


$$O|000\rangle|0\rangle = |000\rangle|0\rangle$$
$$O|001\rangle|0\rangle = |001\rangle|0\rangle$$
$$O|010\rangle|0\rangle = |010\rangle|0\rangle$$
$$O|011\rangle|0\rangle = |011\rangle|0\rangle$$
$$O|100\rangle|0\rangle = |100\rangle|0\rangle$$
$$O|101\rangle|0\rangle = |101\rangle|0\rangle$$
$$O|110\rangle|0\rangle = |110\rangle|1\rangle$$
$$O|111\rangle|0\rangle = |111\rangle|0\rangle$$

Quantum oracles

Idea 2: encode the result in the phase of a qubit.

$$O|\mathbf{x}\rangle = (-1)^{f(\mathbf{x})}|\mathbf{x}\rangle$$



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

Deutsch's algorithm

Motivation: You are given access to an oracle and are promised that it implements one of the following 4 functions:

Name	Action	Name	Action
f_1	$f_1(0) = 0$ $f_1(1) = 0$	f_2	$f_2(0) = 1$ $f_2(1) = 1$
f_3	$f_3(0) = 0$ $f_3(1) = 1$	f_4	$f_4(0) = 1$ $f_4(1) = 0$

Functions f_1 and f_2 are *constant* (same output no matter what the input), and f_3 and f_4 are balanced.

Deutsch's algorithm

How many queries do you need to make to the oracle to determine if the function is constant or balanced? (i.e., either one of f_1/f_2, or one of f_3/f_4).

Name	Action	Name	Action
f_1	$f_1(0) = 0$ $f_1(1) = 0$	f_2	$f_2(0) = 1$ $f_2(1) = 1$
f_3	$f_3(0) = 0$ $f_3(1) = 1$	f_4	$f_4(0) = 1$ $f_4(1) = 0$

Classical solution: 2

We always need to query both inputs 0 and 1 to find out the nature of the function.

Deutsch's algorithm

How many queries do you need to make to the oracle to determine if the function is constant or balanced? (i.e., either one of f_1/f_2, or one of f_3/f_4).

Name	Action	Name	Action
f_1	$f_1(0) = 0$ $f_1(1) = 0$	f_2	$f_2(0) = 1$ $f_2(1) = 1$
f_3	$f_3(0) = 0$ $f_3(1) = 1$	f_4	$f_4(0) = 1$ $f_4(1) = 0$

Quantum solution: 1
How???

Phase kickback

The secret lies in something called *phase kickback*.

What happens when we apply a CNOT to the following state?

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

We get

The control qubit is in $|0\rangle$, so it doesn't have any effect on the target qubit.

Phase kickback

What happens when we apply a CNOT to this state instead?

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

We get

It looks like we've changed the phase of the second qubit.

Phase kickback

But this is a *global* phase, and the math doesn't care which qubit it's attached to. We could equally well write

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

Now it's as if the *target* qubit has done something to the *control* qubit!

We say that the phase has been "kicked back" from the second qubit to the first.

Phase kickback

We can write a general version of this effect:

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

But what does this have to do with Deutsch's algorithm and figuring out if a function is constant or balanced?

Deutsch's algorithm

This is where our oracle comes in. Suppose we have a black box, U_f, that implements any of these four functions, f:

$$U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$$

Setting $|y\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ will allow us to 'extract' the value of $f(0) \oplus f(1)$ with just a single query.

Let's work through the math.

Deutsch's algorithm

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

$$=$$

If $f(x) = 0$, we get

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

If $f(x) = 1$, we get

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

Deutsch's algorithm

So just like the case of the CNOT where we wrote the general version

$$CNOT\left(|b\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\right)=(-1)^b|b\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right),$$

we can write

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

Essentially, before the CNOT was just playing the role of U_f for the specific function $f(0)=0, f(1)=1$.

Deutsch's algorithm

This doesn't look like much on its own - we want to get a *combination* of $f(0)$ and $f(1)$. How can we do this?

By setting $|x\rangle$ to be a superposition!

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

$$=$$

$$=$$

Let's pull out a phase factor of $(-1)^{f(0)}$, since global phase doesn't matter anyways.

Deutsch's algorithm

$$=$$

$$=$$

Now let's look at how this state is different when f is a constant vs. a balanced function.

Deutsch's algorithm

If the function is constant, $f(0) \oplus f(1) = 0$ and the state is

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

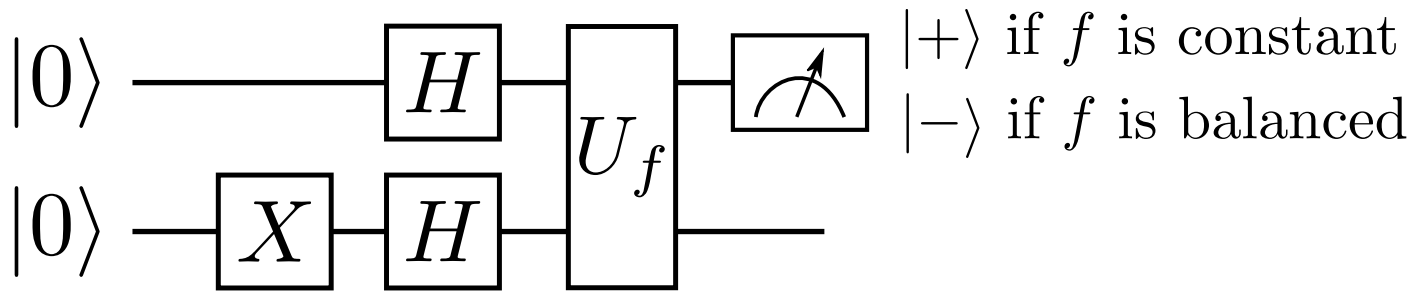
But if the function is balanced, $f(0) \oplus f(1) = 1$ and the state is

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

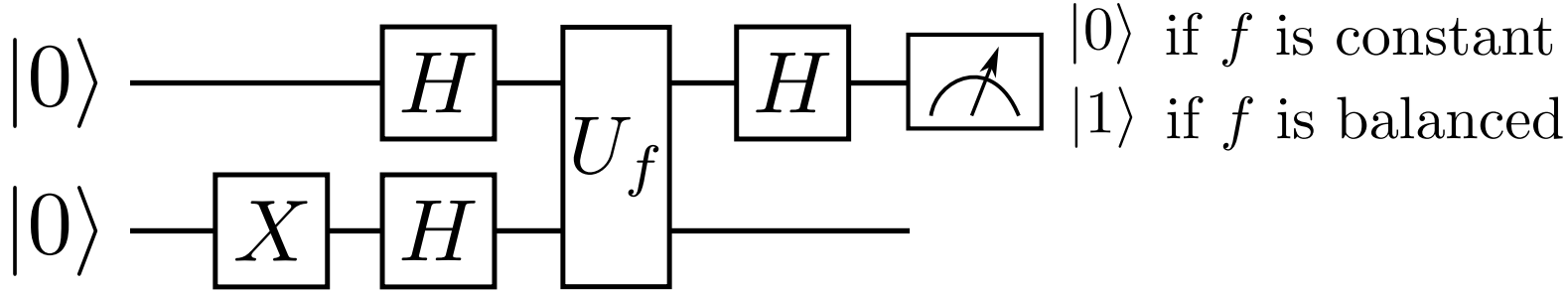
We can measure the first qubit in the *Hadamard basis* to determine exactly the value of $f(0) \oplus f(1)$!

Deutsch's algorithm

As a circuit, Deutsch's algorithm looks like this:

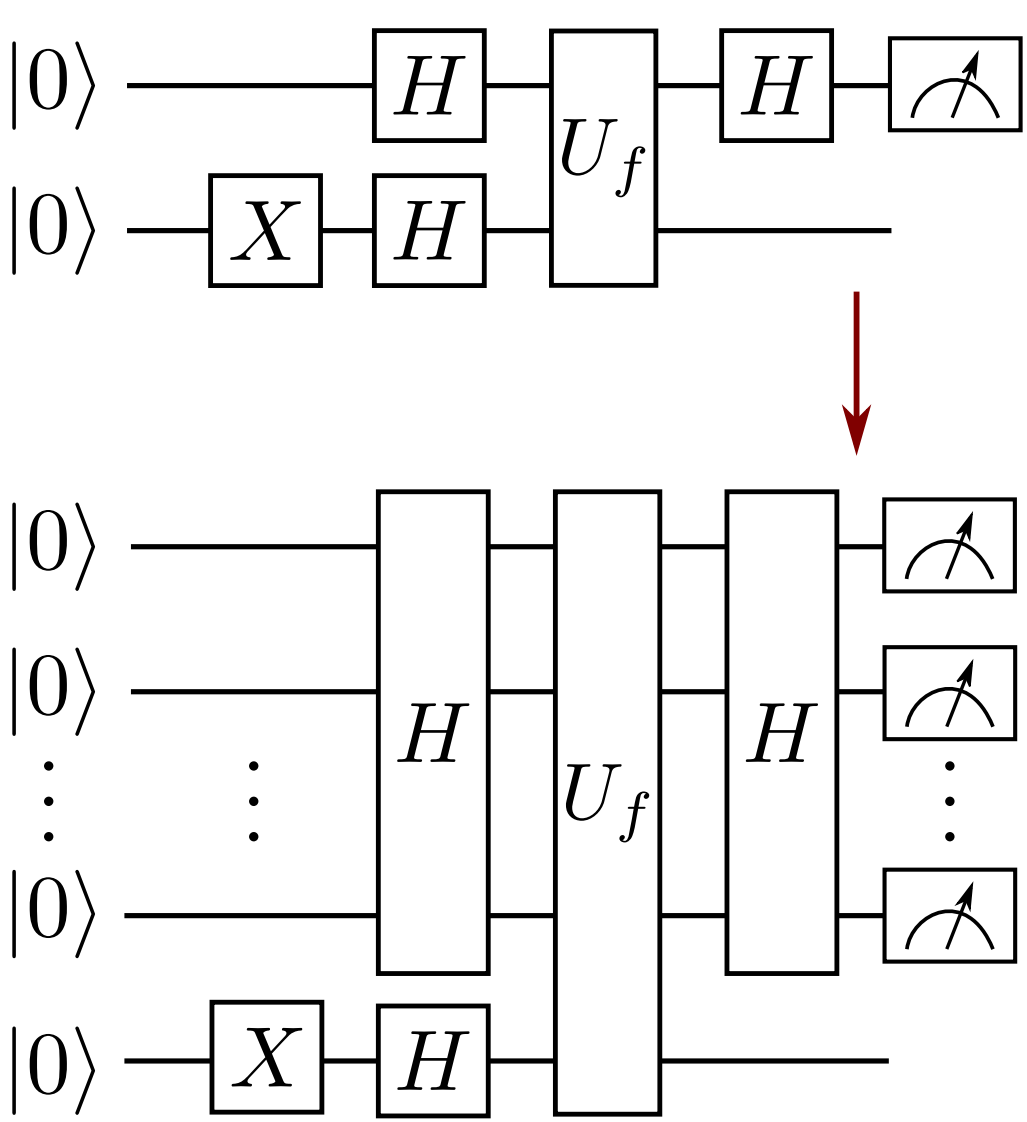


Or equivalently,



We call U_f just once, but obtain information about the value of both $f(0)$ and $f(1)$! Let's implement it.

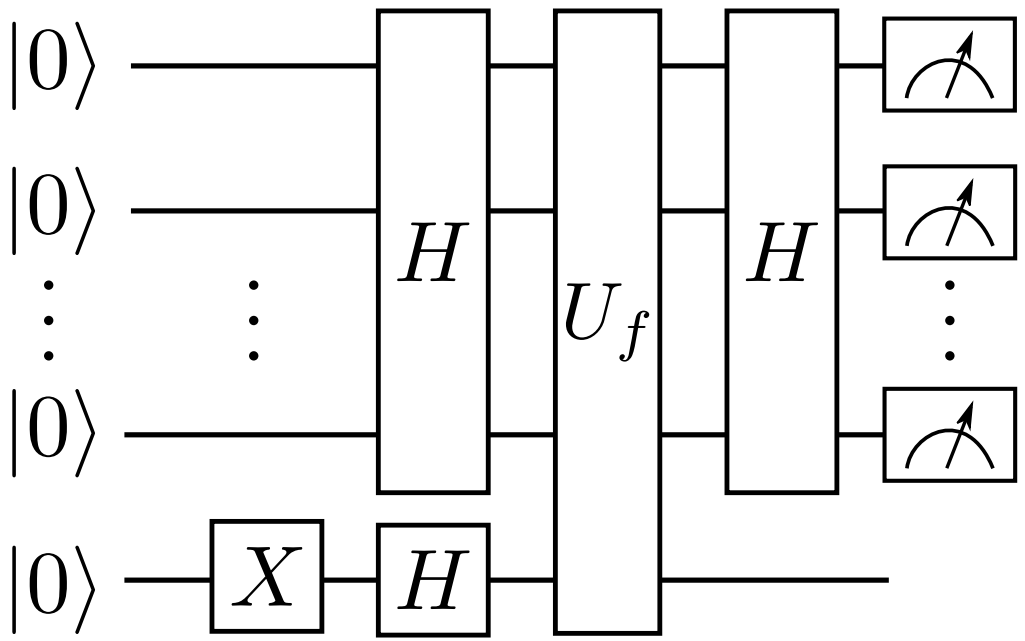
Generalization: Deutsch-Jozsa algorithm



(Challenge: try implementing it yourself to check if this works!)

Generalization: Deutsch-Jozsa algorithm

$2^{n-1} + 1$ classical queries in worst case; still only 1 quantum query.



$|00\cdots0\rangle$ if f is constant

anything but $|00\cdots0\rangle$ if balanced

Recap

- Implement quantum teleportation
- Define and compute the query complexity of a quantum algorithm
- Apply the phase kickback technique
- Implement Deutsch's algorithm in PennyLane

Next time

Content:

- Amplitude amplification
- Grover's algorithm

Action items:

1. Finish up Assignment 1

Recommended reading:

- Codebook nodes I.15, A.1-A.6
- Nielsen & Chuang 1.3.5-1.3.7, 1.4.2-1.4.4, 2.3,

Quiz time...