Introduction to Quantum Computing

Qibo training for DSO

Andrea Pasquale on the behalf of the Qibo team

5th October 2022

Institute of
High Performance
Computing
IHPC

UNIVERSITÀ
DEGLI STUDI
DI MILANO

TII

CQT

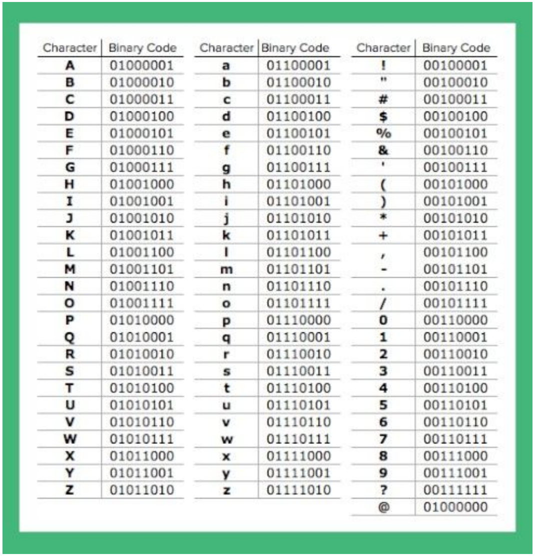
Outline

1. Basic elements of classical logic
2. Brief overview on Quantum Mechanics
3. Quantum Computing
4. Applications
5. Introduction to quantum algorithms
6. Towards Quantum Machine Learning
7. Quantum Technology
8. Languages for quantum computing

Basic elements of classical logic

Binary code

A classical computer operates on string of zeros and ones, also known as binary code.



Character	Binary Code	Character	Binary Code	Character	Binary Code
A	01000001	a	01100001	!	00100001
B	01000010	b	01100010	"	00100010
C	01000011	c	01100011	#	00100011
D	01000100	d	01100100	$	00100100
E	01000101	e	01100101	%	00100101
F	01000110	f	01100110	&	00100110
G	01000111	g	01100111	'	00100111
H	01001000	h	01101000	(	00101000
I	01001001	i	01101001	)	00101001
J	01001010	j	01101010	*	00101010
K	01001011	k	01101011	+	00101011
L	01001100	l	01101100	,	00101100
M	01001101	m	01101101	-	00101101
N	01001110	n	01101110	.	00101110
O	01001111	o	01101111	/	00101111
P	01010000	p	01110000	0	00110000
Q	01010001	q	01110001	1	00110001
R	01010010	r	01110010	2	00110010
S	01010011	s	01110011	3	00110011
T	01010100	t	01110100	4	00110100
U	01010101	u	01110101	5	00110101
V	01010110	v	01110110	6	00110110
W	01010111	w	01110111	7	00110111
X	01011000	x	01111000	8	00111000
Y	01011001	y	01111001	9	00111001
Z	01011010	z	01111010	?	00111111
				@	01000000

Binary code

A binary string is composed by several binary variables, which are categorical variables which can take only one of two values (usually denoted by 0 and 1).

A **bit** is defined as the amount of information carried by a binary variable.

We can represent the state of a bit using the following notation:

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

Therefore the state of 4 classical bits representing 1010 can be represented as

$$|1\rangle\,|0\rangle\,|1\rangle\,|0\rangle$$

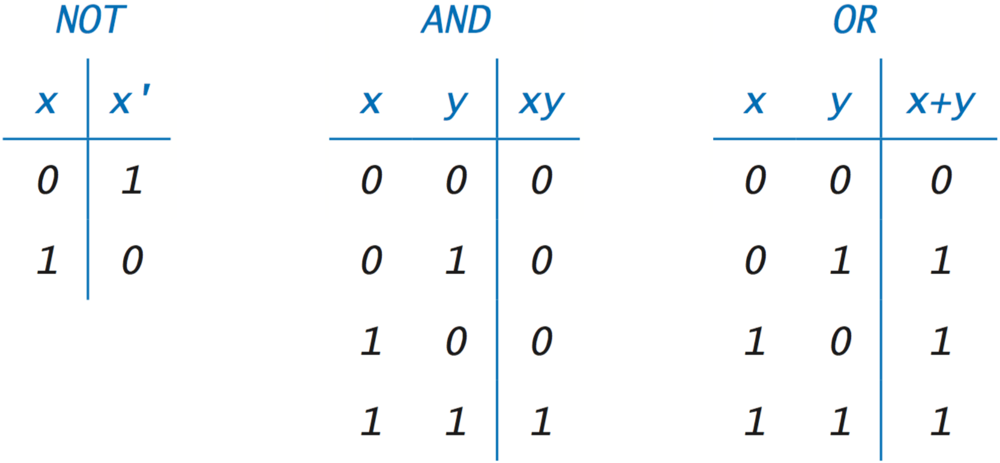
It is also possible to associate with $|0\rangle$ and $|1\rangle$ two column vectors:

$$|0\rangle \rightarrow \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad |1\rangle \rightarrow \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Classical logical operations

We can change the value of a bit using logical opeartions.

It can be shown that **any** logical or arithmetical operation can be obtained by the composition of three elementary logical operations:



NOT

x	x '
0	1
1	0

AND

x	y	xy
0	0	0
0	1	0
1	0	0
1	1	1

OR

x	y	x+y
0	0	0
0	1	1
1	0	1
1	1	1

Reversible logical operations

In Quantum Computing we are mainly interested in reversible operation.

Reversible operation

A logic function is **reversible** if each output arises from a unique input.

What is the only nontrivial reversible operation that we can apply to a single bit?

Reversible logical operations

In Quantum Computing we are mainly interested in reversible operation.

Reversible operation

A logic function is **reversible** if each output arises from a unique input.

What is the only nontrivial reversible operation that we can apply to a single bit?

The **NOT** operation, denoted by the symbol **X** which flips the state of a bit

$$\mathbf{X} : |0\rangle \rightarrow |1\rangle$$
$$\mathbf{X} : |1\rangle \rightarrow |0\rangle$$

Using the vector notation we can represent this gate as a matrix

$$\mathbf{X} \rightarrow \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Not is reversible since if we apply **X** a second time we go back to the original state

$$\mathbf{X}^2 = \mathbf{I}$$

Two-bit reversible operations: SWAP

What about two-bit reversible operations?

The most general reversible operation on two bits is any permutation of their four possible states. We will show two gates **SWAP** and **CNOT**.

Two-bit reversible operations: SWAP

What about two-bit reversible operations?

The most general reversible operation on two bits is any permutation of their four possible states. We will show two gates **SWAP** and **CNOT**.

The **SWAP** gate, S_{ij}, exchanges the states of the bit i and j. For a two-bit system we get the following:

$$\mathsf{S}_{10} |x\rangle |y\rangle = |y\rangle |x\rangle$$

In particular we have:

- $\mathsf{S}_{10} |0\rangle |0\rangle = |0\rangle |0\rangle$
- $\mathsf{S}_{10} |0\rangle |1\rangle = |1\rangle |0\rangle$
- $\mathsf{S}_{10} |1\rangle |0\rangle = |0\rangle |1\rangle$
- $\mathsf{S}_{10} |1\rangle |1\rangle = |1\rangle |1\rangle$

which corresponds to the following matrix representation:

$$\mathsf{S}_{01} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Two-bit reversible operations: CNOT

One of the most important reversible operation, especially for quantum computing, is the *controlled-NOT* operator, $\mathbf{C}_{ij}$, which implement the following operation:

- if $|i\rangle = |0\rangle \Rightarrow$ do nothing
- if $|i\rangle = |1\rangle \Rightarrow$ apply NOT operator to $|j\rangle$

which can be summarized in the following compact form

- $\mathbf{C}_{01} |x\rangle |y\rangle = |x\rangle |y \oplus x\rangle$
- $\mathbf{C}_{01} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

- $\mathbf{C}_{10} |x\rangle |y\rangle = |x \oplus y\rangle |y\rangle$
- $\mathbf{C}_{10} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$

It is clear that the CNOT gate acts as a generalized XOR gate [1].

[1] The operator $\oplus$ corresponds to summing the two bits modulo 2. So for example we have $(0+1)\text{mod}2 = 1\text{mod}2 = 1 = 0$ **XOR** 1.

Manipulating operations on Classical bits

It is also instructive to introduce the operator **N**:

$$\mathbf{N}\left|x\right\rangle = x\left|x\right\rangle$$

which projects onto the state $\left|1\right\rangle$.

And its complementary

$$\tilde{\mathbf{N}} = \mathbf{1} - \mathbf{N}$$

which projects onto the state $\left|0\right\rangle$.

The corresponding matrices are

$$\mathbf{N} \to \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad \tilde{\mathbf{N}} \to \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

Manipulating operations on Classical bits

We can also introduce the **Z** which is defined as

$$\mathbf{Z} = \tilde{\mathbf{N}} - \mathbf{N} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

which anticommutes with **X** since $\mathbf{XZ} = -\mathbf{ZX}$.

Finally we introduce the Hadamard transformation which is defined as

$$\mathbf{H} = \frac{1}{\sqrt{2}}(\mathbf{X} + \mathbf{Z}) \to \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

. From a classical point of view **H** is meaningless since it transforms a single bit state into a linear combinations of states

$$\mathbf{H} |0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \quad \text{and} \quad \mathbf{H} |1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Brief overview on Quantum Mechanics

Postulates of quantum mechanics

The postulates of quantum mechanics are a list of prescription to summarize:

Postulates of quantum mechanics

The postulates of quantum mechanics are a list of prescription to summarize:

1. how to describe the state of a physical system

Postulates of quantum mechanics

The postulates of quantum mechanics are a list of prescription to summarize:

1. how to describe the state of a physical system
2. how to describe the measurement performed on a physical system

Postulates of quantum mechanics

The postulates of quantum mechanics are a list of prescription to summarize:

1. how to describe the state of a physical system
2. how to describe the measurement performed on a physical system
3. how to describe the evolution of a physical system

Postulate 1: State of a quantum system

Each physical system is associated with a complex Hilbert space $\mathcal{H}$. A state is a normalized vector $|\psi\rangle$, which contains all the information about the system. Futhermore, in QM the superposition principle holds:

Superposition principle

If $|\psi\rangle$ and $|\phi\rangle$ are possibile states of a quantum system a state of the form

$$|\xi\rangle = \alpha\,|\psi\rangle + \beta\,|\phi\rangle$$

such that $\langle\xi|\xi\rangle = 1$ is an admissible state of the system, with α and $\beta \in \mathbb{C}$.

For a composite system we have:

$$|\Psi\rangle = |\psi\rangle_1 \otimes \cdots \otimes |\psi\rangle_N \in \mathcal{H}$$

where $\mathcal{H}$ is the tensor product of the Hilbert spaces associated with each system.

Postulate 2: Quantum measurement

Observable quantities are described by Hermitian operators $A = A^\dagger$, therefore the operator A admits a spectral decomposition in terms of real eigenvalues a_i, which are the possible values of the observable.

The probability of obtaining the outcome a_i from the measurement of A in a given state ψ is equal to

$$p(a_i) = |\langle u_i|\psi\rangle|^2$$

The overall expectation value of the observable A is computed as

$$\langle A\rangle = \langle\psi|A|\psi\rangle$$

If we measure the outcome a_i the state of the system collapse to the correspoding eigeinvector of the observable A, such that:

$$A\,|\alpha_i\rangle = a_i\,|\alpha_i\rangle \quad \Rightarrow |\psi\rangle \rightarrow |\alpha_i\rangle$$

Postulate 3: Dynamics of a quantum system

The dynamical evolution of a physical system from an initial time t_0 to a time $t \geq t_0$ is described by a unitary operator $\mathbf{U}(t, t_0)$

$$|\psi_t\rangle = \mathbf{U}(t, t_0) |\psi_0\rangle$$

.

$\mathbf{U}(t, t_0)$ is linked to the *Hamiltonian* of the quantum system through the Schrödinger equation

$$i\hbar \frac{\partial}{\partial t} |\psi_t\rangle = H |\psi_t\rangle$$

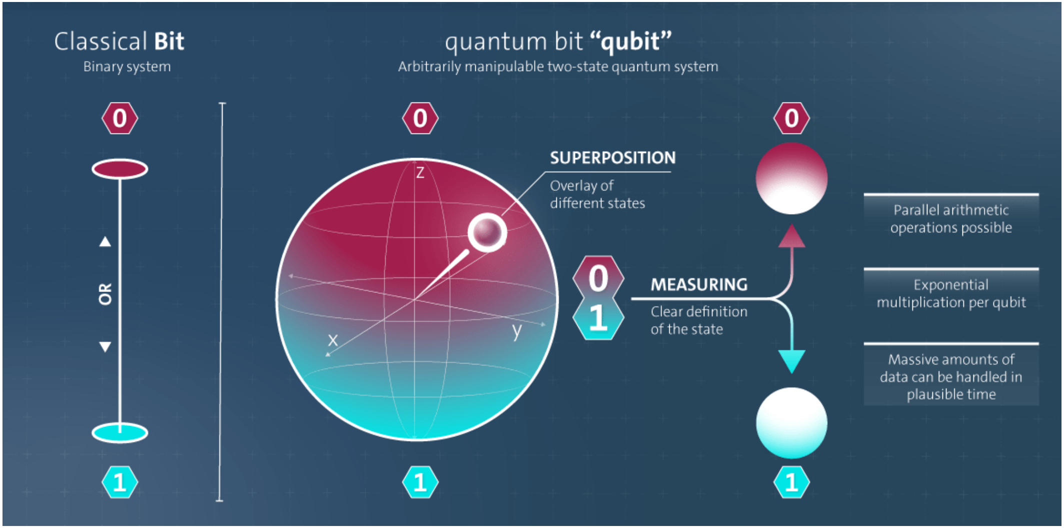
where H is the Hamiltonian of the system.

For example, if the Hamiltonian is time independent, after solving the Schrödinger equation we find the following solution for $\mathbf{U}(t, t_0)$:

$$\mathbf{U}(t, t_0) = \exp(-iH(t - t_0))$$

Quantum Computing

What is a Qubit?


Classical Bit
Binary system
0
OR
1
quantum bit "qubit"
Arbitrarily manipulable two-state quantum system
0
z
x
y
1
SUPERPOSITION
Overlay of different states
0
1
MEASURING
Clear definition of the state
0
1
Parallel arithmetic operations possible
Exponential multiplication per qubit
Massive amounts of data can be handled in plausible time

What is a Qubit?

A state ψ associated with a qubit is a generic two-level quantum system:

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \rightarrow \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

where α and β are two complex numbers constrained by the requirement that $|\psi\rangle$, like $|0\rangle$ and $|1\rangle$ should be a unit vector in the complex vector space.

This means that the following normalization must hold

$$|\alpha|^2 + |\beta|^2 = 1$$

We can see that contrary to the classical case a qubit is associated with multiple states, and the most general expression is a superposition of the classical states $|0\rangle$ and $|1\rangle$.

Bloch sphere

Since $|\alpha|^2 + |\beta|^2 = 1$ we can parametrize the amplitudes of the qubit state in the following way:

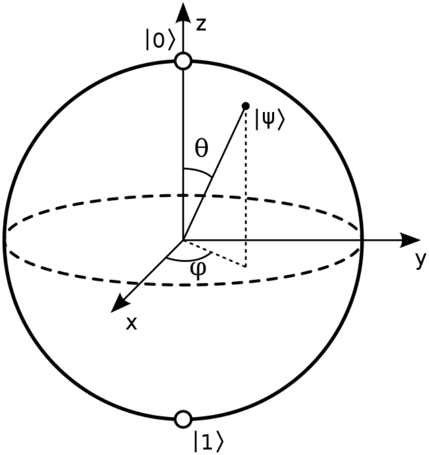
$$\alpha = \cos\frac{\theta}{2} \quad \text{and} \quad \beta = e^{i\phi}\sin\frac{\theta}{2}$$

We can associate with the qubit the following real numbers:

$$r_x = \sin\theta\cos\phi \quad r_y = \sin\theta\sin\phi \quad r_z = \cos\theta$$

which can be seen as the components of a 3-dimensional vector

$$\mathbf{r} = \begin{pmatrix} r_x \\ r_y \\ r_z \end{pmatrix} = \begin{pmatrix} \sin\theta\cos\phi \\ \sin\theta\sin\phi \\ \cos\theta \end{pmatrix}$$



Multiple qubit states

Just as the general state of a single qubit is any normalized superposition of two possible classical state, the general state that the nature allows us to associate with two qubits is any normalized superposition of the four orthogonal classical states

$$|\Psi\rangle = \alpha_{00}\,|00\rangle + \alpha_{01}\,|01\rangle + \alpha_{10}\,|10\rangle + \alpha_{11}\,|11\rangle$$

with the normalization condition $|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$

For a n qubit system we have the following expression

$$\Psi = \sum_{x=0}^{2^n-1} \alpha_x\,|x\rangle_n \quad \text{with} \quad \sum_{x=0}^{2^n-1} |\alpha_x|^2 = 1$$

We observe that the we are summing over 2^n values since there are 2^n classical states that one could obtain by looking at all the possible tensor products.

Reversible operations on qubits

We have already observed that the only nontrivial reversible operation that can performed on a classical bit is the NOT operation **X**.

In QM we can modify a state by applying an *unitary* transformation, see postulate 3, **u**, which by definition satisfy the following condition

$$\mathbf{u}\mathbf{u}^\dagger = \mathbf{u}^\dagger\mathbf{u} = 1$$

Since any unitary transformation has a unitary inverse, such actions of a quantum computer on a qubit are fully reversible.

The previous equation holds for a generic system with n qubits, in particular we have that in order to modify a n qubit system we can act with a $2^n \times 2^n$ unitary matrix.

Quantum logic gates

It is possible to represent schematically the action of a unitary transformation $\mathbf{u}$ on a qubit state $|\psi\rangle$ using the following picture

$$|\psi\rangle \;\rule[0.5ex]{3em}{0.4pt}\boxed{\mathbf{u}}\rule[0.5ex]{3em}{0.4pt}\; \mathbf{u}\,|\psi\rangle = |\psi'\rangle$$

which in quantum computing is called a **quantum circuit**.

What happens if we apply more than one gate?

Suppose that we apply two unitary transformations $\mathbf{u}_1$ and $\mathbf{u}_2$ on the state $|\psi\rangle$

$$|\psi'\rangle = \mathbf{u}_1\mathbf{u}_2\,|\psi\rangle$$

We can again represent these transformations using a quantum circuit.

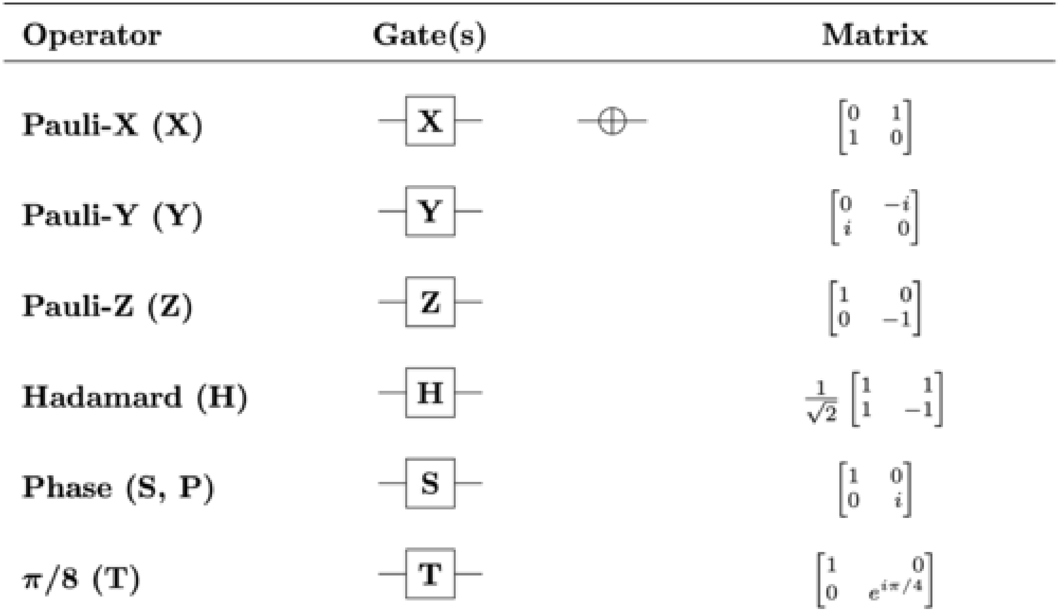
$$|\psi\rangle \;\rule[0.5ex]{3em}{0.4pt}\boxed{\mathbf{u}_2}\rule[0.5ex]{3em}{0.4pt}\boxed{\mathbf{u}_1}\rule[0.5ex]{3em}{0.4pt}\; |\psi'\rangle$$

Observation

The sequence of symbols is reverse from the sequence in which they appear in the mathematical expression!

One qubit gates

Here are the most common used one qubit gates.



Operator	Gate(s)	Matrix
Pauli-X (X)	$-\boxed{\mathbf{X}}-$ $-\oplus-$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y (Y)	$-\boxed{\mathbf{Y}}-$	$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z (Z)	$-\boxed{\mathbf{Z}}-$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Hadamard (H)	$-\boxed{\mathbf{H}}-$	$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Phase (S, P)	$-\boxed{\mathbf{S}}-$	$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$ (T)	$-\boxed{\mathbf{T}}-$	$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$

Lets have a closer look to some of them.

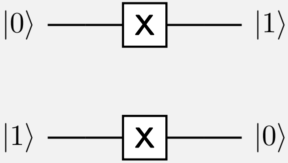
X and Z gates

X gate

The **X** gate acts as the classical NOT gate, it is represented by σ_x

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

therefore we get

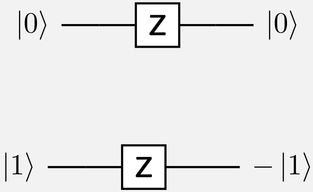


Z gate

The **Z** gate flips the sign of $|1\rangle$, it is represented by the σ_z

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

therefore we get

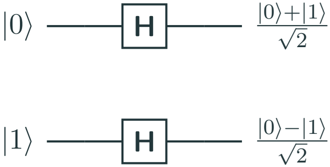


Hadamard gate

The Hadamard gate **H** is represented by the following matrix

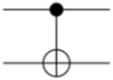
$$\mathbf{H} \rightarrow \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

It is extremely important in quantum computing since it can create a superposition of states


$|0\rangle$
H
$\frac{|0\rangle+|1\rangle}{\sqrt{2}}$
$|1\rangle$
H
$\frac{|0\rangle-|1\rangle}{\sqrt{2}}$

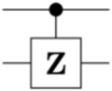
Two-qubits gate

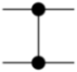
Controlled Not (CNOT, CX)



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

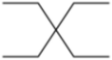
Controlled Z (CZ)

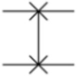




$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

SWAP



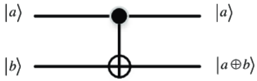


$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

CNOT gate

The prototypical multi-qubit quantum logic gate is the controlled-**NOT** or **CNOT** gate. This gate has two input qubits, known as the *control* qubit and the *target* qubit.

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$



In the case where the first qubit is the control qubit we get the matrix $\mathbf{C}_{01}$ if we swap target and control qubit we get $\mathbf{C}_{10}$.

We can observe that $\mathbf{C}_{01}$ can be rewritten in the following way:

$$\mathbf{C}_{01} = \begin{pmatrix} \mathbf{1} & 0 \\ 0 & \sigma_x \end{pmatrix} = \mathbf{CNOT}$$

.

Can we generalize this expression for a generic unitary matrix **U**?

Generic controlled gate

The conditional application of a unitary transformation **U** to a qubit namely

$$c\mathbf{U}\left|x\right\rangle\left|y\right\rangle = \mathbf{1}\otimes\mathbf{U}_x\left|x\right\rangle\left|y\right\rangle$$

Can be represented through the following 4x4 matrix:

$$c\mathbf{U} \rightarrow \begin{pmatrix} \mathbf{1} & 0 \\ 0 & \mathbf{U} \end{pmatrix}$$

under the assumption the first qubit is the control qubit.

As an example the **CZ** can be represented as follows:

$$\mathbf{CZ} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} = \begin{pmatrix} \mathbf{1} & 0 \\ 0 & \sigma_z \end{pmatrix}$$

Gates with more than 2 qubits?

What about gates with more that 2 qubits? As an example we present the **TOFFOLI** gate, which is represented by the unitary operator **T**.

The **T** gate acts on the computational basis as follows:

$$\mathbf{T} |x\rangle |y\rangle |z\rangle = |x\rangle |y\rangle |z \oplus xy\rangle$$

It is particulary useful in quantum computing since it enables us to perform an **AND** gate of two bits and the **NOT** gate, in fact

$$\textbf{AND} \rightarrow \mathbf{T} |x\rangle |y\rangle |0\rangle = |x\rangle |y\rangle |xy\rangle \equiv |x\rangle |y\rangle |x \wedge y\rangle$$

$$\textbf{NOT} \rightarrow \mathbf{T} |1\rangle |1\rangle |x\rangle = |1\rangle |1\rangle |x \oplus 1\rangle \equiv |1\rangle |1\rangle |\bar{x}\rangle$$

Since all logical and mathematical operations can be built out of AND and NOT, we have shown that using the TOFFOLI gate we can reproduce all these operations on a quantum computer, i.e. reversibly.

Measurements

For a quantum system, in order to extract information, we need to make a **measurement**, which corresponds to perfoming a certain opeartion on each qubit, the outcome of which is either 0 and 1.[2]

Futhermore, we know from postulate 2, that the state determines only the *probability* of the possible outcomes which is given the by squared magnitude of the amplitudes.

If we have the following state:

$$|\psi\rangle = \sum_{x=0}^{2^n-1} \alpha_x |x\rangle$$

the probabiility of measuring a specific state $|y\rangle$ is given by

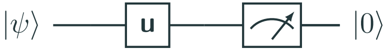
$$p(y) = |\langle y|\psi\rangle|^2 = |\alpha_y|^2 .$$

It is now clear that measuring a qubit requires a *statistical* approach, this is why we talk about expected values of the observables.

[2] The observable that we are measuring in this case is the **Z** operator.

Irreverisbility of measurements

Contrary to all the other operations, the measurement is an **irreversible** operation.

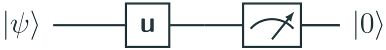


In particular, this operation is irreversible since we are losing information about the system.

What information exactly?

Irreverisbility of measurements

Contrary to all the other operations, the measurement is an **irreversible** operation.



In particular, this operation is irreversible since we are losing information about the system.

What information exactly?

After we measure a quantum system, the state collapse to the eigenstate associated with the measurement outcome. Therefore, we no longer have access to the information available in the amplitudes of the state $|\psi\rangle$.

State preparation

Supppose that we want to prepare our quantum system in a specific state, how can we produce that particular state?

State preparation

Supppose that we want to prepare our quantum system in a specific state, how can we produce that particular state?

We can do it by **measuring** the system.

State preparation

Supppose that we want to prepare our quantum system in a specific state, how can we produce that particular state?

We can do it by **measuring** the system.

This initial action of the measurement gate is called **state preparation**, since after this first step we have created a definite state.

Is there another way?

State preparation

Supppose that we want to prepare our quantum system in a specific state, how can we produce that particular state?

We can do it by **measuring** the system.

This initial action of the measurement gate is called **state preparation**, since after this first step we have created a definite state.

Is there another way?

For some particular physical realizations of qubits, yes.

For example, if each qubit is an atom, by cooling the system to an approriate low temperature we can produce the state $|0\rangle_n$

Applications

CNOT and No-cloning theorem

One of the most interesting result in QM is the no-cloning theorem which state the following:

No-cloning theorem

Assume we have a unitary operator U_{cl} and two quantum states $|\phi\rangle$ and $|\psi\rangle$ which U_{cl} copies,

$$U_{cl} |\phi\rangle \otimes |0\rangle = |\phi\rangle \otimes |\phi\rangle$$

$$U_{cl} |\psi\rangle \otimes |0\rangle = |\psi\rangle \otimes |\psi\rangle$$

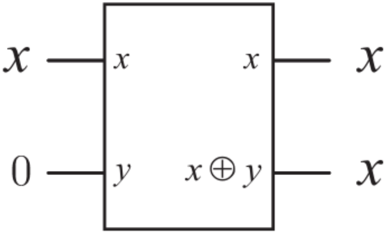
then $\langle\phi|\psi\rangle$ is 0 or 1.

This theorem is telling us that we cannot create an identical copy of an arbitrary unknown quantum state.

It is possible to verify this theorem using quantum computing, in particular the **CNOT** gate.

CNOT and No-cloning theorem

Copying a classical bit is fairly easy using a **XOR** or **CNOT** gate



Lets try to do the same thing with a quantum circuit.

CNOT and No-cloning theorem

Suppose that we try to copy a qubit in the unknown state $|\psi\rangle = a|0\rangle + b|1\rangle$ in the same manner.

$$|\psi\rangle|0\rangle = (a|0\rangle + b|1\rangle)|0\rangle = a|00\rangle + b|10\rangle$$

If we apply the **CNOT** gate we get the following:

$$\textbf{CNOT} a|00\rangle + b|10\rangle = a|00\rangle + b|11\rangle$$

Have we actually copied the state $|\psi\rangle$?

Lets compute the state $|\psi\rangle|\psi\rangle$:

$$|\psi\rangle|\psi\rangle = a^2|00\rangle + ab|01\rangle + ab|10\rangle + b^2|11\rangle$$

Unless $ab = 0$, i.e. the state are orthogonal, we are not able to copy the state ψ which is exactly the statement of the theorem.

Entanglement and Bell states

A key ingredient in quantum computing is the possibility to create entanglement. But what is entanglement?

If for a multi-qubit system we are able to write the state as a tensor product of single-qubit states

$$|\Psi\rangle_N = |\psi\rangle_1 \otimes \cdots \otimes |\psi\rangle_N$$

we say that the state $|\Psi\rangle$ is *factorized* or *separable*.

Entagled state

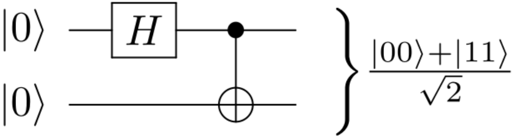
A state which is **not** separable is called *entangled*.

For example the following state is entangled:

$$|\Psi\rangle = \frac{|0\rangle\,|0\rangle + |1\rangle\,|1\rangle}{\sqrt{2}}$$

Bell states

We can recreate easily the entagled state of the previous slide using the following circuit.



This state is one of the four *Bell states* or *EPR state*, which are maximally entangled two qubits state. The other ones can be generated by acting with the same circuit on different initial state:

- $|00\rangle \rightarrow |\beta_{00}\rangle = \frac{|0\rangle|0\rangle+|1\rangle|1\rangle}{\sqrt{2}}$
- $|01\rangle \rightarrow |\beta_{01}\rangle = \frac{|0\rangle|1\rangle+|1\rangle|0\rangle}{\sqrt{2}}$
- $|10\rangle \rightarrow |\beta_{10}\rangle = \frac{|0\rangle|0\rangle-|1\rangle|1\rangle}{\sqrt{2}}$
- $|11\rangle \rightarrow |\beta_{11}\rangle = \frac{|0\rangle|1\rangle-|1\rangle|0\rangle}{\sqrt{2}}$

Introduction to quantum algorithms

The general computational process

- A suitable programmed quantum computer should actor on a number x to produce another number $f(x)$ for some specified function f.
- Since we expect f to be not reversible we shall need at least $n+m$ qubits, assuming x is a n bit integers and $f(x)$ an m-bit integer. Usually we call *input register* the set of qubits that represent x and *output register* the set of qubits that represent $f(x)$.
- To perform the calculation of $f(x)$ we apply a suitable tranformation $\mathbf{U}_f$ to our set of $n+m$ qubits:

$$\mathbf{U}_f(|x\rangle_n |y\rangle_m) = |x\rangle_n |y \oplus f(x)\rangle_m$$

  where $\oplus$ indicates the modulo-2 bitwise addiction or the XOR .
  If the initial value of the output register is $y = 0$ we can obtain the actual value of $f(x)$

$$\mathbf{U}_f(|x\rangle_n |0\rangle_m) = |x\rangle_n |f(x)\rangle_m$$

The general computational process

We can also observe that $\mathbf{U}_f$ is clearly invertible, as expected

$$\mathbf{U}_f\mathbf{U}_f(|x\rangle\,|y\rangle) = \mathbf{U}_f\,|x\rangle\,|y \oplus f(x)\rangle = |x\rangle\,|y\rangle$$

since $z \oplus z = 0$ for any z.

Useful identities

Before looking at some quantum algorithms it is useful to learn a few identities.

For a single qubit we write the action of the Hadamard gate in the following form:

$$\mathbf{H}\left|x\right\rangle = \frac{\left|0\right\rangle + (-)^{x}\left|1\right\rangle}{\sqrt{2}} = \frac{1}{\sqrt{2}}\sum_{z=0,1}(-)^{xz}\left|z\right\rangle$$

For a n qubits system we get

$$\mathbf{H}^{\otimes n}\left|x\right\rangle_{n} = \frac{\left|0\right\rangle + (-)^{x_{n-1}}\left|1\right\rangle}{\sqrt{2}} \otimes \cdots \otimes \frac{\left|0\right\rangle + (-)^{x_{0}}\left|1\right\rangle}{\sqrt{2}} = \frac{1}{2^{n/2}}\sum_{z=0}^{2^{n}-1}(-)^{xz}\left|z\right\rangle$$

Finally if $f(x) \in \{0,1\}$ then we have

$$\mathbf{U}_{f}(\mathbf{I}\otimes\mathbf{H})\left|x\right\rangle\left|1\right\rangle = \left|x\right\rangle\frac{\left|f(x)\right\rangle - \left|1\oplus f(x)\right\rangle}{\sqrt{2}} = (-)^{f(x)}\left|x\right\rangle\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}$$

Deutsch algorithm

As a first example lets have a look at the Deutsch problem

Deutsch problem

Given a function $f : \{0, 1\} \to \{0, 1\}$ we want to know whether if $f(0) = f(1)$.

From a classical point of view of how many times do we need to evaluate $f(x)$?

Deutsch algorithm

As a first example lets have a look at the Deutsch problem

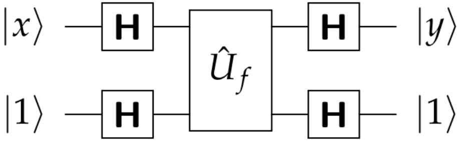
Deutsch problem

Given a function $f : \{0,1\} \rightarrow \{0,1\}$ we want to know whether if $f(0) = f(1)$.

From a classical point of view of how many times do we need to evaluate $f(x)$?

Correct answer: 2

We are going to show that a quantum algorithm can tell us the answer using just one evaluation of the function f using the following circuit:



Deutsch algorithm

Lets apply the quantum circuit of the previous slide step by step:

1. Apply **H** to both qubits in the initial state

$$\mathbf{H} \otimes \mathbf{H} \left|x\right\rangle \left|1\right\rangle = \sum_z \frac{(-)^{xz}}{\sqrt{2}} \left|z\right\rangle \frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}$$

   where we used one the identities from the previous slides.

2. Apply $\mathbf{U}_f$

$$\mathbf{U}_f \sum_z \frac{(-)^{xz}}{\sqrt{2}} \left|z\right\rangle \frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}} = \sum_z \frac{(-)^{xz+f(z)}}{\sqrt{2}} \left|z\right\rangle \frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}$$

3. Apply again the Hadamard transformations

$$(\mathbf{H} \otimes \mathbf{H})\mathbf{U}_f(\mathbf{H} \otimes \mathbf{H}) \left|x\right\rangle \left|1\right\rangle = \sum_{s=0,1} c_f(x,s) \left|s\right\rangle \left|1\right\rangle$$

   where we have introduce the coefficients:

$$c_f(x,s) = \frac{1}{2}(-)^{f(0)}\left[1 + (-)^{x+s}(-)^{f(1)-f(0)}\right]$$

Deutsch algorithm

It is straightforward to verify that

- if $f(1) = f(0) \Rightarrow |c_f(x,x)|^2 = 1$ and $|c_f(x,\bar{x})|^2 = 0$
- if $f(1) \neq f(0) \Rightarrow |c_f(x,x)|^2 = 0$ and $|c_f(x,\bar{x})|^2 = 1$

Therefore, if a measurement on the input register gives result $|x\rangle$ we can conclude that $f(0) = f(1)$ if it leads to $|\bar{x}\rangle$ we have $f(1) \neq f(0)$.

All of this only with a single query of $\mathbf{U}_f$!

Quantum algorithms

The Deutsch algorithm is just one of many quantum algorithms that are currently known.

Gate Circuits

- Search (Grover)
- QFT (Shor)
- Deutsch
- . . .

Variational

- Eigeinsolvers
- Autoencoders
- Classifiers
- . . .

Annealing

- Direct annealing
- Adiabatic evolution
- QAOA
- . . .

Towards Quantum Machine Learning

Variational Quantum Circuits

Getting inspiration from **AI**:

- Supervised learning ⇒ Regression and classification
- Unsupervised learning ⇒ Generative models, autoencoders
- Reinforcement learning ⇒ Quantum RL / Q-learning

Define new parametric model architectures for quantum hardware:

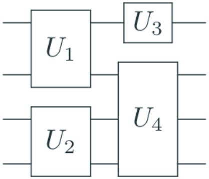
⇒ Variational Quantum Circuits / Quantum Machine Learning

Rational for Variational Quantum Circuits

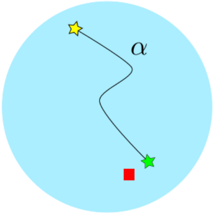
Rational:

Deliver a variational quantum state that can explore a large Hilbert space

$$U(\vec{\alpha}) = U_n \ldots U_2 U_1$$



Near optimal solution



Idea: Quantum Computer is a machine that generates variational states.

⇒ **Variational Quantum Computer**

Solovay-Kitaev Theorem

Let $\{U_i\}$ be a dense set of unitaries.
Define a circuit approximation to V:

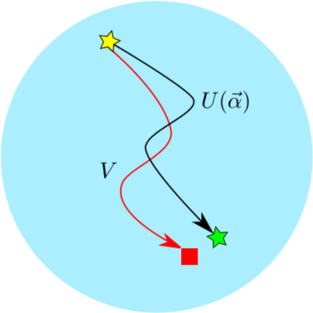
$$|U_k \ldots - V| < \delta$$

Scaling to best approximation

$$k \sim \mathcal{O}\Big(\log^c \frac{1}{\delta}\Big)$$

where $c < 4$.

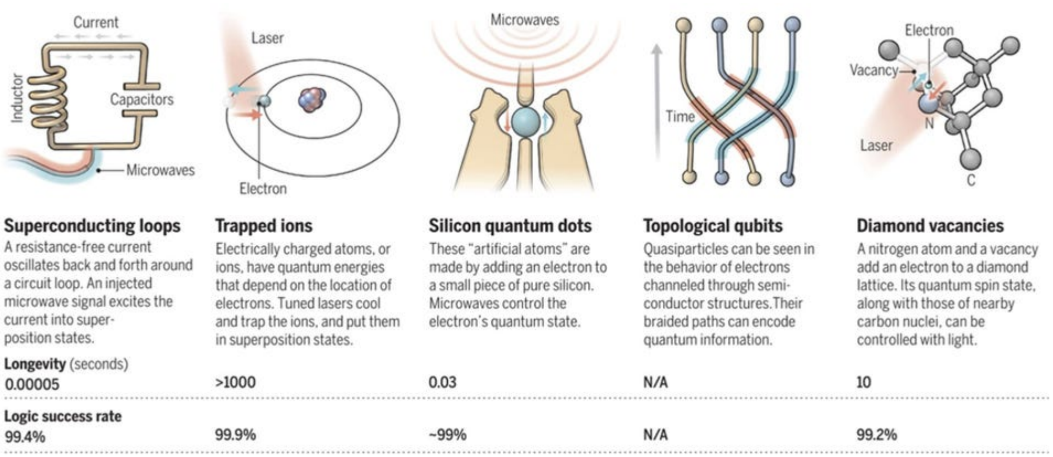
Optimal solution



⇒ The approximation is **efficient** and requires a **finite** number of gates

Quantum Technology

Quantum technologies



Superconducting loops
A resistance-free current oscillates back and forth around a circuit loop. An injected microwave signal excites the current into superposition states.

Trapped ions
Electrically charged atoms, or ions, have quantum energies that depend on the location of electrons. Tuned lasers cool and trap the ions, and put them in superposition states.

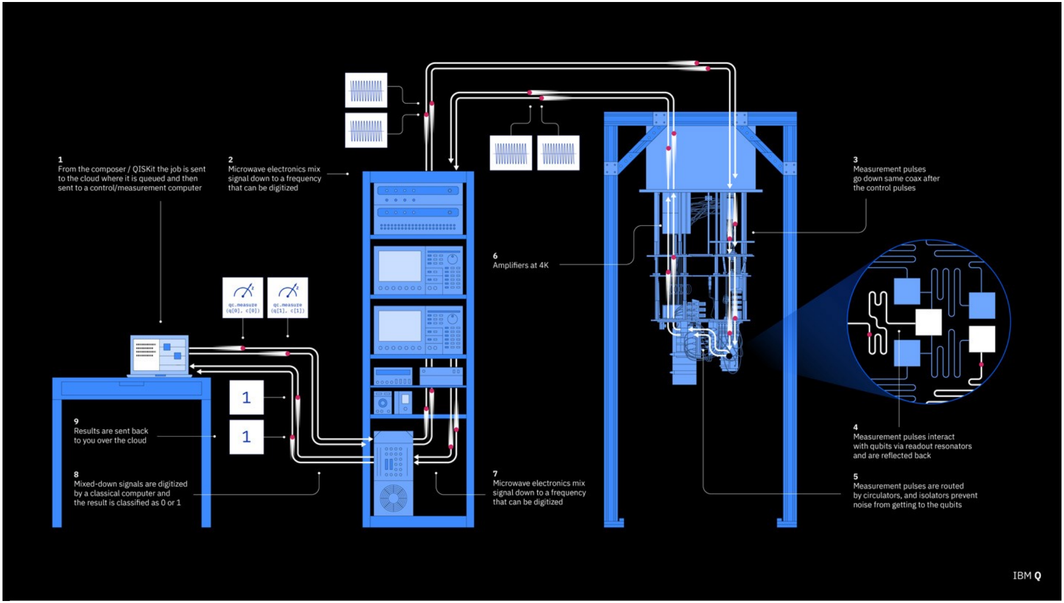
Silicon quantum dots
These "artificial atoms" are made by adding an electron to a small piece of pure silicon. Microwaves control the electron's quantum state.

Topological qubits
Quasiparticles can be seen in the behavior of electrons channeled through semiconductor structures. Their braided paths can encode quantum information.

Diamond vacancies
A nitrogen atom and a vacancy add an electron to a diamond lattice. Its quantum spin state, along with those of nearby carbon nuclei, can be controlled with light.

	Superconducting loops	Trapped ions	Silicon quantum dots	Topological qubits	Diamond vacancies
Longevity (seconds)	0.00005	>1000	0.03	N/A	10
Logic success rate	99.4%	99.9%	~99%	N/A	99.2%

Physical implementation



1
From the composer / QISKit the job is sent to the cloud where it is queued and then sent to a control/measurement computer

2
Microwave electronics mix signal down to a frequency that can be digitized

3
Measurement pulses go down same coax after the control pulses

4
Measurement pulses interact with qubits via readout resonators and are reflected back

5
Measurement pulses are routed by circulators, and isolators prevent noise from getting to the qubits

6
Amplifiers at 4K

7
Microwave electronics mix signal down to a frequency that can be digitized

8
Mixed-down signals are digitized by a classical computer and the result is classified as 0 or 1

9
Results are sent back to you over the cloud

Physical implementation

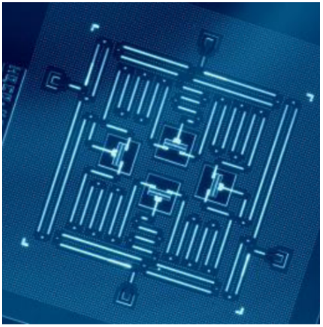


Figure 1: Superconducting device assembled by IBM

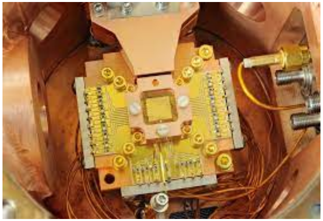
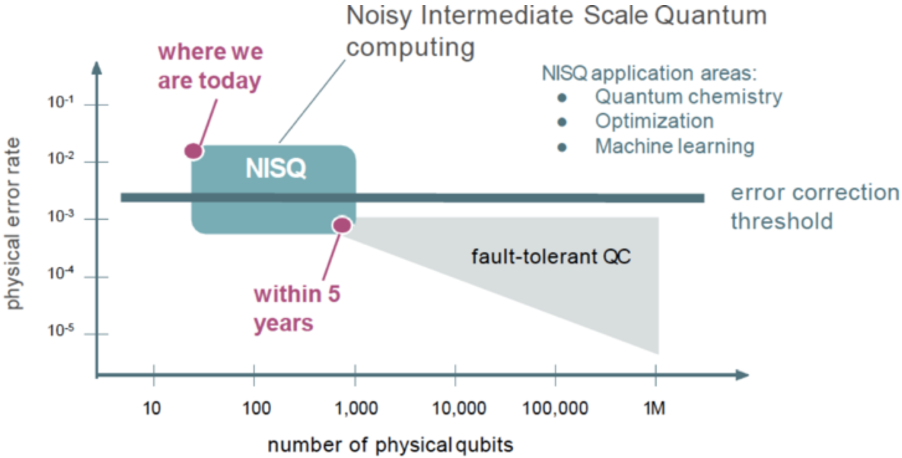


Figure 2: Chip based on trapped ions technology

NISQ era



Noisy Intermediate Scale Quantum computing

where we are today

within 5 years

error correction threshold

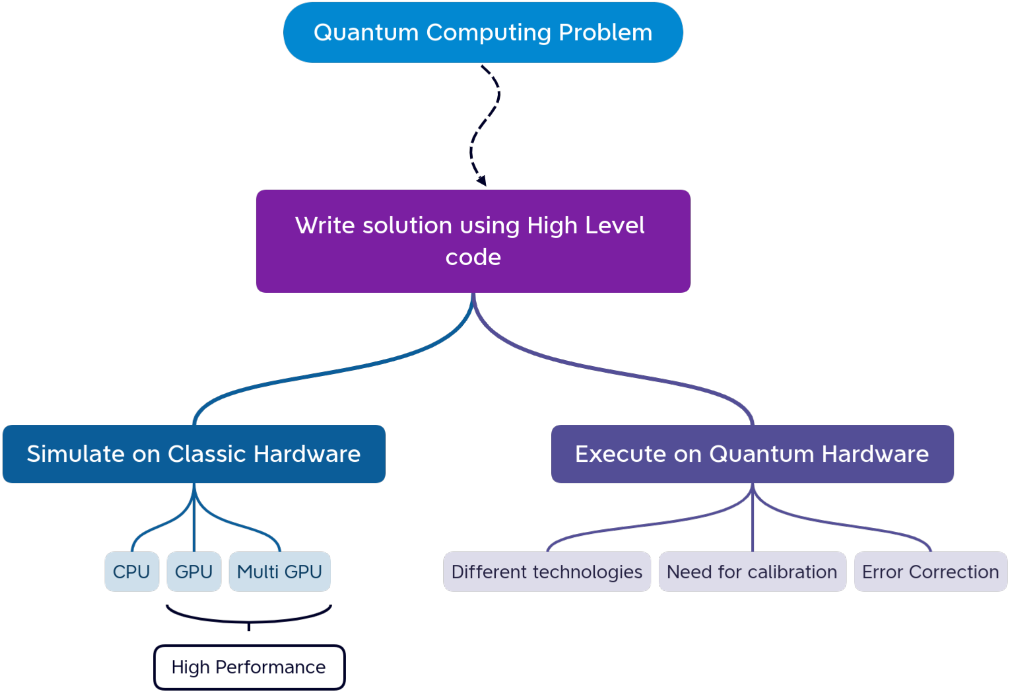
NISQ application areas:

- Quantum chemistry
- Optimization
- Machine learning

“Quantum computing in the NISQ era and beyond” Preskill, 2018 https://arxiv.org/abs/1801.00862

Languages for quantum computing

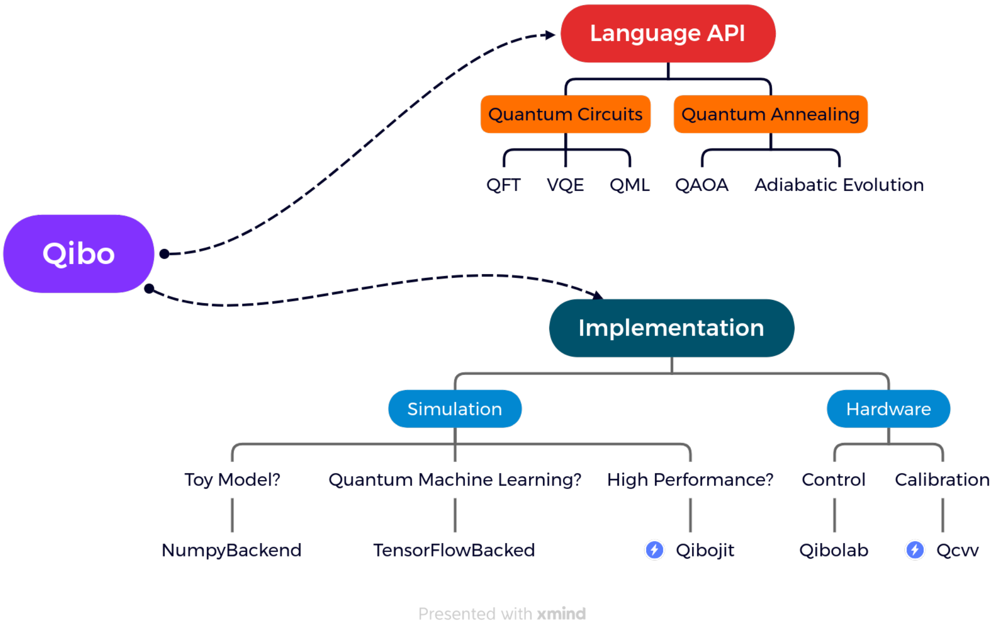
Challenge


Quantum Computing Problem
Write solution using High Level code
Simulate on Classic Hardware
Execute on Quantum Hardware
CPU
GPU
Multi GPU
High Performance
Different technologies
Need for calibration
Error Correction

Presented with xmind

Introducing Qibo

Qibo is an **open-source** full stack API for quantum simulation and quantum hardware control and calibration.



Thanks for listening!