A Practical Introduction to Quantum Computing: From Qubits to Quantum Machine Learning and Beyond

Vishnu Sivan
mail.vishnu.sivan@gmail.com

Software Developer

December 2021

Part I

Introduction: quantum computing... the end of the world as we know it?

I, for one, welcome our new quantum overlords

NEWS QUANTUM PHYSICS

Google officially lays claim to quantum supremacy

A quantum computer reportedly beat the most powerful supercomputers at one type of calculation

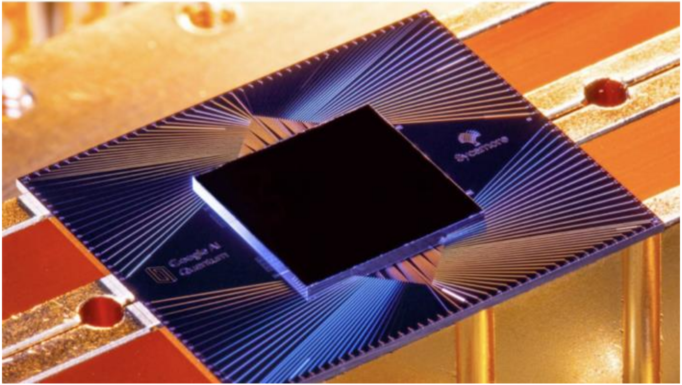


Image credits: sciencenews.org

Philosophy of the course

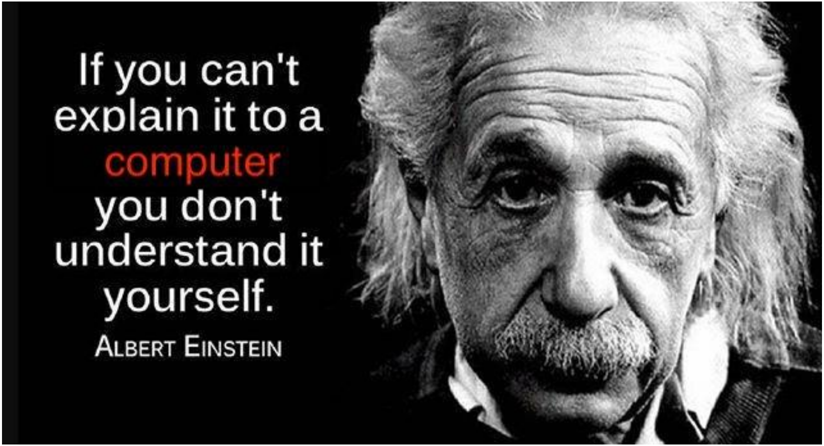


Image credits:Modified from an Instagram image by Bob MacGuffie

Tools and resources

- Jupyter Notebooks
  - Web application to create and execute notebooks that include code, images, text and formulas
  - They can be used locally (Anaconda) or in the cloud (mybinder.org, Google Colab...)
- IBM Quantum Experience
  - Free online access to quantum simulators (up to 32 qubits) and **actual quantum computers** (1, 5 and 15 qubits) with different topologies
  - Programmable with a visual interface and via different languages (python, qasm, Jupyter Notebooks)
  - Launched in May 2016
  - https://quantum-computing.ibm.com/

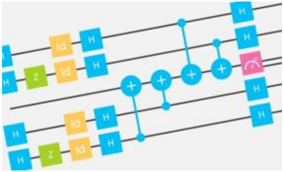


Image credits:IBM

Tools and resources (2)

- Quirk
  - Online simulator (up to 16 qubits)
  - Lots of different gates and visualization options
  - `http://algassert.com/quirk`
- D-Wave Leap
  - Access to D-Wave quantum computers
  - Ocean: python library for quantum annealing
  - Problem specific (QUBO, Ising model...)
  - `https://www.dwavesys.com/take-leap`

The shape of things to come

Image credits: Created with wordclouds.com

What is quantum computing?

Quantum computing

Quantum computing is a computing paradigm that exploits quantum mechanical properties (superposition, entanglement, interference...) of matter in order to do calculations

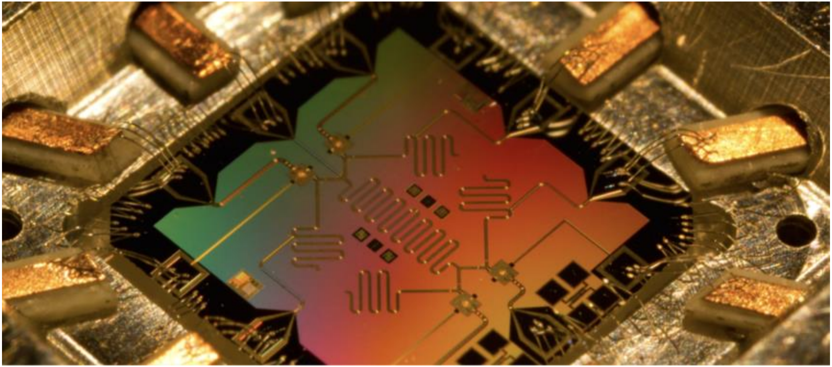


Image credits:Erik Lucero

Models of quantum computing

- There are several models of quantum computing (they're all equivalent)
  - Quantum Turing machines
  - **Quantum circuits**
  - Measurement based quantum computing (MBQC)
  - Adiabatic quantum computing
  - Topological quantum computing
- Regarding their **computational capabilities**, they are equivalent to classical models (Turing machines)

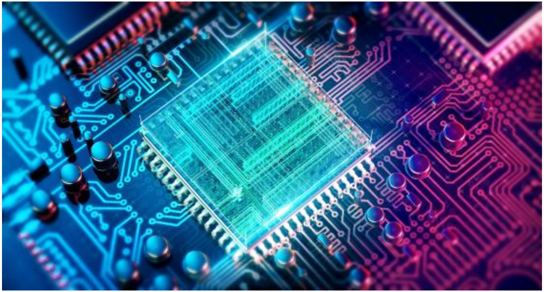


Image credits:Getty Images

Quantum and classical computational complexity

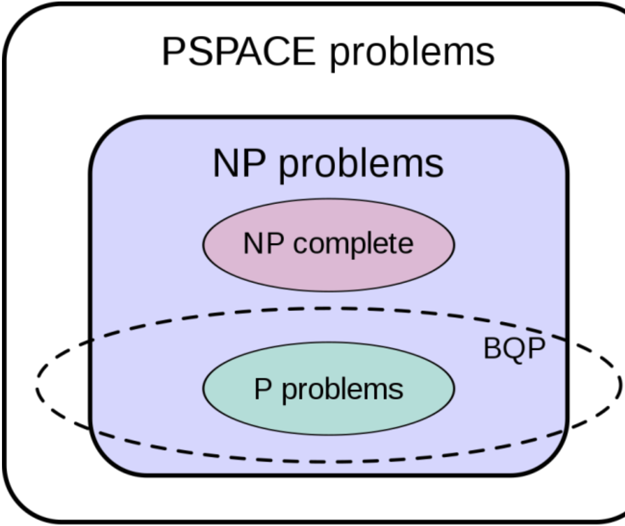


Image credits:wikipedia.org

What technologies are used to build quantum computers?

	Superconducting loops	Trapped ions	Silicon quantum dots	Topological qubits	Diamond vacancies
Company support	Google, IBM, Quantum Circuits	ionQ	Intel	Microsoft, Bell Labs	Quantum Diamond Technologies
Pros	Fast working. Build on existing semiconductor industry.	Very stable. Highest achieved gate fidelities.	Stable. Build on existing semiconductor industry.	Greatly reduce errors.	Can operate at room temperature.
Cons	Collapse easily and must be kept cold.	Slow operation. Many lasers are needed.	Only a few entangled. Must be kept cold.	Existence not yet confirmed.	Difficult to entangle.

Image credits: Graphic by C. Bickle/Science data by Gabriel Popkin

What is a quantum computer like?

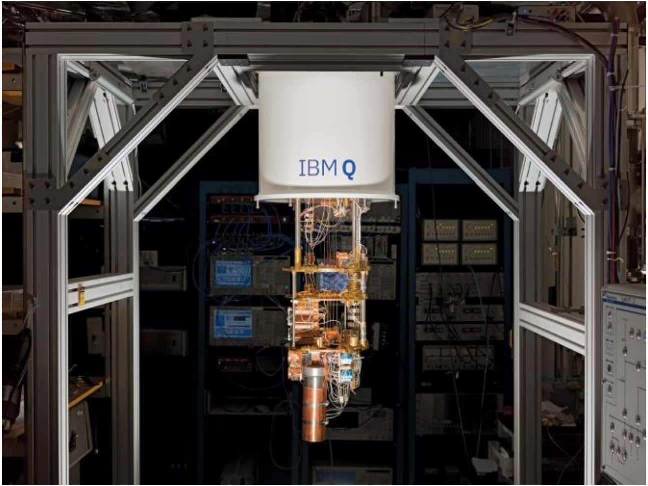


Image credits:IBM

The Sounds of IBM: IBM Q

Programming a quantum computer

- Different frameworks and programming languages:
  - qasm
  - Qiskit (IBM)
  - Cirq (Google)
  - Forest/pyqil (Rigetti)
  - Q# (Microsoft)
  - Ocean (D-Wave)
  - ...
- Most of them for quantum circuit specification

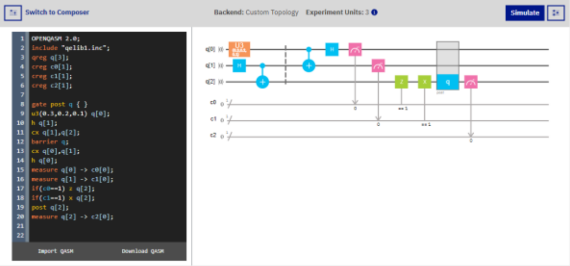


Image credits:IBM

What are the elements of a quantum circuit?

- Every computation has three elements: data, operations and results
- In quantum circuits:
  - Data = **qubits**
  - Operations = **quantum gates** (unitary transformations)
  - Results = **measurements**

Image credits:Adobe Stock

Part II

One-qubit systems: one qubit to rule them all

What is a qubit?

- A classical bit can take two different values (0 or 1). It is discrete.
- A qubit can “take” **infinitely** many different values. It is continuous.
- Qubits live in a **Hilbert vector space** with a basis of two elements that we denote $|0\rangle$ y $|1\rangle$.
- A generic qubit is in a **superposition**

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

where α and β are **complex numbers** such that

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

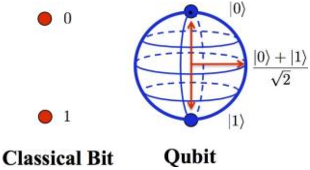


Image credits:https://prateekvjoshi.com/

Measuring a qubit

- The way to know the value of a qubit is to perform a measurement. However
  - The result of the measurement is random
  - When we measure, we only obtain one (classical) bit of information
- If we measure the state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ we get 0 with probability $|\alpha|^2$ and 1 with probability $|\beta|^2$.
- Moreover, the new state after the measurement will be $|0\rangle$ or $|1\rangle$ depending of the result we have obtained (wavefunction colapse)
- We cannot perform several independent measurements of $|\psi\rangle$ because we cannot copy the state (**no-cloning theorem**)

The Bloch sphere

- A common way of representing the state of a qubit is by means of a point in the surface of the Bloch sphere
- If $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$ we can find angles γ, δ, θ such that

$$\alpha = e^{i\gamma}\cos\frac{\theta}{2}$$

$$\beta = e^{i\delta}\sin\frac{\theta}{2}$$

- Since an overall phase is physically irrelevant, we can rewrite

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

with $0 \le \theta \le \pi$ and $0 \le \phi < 2\pi$.

The Bloch sphere (2)

- From $|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle$ we can obtain spherical coordinates for a point in R^3

$$(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$$

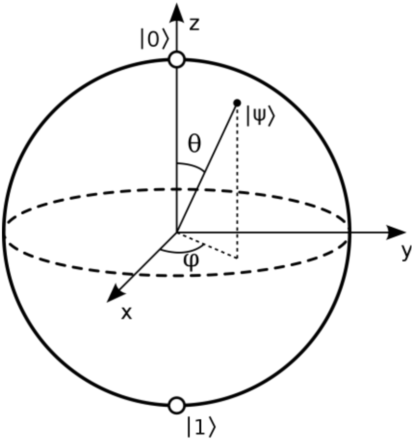


Image credits:wikipedia.org

What are quantum gates?

- Quantum mechanics tells us that the evolution of an isolated state is given by the Schrödinger equation

$$H(t)|\psi(t)\rangle = i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle$$

- In the case of quantum circuits, this implies that the operations that can be carried out are given by unitary matrices. That is, matrices U of complex numbers verifying

$$UU^\dagger = U^\dagger U = I$$

  where $U^\dagger$ is the conjugate transpose of U.

- Each such matrix is a possible quantum gate in a quantum circuit

Reversible computation

- As a consequence, all the operations have an inverse: **reversible computing**
- Every gate has the same number of inputs and outputs
- We cannot directly implement some classical gates such as *or*, *and*, *nand*, *xor*...
- But we can simulate any classical computation with small overhead
- Theoretically, we could compute without wasting energy (Landauer's principle, 1961)

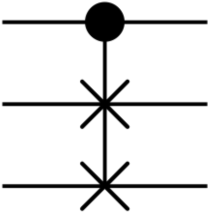


Image credits: wikipedia.org

One-qubit gates

- When we have just one qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, we usually represent it as a column vector $\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$
- Then, a one-qubit gate can be identified with a matrix $U = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ that

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} \bar{a} & \bar{c} \\ \bar{b} & \bar{d} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

where $\bar{a}, \bar{b}, \bar{c}, \bar{d}$ are the conjugates of complex numbers a, b, c, d.

Action of a one-qubit gate

- A state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ is transformed into

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} a\alpha + b\beta \\ c\alpha + d\beta \end{pmatrix}$$

  that is, into the state $|\psi\rangle = (a\alpha + b\beta)\,|0\rangle + (c\alpha + d\beta)\,|1\rangle$
- Since U is unitary, it holds that

$$|(a\alpha + b\beta)|^2 + |(c\alpha + d\beta)|^2 = 1$$

The X or NOT gate

- The X gate is defined by the (unitary) matrix

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

- Its action (in quantum circuit notation) is

$$|0\rangle \; -\boxed{X}- \; |1\rangle$$

$$|1\rangle \; -\boxed{X}- \; |0\rangle$$

  that is, it acts like the classical NOT gate
- On a general qubit its action is

$$\alpha\,|0\rangle + \beta\,|1\rangle \; -\boxed{X}- \; \beta\,|0\rangle + \alpha\,|1\rangle$$

The Z gate

- The Z gate is defined by the (unitary) matrix

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

- Its action is

$$|0\rangle \; -\boxed{Z}- \; |0\rangle$$

$$|1\rangle \; -\boxed{Z}- \; -|1\rangle$$

The H or Hadamard gate

- The H or Hadamard gate is defined by the (unitary) matrix

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

- Its action is

$$|0\rangle \;—\boxed{H}—\; \frac{|0\rangle+|1\rangle}{\sqrt{2}}$$

$$|1\rangle \;—\boxed{H}—\; \frac{|0\rangle-|1\rangle}{\sqrt{2}}$$

- We usually denote

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

and

$$|-\rangle := \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Other important gates

- Y gate
$$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$
- S gate
$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{bmatrix}$$
- T gate
$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix}$$
- The gates X, Y and Z are also called, together with the identity, the Pauli gates. An alternative notation is σ_X, σ_Y, σ_Z.

Rotation gates

- We can define the following rotation gates

$$R_X(\theta) = e^{-i\frac{\theta}{2}X} = \cos\frac{\theta}{2} I - i\sin\frac{\theta}{2} X = \begin{bmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}$$

$$R_Y(\theta) = e^{-i\frac{\theta}{2}Y} = \cos\frac{\theta}{2} I - i\sin\frac{\theta}{2} Y = \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}$$

$$R_Z(\theta) = e^{-i\frac{\theta}{2}Z} = \cos\frac{\theta}{2} I - i\sin\frac{\theta}{2} Z = \begin{bmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{bmatrix} \equiv \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}$$

- Notice that $R_X(\pi) \equiv X$, $R_Y(\pi) \equiv Y$, $R_Z(\pi) \equiv Z$, $R_Z(\frac{\pi}{2}) \equiv S$, $R_Z(\frac{\pi}{4}) \equiv T$

Using rotation gates to generate one-qubit gates

- For any one-qubit gate U there exist a unit vector $r = (r_x, r_y, r_z)$ and an angle θ such that

$$U \equiv e^{-i\frac{\theta}{2} r \cdot \sigma} = \cos\frac{\theta}{2} I - i \sin\frac{\theta}{2}(r_x X + r_y Y + r_z Z)$$

- For instance, choosing $\theta = \pi$ and $r = (\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}})$ we can see that

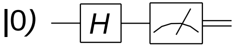
$$H \equiv e^{-i\frac{\theta}{2} r \cdot \sigma} = -i \frac{1}{\sqrt{2}}(X + Z)$$

- Additionally, it can also be proved that there exist angles α, β and γ such that

$$U \equiv R_Z(\alpha) R_Y(\beta) R_Z(\gamma)$$

Hello, quantum world!

- Our very first quantum circuit!



- After applying the *H* gate the qubit state is

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

- When we measure, we obtain 0 or 1, each with 50% probability: we have a circuit that generates perfectly uniform random bits!