Calcul Québec

Programme d’été en informatique quantique

Juin 2025

Juliette Geoffrion, Kim Renaud

Today

- Introduction to quantum information
    - Notebook 1 : Single qubit quantum simulator

- How to program a quantum computer?
    - Notebook 2 : Intro to PennyLane

Connecting to the Magic Castle (virtual machine)

1. Create your username and password on Mokey

2. Use them to connect to the virtual machine

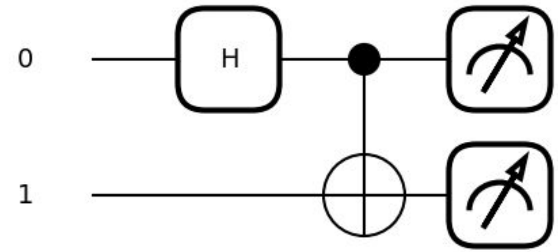
You can save your work on the virtual machine as you work (VM). However, **we strongly recommend that you also make regular backups** locally or on GitHub.

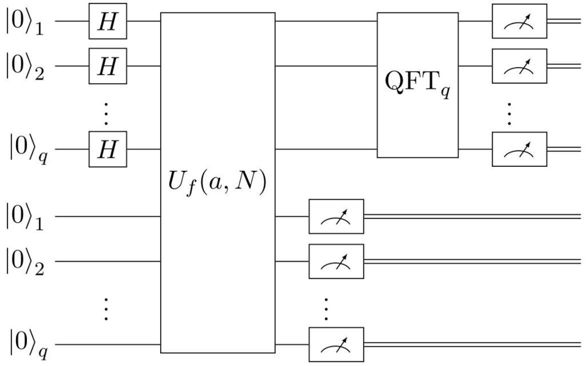
Key Dates
*Dates tentative

June 10	10:00 - 12:00	**Workshop** : Intro to PennyLane
June 16 and 17	TBD	**Individual Project Consultation** : 30-minute 1-on-1 meetings to discuss project scope
July 17	10:00 - 11:00	**Group Roundtable** : Brief project update
August 21	17:00 - 19:00	**Final project presentations** and **MonarQ visit**

Suggested workflow

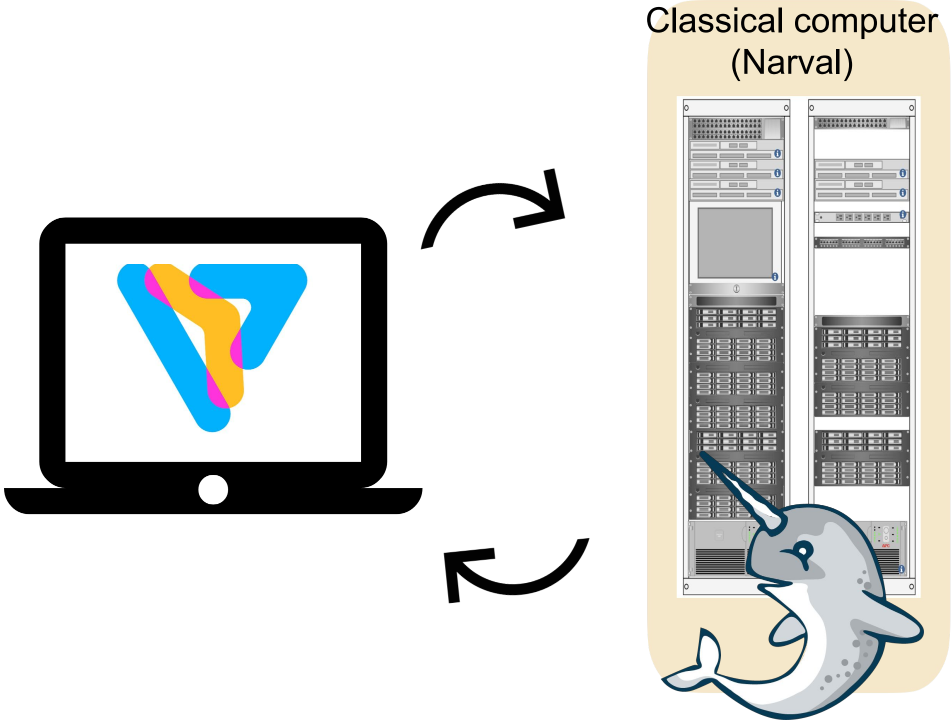
Design quantum circuits in PennyLane



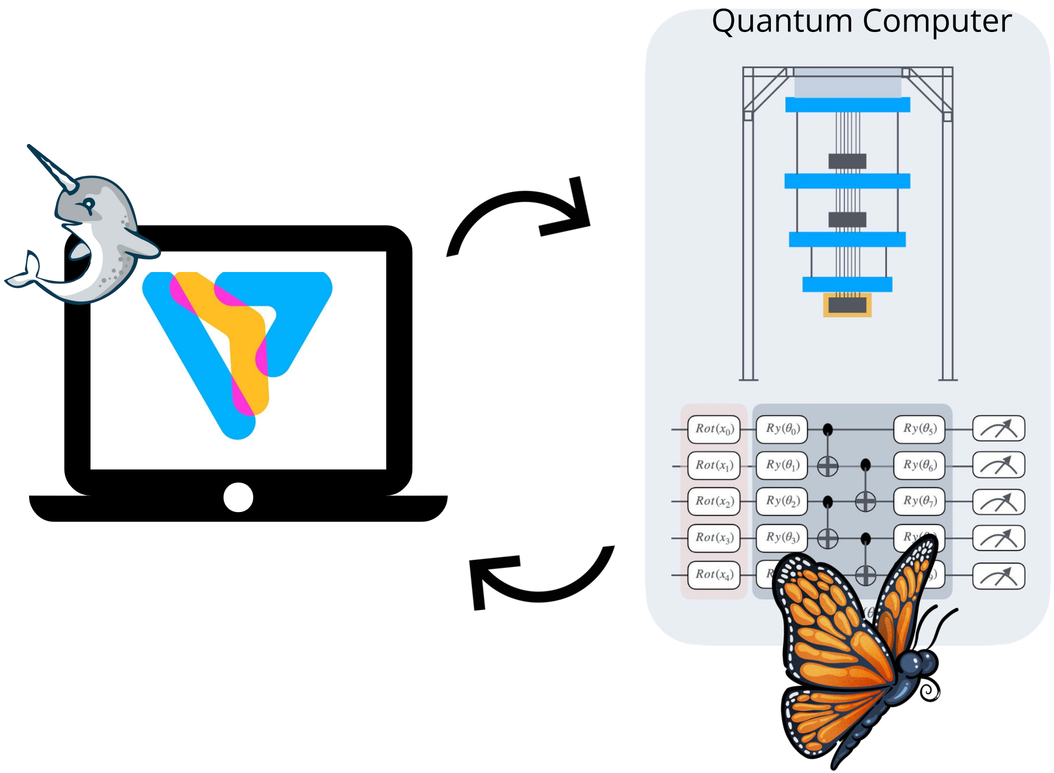


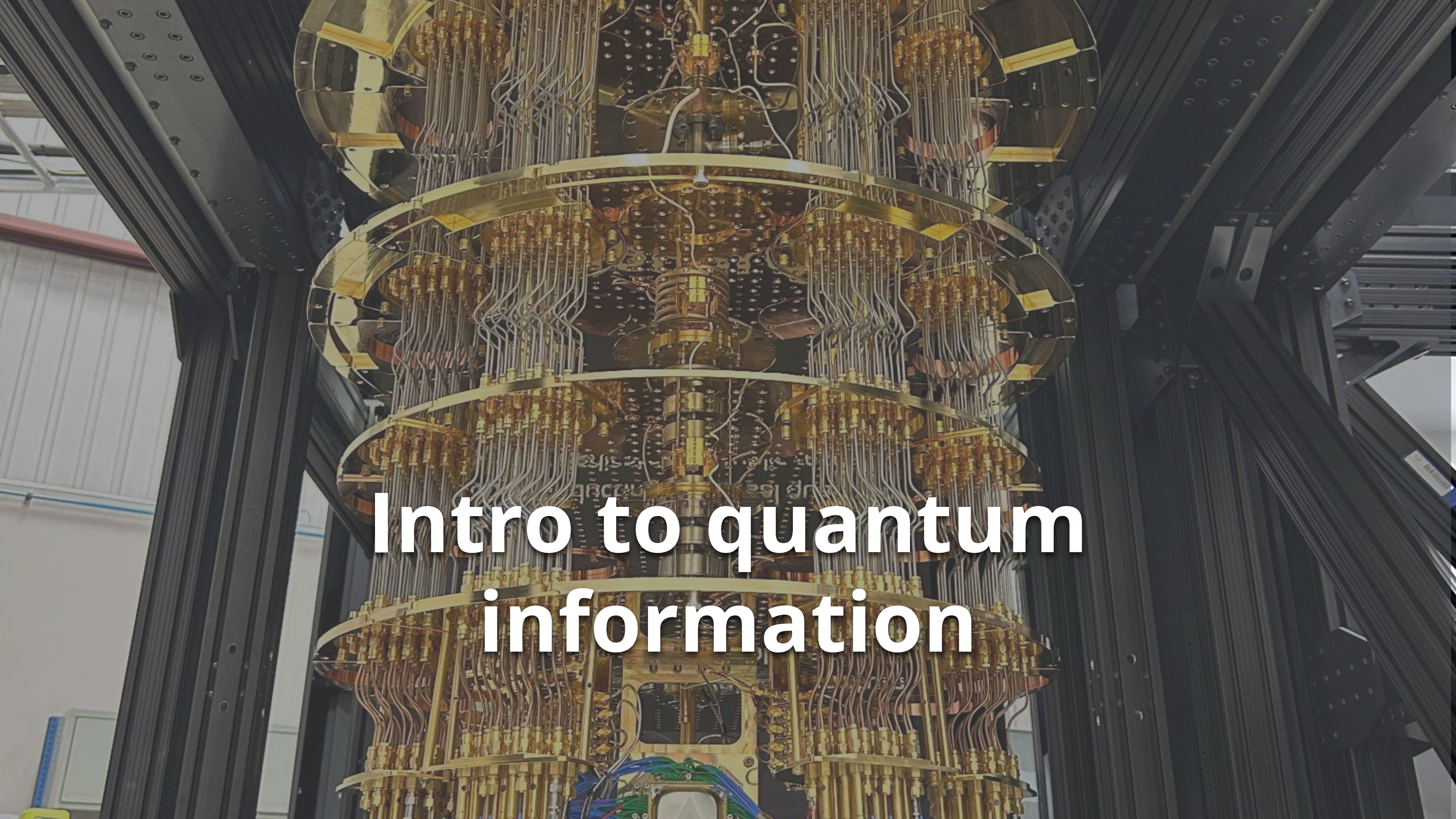
Suggested workflow

Simulate your quantum circuits on a classical supercomputer



Run your quantum circuits on a quantum computer



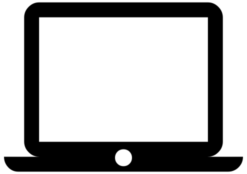


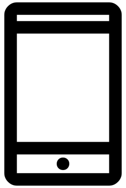
Intro to quantum information

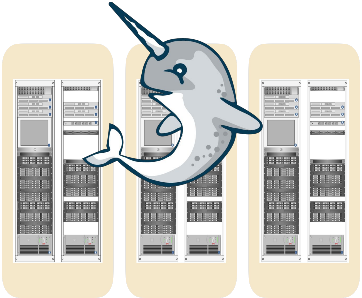
Overview

Classical information

- Uses binary states or **bits** (0 and 1)
- Output is deterministic
- Laptops, cell phones, computing clusters etc.







Quantum information

- Uses quantum bits or **qubits**
- Depend on the properties of quantum mechanics : superposition, interference and entanglement
- Output is probabilistic
- MonarQ

Qubit : the mathematical object

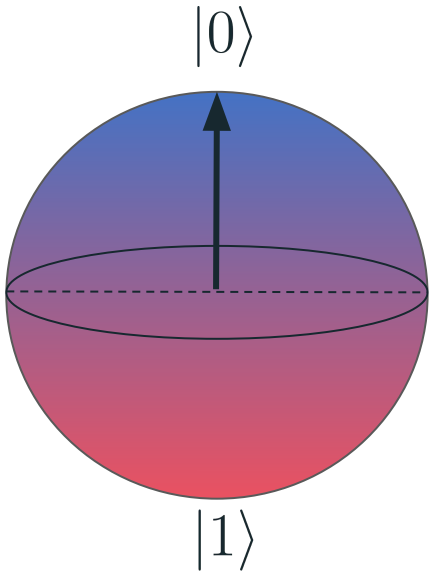
Qubits are the physical realization of a two-level quantum system, which stores and processes quantum information.

Computational basis

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Mathematically, a qubit is a vector in a complex vector space. It can be visualized on the Bloch sphere.

Two **classical** bits



One **quantum** bit (qubit)

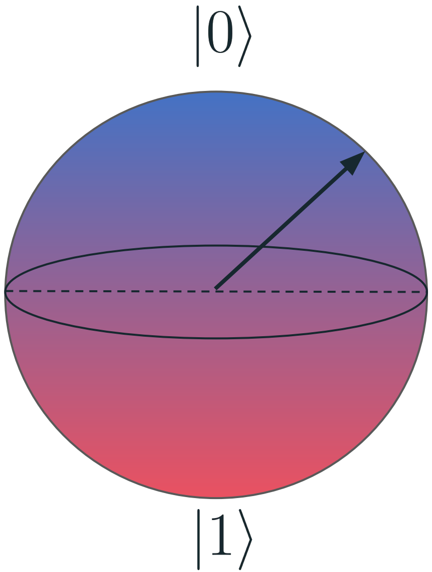
Qubit : the mathematical object

Superposition

$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$

A superposition is a linear combination of basis states like $|0\rangle$ and $|1\rangle$

Two **classical** bits



One **quantum** bit (qubit)

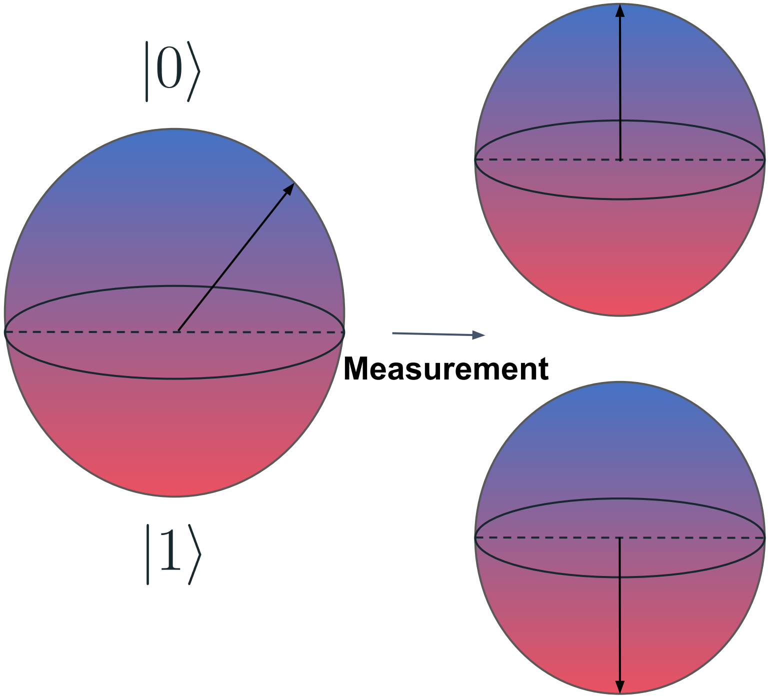
Qubit : the mathematical object

Probabilistic output at measurement

$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$

$$P(|0\rangle) = |\alpha|^2 = \alpha^*\alpha$$

$$P(|1\rangle) = |\beta|^2 = \beta^*\beta$$



Qubit : the mathematical object

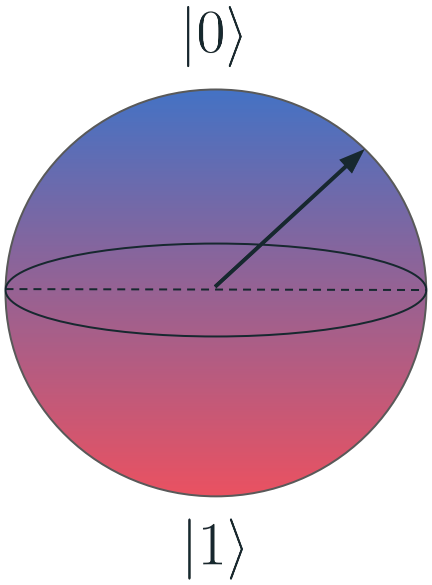
Interference

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \text{ with } |\alpha|^2 + |\beta|^2 = 1$$

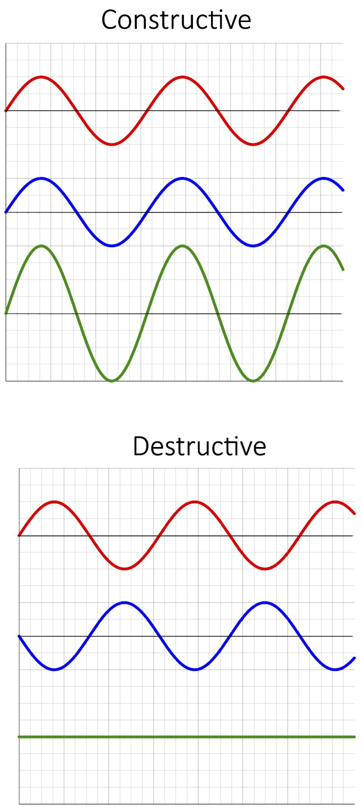
$$P(|0\rangle) = |\alpha|^2$$

$$P(|1\rangle) = |\beta|^2$$

Amplitudes and probabilities can be manipulated with constructive and destructive interference.



One **quantum** bit
(qubit)

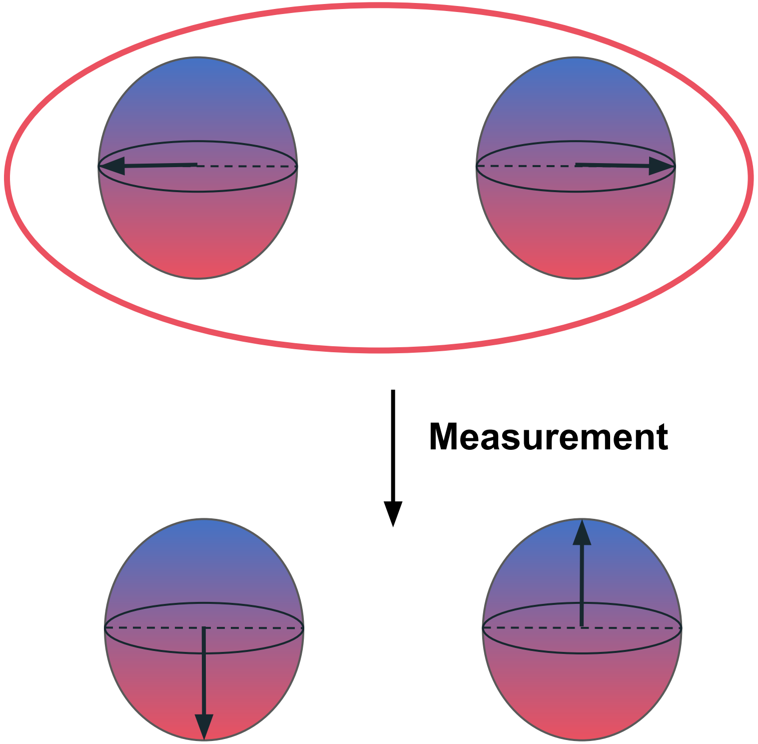


Qubit : the mathematical object

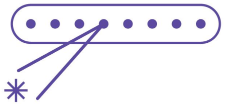
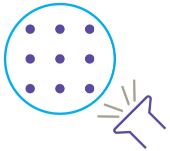
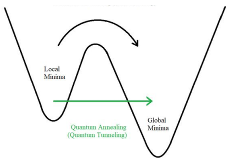
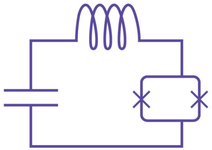
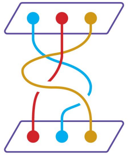
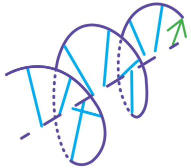
Entanglement

Entangled qubits are correlated. Measurement of one modifies the measurement of the others.

There is no analogue in classical mechanics.



Qubit : the physical object

Atoms		Electrons			Photons
Trapped ions	Cold atoms	Annealing	Superconducting	Topological	Photons
					
IONQ QUANTINUUM	QuEra Computing Inc. Pasqal	D-Wave	IBM amazon Google ANYON IQM Nord Quantique	Microsoft	XANADU PsiQuantum QUANDELA

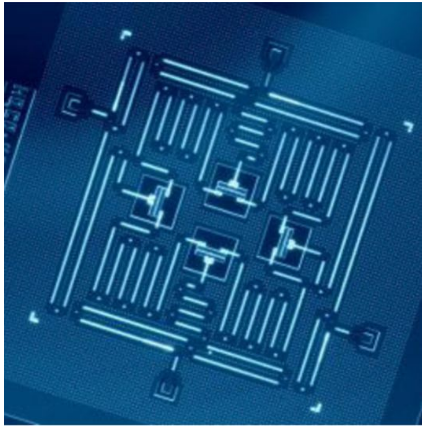
Qubits

- Fundamental element of computation exhibiting the properties of quantum mechanics.

- The Boolean states *0* and *1* are represented by a pair of normalised and orthogonal quantum states $|0\rangle$ and $|1\rangle$

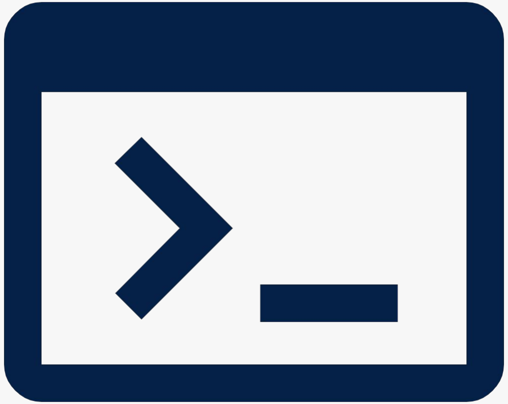
- In practice, a qubit is typically and microscopic system such as an atom, an electron, a spin or a polarised photon.

MonarQ's qubits are **transmons** made from superconducting material operating at extremely low temperatures and manipulated by microwave pulses.



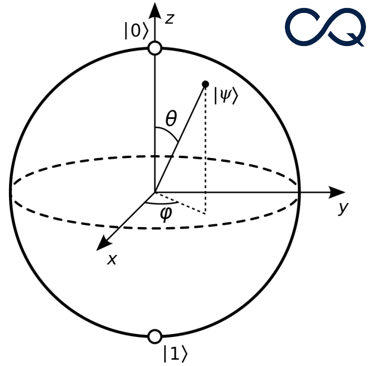
https://www.nature.com/articles/s41534-016-0004-0

Coding break



Notebook 1 :
Building a quantum simulator

Notebook 1 cheat sheet



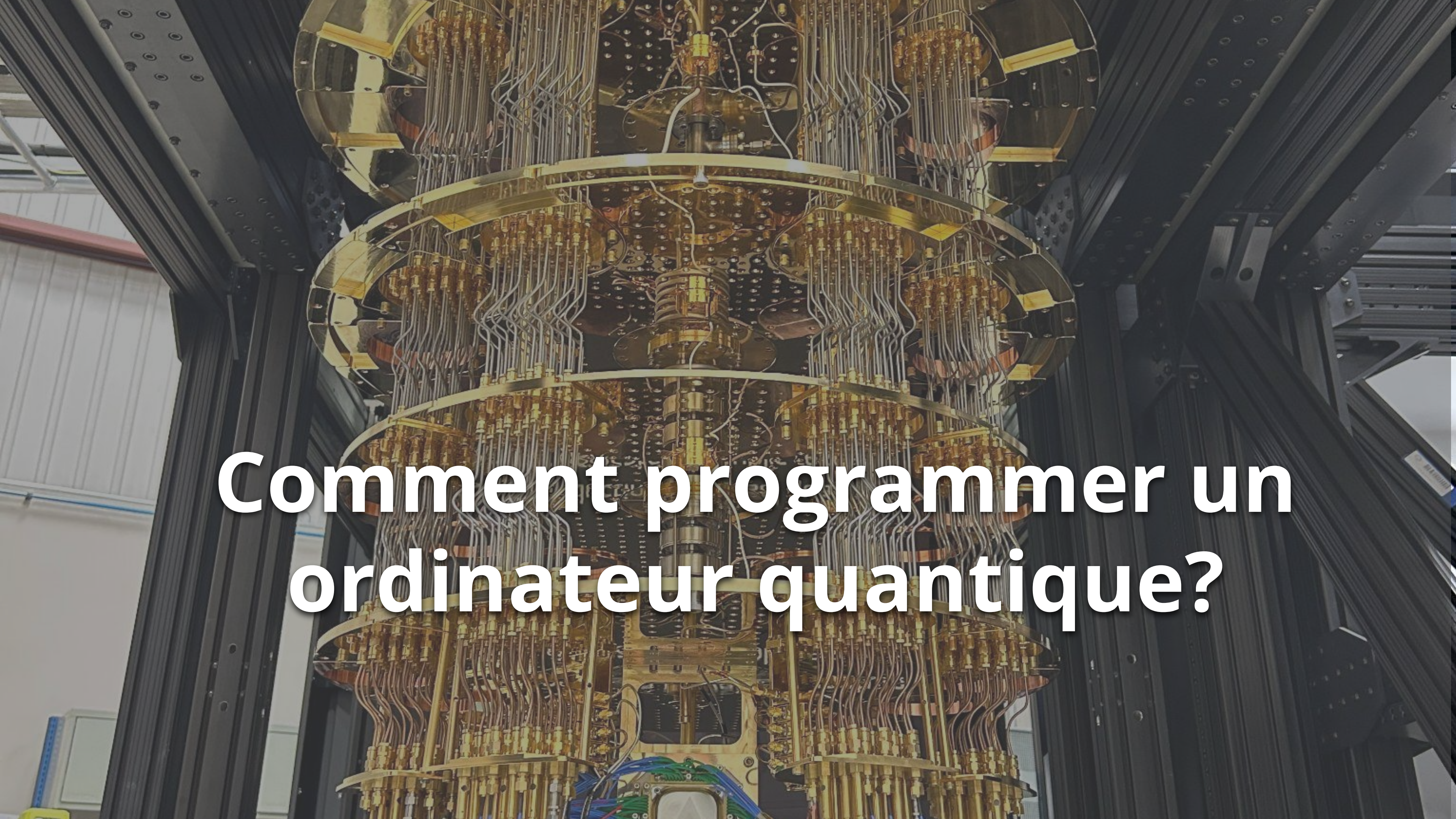
1. **Qubits** can be in a **superposition state**
   - $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$
   - Represented by a vector in a complex vector space

2. **Dirac** notation
   - A **ket** $|\psi\rangle$ is a column vector and its associated **bra** $\langle\psi|$ is a row vector obtained by taking the complex conjugate transpose.

3. **Bloch sphere**
   - **Visual representation of a qubit** as a vector on a sphere with θ and ϕ as coordinates for the state : $|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle$

4. Quantum **gates** are **unitary matrices** and act on qubits like **rotations** on the Bloch sphere.

5. A **quantum circuit** is a sequence of unitary transformations (gates) applied to an initial state.
   - $|\psi_{\text{final}}\rangle = U_n \cdots U_2 U_1 |\psi_{\text{initial}}\rangle$
   - **Measurement** gives **probabilistic** results based on the final state of the qubit



Comment programmer un ordinateur quantique?

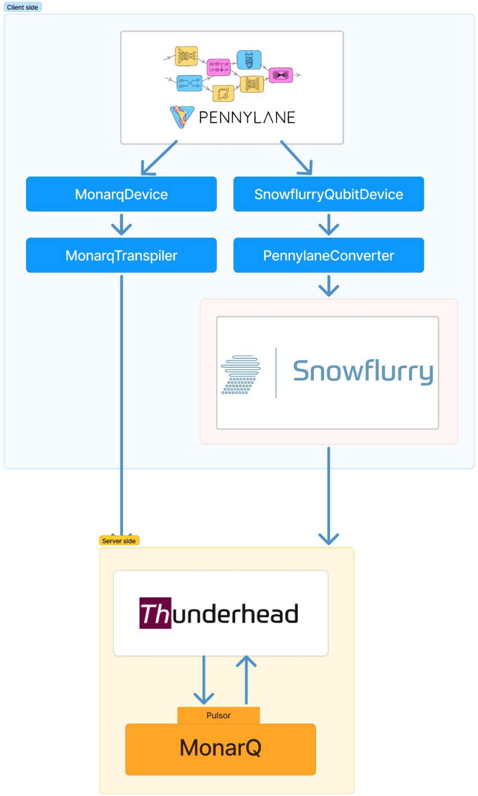
Quantum software development kits

- **PennyLane**
  - Developed in Python by Xanadu
  - Interface directly with MonarQ

- **Snowflurry**
  - Developed in Julia by Anyon Systems
  - Interface directly with MonarQ

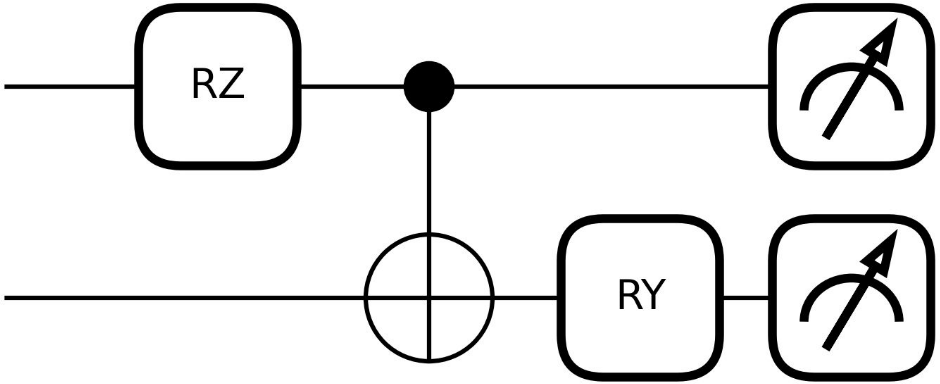
- **Qiskit**
  - Developed in Python by IBM
  - Qiskit-calculquebec plugin in development

write circuits and more



Quantum circuits

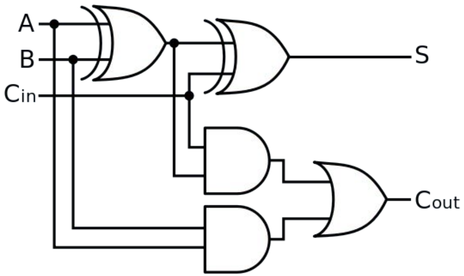
```
dev = qml.device('default.qubit', wires = 2)
@qml.qnode(dev) # qnode decorator
def quantum_function(x, y):
    qml.RZ(x, wires=0)
    qml.CNOT(wires=[0,1])
    qml.RY(y, wires=1)
    return qml.state()
```



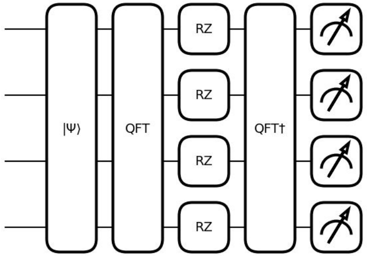
Quantum circuits

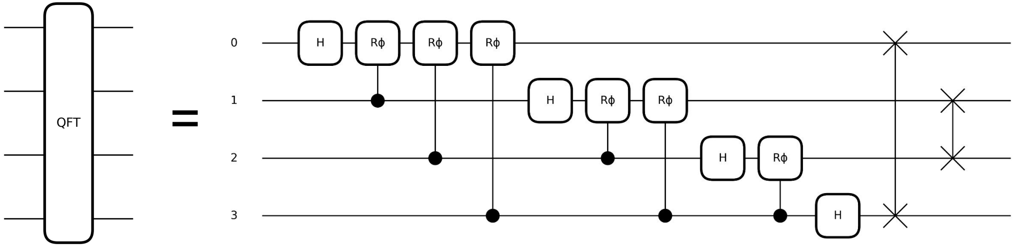
Classical computers also use circuits, but we do not generally think about these explicitly.

- **Classical** boolean circuit for addition



- **Quantum** circuit for addition



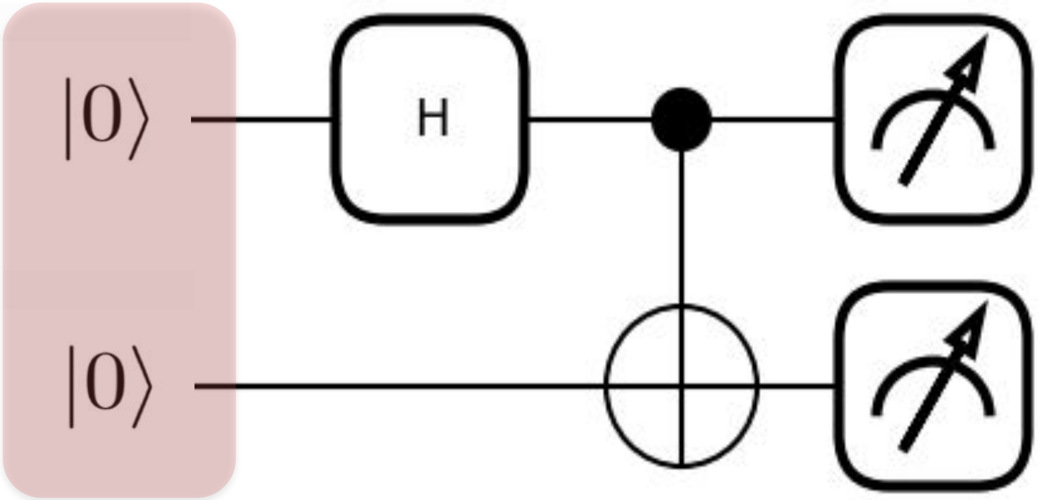


Key components of a quantum circuit

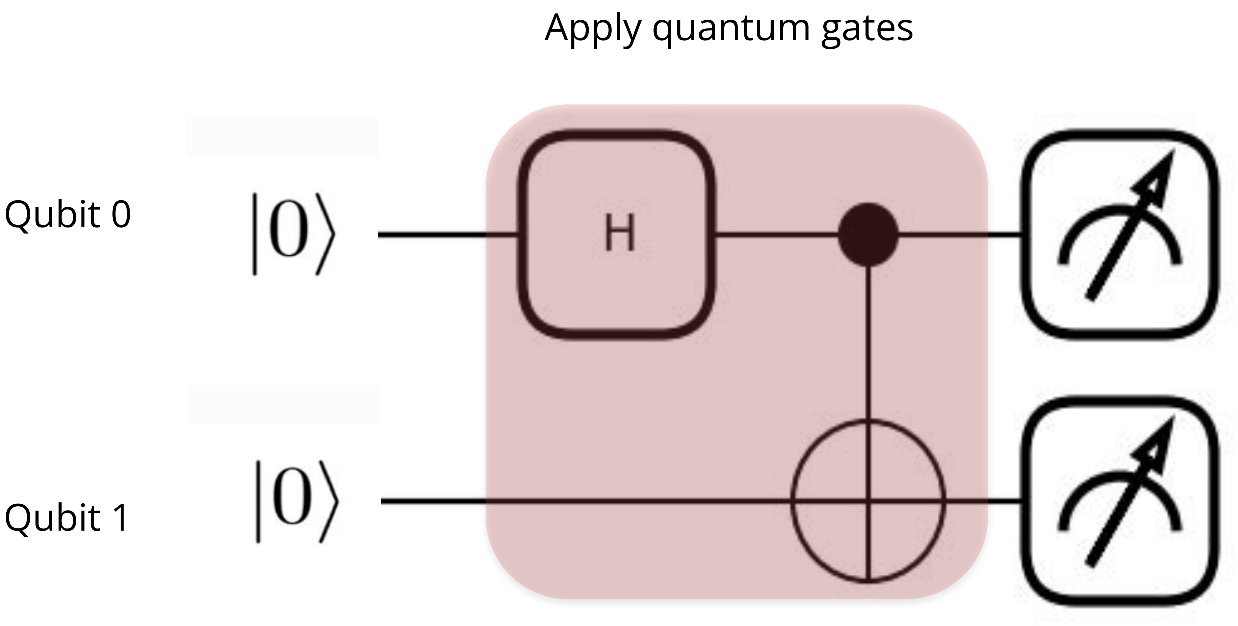
Each qubit is initialized
to state 0 by default

Qubit 0

Qubit 1

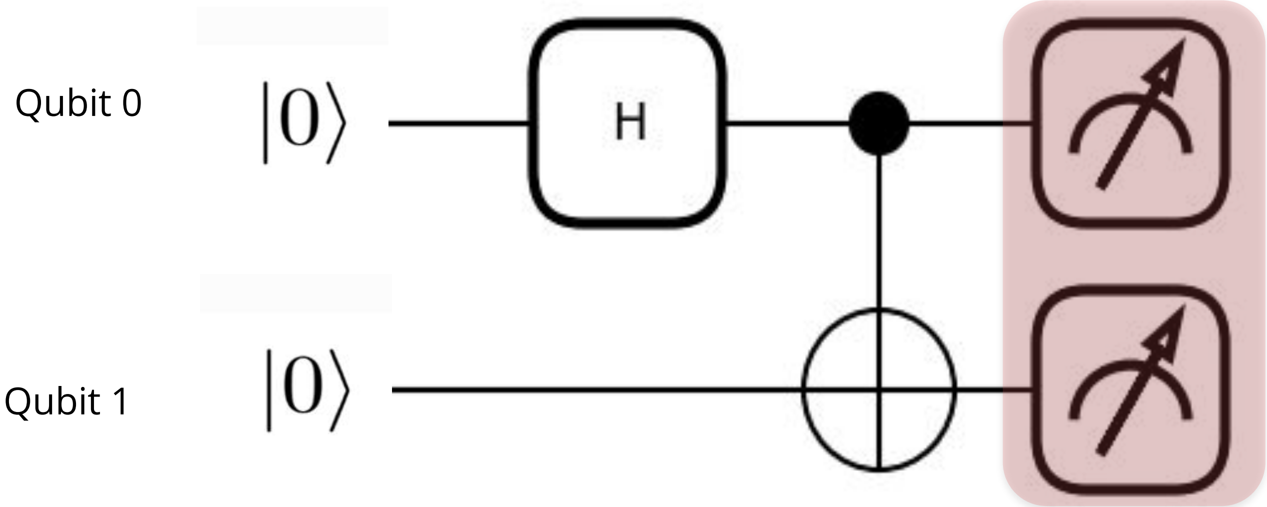


Key components of a quantum circuit



Key components of a quantum circuit

Measure the state of one or more qubit



Coding break

Notebook 2 :
PennyLane Basics

Notebook 2 cheat sheet

1. **Quantum circuits are composed of gates**

Gate	Circuit Element	Matrix Representation	Action on Basis States
Hadamard Gate H	$-\boxed{H}-$	$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$	$H\|0\rangle = \frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle)$ $H\|1\rangle = \frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle)$
Pauli-X Gate X	$-\boxed{X}-$	$\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$	$X\|0\rangle = \|1\rangle$ $X\|1\rangle = \|0\rangle$
Pauli-Y Gate Y	$-\boxed{Y}-$	$\begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}$	$Y\|0\rangle = i\|1\rangle$ $Y\|1\rangle = -i\|0\rangle$
Pauli-Z Gate Z	$-\boxed{Z}-$	$\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}$	$Z\|0\rangle = \|0\rangle$ $Z\|1\rangle = -\|1\rangle$
CNOT Gate	$\bullet$ / $\oplus$	$\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix}$	$CNOT\|00\rangle = \|00\rangle$ $CNOT\|01\rangle = \|01\rangle$ $CNOT\|10\rangle = \|11\rangle$ $CNOT\|11\rangle = \|10\rangle$

2. **Quantum circuits must return a measurement**
   (`qml.state`, `qml.expval`, `qml.probs`, `qml.counts`)

3. **Link a circuit and a device together with a Qnode**

Qnode

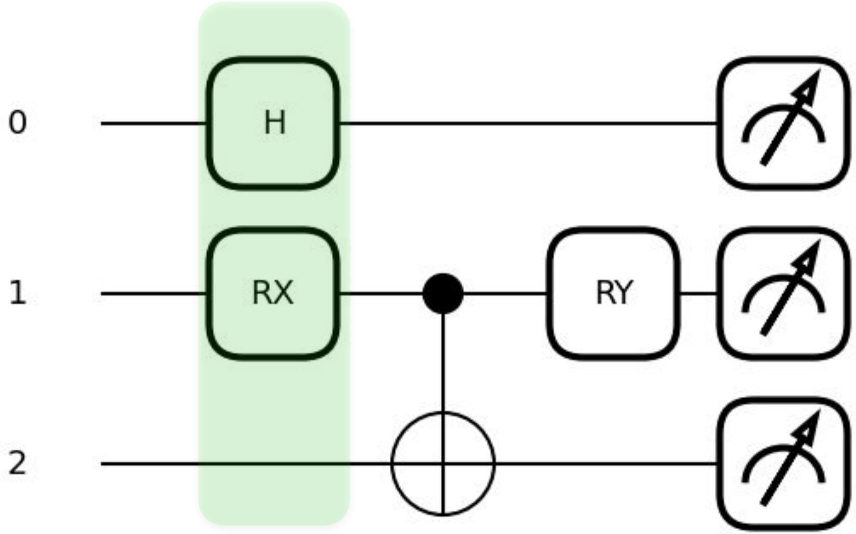
Quantum function

```
def circuit():
  …
 return qml.counts(0)
```

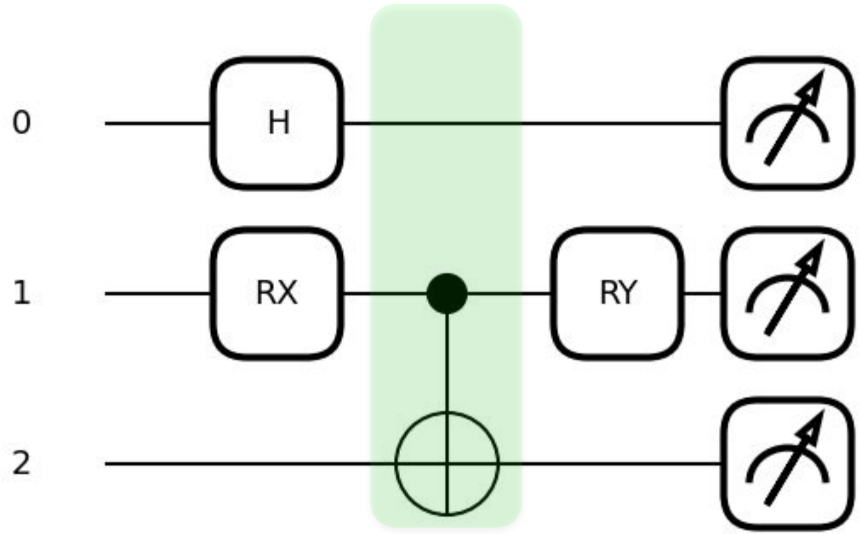
Device

```
qml.device(...)
```

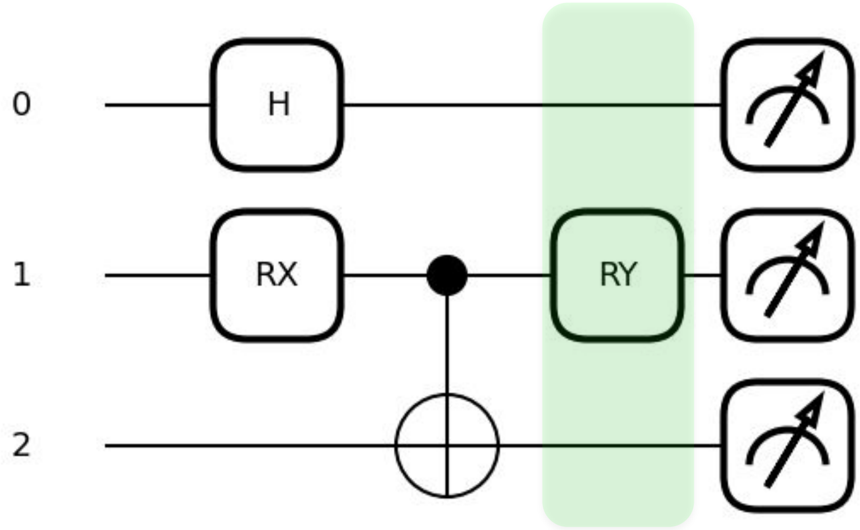
Circuit depth



Circuit depth

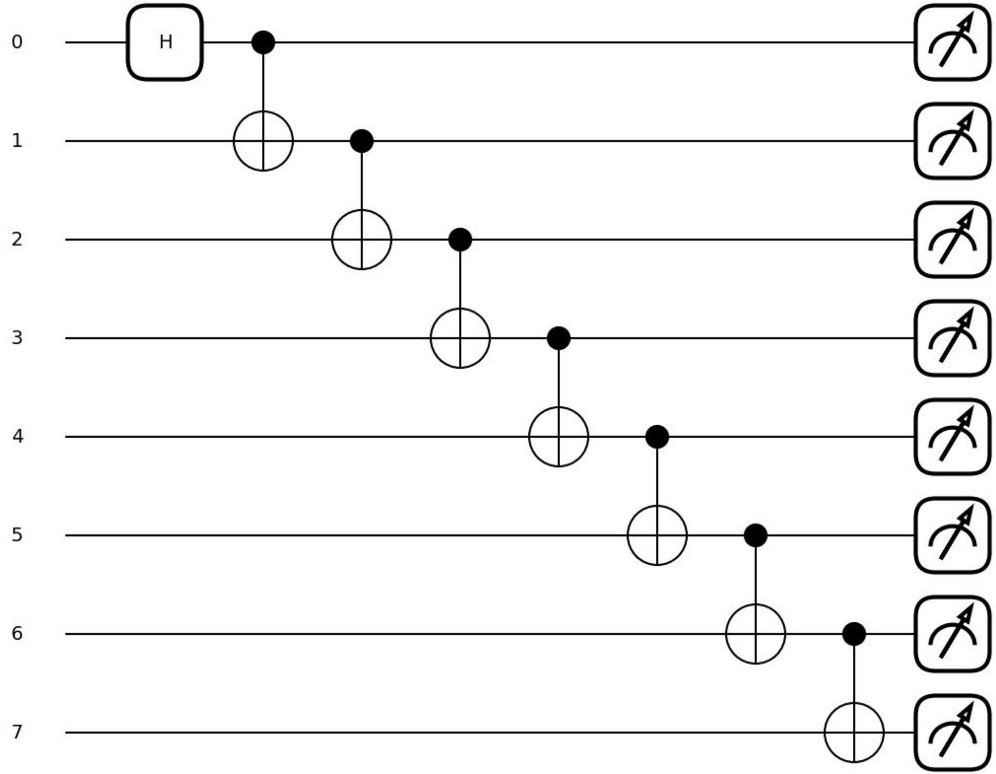


Circuit depth



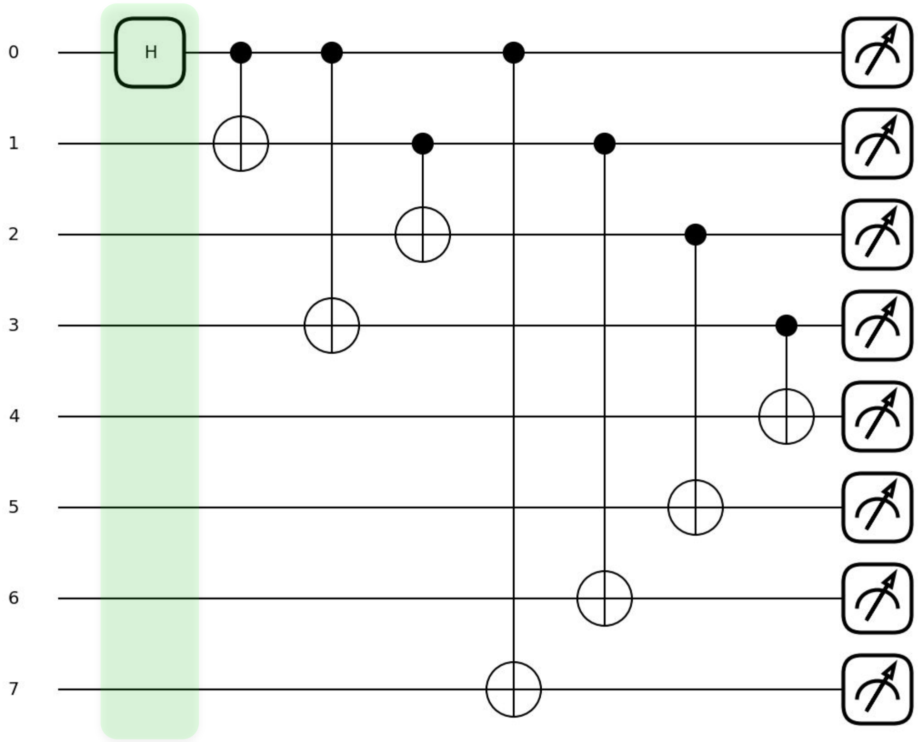
The depth of this circuit is 3.

Circuit depth

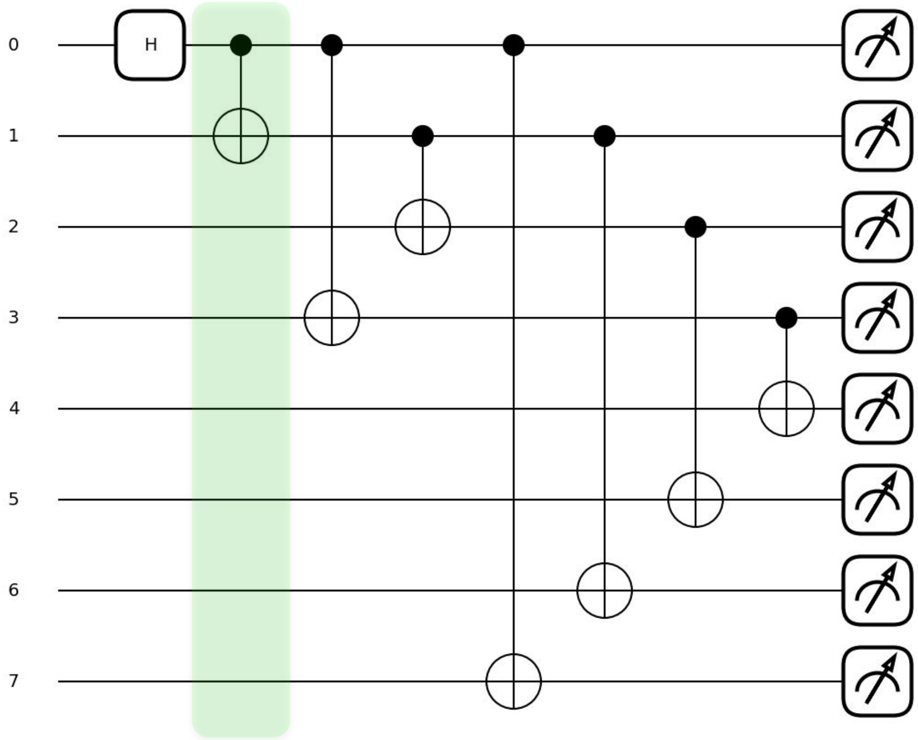


What is the depth of this circuit?

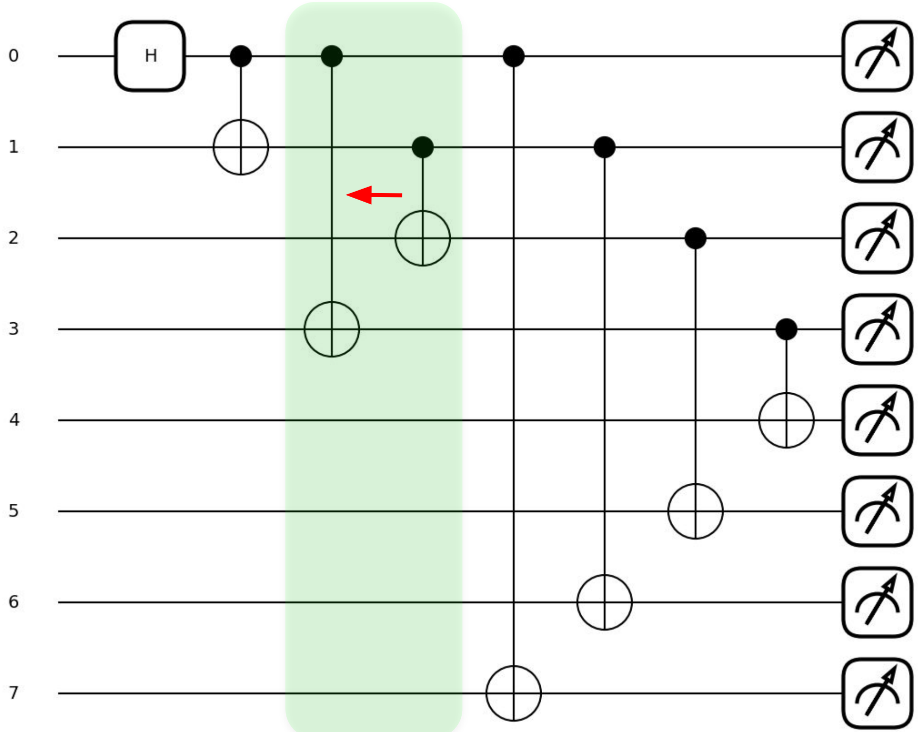
Circuit depth



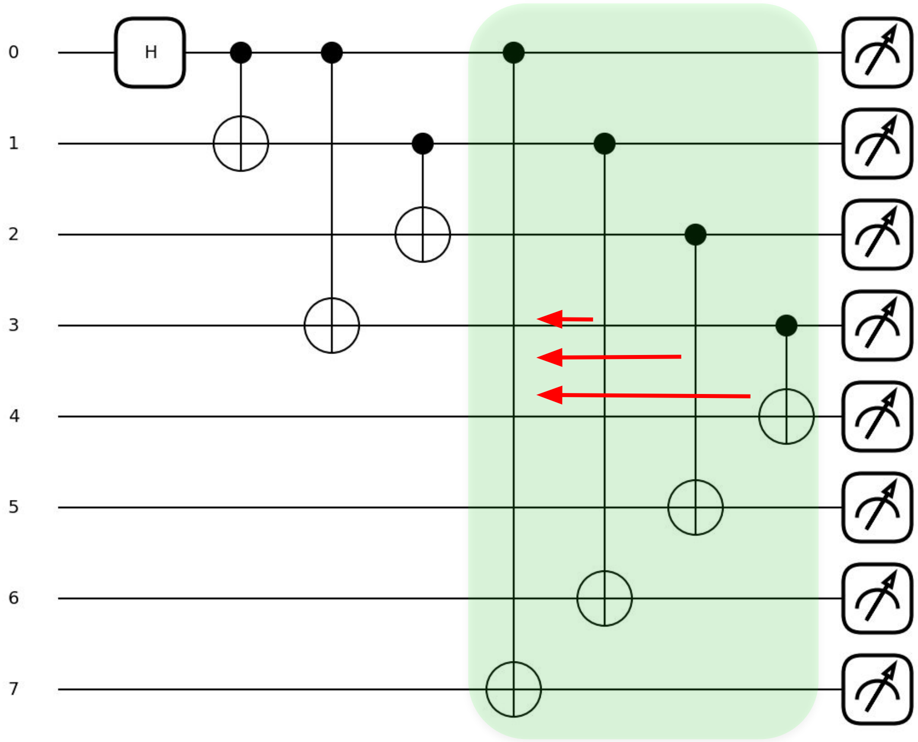
Circuit depth



Circuit depth

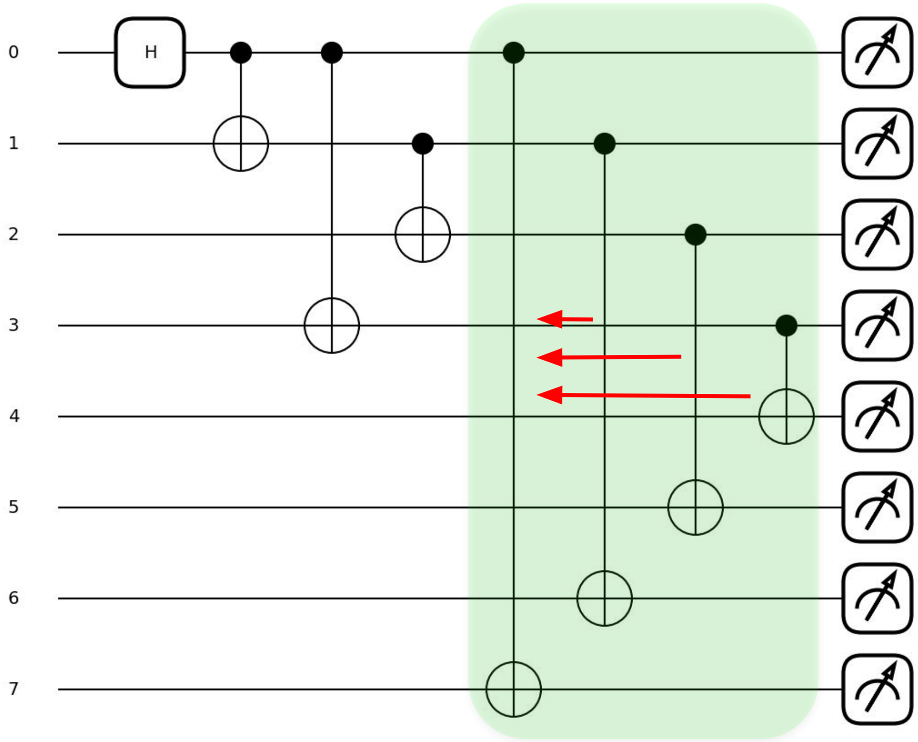


Circuit depth



The depth of this circuit is 4.

Circuit depth



Calcul Québec