

# Algoritmo de diseño matricial de circuitos cuánticos



Ponente: Javier París Uhryn  
Tutora: Estrella Pulido Cañabate





# Índice

- Computación cuántica

- ¿Qué es?
- ¿Cómo funciona?
- ¿Para qué sirve?

- Algoritmos cuánticos

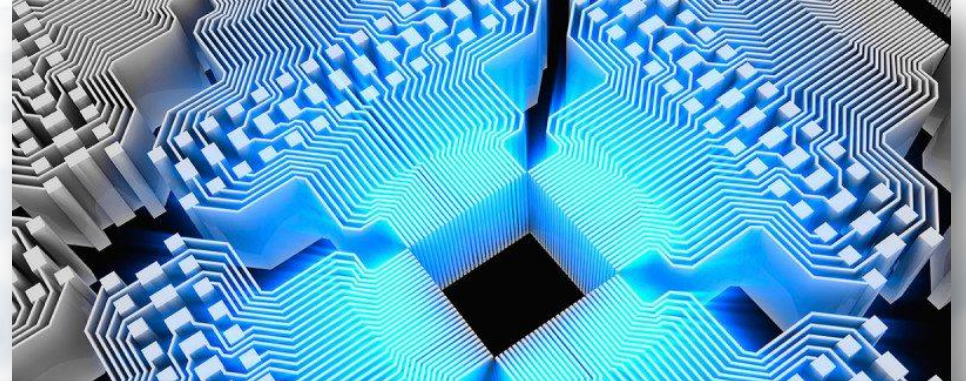
## ¿Qué es la computación cuántica?

- ¿Cómo se pueden usar?

- Diseño del algoritmo QCMD

- ¿Cuál es su importancia?
- ¿Cómo se demuestra su funcionamiento?

- Conclusiones





# Computación cuántica



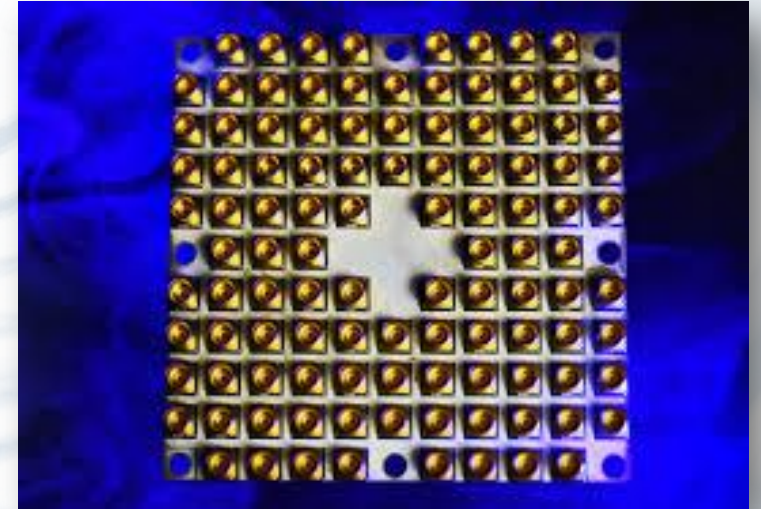
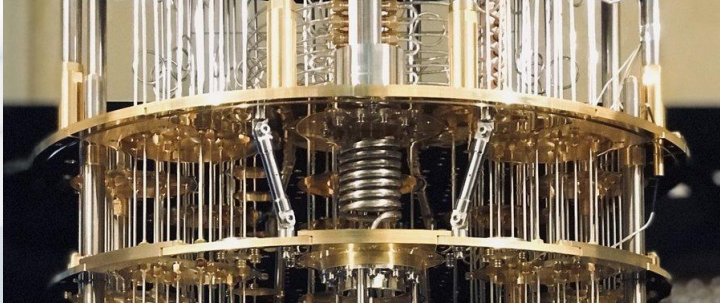
¿Este paradigma es usable en el mundo real?





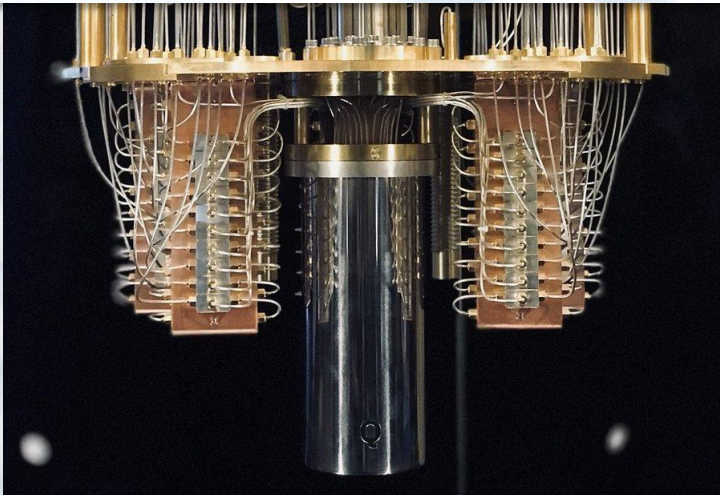
# Computación cuántica

## Ordenador cuántico



Intel Quantum Microprocessor

**¿Qué hace especial a la computación cuántica?**



IBM Quantum Computer



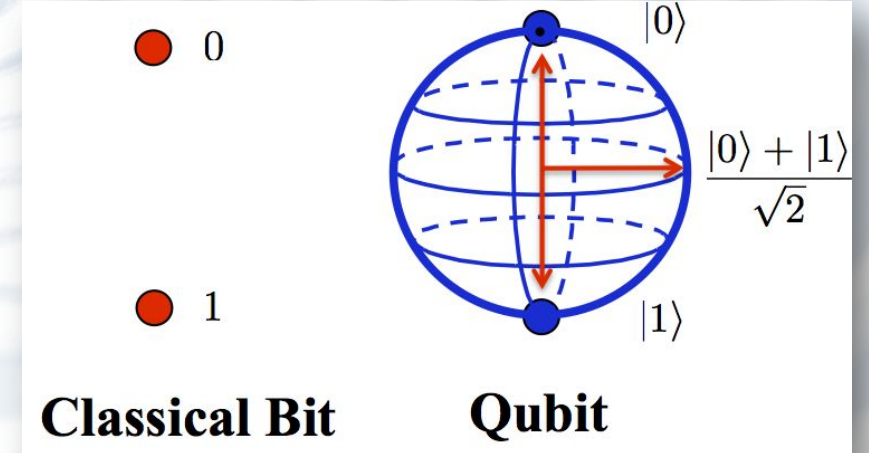
D-Wave Quantum Computer



# Computación cuántica

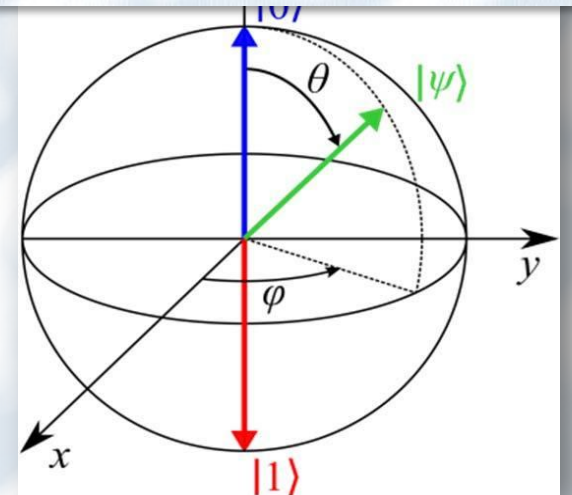
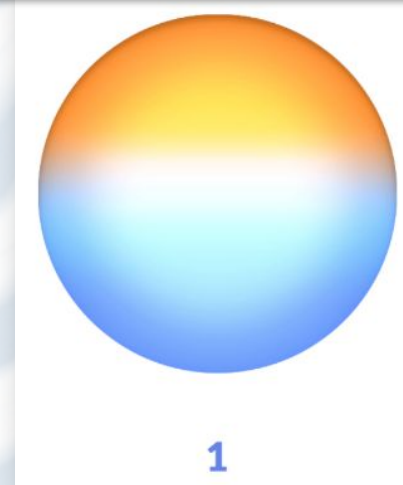
## Qubit

- Puede ser 0 y 1 a la vez
- La medida dará 0 o 1 con



## ¿Qué se puede hacer con un qubit?

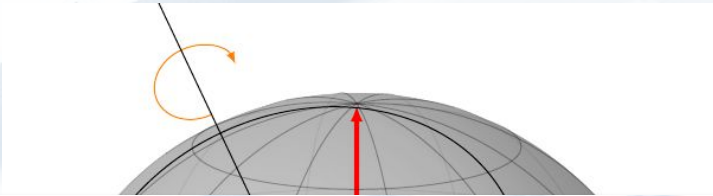
- Se parametriza con 2 variables



# Computación cuántica

## Operaciones

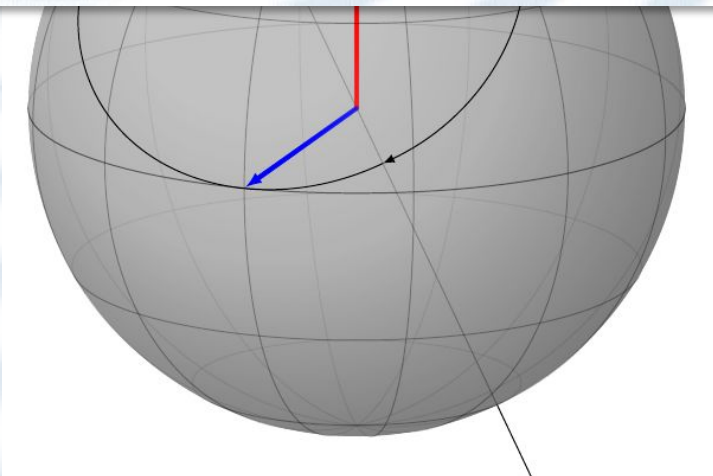
- Giros sobre la esfera



- Se representa por una matriz 2x2

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

¿Para qué sirve si solo se puede obtener un 1 o un 0?



Giro de Hadamard

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

Representación  
matemática

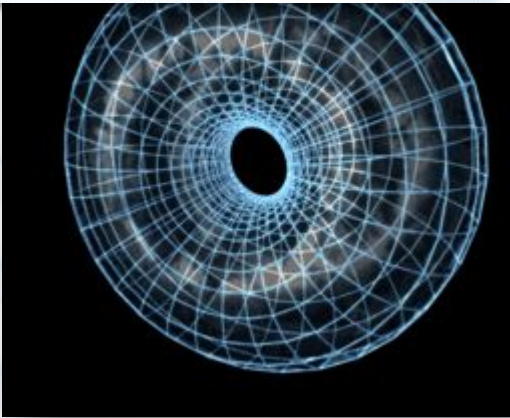


# Computación cuántica

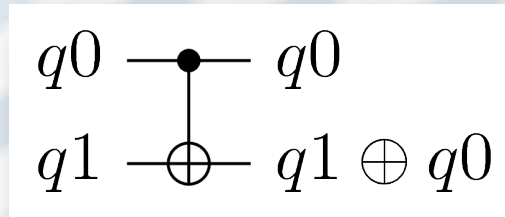
## Más de un qubit

- Cada qubit duplica el número de dimensiones
- Cada matriz será de tamaño  $2^{(n \text{ qubits})}$

¿Cómo se puede usar estos qubits?



Representación abstracta de sistemas de más de 3 dimensiones



Representación de puerta XNOT en circuito

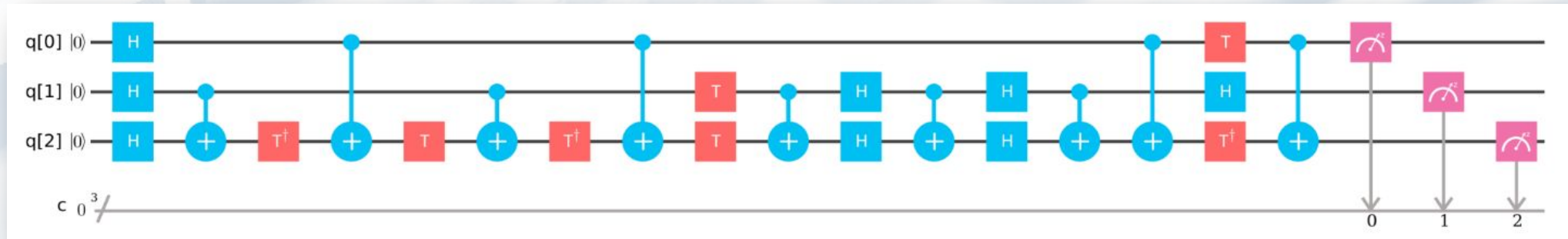
$$\begin{matrix} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{matrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Matriz de puerta XNOT

# Computación cuántica

## Circuito cuántico I

- Una concatenación de giros y puertas XNOT generan un circuito cuántico



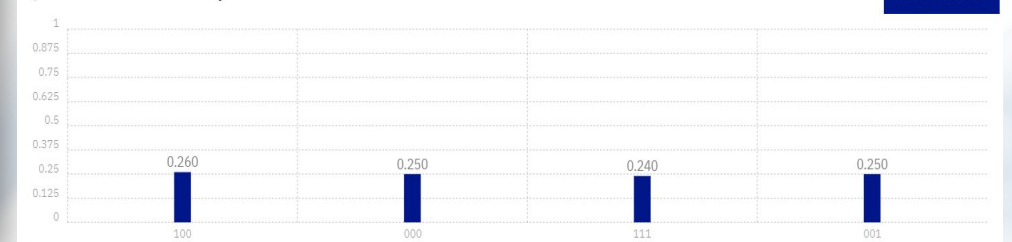
Circuito que genera una puerta Toffoli

```

1  include "qelib1.inc";
2  qreg q[3];
3  creg c[3];
4
5  h q[0];
6  h q[1];
7  h q[2];
8  cx q[1],q[2];
9  tdg q[2];
10 cx q[0],q[2];
11 t q[2];
12 cx q[1],q[2];
13 tdg q[2];
14 cx q[0],q[2];
15 t q[1];
16 t q[2];
17 cx q[1],q[2];
18 h q[1];
19 h q[2];
20 cx q[1],q[2];
21 h q[1];
22 h q[2];
23 cx q[1],q[2];
24 cx q[0],q[2];
25 t q[0];
26 h q[1];
27 tdg q[2];
28 cx q[0],q[2];
29 measure q[0] -> c[0];
30 measure q[1] -> c[1];
31 measure q[2] -> c[2];

```

Quantum State: Computation Basis

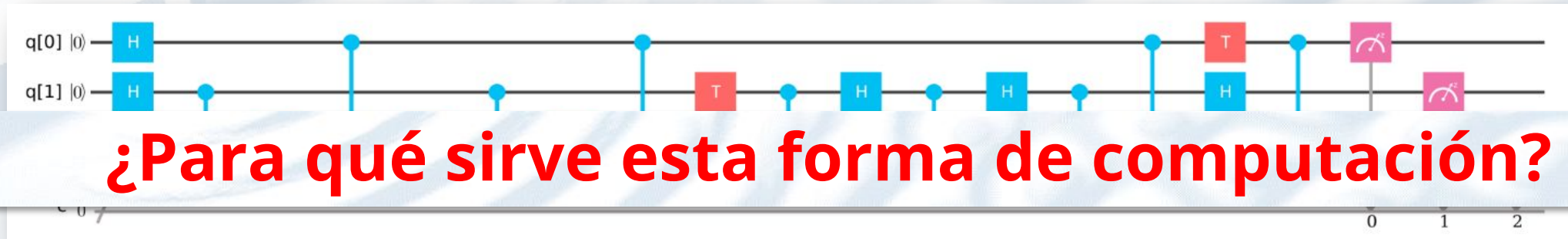




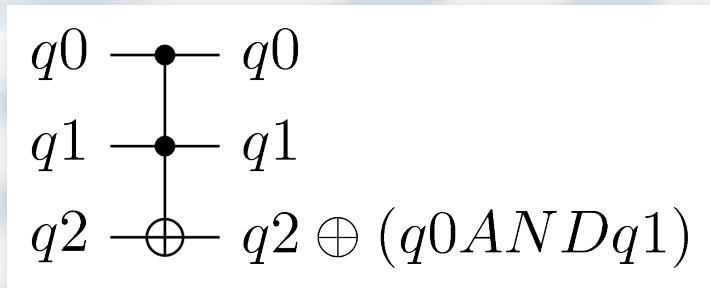
# Computación cuántica

## Circuito cuántico II

- Una concatenación de giros y puertas XNOT generan un circuito cuántico



Circuito que genera una puerta Toffoli

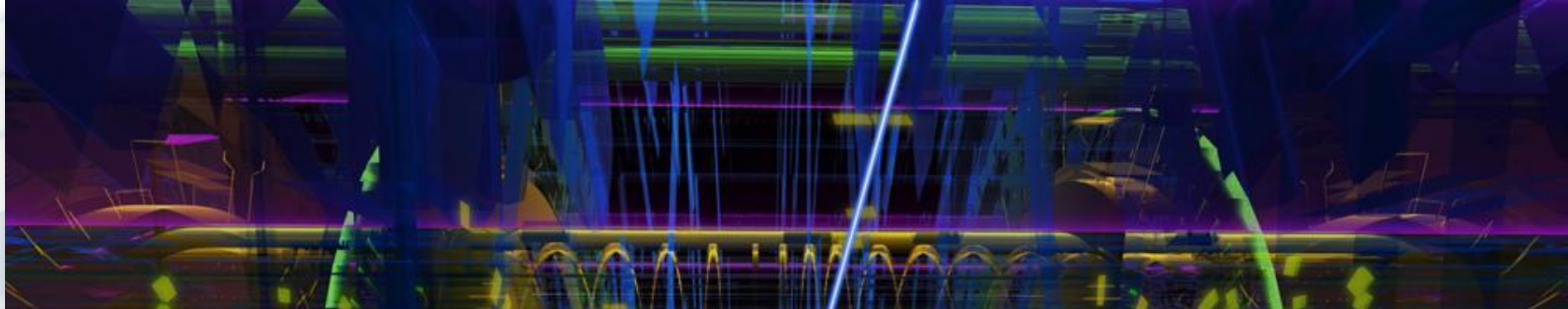


1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	0

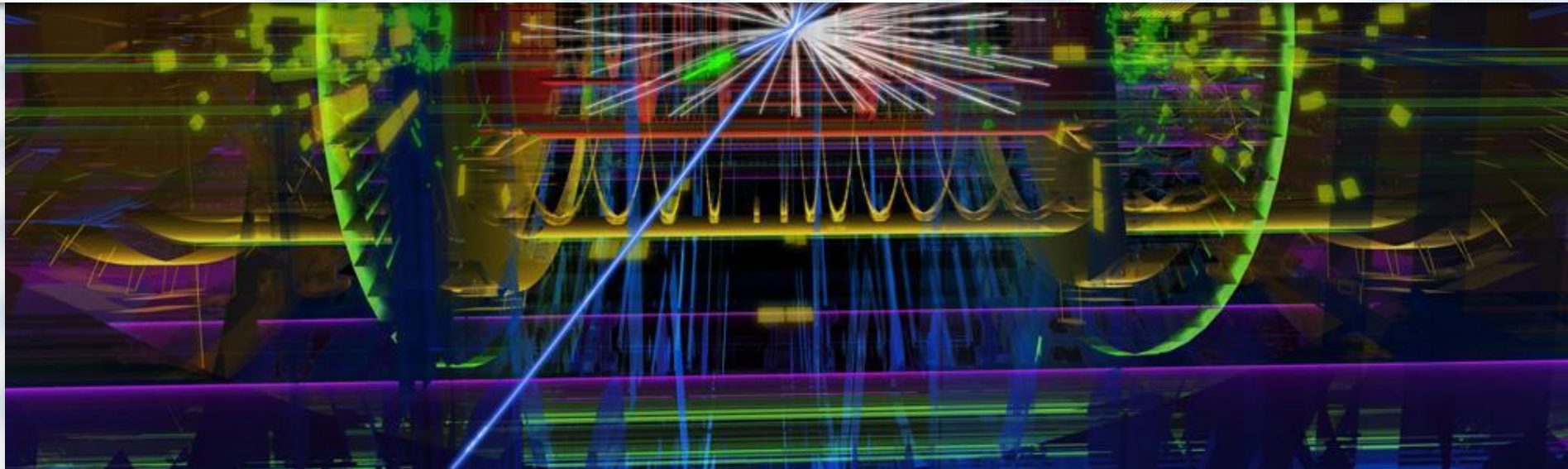
Matriz de la puerta Toffoli



# Algoritmos cuánticos



**¿Qué problemas se pueden resolver con estos algoritmos?**



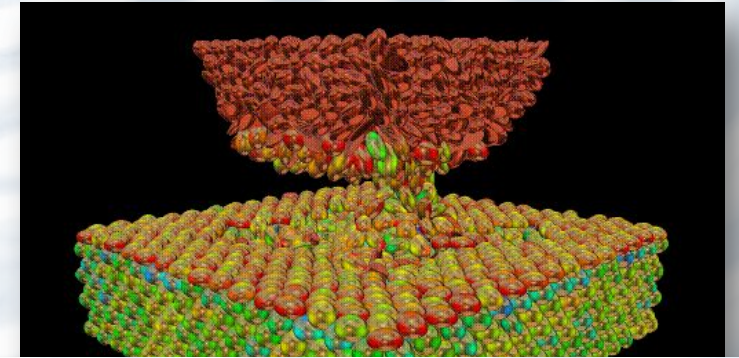


# Algoritmos cuánticos

## Utilidades

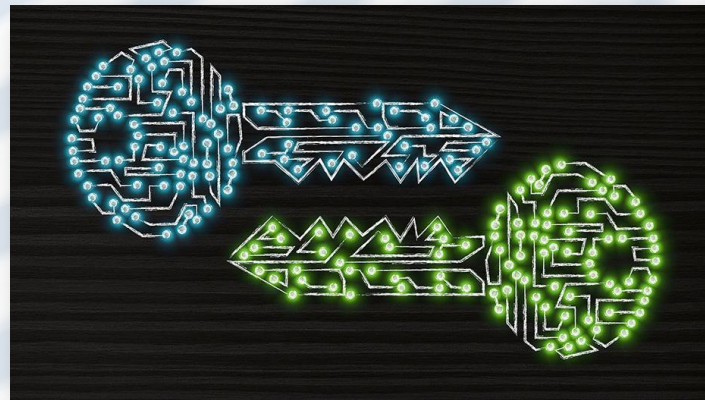


Algoritmos de búsqueda



Simulación de sistemas cuánticos  
(químicos y físicos)

**¿Qué se puede buscar si no es posible guardar información?**



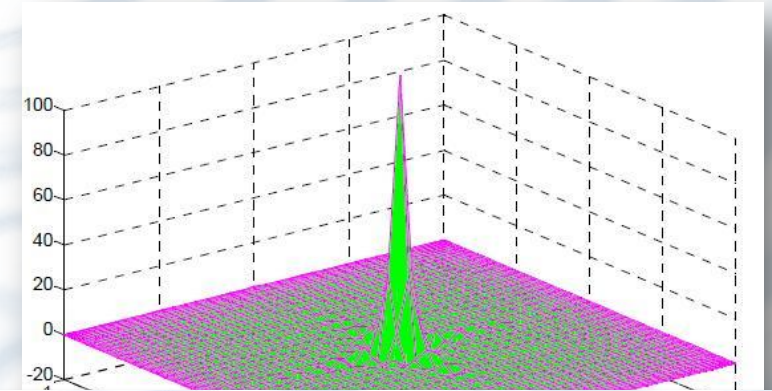
Factorización de números



# Algoritmos cuánticos

## Algoritmo de Grover - uso

- Búsqueda de elementos con características concretas entre N elementos
- Menor complejidad que cualquier algoritmo

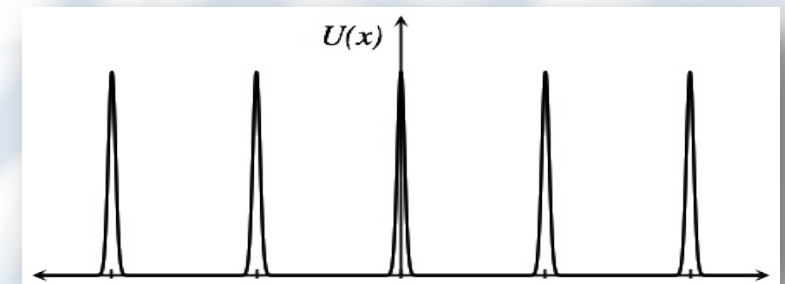


¿Cómo hace este algoritmo para buscar?

$$A_{grover} \approx \frac{\pi\sqrt{N}}{4} = \mathbf{O}(\sqrt{N})$$

$$A_{search} \approx \frac{N}{2} = \mathbf{O}(N)$$

Búsqueda de 1 elemento



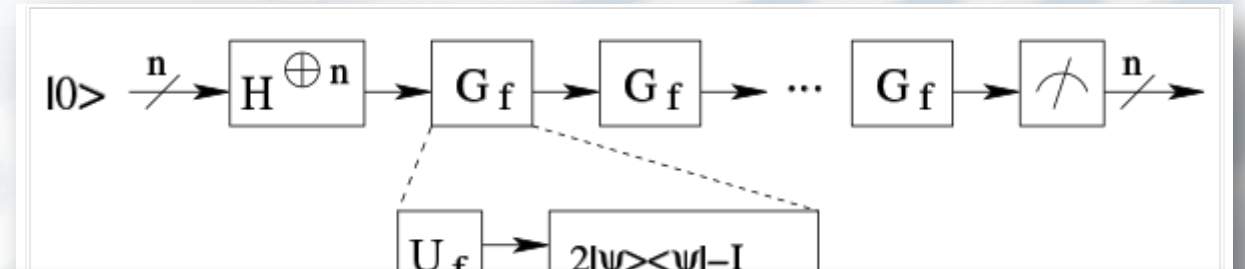
Búsqueda de varios elementos



# Algoritmos cuánticos

## Algoritmo de Grover - funcionamiento

- Estado de superposición
- Iteración de Grover
- Oráculo



**¿Cómo se puede implementar este algoritmo?**

$$U_f = (-1)^{f(x)}$$

$$\begin{bmatrix} a_0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & a_{n-1} \end{bmatrix}$$

Matriz oráculo

$$\begin{bmatrix} \frac{1}{\sqrt{N}} - 1 & \cdots & \frac{1}{\sqrt{N}} \\ \vdots & \ddots & \vdots \\ \frac{1}{\sqrt{N}} & \cdots & \frac{1}{\sqrt{N}} - 1 \end{bmatrix}$$

Matriz amplificación de amplitud



# Algoritmo de diseño matricial de circuitos cuánticos



¿Cómo funciona el algoritmo?





# Algoritmo QCMD

## Fundamento teórico

- Dividir la matriz en submatrices de 2x2
- Se repite este proceso hasta llegar a la matriz identidad

$$\begin{bmatrix} a_{00}^* & a_{10}^* & 0 & 0 \\ a_{10} & -a_{00} & 0 & 0 \end{bmatrix} \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a'_{00} & a'_{01} & a'_{02} & a'_{03} \\ 0 & a'_{11} & a'_{12} & a'_{13} \\ a'_{20} & a'_{21} & a'_{22} & a'_{23} \\ a'_{30} & a'_{31} & a'_{32} & a'_{33} \end{bmatrix}$$

$$\sum_{i=1}^{N-1} i = \frac{N(N-1)}{2} = \frac{2^{2n} - 2^n}{2}$$

**¿Cómo se llega a estas matrices más pequeñas?**

Número de submatrices

$$\begin{bmatrix} a'_{00} & 0 & a'_{20} & 0 \\ 0 & 1 & 0 & 0 \\ a'_{20} & 0 & -a'_{00} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a'_{00} & a'_{01} & a'_{02} & a'_{03} \\ 0 & a'_{11} & a'_{12} & a'_{13} \\ a'_{20} & a'_{21} & a'_{22} & a'_{23} \\ a'_{30} & a'_{31} & a'_{32} & a'_{33} \end{bmatrix} = \begin{bmatrix} a''_{00} & a''_{01} & a''_{02} & a''_{03} \\ 0 & a'_{11} & a'_{12} & a'_{13} \\ 0 & a'_{21} & a'_{22} & a'_{23} \\ a'_{30} & a'_{31} & a'_{32} & a'_{33} \end{bmatrix}$$

$$O(n12^n)$$

Complejidad total del  
algoritmo

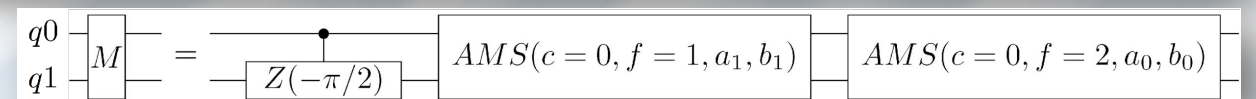
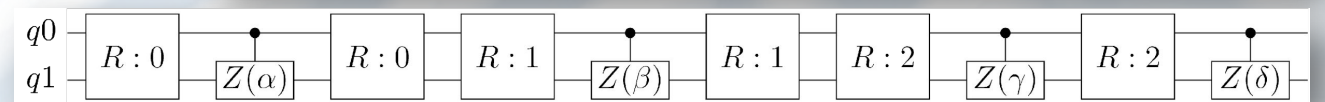
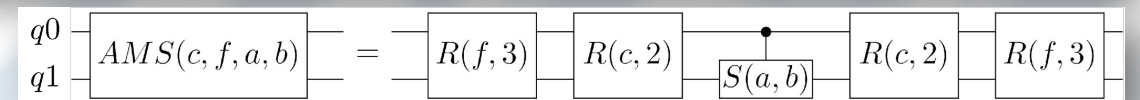
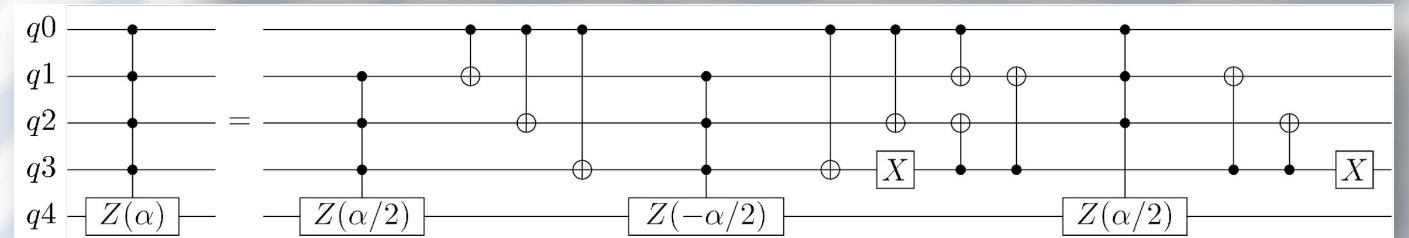
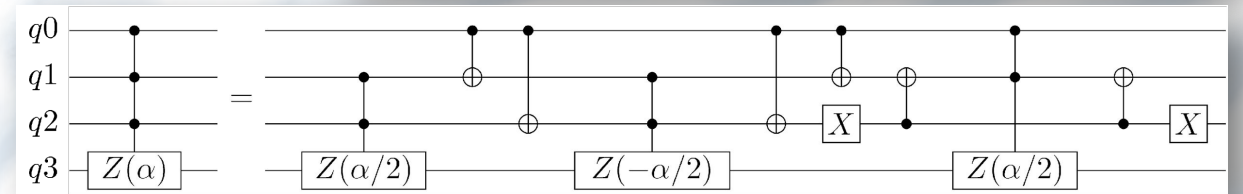
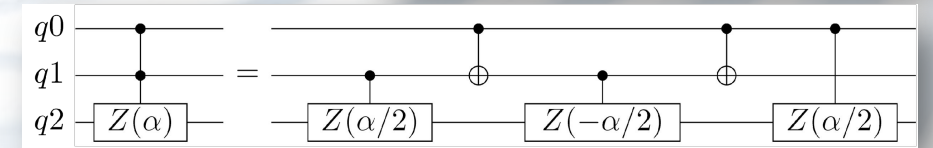
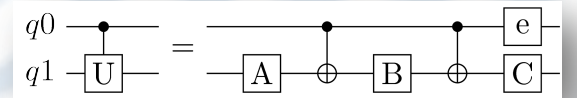
$$\begin{bmatrix} a''_{00} & 0 & 0 & a''_{30} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ a'_{30} & 0 & 0 & -a''_{00} \end{bmatrix} \begin{bmatrix} a''_{00} & a''_{01} & a''_{02} & a''_{03} \\ 0 & a'_{11} & a'_{12} & a'_{13} \\ 0 & a'_{21} & a'_{22} & a'_{23} \\ a'_{30} & a'_{31} & a'_{32} & a'_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & a'_{11} & a'_{12} & a'_{13} \\ 0 & a'_{21} & a'_{22} & a'_{23} \\ 0 & a'_{31} & a'_{32} & a'_{33} \end{bmatrix}$$



# Algoritmo QCMD

## Ejemplo de circuitos intermedios I

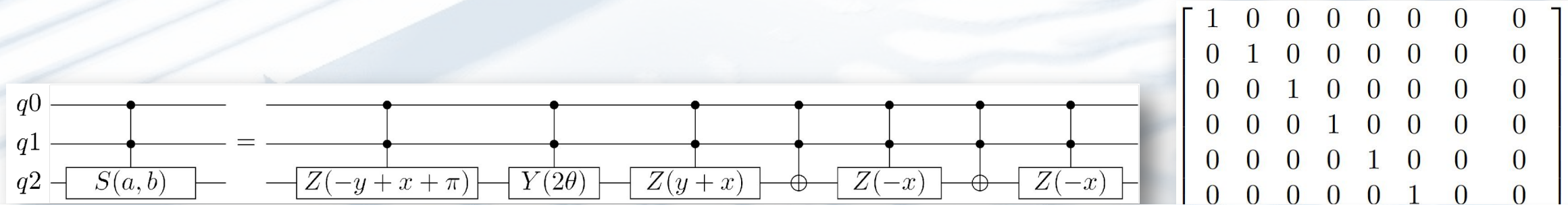
- Puertas básicas
  - Puerta de giro
  - Puerta XNOT
- Puerta de giro condicionado simple
- Puerta de giro condicionado múltiple
- Puerta giro especial
- Puerta de intercambio
- Puerta de ajuste matricial simple
- Puerta de ajuste diagonal
- Puerta de ajuste matricial



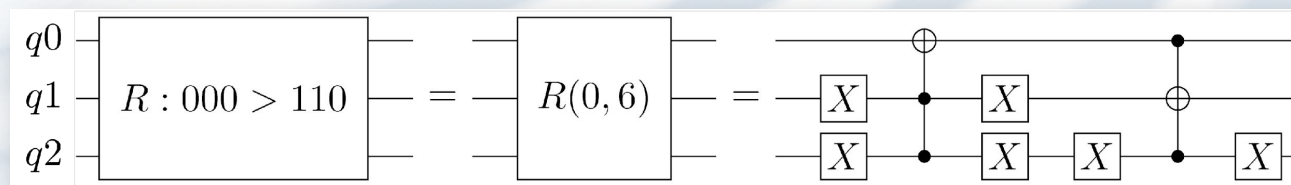


# Algoritmo QCMD

## Ejemplo de circuitos intermedios II



¿Cómo se puede estar seguro de que funciona?



Diseño y matriz para la puerta de *intercambio*

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



# Algoritmo QCMD

## Implementación

```
# inicializa un sistema de diccionarios para 4 qubits
dic = DictionarySearcher(nQubits=4)

# genera una puerta de giro condicionado multiple y las subpuertas necesarias
gateId = AQP.generateMultipleTurn(sources=[0,1,2], target=3, angle=math.pi/2,
    turnType=TurnType.Z, dic=dic)
```

Código  
para crear  
una puerta  
de giro  
múltiple

## ¿Cómo se sabe que la matriz no es la que te dan?

```
// Assembler file: multiConditionalGate.qasm for id: 35 //

OPENQASM 2.0;
include "qelib1.inc";
qreg q[4];
creg c[4];

// X-NOT - source: 2 target: 3 //
cx q[1], q[0];
// X-NOT - source: 2 target: 3 //
```

Código  
qasm para  
ejecutar un  
circuito  
cuántico

# Algoritmo QCMD

## Resultados simulados I

```
Assembler written with 433 gates

Generated matrix
[[-0.5+0.j  0.5-0.j  0.5-0.j  0.5-0.j]
 [ 0.5-0.j -0.5+0.j  0.5-0.j  0.5-0.j]
 [ 0.5-0.j  0.5-0.j -0.5+0.j  0.5-0.j]
 [ 0.5-0.j  0.5-0.j  0.5-0.j -0.5+0.j]]

Simulated matrix
[[-0.5+0.j  0.5-0.j  0.5-0.j  0.5-0.j]
 [ 0.5-0.j -0.5+0.j  0.5-0.j  0.5-0.j]
 [ 0.5-0.j  0.5-0.j -0.5+0.j  0.5-0.j]
 [ 0.5-0.j  0.5-0.j  0.5-0.j -0.5+0.j]]

CLOSE
```

Resultado del diseño para una  
matriz de amplificación de  
amplitud de 2 qubits





# Algoritmo QCMD

## Resultados simulados II

```
Assembler written with 9777 gates

Generated matrix
[[-0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j]]

Simulated matrix
[[-0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j  0.25-0.j]
 [ 0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j  0.25-0.j -0.75+0.j]]

CLOSE
```

Resultado del diseño para una matriz de amplificación de amplitud de 3 qubits





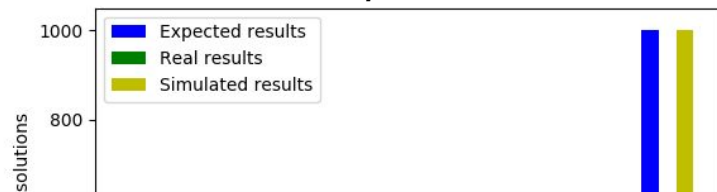




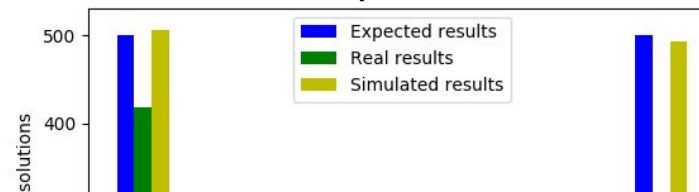
# Algoritmo QCMD

## Resultados en el ordenador cuántico

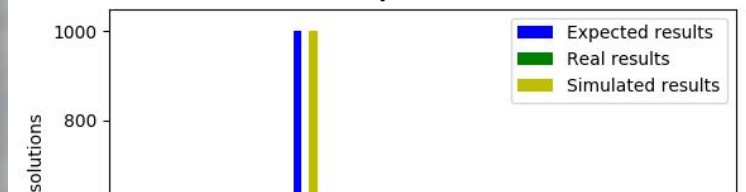
2 qubits



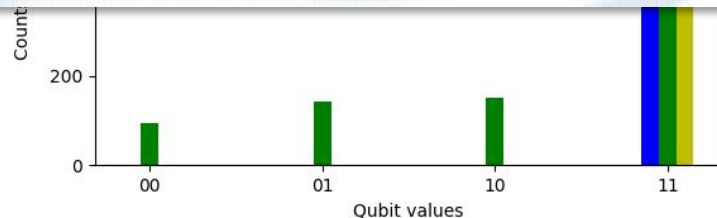
2 qubits



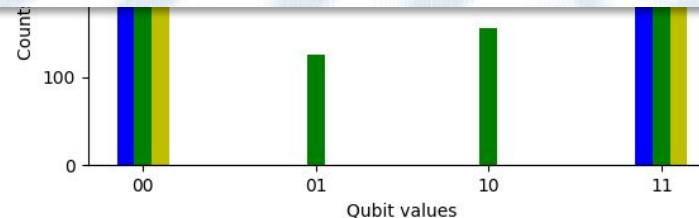
3 qubits



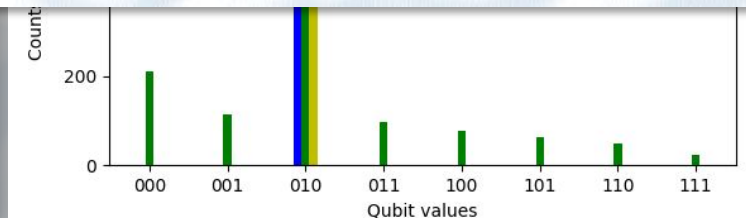
¿Qué futuro tiene?



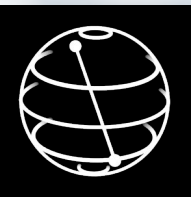
Resultado de una puerta de intercambio entre los estados 0 y 3



Resultado de una puerta de inicialización de estado de superposición para los estados 0 y 3



Resultado de una puerta de intercambio entre los estados 0 y 2



# Conclusiones



If quantum mechanics hasn't profoundly shocked you, you haven't understood it yet.  
Niels Bohr



# Referencias

- Índice
  - Procesador
    - <https://www.newscientist.com/article/2151032-googles-quantum-computing-plans-threatened-by-ibm-curveball/>
- Computación cuántica
  - Átomo
    - <https://www.wired.com/story/the-era-of-quantum-computing-is-here-outlook-cloudy/>
- Ordenador cuántico
  - IBM
    - <https://www.flickr.com/photos/criminalintent/39660636671>
  - Intel
    - <http://www.techeblog.com/index.php/tech-gadget/fascinating-look-at-intel-s-49-qubit-superconducting-quantum-cpu>
  - D-Wave
    - <http://androgeek.hu/hir/dwave-2x-szazmillioszor-gyorsabb-mint-egy-pc>
- Qubit
  - Esfera de Bloch
    - [https://www.researchgate.net/figure/Bloch-sphere-representation-of-a-qubit\\_fig1\\_317573486](https://www.researchgate.net/figure/Bloch-sphere-representation-of-a-qubit_fig1_317573486)
  - Degradado
    - <https://hackernoon.com/a-brief-introduction-to-quantum-computing-d21e578cb7ed>
  - Bit vs Qubit
    - <https://www.quora.com/How-is-computing-with-qubits-more-difficult-and-efficient-than-with-classical-BITS>
- Operaciones
  - Gif 4 dimensiones
    - [https://en.wikipedia.org/wiki/Four-dimensional\\_space#/media/File:Clifford-torus.gif](https://en.wikipedia.org/wiki/Four-dimensional_space#/media/File:Clifford-torus.gif)
- Algoritmos cuánticos
  - Colisión
    - <https://arstechnica.com/science/2017/10/quantum-algorithm-finds-higgs-needle-in-photon-haystack/>
- Utilidades
  - DNA
    - <https://www.lanacion.com.ar/1929021-describen-el-mapa-de-mutaciones-geneticas-y-su-frecuencia-en-la-especie-humana>
  - Compuesto químico
    - <https://www.cc.gatech.edu/scivis/research/atom/atom.html>
  - Criptografía
    - <https://coincentral.com/cryptography-a-brief-history-from-symmetry-to-bitcoin/>
- Grover – funcionamiento
  - Esquema
    - [https://commons.wikimedia.org/wiki/File:Grover\\_algorithm\\_circuit\\_3.svg](https://commons.wikimedia.org/wiki/File:Grover_algorithm_circuit_3.svg)
- Grover – uso
  - Tridimensional
    - [http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S1679-78252016000901763](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1679-78252016000901763)
  - Bidimensional
    - <http://jhwilson.com/2013/06/14/delta-function-lattice/>
- QCMD Alg
  - Números y qubits
    - <https://ipcsdemo.wordpress.com/2016/03/07/quantum-hacking-is-now-possible-with-shors-algorithm-implementation/>
- Resultados
  - Logo Qiskit
    - <https://qiskit.org/>
- Conclusiones
  - Átomo
    - <https://www.araposts.com/2016/08/quantum-mechanics.html>
- Fondo
  - <https://spiritualunderground.com/2017/01/30/time-and-space-are-not-what-you-think/>