Simon Algorithm

Terminology

DUP	Duplicate
Hadamard gate	Averiguar la bibliografia de Hadamard

Revisar lo estudiado em la materia de Investigación visto en UTEL.

1. Introduction

Disjunctive Normal Form

Any Boolean function $f : \{0,1\}^n \rightarrow \{0,1\}$ can be constructed out of AND, OR, NOT and DUP gates.

One way to see this is the Disjunctive Normal Form.

For every $x \in \{0,1\}^n$ there is a clause that evaluates to one on and only on x.

Example: $x_1 \wedge \neg x_2 \wedge x_3$ evaluates to one only on the string 101

We can take the OR of all such conjunctions for the strings on which f evaluates to one.

How efficient this representation is?

Every clause is going to be of size n.

The number of gates in a disjunctive normal form is $O(n2^n)$

Shannon demonstrates that most Boolean functions require at least $\frac{2^n}{3n}$ gates.

A function (family) is considered efficiently computable if it can be constructed with n^k gates for constant k.

Reversible Circuits

Reversible circuits were studied in the 1960s and 70s to reduce energy dissipation in circuits.

In a reversible circuit every gate has to be invertible.

In particular, the number of inputs equals the number of outputs.

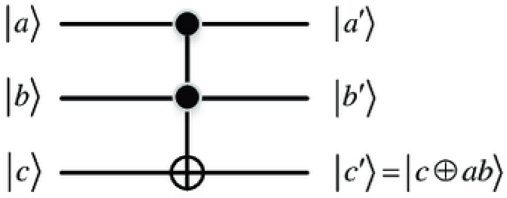
AND and OR are not invertible, so we need to find new basic gates.

Very nice replacement is the Toffoli Gate.

Toffoli Gate

The Toffoli gate or controlled controlled NOT (**CCNOT**) gate acts on 3 bits and has the following behavior: if the first two bits are one then negate the third bit.

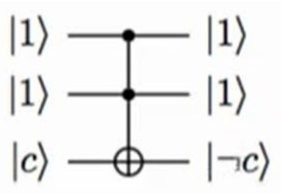
input	output
000	0
001	1
010	0
011	1
100	0
101	1
110	1
111	0



Ancilla bits

For reversible circuits we allow “extra” wires that carry hard-coded bits.

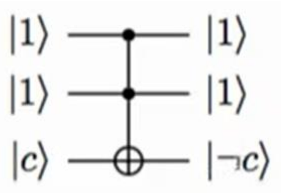
Using ancilla bits we see that Toffoli gate can compute NOT.



We hard-code the first two inputs to be one.

Garbage bits

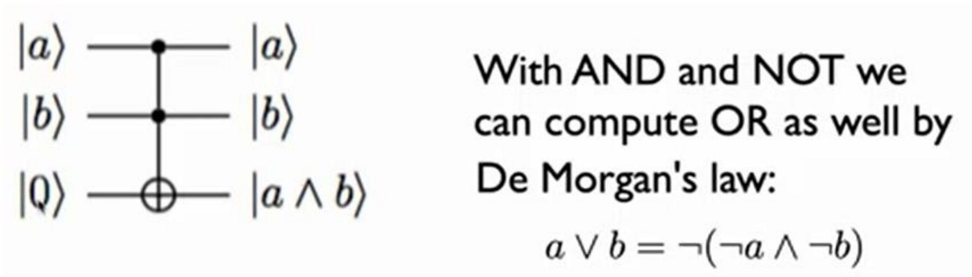
For reversible circuits we will also (potentially) only be interested in some of the outputs.



We read some of the output bits for our answer. The rest are “garbage” bits.

Toffoli Gate

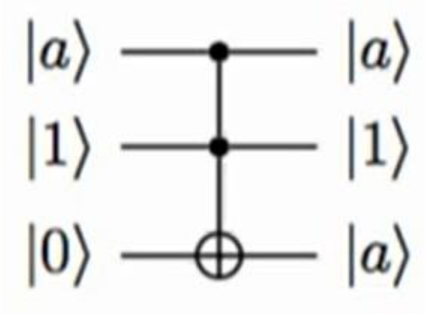
Allowing ancilla bits and garbage bits, there is a reversible circuit using only Toffoli gates that simulates any AND, OR, NOT and DUP circuit.



With AND and NOT we can compute OR as well by De Morgan's law:

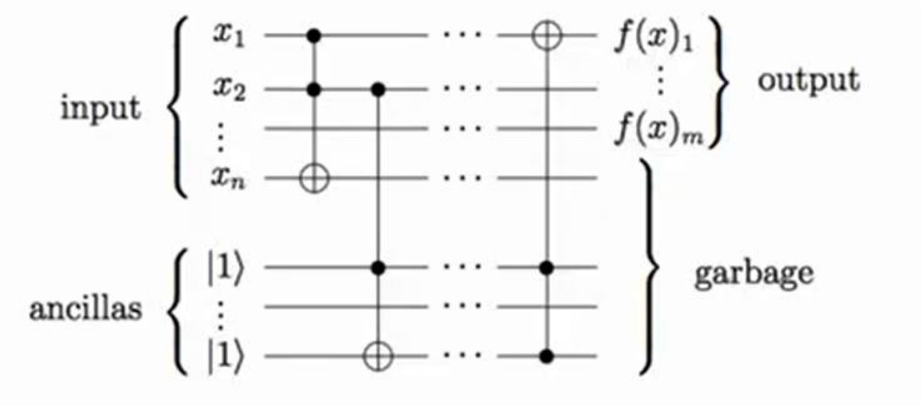
$$a \vee b = \neg(\neg a \wedge \neg b)$$

Finally, we can also simulate DUP gates.



Classical Reversible Circuit

A classical reversible circuit thus looks as follows:



Reading a circuit

- Input is in input registers $1, \dots, n$
- Ancillas in input registers $n+1, \dots, n+c$ initialized to $|1\rangle$
- Read output in output registers $1, \dots, m$

The reading from left to right we obtain a sequence of operations:

$$x_n \leftarrow CCNOT(x_1, x_2, x_n)$$

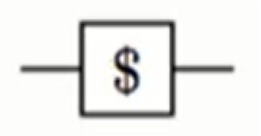
$$x_{n+c} \leftarrow CCNOT(x_2, x_{n+1}, x_{n+c})$$

$$x_1 \leftarrow CCNOT(x_{n+1}, x_{n+c}, x_1)$$

Randomized circuits

Now we discuss adding randomization to our circuits.

We augment the circuit with a “coin toss” gate that outputs zero or one uniformly at random.

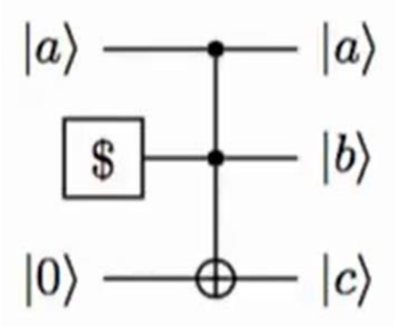


We can push all coin tosses to the very beginning of the algorithm.

We thus image a circuit that has n input wires, c ancilla wires and r coin toss wires.

Let's play with randomized circuits to build up to analyzing quantum circuits.

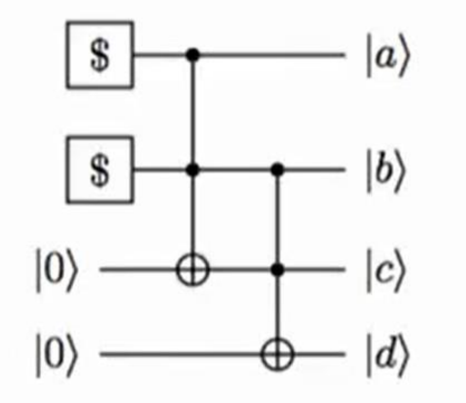
What is the output $|c\rangle$ of this circuit?

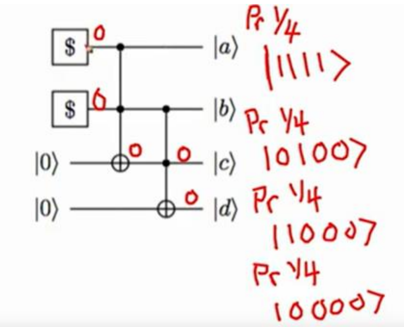


If $|a\rangle = |0\rangle$ then $|c\rangle = |0\rangle$

If $|a\rangle = |1\rangle$ then $|c\rangle = |0\rangle$ with probability 0.5 and $|c\rangle = |1\rangle$ with probability 0.5.

What is the probability distribution over outputs in this circuit?





input	probability	output
$|0000\rangle$	0.25	$|0000\rangle$
$|0100\rangle$	0.25	$|0100\rangle$
$|1000\rangle$	0.25	$|1000\rangle$
$|1100\rangle$	0.25	$|1111\rangle$

We can describe the output as the "superposition":

$$0.25 \cdot |0000\rangle + 0.25 \cdot |0100\rangle + 0.25 \cdot |1000\rangle + 0.25 \cdot |1111\rangle$$

Quantum Circuits

What should we choose as elementary gates for quantum circuits?

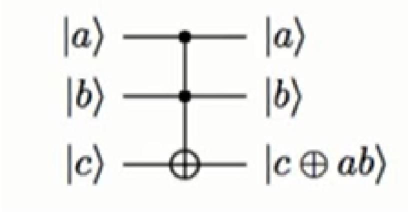
Just for one qubit gates there are already continuously many possibilities.

Desirable properties:

- Finite gate set.
- Generalizes classical reversible and randomized circuits.
- "Universality": any unitary can be approximated by a large enough circuit.

Gate Set

To generalize classical reversible computation it is natural to include the CCNOT gate.



To generalize randomized circuits, we also need to simulate a coin toss. How can we do this?

Hadamard Gate

The Hadamard gate is nothing more than a 2×2 Discrete Fourier Transform matrix (two-point DFT). That is the reason that $\boldsymbol{H}\left(\frac{|0\rangle+|1\rangle}{\mathbf{2}}\right) = |0\rangle$ and $\boldsymbol{H}\left(\frac{|0\rangle-|1\rangle}{\mathbf{2}}\right) = |1\rangle$, this is "periodicity".

$\boldsymbol{H} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ → two-point DFT

$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ → these are the standard euclidean basis.

So natural we take them as the computational basis for quantum computation:

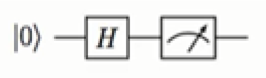
$$\boldsymbol{H}|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \qquad \boldsymbol{H}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

The equation for Hadamard gate transformation on multiple qubits is:

$$\boldsymbol{H}^{\otimes n}|x_1, \dots, x_n\rangle = \frac{\sum_{z_1,\dots,z_n}(-1)^{x_1 z_1 + \cdots + x_n z_n}|z_1, \dots, z_n\rangle}{\sqrt{2^n}} = \frac{1}{\sqrt{2^n}}\sum_{z=0}^{2^n-1}(-1)^{x_n \cdot z}|z\rangle$$

This returns a uniformly random bit, i.e., simulates a coin toss.

Deferred measurement



This returns a uniformly random bit, i.e. simulates a coin gate.

One drawback to this simulation is that it requires measurement in the middle of the computation.

On the problem set you will show how to simulate randomized circuits even deferring all measurements to the end.

Universality

Hadamard and Toffoli gates can efficiently approximate a quantum circuit on any other gate set.

Gate: Unitary on < 4 qubits.

Efficiently: $O(m \cdot polylog(n, m, \frac{1}{\varepsilon}))$ Hadamards and Toffolis to approximate a circuit on n qubits with m gates to error ε.

A relaxed notion of approximation is used here where extra ancillas are allowed.

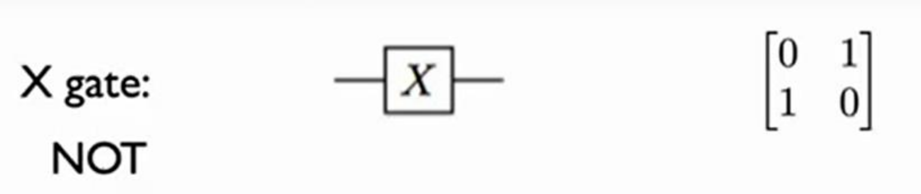
We have a circuit U and n qubits and we want to approximate the circuit U to $\tilde{\mathrm{U}}$ also on n qubits. Applying the operator norm or spectral norm we have the expression:

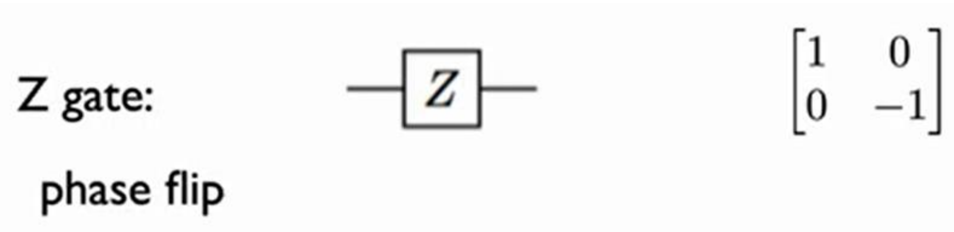
$$\|\mathrm{U} - \mathrm{U}\| \leq \varepsilon$$

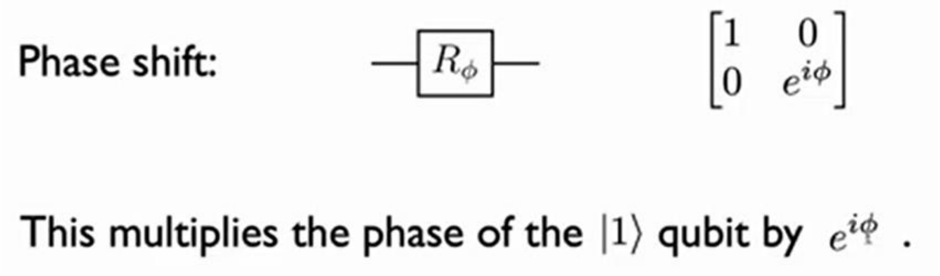
If there is an ancilla state $|\psi\rangle$ such that for every unit vector $|\phi\rangle$:

$$\left\|\tilde{\mathrm{U}}(|\phi\rangle \otimes |\psi\rangle) - \mathrm{U}(|\phi\rangle \otimes |\psi\rangle)\right\| \leq \varepsilon$$

One qubit gates







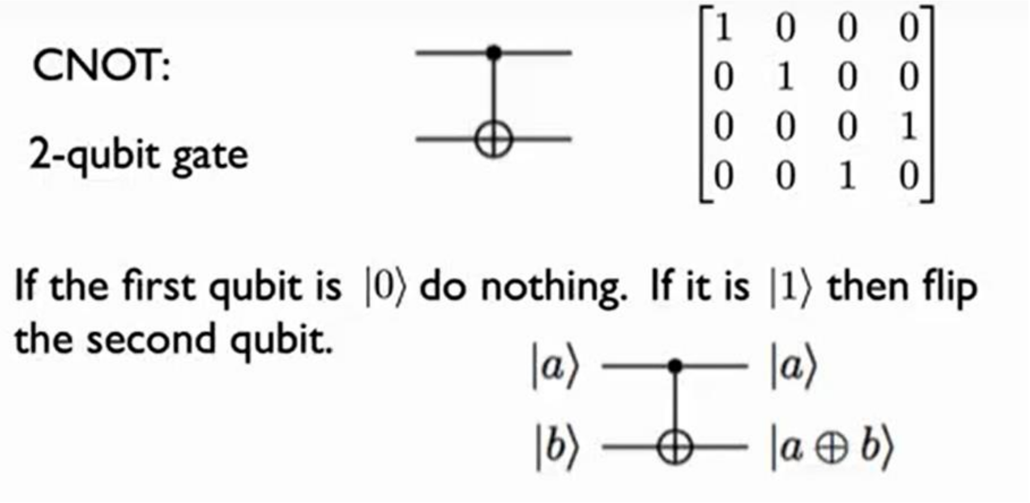
Phase shift:

$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{bmatrix}$$

This multiplies the phase of the $|1\rangle$ qubit by $e^{i\phi}$.

Of particular interest is the $T = \frac{R_\pi}{4}$ gate.

Two qubit gates



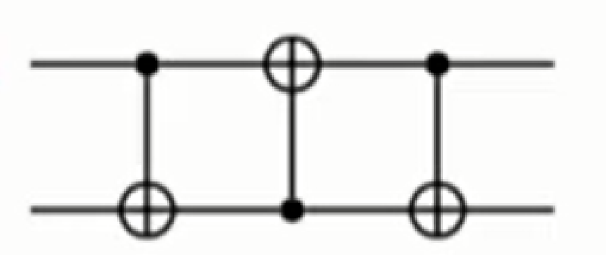
CNOT:

2-qubit gate

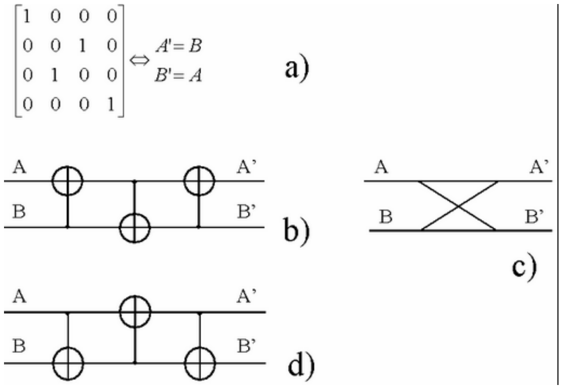
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

If the first qubit is $|0\rangle$ do nothing. If it is $|1\rangle$ then flip the second qubit.

Example Circuit



SWAP gate



2. Early Quantum Algorithms

In 1993: Simon's algorithm

In 1994: Shor's factoring algorithm

In 1996: Grover's search algorithm

Preliminaries

Action of the Hadamard transformation

Explicar Hadamard aqui

Oracle model (module oracle)

Quantum algorithms are often studied in a black-box circuit or oracle model. We want to use this black box as few number of times as possible in order to determine some property of the black box. Basically, the black box encodes some function from zero one to end to zero one.

$$f: \{0,1\}^n \rightarrow \{0,1\}$$

The behavior of black box is determined by the unitary map O_f where:

$$O_f|x\rangle|b\rangle = |x\rangle|b \oplus f(x)\rangle$$

where:

x	inbit string, $x \in \{0,1\}^n$
b	bit, b $\in \{0,1\}$
$|x\rangle|b\rangle$	x tensor b

Querying the oracle

We can see the potential of quantum algorithms by creating the uniform superposition and querying the oracle.

$$O_f\left(\boldsymbol{H}^{\otimes n}\otimes I\right)|0^n\rangle|0\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\,\in\{0,1\}^n}|x\rangle|f(x)\rangle$$

Phase kickback trick (phase oracle)

We now cover another useful trick you often see in quantum algorithms. Sometimes it is more convenient to compute the function f in the phase.

$$O_{f,\pm}|x\rangle|b\rangle = (-1)^{b\cdot f(x)}|x\rangle|b\rangle$$

We can easily implement this using O_f. (agregar la demostración)

3. Simon Algorithm

Oracle

In Simon's algorithm are given **oracle** access to a function $f\colon\{0,1\}^n \rightarrow \{0,1\}$.

$$O_f|x\rangle|z\rangle = |x\rangle|z \oplus f(x)\rangle$$

where:

$\otimes$	bitwise XOR (not carrying bit)
	$z \oplus f(x) \in \{0,1\}^n$
$\lvert x\rangle\lvert b\rangle$	x tensor b

$$(z \oplus f(x))_i = z_i \oplus f(x)_i$$

Now the promise on f is that there is an s $(n\ bit\ string) \in \{0,1\}^n$, not all zero, such that $f(x) = f(y)$ if and only if either:

1) $x = y$
2) $x = y \oplus s$

Thus for every value z in the range of f there are exactly two distinct strings x, y such that $z = f(x) = f(y)$

The goal is to determine the hidden string s.

3.1.Example $\boldsymbol{n = 3\ bits, s = 011}$

domain: x	range: $f(x)$
000	red
001	blue
010	blue
011	red
100	green
101	black
110	black

111	green

How could you solve this problem classically?

If you find $x \neq y$ with $f(x) = f(y)$ then $s = x \oplus y$

$$x = y \oplus s$$

$$y \oplus x = y \oplus s \oplus y$$

$$y \oplus x = 0 \oplus s$$

$$x \oplus y = s$$

A probabilistic argument (birthday problem – obtener más información de los enlaces que tengo registrado). shows such a pair exists with constant probability in a random set of size $O(2^{n/2})$ (debo agregar información del curso de Python)

You can also show that any successful randomized algorithm must query the oracle $\Omega(2^{n/2})$ times.

There is also a cute $O(2^{n/2})$ deterministic algorithm.

To find the most significant there is a query $f(x)$ for all $x = 0 \ldots 0x_{n/2-1} \ldots x_1x_0$, $2^{n/2}$

To find the least significant there is a query $f(x)$ for all $x = x_{n-1} \ldots x_{n/2}0 \ldots 0$, $2^{n/2}$

Total of $2^{n/2+1}$ many queries. Como?

Pair XORs to secret

Now consider the secret $s = s_{n-1} \ldots s_1s_0$

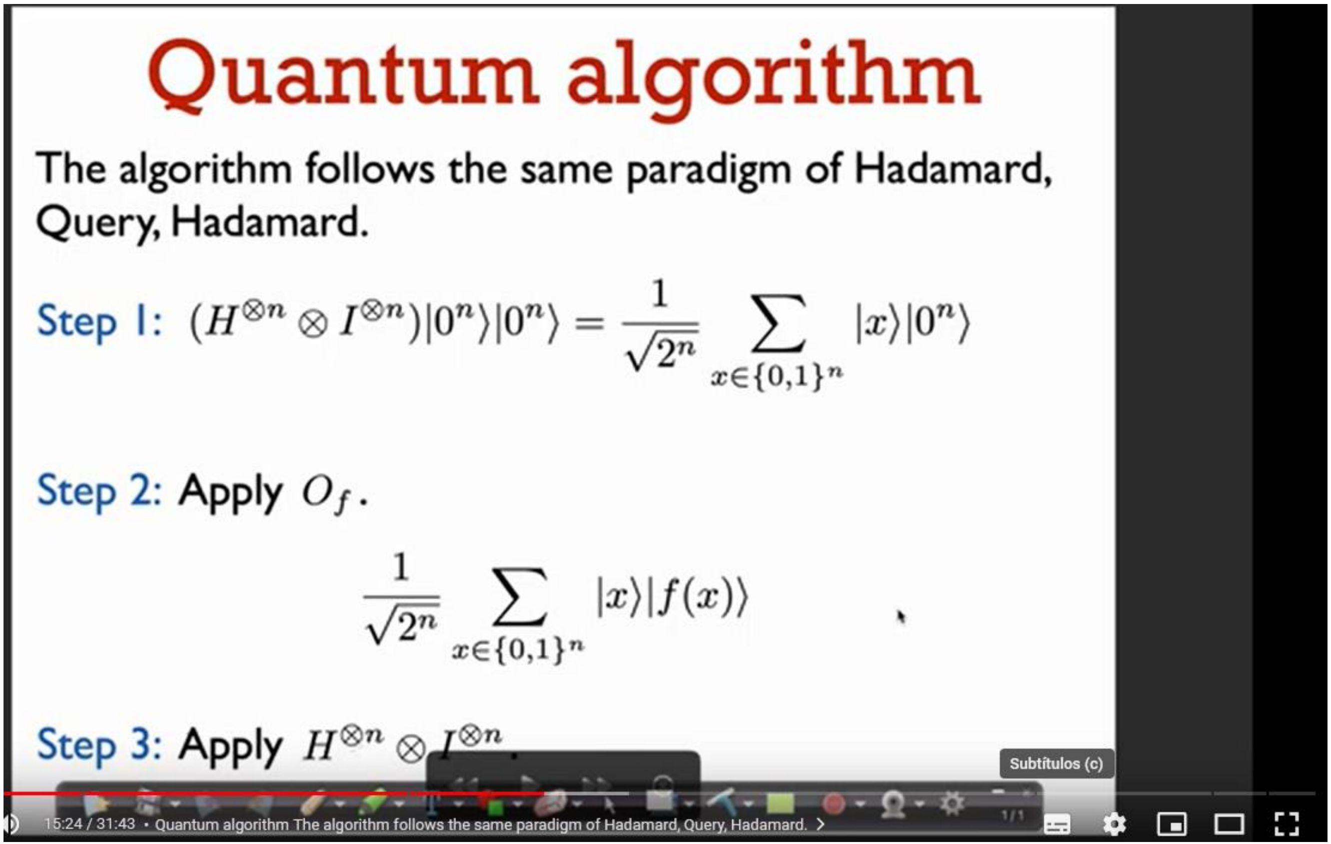
We queried $l = 0 \ldots 0s_{n/2-1} \ldots s_1s_0$

We also queried $h = s_{n-1} \ldots s_{n/2}0 \ldots 0$

$h \neq l$

and $h \oplus l = s$ thus $f(h) = f(l)$

This means that amongst the queries we will find a pair x, y with $f(x) = f(y)$ and can determine the secret s.



Quantum algorithm

The algorithm follows the same paradigm of Hadamard, Query, Hadamard.

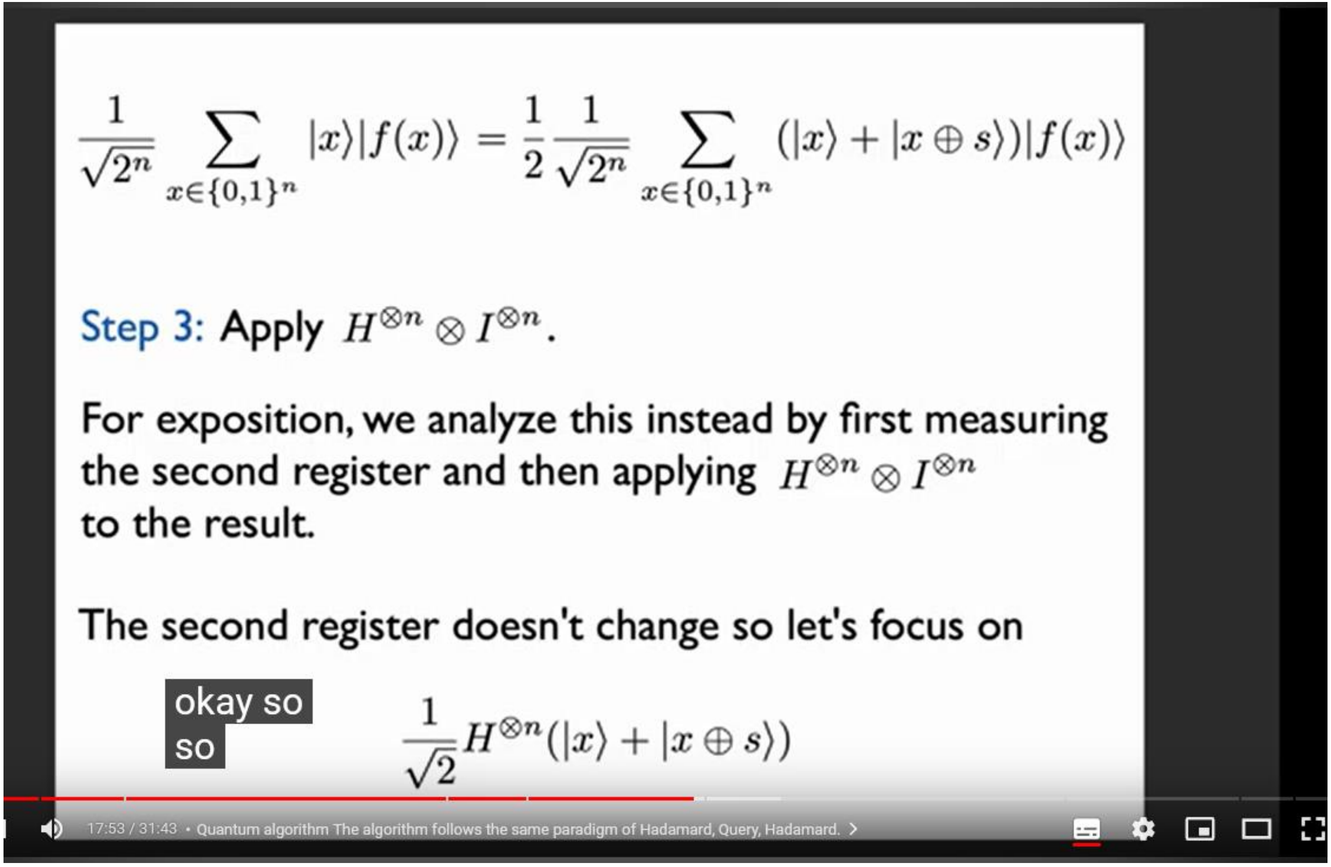
Step 1: $(H^{\otimes n} \otimes I^{\otimes n})|0^n\rangle|0^n\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle|0^n\rangle$

Step 2: Apply O_f.

$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle|f(x)\rangle$$

Step 3: Apply $H^{\otimes n} \otimes I^{\otimes n}$.

Subtítulos (c)
1/1
15:24 / 31:43 • Quantum algorithm The algorithm follows the same paradigm of Hadamard, Query, Hadamard. >



$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle|f(x)\rangle = \frac{1}{2}\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (|x\rangle + |x \oplus s\rangle)|f(x)\rangle$$

Step 3: Apply $H^{\otimes n} \otimes I^{\otimes n}$.

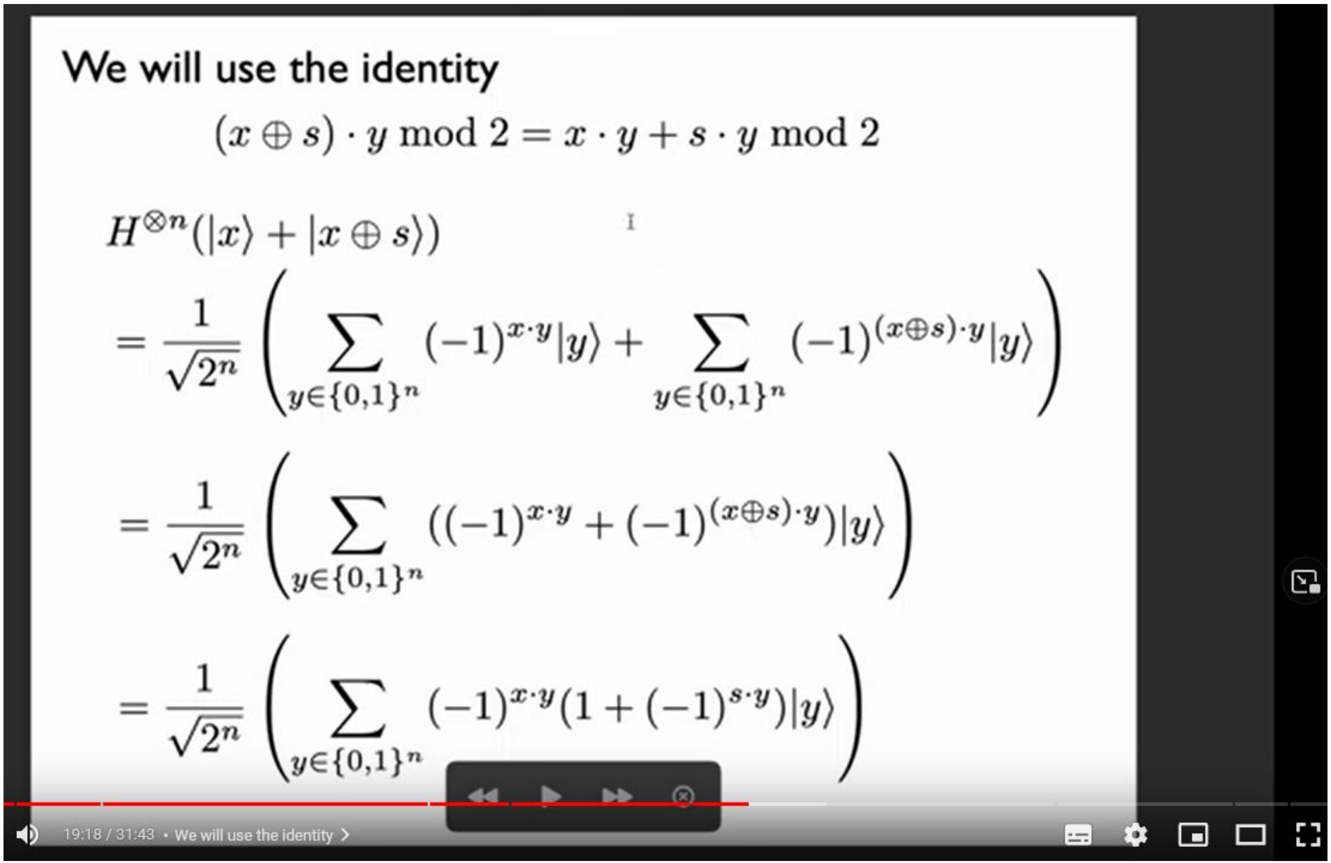
For exposition, we analyze this instead by first measuring the second register and then applying $H^{\otimes n} \otimes I^{\otimes n}$ to the result.

The second register doesn't change so let's focus on

$$\frac{1}{\sqrt{2}} H^{\otimes n}(|x\rangle + |x \oplus s\rangle)$$

okay so
so

17:53 / 31:43 • Quantum algorithm The algorithm follows the same paradigm of Hadamard, Query, Hadamard. >



We will use the identity

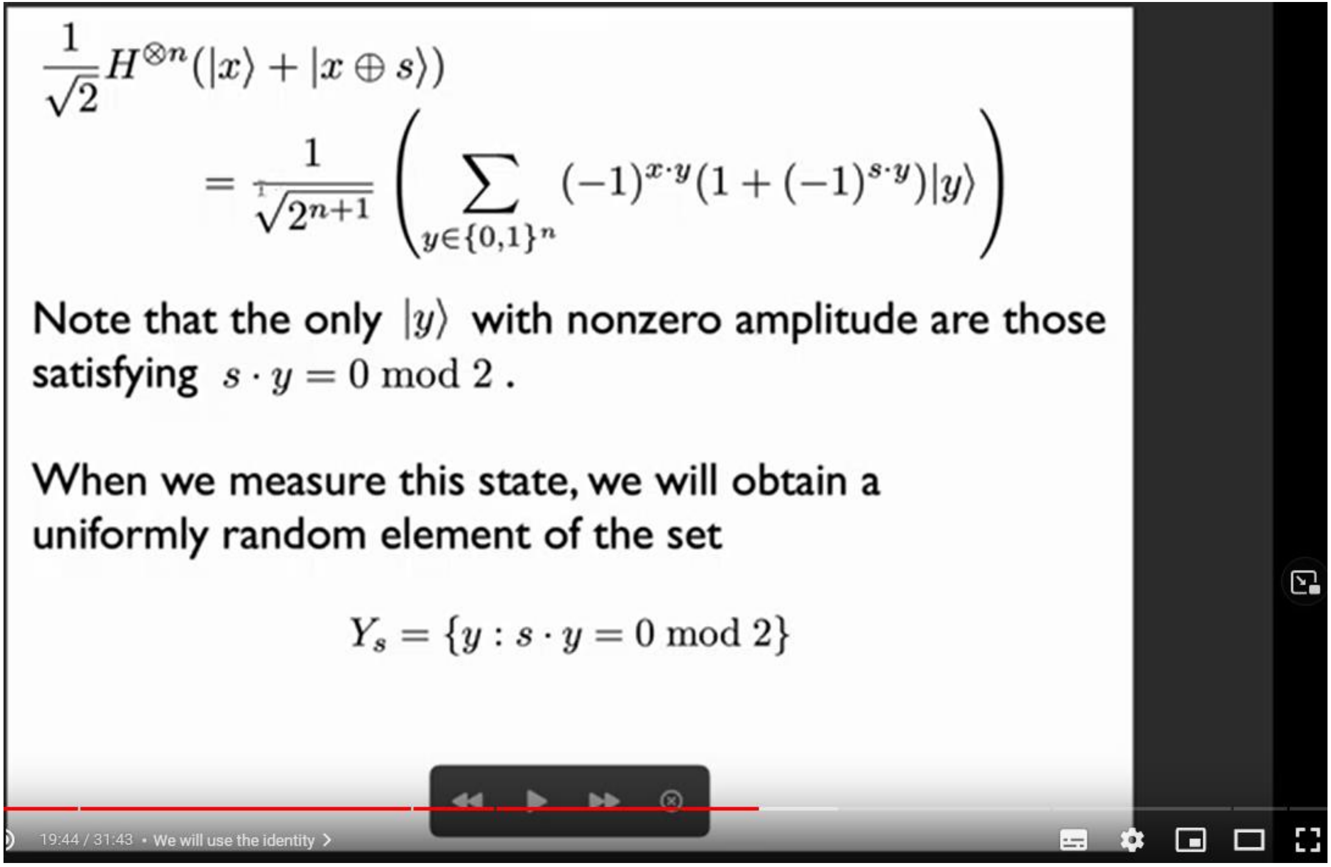
$$(x \oplus s) \cdot y \bmod 2 = x \cdot y + s \cdot y \bmod 2$$

$$H^{\otimes n}(|x\rangle + |x \oplus s\rangle)$$

$$= \frac{1}{\sqrt{2^n}} \left(\sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle + \sum_{y \in \{0,1\}^n} (-1)^{(x \oplus s) \cdot y} |y\rangle \right)$$

$$= \frac{1}{\sqrt{2^n}} \left(\sum_{y \in \{0,1\}^n} ((-1)^{x \cdot y} + (-1)^{(x \oplus s) \cdot y}) |y\rangle \right)$$

$$= \frac{1}{\sqrt{2^n}} \left(\sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} (1 + (-1)^{s \cdot y}) |y\rangle \right)$$



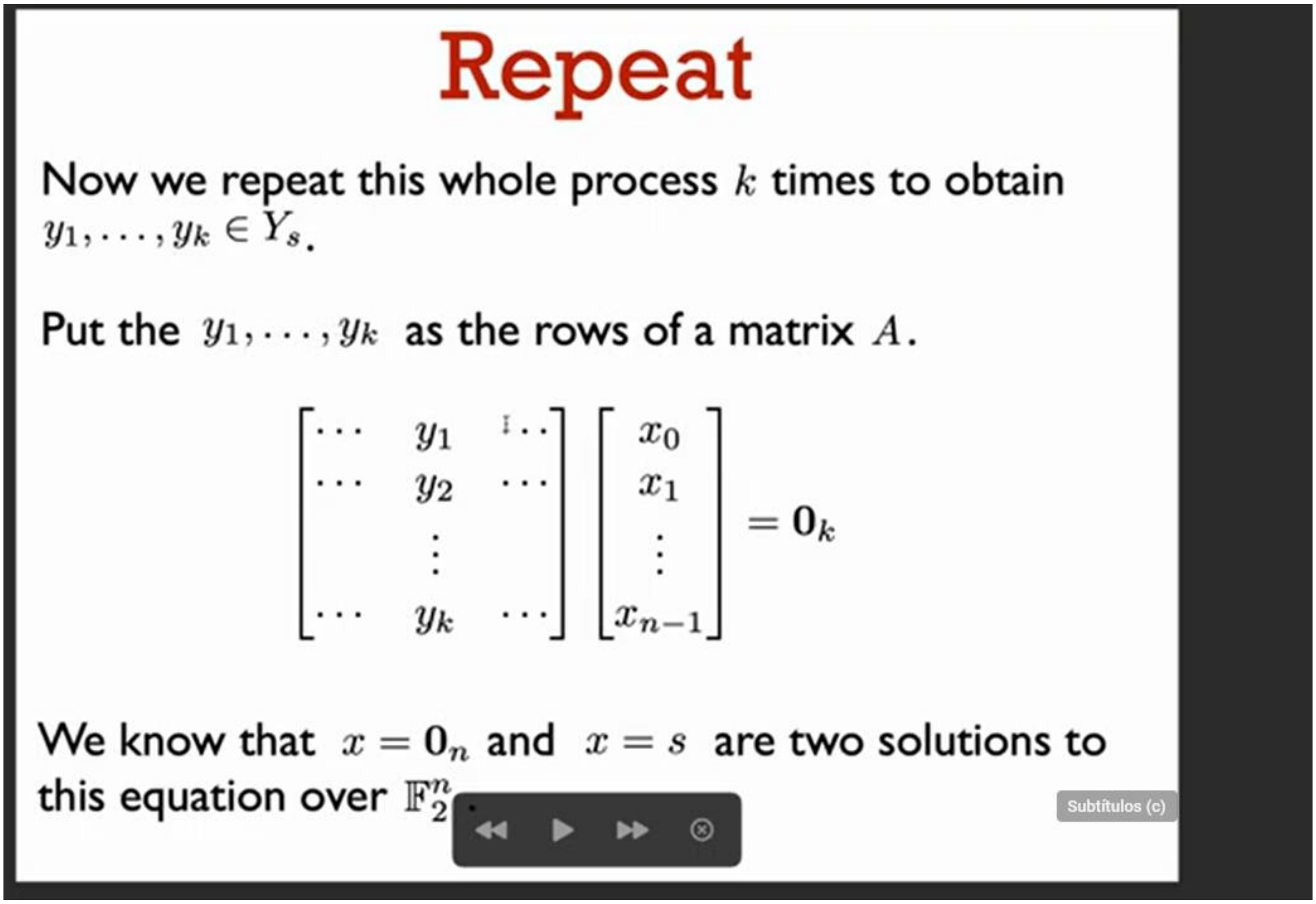
$$\frac{1}{\sqrt{2}} H^{\otimes n}(|x\rangle + |x \oplus s\rangle)$$

$$= \frac{1}{\sqrt{2^{n+1}}} \left(\sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} (1 + (-1)^{s \cdot y}) |y\rangle \right)$$

Note that the only $|y\rangle$ with nonzero amplitude are those satisfying $s \cdot y = 0 \bmod 2$.

When we measure this state, we will obtain a uniformly random element of the set

$$Y_s = \{y : s \cdot y = 0 \bmod 2\}$$



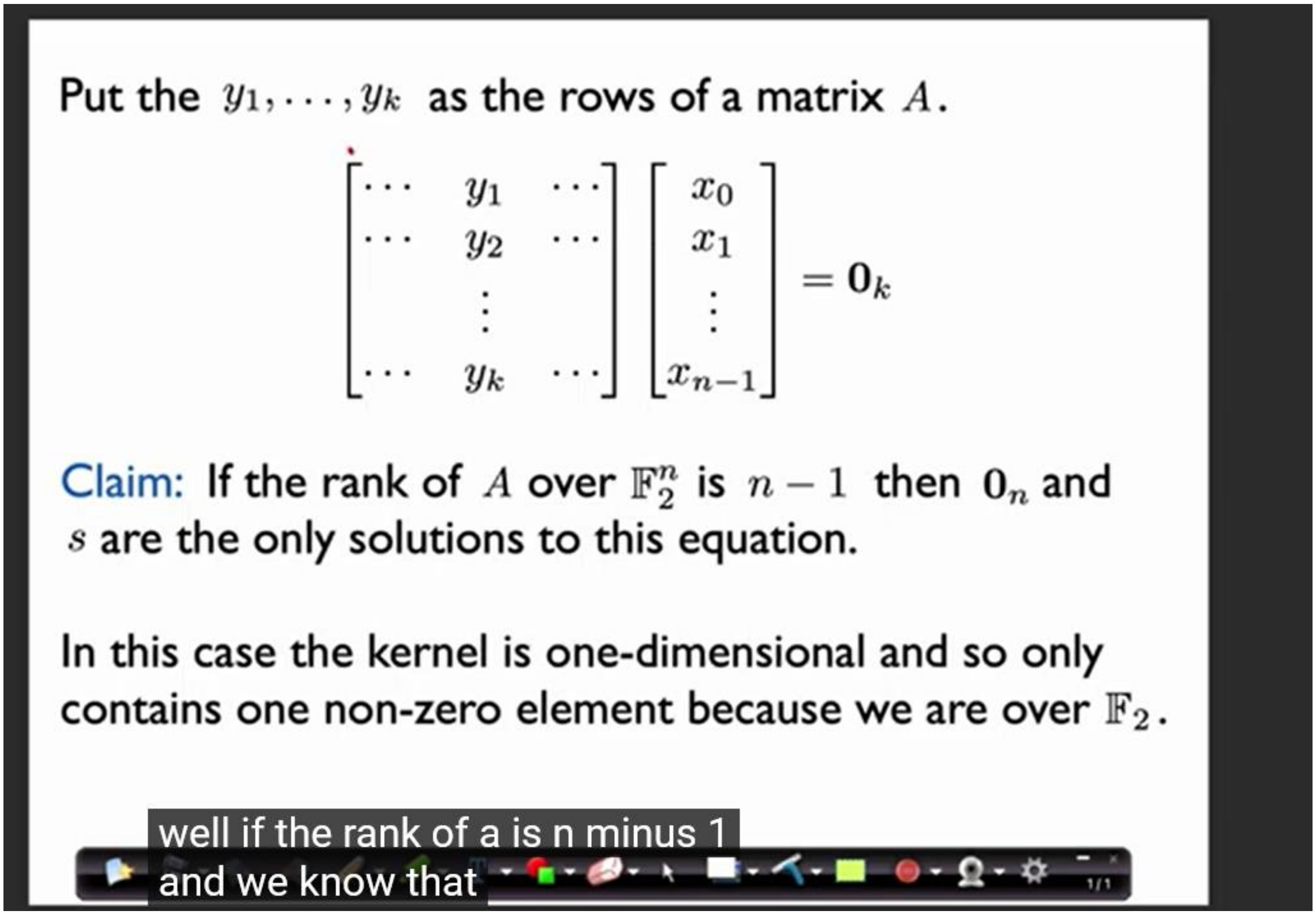
Repeat

Now we repeat this whole process k times to obtain $y_1, \ldots, y_k \in Y_s$.

Put the $y_1, \ldots, y_k$ as the rows of a matrix A.

$$\begin{bmatrix} \cdots & y_1 & \cdots \\ \cdots & y_2 & \cdots \\ & \vdots & \\ \cdots & y_k & \cdots \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{bmatrix} = \mathbf{0}_k$$

We know that $x = \mathbf{0}_n$ and $x = s$ are two solutions to this equation over $\mathbb{F}_2^n$.



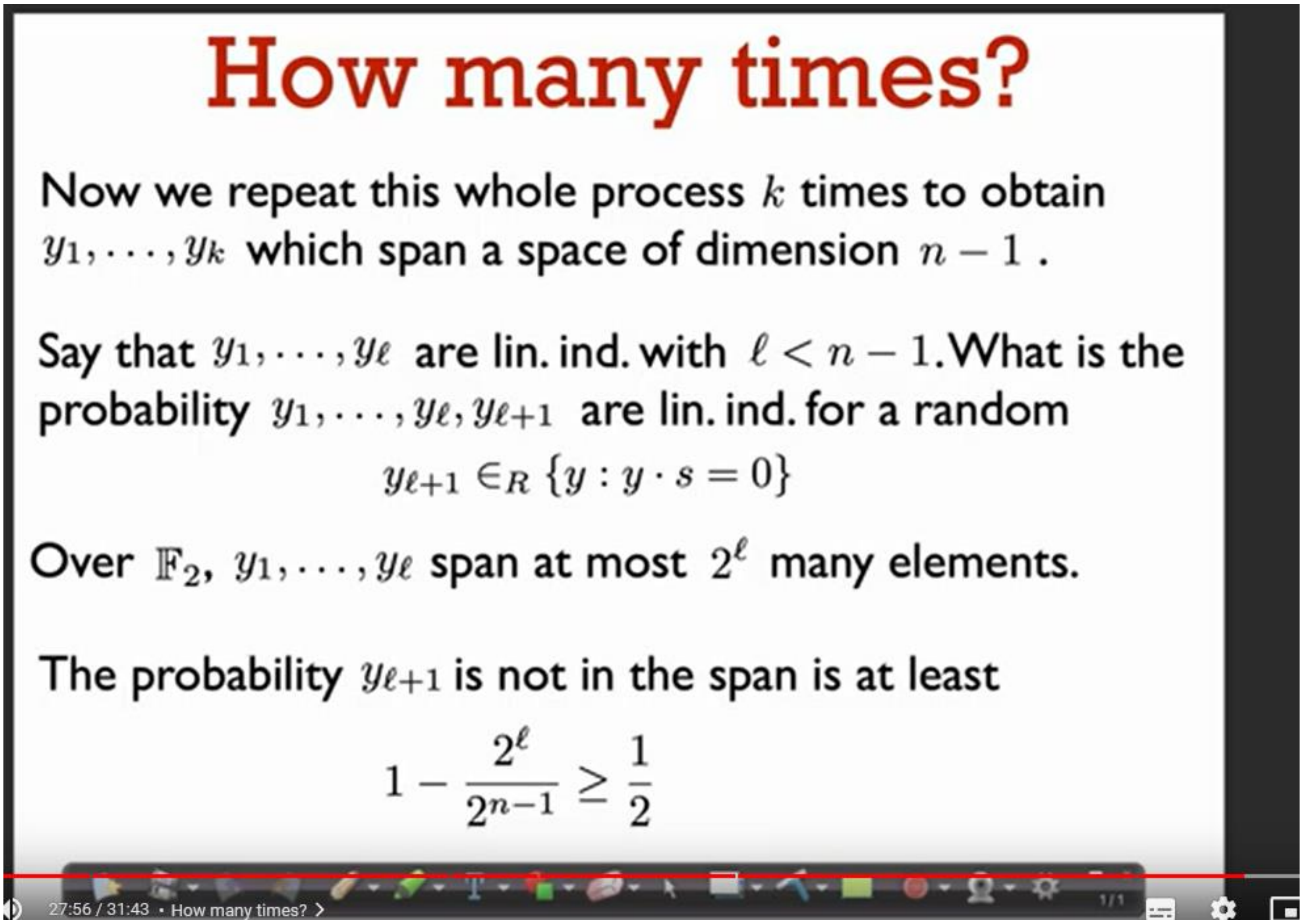
Put the $y_1, \dots, y_k$ as the rows of a matrix A.

$$\begin{bmatrix} \cdots & y_1 & \cdots \\ \cdots & y_2 & \cdots \\ & \vdots & \\ \cdots & y_k & \cdots \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{bmatrix} = \mathbf{0}_k$$

Claim: If the rank of A over $\mathbb{F}_2^n$ is $n-1$ then $\mathbf{0}_n$ and s are the only solutions to this equation.

In this case the kernel is one-dimensional and so only contains one non-zero element because we are over $\mathbb{F}_2$.

well if the rank of a is n minus 1
and we know that



How many times?

Now we repeat this whole process k times to obtain $y_1, \cdots, y_k$ which span a space of dimension $n-1$.

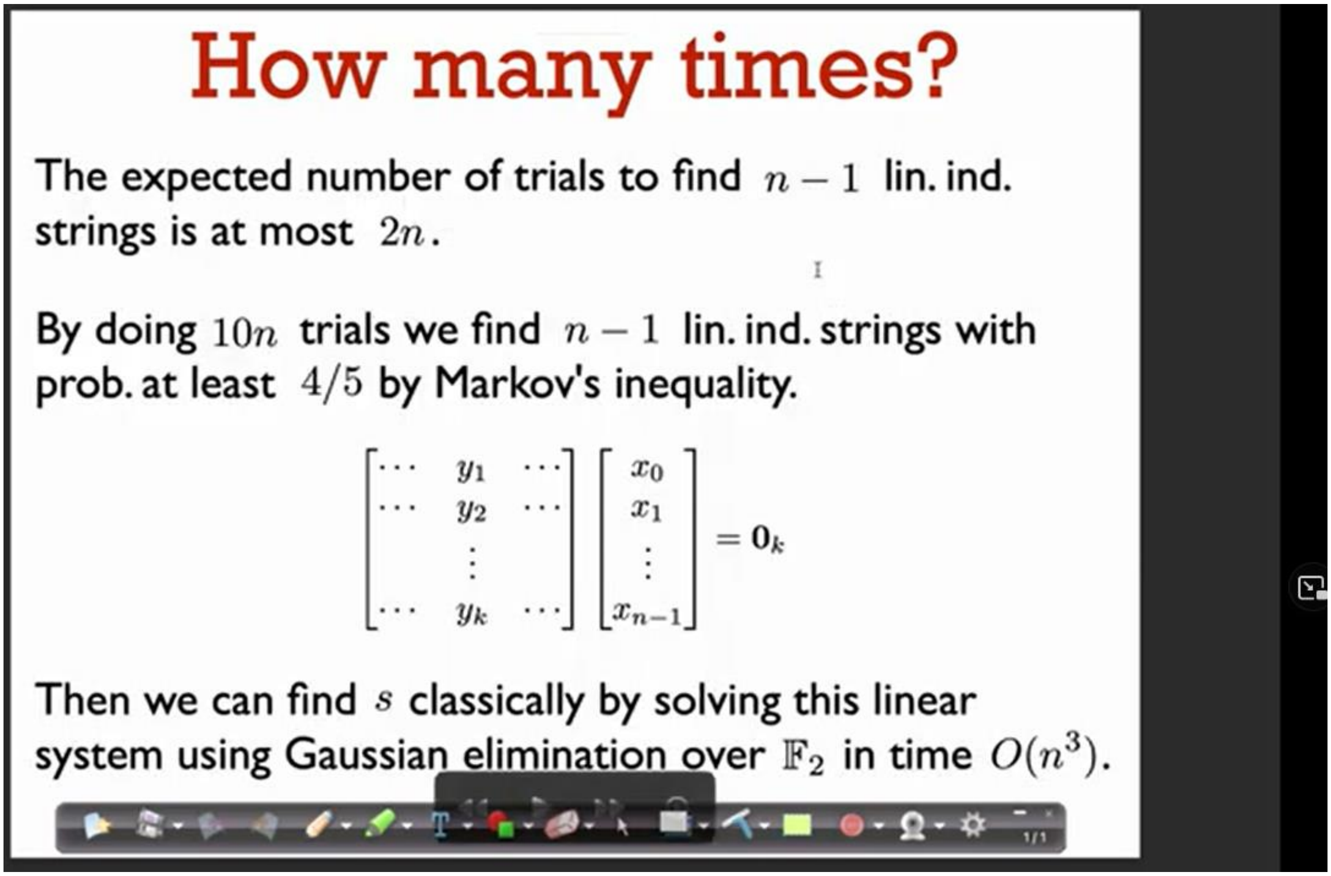
Say that $y_1, \cdots, y_\ell$ are lin. ind. with $\ell < n-1$.What is the probability $y_1, \cdots, y_\ell, y_{\ell+1}$ are lin. ind. for a random

$$y_{\ell+1} \in_R \{y : y \cdot s = 0\}$$

Over $\mathbb{F}_2$, $y_1, \cdots, y_\ell$ span at most 2^ℓ many elements.

The probability $y_{\ell+1}$ is not in the span is at least

$$1 - \frac{2^\ell}{2^{n-1}} \geq \frac{1}{2}$$



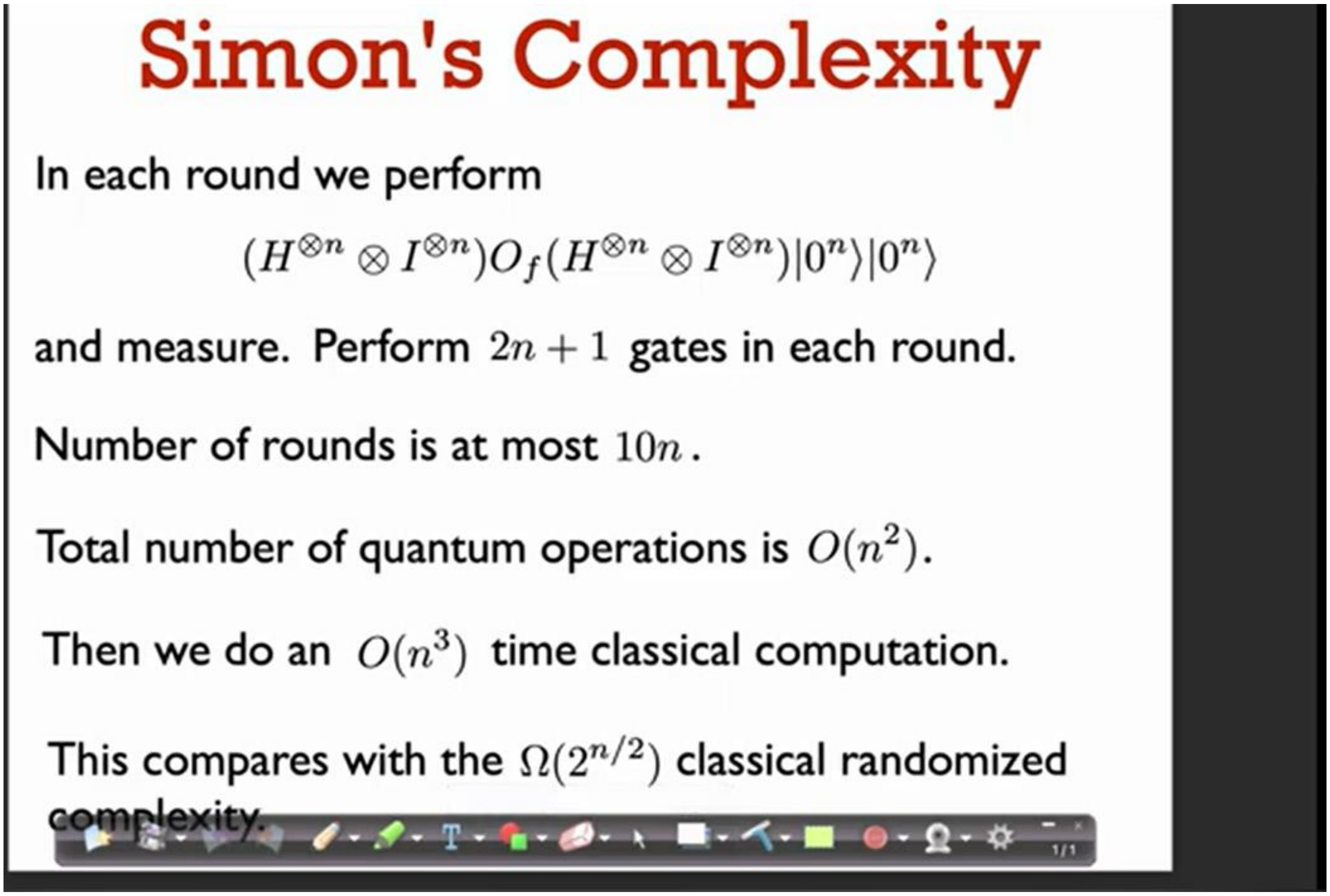
How many times?

The expected number of trials to find $n-1$ lin. ind. strings is at most $2n$.

By doing $10n$ trials we find $n-1$ lin. ind. strings with prob. at least $4/5$ by Markov's inequality.

$$\begin{bmatrix} \cdots & y_1 & \cdots \\ \cdots & y_2 & \cdots \\ & \vdots & \\ \cdots & y_k & \cdots \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{bmatrix} = \mathbf{0}_k$$

Then we can find s classically by solving this linear system using Gaussian elimination over $\mathbb{F}_2$ in time $O(n^3)$.



Simon's Complexity

In each round we perform

$$(H^{\otimes n} \otimes I^{\otimes n})O_f(H^{\otimes n} \otimes I^{\otimes n})|0^n\rangle|0^n\rangle$$

and measure. Perform $2n+1$ gates in each round.

Number of rounds is at most $10n$.

Total number of quantum operations is $O(n^2)$.

Then we do an $O(n^3)$ time classical computation.

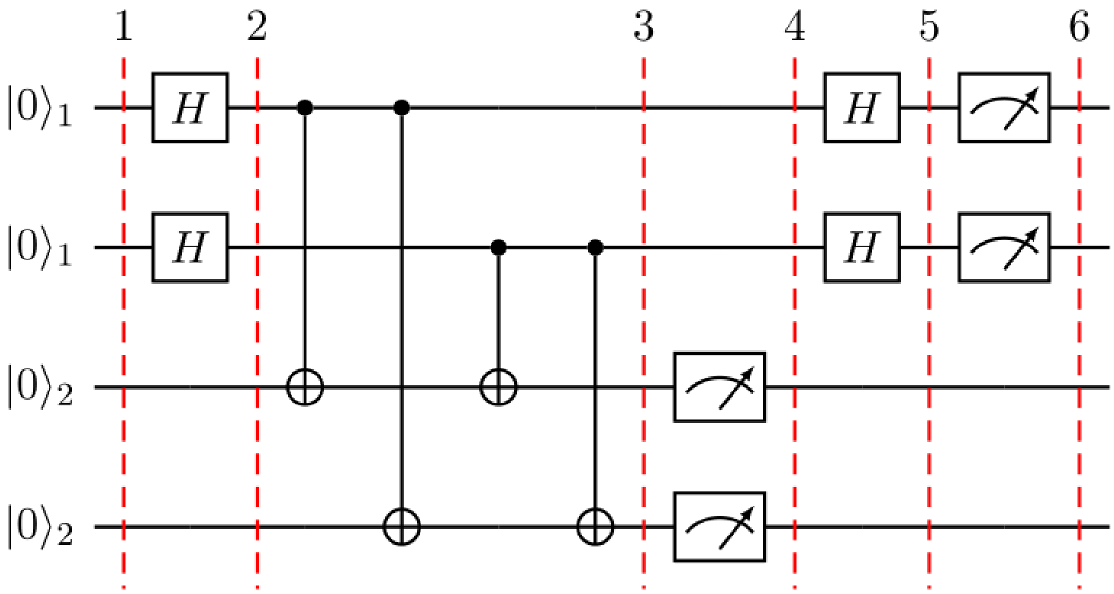
This compares with the $\Omega(2^{n/2})$ classical randomized complexity.

3.2.Example $\boldsymbol{n = 2\ bits, c = 11}$

- **Enunciado del ejercicio**

$n = 2$

- **Diagrama cuántico del ejercicio**



Se debe corregir los estados.

- **Solución del problema**

$n = 2$

- **First status**

$|\varphi_0\rangle = |0\rangle_A{}^{\otimes n}\ |0\rangle_B{}^{\otimes n}$

$|\varphi_0\rangle = |00\rangle|00\rangle$

- **Second status**

Aquí debo agregar un enlace a otra pestaña de la presentación indicando que Hadamard gate lo que da es una lista de la superposición de los estados de un qubit.

$$\boxed{|\varphi_1\rangle = \boldsymbol{H}^{\otimes n}|0\rangle_A{}^{\otimes n}\ |0\rangle_B{}^{\otimes n} = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle_A = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle_A = \frac{1}{\sqrt{2^2}}\sum_{x=0}^{3}|x\rangle_A}$$

$$|\varphi_1\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)_A$$

- **Third status**

$$|\varphi_2\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle_1 \cdot |y \otimes f(x)\rangle_2 = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle \cdot |0 \oplus f(x)\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle \cdot |f(x)\rangle$$

$$|\varphi_2\rangle = \frac{1}{\sqrt{2^2}}\{|00\rangle \cdot f(00) + |01\rangle \cdot f(01) + |10\rangle \cdot f(10) + |11\rangle \cdot f(11)\}$$

$$|\varphi_2\rangle = \frac{1}{\sqrt{2^2}}\{|00\rangle \cdot |01\rangle + |01\rangle \cdot |10\rangle + |10\rangle \cdot |10\rangle + |11\rangle \cdot |01\rangle\}$$

- **Fourth status**

$$|\varphi_3\rangle = \frac{1}{\sqrt{2}}(|x\rangle + |x \oplus a\rangle)$$

- **Fifth status**

$$|\varphi_4\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}\sum_{z=0}^{2^n-1}(-1)^{x_n \cdot z_n}\,|z\rangle|f(x)\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}\sum_{z=0}^{2^n-1}(-1)^{x_1 \cdot z_1 + x_2 \cdot z_2}\,|z\rangle|f(x)\rangle$$

$$|\varphi_4\rangle = \frac{1}{\sqrt{2^3}}\left\{|\varphi_4\rangle_{state\ 1} + |\varphi_4\rangle_{state\ 2} + |\varphi_4\rangle_{state\ 3} + |\varphi_4\rangle_{state\ 4}\right\}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 1} = {} & [(-1)^{0\cdot0+0\cdot0}|00\rangle + (-1)^{0\cdot0+1\cdot0}|01\rangle + (-1)^{1\cdot0+0\cdot0}|10\rangle + (-1)^{1\cdot0+1\cdot0}|11\rangle] \\ & \otimes |f(00)\rangle = [|00\rangle + |01\rangle + |10\rangle + |11\rangle] \otimes |01\rangle\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 2} = {} & [(-1)^{0\cdot0+0\cdot1}|00\rangle + (-1)^{0\cdot0+1\cdot1}|01\rangle + (-1)^{1\cdot0+0\cdot1}|10\rangle + (-1)^{1\cdot0+1\cdot1}|11\rangle] \\ & \otimes |f(01)\rangle = [|00\rangle - |01\rangle + |10\rangle - |11\rangle] \otimes |10\rangle\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 3} = {} & [(-1)^{0\cdot1+0\cdot0}|00\rangle + (-1)^{0\cdot1+1\cdot0}|01\rangle + (-1)^{1\cdot1+0\cdot0}|10\rangle + (-1)^{1\cdot1+1\cdot0}|11\rangle] \\ & \otimes |f(10)\rangle = [|00\rangle + |01\rangle - |10\rangle - |11\rangle] \otimes |10\rangle\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 4} = {} & [(-1)^{0\cdot1+0\cdot1}|00\rangle + (-1)^{0\cdot1+1\cdot1}|01\rangle + (-1)^{1\cdot1+0\cdot1}|10\rangle + (-1)^{1\cdot1+1\cdot1}|11\rangle] \\ & \otimes |f(11)\rangle = [|00\rangle - |01\rangle - |10\rangle + |11\rangle] \otimes |01\rangle\end{aligned}$$

$$|\varphi_4\rangle = \frac{1}{\sqrt{2^3}}\left\{|\varphi_4\rangle_{state\ 1} + |\varphi_4\rangle_{state\ 4} + |\varphi_4\rangle_{state\ 2} + |\varphi_4\rangle_{state\ 3}\right\}$$

$$\begin{aligned}|\varphi_4\rangle = \frac{1}{\sqrt{2^3}}\Big\{ & \big[[|00\rangle + |01\rangle + |10\rangle + |11\rangle] \otimes |01\rangle + [|00\rangle - |01\rangle - |10\rangle + |11\rangle] \otimes |01\rangle\big] \\ & + \big[[|00\rangle - |01\rangle + |10\rangle - |11\rangle] \otimes |10\rangle + [|00\rangle + |01\rangle - |10\rangle - |11\rangle] \otimes |10\rangle\big]\Big\}\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle = \frac{1}{\sqrt{2^3}}\{ & [|00\rangle + |11\rangle] \otimes |01\rangle + [|00\rangle + |11\rangle] \otimes |01\rangle + [|00\rangle - |11\rangle] \otimes |10\rangle \\ & + [|00\rangle - |11\rangle] \otimes |10\rangle\}\end{aligned}$$

$$|\varphi_4\rangle = \frac{1}{\sqrt{2^3}}\{2 \cdot |00\rangle \otimes [|01\rangle + |10\rangle] + 2 \cdot |11\rangle \otimes [|01\rangle - |10\rangle]\}$$

$\langle 00, c\rangle = 0$

Si escribimos $c = c_1c_2$ tenemos un sistema de ecuaciones:

1	$(0 \times c_1) \oplus (0 \times c_2) = 0$	Solución Trivial
2	$(1 \times c_1) \oplus (1 \times c_2) = 0$	$c_1 \oplus c_2 = 0 :\ c_1 = c_2 = 0, 1$

Tomando 2 y si $c_1 = c_2 = 0$

$(1 \times 0) \oplus (1 \times 0) = 0 : Solución\ trivial$

Tomando 2 y si $c_1 = c_2 = 0, c_2 = 1$

$(1 \times 1) \oplus (1 \times 1) = 0$

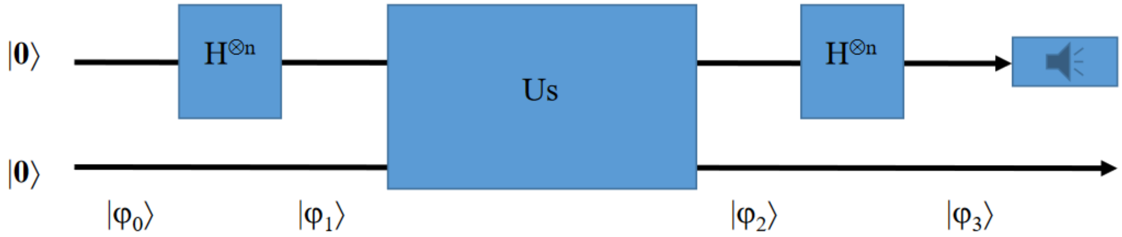
Por tanto: $c = c_1 c_2 = 11 \rightarrow período\ de\ la\ función$

3.3.Example $\boldsymbol{n = 3\ bits, c = 101}$

- **Enunciado del ejercicio**

$n = 3$

- **Diagrama cuántico del ejercicio**



Se debe agregar más estados en el gráfico.

- **Solución del problema**

$n = 3\ bits$

- **First status**

$|\varphi_0\rangle = |0\rangle_A{}^{\otimes n}\, |0\rangle_B{}^{\otimes n}$

$|\varphi_0\rangle = |000\rangle|000\rangle$

- **Second status**

Aquí debo agregar un enlace a otra pestaña de la presentación indicando que Hadamard gate lo que da es una lista de la superposición de los estados de un qubit.

$$\boxed{|\varphi_1\rangle = \boldsymbol{H}^{\otimes n}|0\rangle_A{}^{\otimes n}\, |0\rangle_B{}^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle_A = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle_A = \frac{1}{\sqrt{2^3}} \sum_{x=0}^{7} |x\rangle_A}$$

$$|\varphi_1\rangle = \frac{1}{\sqrt{2^3}}(|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)_A$$

- **Third status**

$$|\varphi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle_1 \cdot |y \otimes f(x)\rangle_2 = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \cdot |0 \oplus f(x)\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle \cdot |f(x)\rangle$$

$$\begin{aligned} |\varphi_2\rangle = \frac{1}{\sqrt{2^3}} \{ & |000\rangle \cdot f(000) + |001\rangle \cdot f(001) + |010\rangle \cdot f(010) + |011\rangle \cdot f(011) \\ & + |100\rangle \cdot f(100) + |101\rangle \cdot f(101) + |110\rangle \cdot f(110) + |111\rangle \cdot f(111)\} \end{aligned}$$

$$\begin{aligned} |\varphi_2\rangle = \frac{1}{\sqrt{8}} \{ & |000\rangle \cdot |100\rangle + |001\rangle \cdot |001\rangle + |010\rangle \cdot |101\rangle + |011\rangle \cdot |111\rangle + |100\rangle \cdot |001\rangle \\ & + |101\rangle \cdot |100\rangle + |110\rangle \cdot |111\rangle + |111\rangle \cdot |101\rangle\} \end{aligned}$$

- **Fourth status**

$$|\varphi_3\rangle = \frac{1}{\sqrt{2}} (|x\rangle + |x \oplus a\rangle)$$

- **Fifth status**

$$|\varphi_4\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} \sum_{z=0}^{2^n-1} (-1)^{x_n \cdot z_n} |z\rangle |f(x)\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} \sum_{z=0}^{2^n-1} (-1)^{x_1 \cdot z_1 + x_2 \cdot z_2 + x_3 \cdot z_3} |z\rangle |f(x)\rangle$$

$$\begin{aligned} |\varphi_4\rangle = \frac{1}{\sqrt{2^3}} \{ & |\varphi_4\rangle_{state\ 1} + |\varphi_4\rangle_{state\ 2} + |\varphi_4\rangle_{state\ 3} + |\varphi_4\rangle_{state\ 4} + |\varphi_4\rangle_{state\ 5} + |\varphi_4\rangle_{state\ 6} \\ & + |\varphi_4\rangle_{state\ 7} + |\varphi_4\rangle_{state\ 8}\} \end{aligned}$$

$$\begin{aligned} |\varphi_4\rangle_{state\ 1} = & [(-1)^{0\cdot0+0\cdot0+0\cdot0}|000\rangle + (-1)^{0\cdot0+0\cdot0+1\cdot0}|001\rangle + (-1)^{0\cdot0+1\cdot0+0\cdot0}|010\rangle \\ & + (-1)^{0\cdot0+1\cdot0+1\cdot0}|011\rangle + (-1)^{0\cdot1+0\cdot0+0\cdot0}|100\rangle + (-1)^{0\cdot1+0\cdot0+1\cdot0}|101\rangle \\ & + (-1)^{1\cdot0+1\cdot0+0\cdot0}|110\rangle + (-1)^{1\cdot0+1\cdot0+1\cdot0}|111\rangle] \otimes |f(000)\rangle \\ & = [|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle] \otimes |100\rangle \end{aligned}$$

$$\begin{aligned} |\varphi_4\rangle_{state\ 2} = & [(-1)^{0\cdot0+0\cdot0+0\cdot1}|000\rangle + (-1)^{0\cdot0+0\cdot0+1\cdot1}|001\rangle + (-1)^{0\cdot0+1\cdot0+0\cdot1}|010\rangle \\ & + (-1)^{0\cdot0+1\cdot0+1\cdot1}|011\rangle + (-1)^{0\cdot1+0\cdot0+0\cdot1}|100\rangle + (-1)^{0\cdot1+0\cdot0+1\cdot1}|101\rangle \\ & + (-1)^{1\cdot0+1\cdot0+0\cdot1}|110\rangle + (-1)^{1\cdot0+1\cdot0+1\cdot1}|111\rangle] \otimes |f(001)\rangle \\ & = [|000\rangle - |001\rangle + |010\rangle - |011\rangle + |100\rangle - |101\rangle + |110\rangle - |111\rangle] \otimes |001\rangle \end{aligned}$$

$$\begin{aligned} |\varphi_4\rangle_{state\ 3} = & [(-1)^{0\cdot0+0\cdot1+0\cdot0}|000\rangle + (-1)^{0\cdot0+0\cdot1+1\cdot0}|001\rangle + (-1)^{0\cdot0+1\cdot1+0\cdot0}|010\rangle \\ & + (-1)^{0\cdot0+1\cdot1+1\cdot0}|011\rangle + (-1)^{0\cdot1+0\cdot1+0\cdot0}|100\rangle + (-1)^{0\cdot1+0\cdot1+1\cdot0}|101\rangle \\ & + (-1)^{1\cdot0+1\cdot1+0\cdot0}|110\rangle + (-1)^{1\cdot0+1\cdot1+1\cdot0}|111\rangle] \otimes |f(010)\rangle \\ & = [|000\rangle + |001\rangle - |010\rangle - |011\rangle + |100\rangle + |101\rangle - |110\rangle - |111\rangle] \otimes |101\rangle \end{aligned}$$

$$\begin{aligned} |\varphi_4\rangle_{state\ 4} = & [(-1)^{0\cdot0+0\cdot1+0\cdot1}|000\rangle + (-1)^{0\cdot0+0\cdot1+1\cdot1}|001\rangle + (-1)^{0\cdot0+1\cdot1+0\cdot1}|010\rangle \\ & + (-1)^{0\cdot0+1\cdot1+1\cdot1}|011\rangle + (-1)^{0\cdot1+0\cdot1+0\cdot1}|100\rangle + (-1)^{0\cdot1+0\cdot1+1\cdot1}|101\rangle \\ & + (-1)^{1\cdot0+1\cdot1+0\cdot1}|110\rangle + (-1)^{1\cdot0+1\cdot1+1\cdot1}|111\rangle] \otimes |f(011)\rangle \\ & = [|000\rangle - |001\rangle - |010\rangle + |011\rangle + |100\rangle - |101\rangle - |110\rangle + |111\rangle] \otimes |111\rangle \end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 5} = {} & [(-1)^{0\cdot 1+0\cdot 0+0\cdot 0}|000\rangle + (-1)^{0\cdot 1+0\cdot 0+1\cdot 0}|001\rangle + (-1)^{0\cdot 1+1\cdot 0+0\cdot 0}|010\rangle \\ & + (-1)^{0\cdot 1+1\cdot 0+1\cdot 0}|011\rangle + (-1)^{1\cdot 1+0\cdot 0+0\cdot 0}|100\rangle + (-1)^{1\cdot 1+0\cdot 0+1\cdot 0}|101\rangle \\ & + (-1)^{1\cdot 1+1\cdot 0+0\cdot 0}|110\rangle + (-1)^{1\cdot 1+1\cdot 0+1\cdot 0}|111\rangle] \otimes |f(100)\rangle \\ & = [|000\rangle + |001\rangle + |010\rangle + |011\rangle - |100\rangle - |101\rangle - |110\rangle - |111\rangle] \otimes |001\rangle\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 6} = {} & [(-1)^{0\cdot 1+0\cdot 0+0\cdot 1}|000\rangle + (-1)^{0\cdot 1+0\cdot 0+1\cdot 1}|001\rangle + (-1)^{0\cdot 1+1\cdot 0+0\cdot 1}|010\rangle \\ & + (-1)^{0\cdot 1+1\cdot 0+1\cdot 1}|011\rangle + (-1)^{1\cdot 1+0\cdot 0+0\cdot 1}|100\rangle + (-1)^{1\cdot 1+0\cdot 0+1\cdot 1}|101\rangle \\ & + (-1)^{1\cdot 1+1\cdot 0+0\cdot 1}|110\rangle + (-1)^{1\cdot 1+1\cdot 0+1\cdot 1}|111\rangle] \otimes |f(101)\rangle \\ & = [|000\rangle - |001\rangle + |010\rangle - |011\rangle - |100\rangle + |101\rangle - |110\rangle + |111\rangle] \otimes |100\rangle\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 7} = {} & [(-1)^{0\cdot 1+0\cdot 1+0\cdot 0}|000\rangle + (-1)^{0\cdot 1+0\cdot 1+1\cdot 0}|001\rangle + (-1)^{0\cdot 1+1\cdot 1+0\cdot 0}|010\rangle \\ & + (-1)^{0\cdot 1+1\cdot 1+1\cdot 0}|011\rangle + (-1)^{1\cdot 1+0\cdot 1+0\cdot 0}|100\rangle + (-1)^{1\cdot 1+0\cdot 1+1\cdot 0}|101\rangle \\ & + (-1)^{1\cdot 1+1\cdot 1+0\cdot 0}|110\rangle + (-1)^{1\cdot 1+1\cdot 1+1\cdot 0}|111\rangle] \otimes |f(110)\rangle \\ & = [|000\rangle + |001\rangle - |010\rangle - |011\rangle - |100\rangle - |101\rangle + |110\rangle + |111\rangle] \otimes |111\rangle\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle_{state\ 8} = {} & [(-1)^{0\cdot 1+0\cdot 1+0\cdot 1}|000\rangle + (-1)^{0\cdot 1+0\cdot 1+1\cdot 1}|001\rangle + (-1)^{0\cdot 1+1\cdot 1+0\cdot 1}|010\rangle \\ & + (-1)^{0\cdot 1+1\cdot 1+1\cdot 1}|011\rangle + (-1)^{1\cdot 1+0\cdot 1+0\cdot 1}|100\rangle + (-1)^{1\cdot 1+0\cdot 1+1\cdot 1}|101\rangle \\ & + (-1)^{1\cdot 1+1\cdot 1+0\cdot 1}|110\rangle + (-1)^{1\cdot 1+1\cdot 1+1\cdot 1}|111\rangle] \otimes |f(111)\rangle \\ & = [|000\rangle - |001\rangle - |010\rangle + |011\rangle - |100\rangle + |101\rangle + |110\rangle - |111\rangle] \otimes |101\rangle\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle = \frac{1}{\sqrt{2^3}} & \{|\varphi_4\rangle_{state\ 1} + |\varphi_4\rangle_{state\ 6} + |\varphi_4\rangle_{state\ 2} + |\varphi_4\rangle_{state\ 5} + |\varphi_4\rangle_{state\ 3} + |\varphi_4\rangle_{state\ 8} \\ & + |\varphi_4\rangle_{state\ 4} + |\varphi_4\rangle_{state\ 7}\}\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle = \frac{1}{\sqrt{2^3}} & \{[[|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle] \otimes |100\rangle \\ & + [|000\rangle - |001\rangle + |010\rangle - |011\rangle - |100\rangle + |101\rangle - |110\rangle + |111\rangle] \otimes |100\rangle] \\ & + [[|000\rangle - |001\rangle + |010\rangle - |011\rangle + |100\rangle - |101\rangle + |110\rangle - |111\rangle] \otimes |001\rangle \\ & + [|000\rangle + |001\rangle + |010\rangle + |011\rangle - |100\rangle - |101\rangle - |110\rangle - |111\rangle] \otimes |001\rangle] \\ & + [[|000\rangle + |001\rangle - |010\rangle - |011\rangle + |100\rangle + |101\rangle - |110\rangle - |111\rangle] \otimes |101\rangle \\ & + [|000\rangle - |001\rangle - |010\rangle + |011\rangle - |100\rangle + |101\rangle + |110\rangle - |111\rangle] \otimes |101\rangle] \\ & + [[|000\rangle - |001\rangle - |010\rangle + |011\rangle + |100\rangle - |101\rangle - |110\rangle + |111\rangle] \otimes |111\rangle \\ & + [|000\rangle + |001\rangle - |010\rangle - |011\rangle - |100\rangle - |101\rangle + |110\rangle + |111\rangle] \otimes |111\rangle]\}\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle = \frac{1}{\sqrt{2^3}} & \{[[|000\rangle + |010\rangle + |101\rangle + |111\rangle] \otimes |100\rangle + [|000\rangle + |010\rangle + |101\rangle + |111\rangle] \\ & \otimes |100\rangle] \\ & + [[|000\rangle + |010\rangle - |101\rangle - |111\rangle] \otimes |001\rangle + [|000\rangle + |010\rangle - |101\rangle - |111\rangle] \\ & \otimes |001\rangle] \\ & + [[|000\rangle - |010\rangle + |101\rangle - |111\rangle] \otimes |101\rangle + [|000\rangle - |010\rangle + |101\rangle|111\rangle] \\ & \otimes |101\rangle] \\ & + [[|000\rangle - |010\rangle - |101\rangle + |111\rangle] \otimes |111\rangle + [|000\rangle - |010\rangle - |101\rangle + |111\rangle] \\ & \otimes |111\rangle]\}\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle = \frac{1}{\sqrt{2^3}}\{&[[2\cdot|000\rangle + 2\cdot|010\rangle + 2\cdot|101\rangle + 2\cdot|111\rangle]\otimes|100\rangle]\\ &+ [[2\cdot|000\rangle + 2\cdot|010\rangle - 2\cdot|101\rangle - 2\cdot|111\rangle]\otimes|001\rangle]\\ &+ [[2\cdot|000\rangle - 2\cdot|010\rangle + 2\cdot|101\rangle - 2\cdot|111\rangle]\otimes|101\rangle]\\ &+ [[2\cdot|000\rangle - 2\cdot|010\rangle - 2\cdot|101\rangle + 2\cdot|111\rangle]\otimes|111\rangle]\}\end{aligned}$$

$$\begin{aligned}|\varphi_4\rangle = \frac{1}{\sqrt{2^3}}\{&2\cdot|000\rangle\otimes[|100\rangle + |001\rangle + |101\rangle + |111\rangle] + 2\cdot|001\rangle\\ &\otimes[|100\rangle + |001\rangle - |101\rangle - |111\rangle] + 2\cdot|101\rangle\otimes[|100\rangle - |001\rangle + |101\rangle - |111\rangle]\\ &+ 2\cdot|111\rangle\otimes[|100\rangle - |001\rangle - |101\rangle + |111\rangle]\}\end{aligned}$$

$\langle 000, c\rangle = 0$

Si escribimos $c = c_1c_2c_3$ tenemos un sistema de ecuaciones:

1	$(0\times c_1)\oplus(0\times c_2)\oplus(0\times c_3) = 0$	Solución Trivial
2	$(0\times c_1)\oplus(1\times c_2)\oplus(0\times c_3) = 0$	$c_2 = 0$
3	$(1\times c_1)\oplus(0\times 0)\oplus(1\times c_3) = 0$	$c_1 \oplus c_3 = 0 :\ c_1 = c_3 = 0, 1$
4	$(1\times c_1)\oplus(1\times c_2)\oplus(1\times c_3) = 0$	

Tomando 4 y si $c_1 = c_3 = 0, c_2 = 0$

$(1\times 0)\oplus(1\times 0)\oplus(1\times 0) = 0 : \mathit{Solución\ trivial}$

Tomando 4 y si $c_1 = c_3 = 1, c_2 = 0$

$(1\times 1)\oplus(1\times 0)\oplus(1\times 1) = 0$

Por tanto: $c = c_1c_2c_3 = 101 \rightarrow \mathit{período\ de\ la\ función}$

4. Cálculos matemáticos (ejemplos y demostraciones)

4.1. Hadamard Gate

$$\boldsymbol{H}|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\times 1 + 1\times 0\\ 1\times 1 - 1\times 0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} = \frac{1}{\sqrt{2}}\left[\begin{pmatrix}1\\0\end{pmatrix} + \begin{pmatrix}0\\1\end{pmatrix}\right] = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

$$\boldsymbol{H}|1\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\times 0 + 1\times 1\\ 1\times 0 - 1\times 1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\-1\end{pmatrix} = \frac{1}{\sqrt{2}}\left[\begin{pmatrix}1\\0\end{pmatrix} - \begin{pmatrix}0\\1\end{pmatrix}\right] = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

- Calculate $|0\rangle$ using Hadamard gate transformation on multiple qubits.

$n = 1$

$|x\rangle = |0\rangle \rightarrow x_1 = 0$

$$\boldsymbol{H}^{\otimes 1}|x_1\rangle = \frac{1}{\sqrt{2^n}}\sum_{z=0}^{2^n-1}(-1)^{x_1\cdot z}\,|z\rangle = \frac{1}{\sqrt{2^1}}[(-1)^{0\cdot 0}|0\rangle + (-1)^{0\cdot 1}|1\rangle] = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

- Calculate $|1\rangle$ using Hadamard gate transformation on multiple qubits.

$n = 1$

$|x\rangle = |1\rangle \rightarrow x_1 = 1$

$$\boldsymbol{H}^{\otimes 1}|x_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{z=0}^{2^n-1} (-1)^{x_1 \cdot z}\,|z\rangle = \frac{1}{\sqrt{2^1}}[(-1)^{1\cdot 0}|0\rangle + (-1)^{1\cdot 1}|1\rangle] = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

- Calculate $|00\rangle$ using Hadamard gate transformation on multiple qubits.

$n = 2$

$|x\rangle = |00\rangle \rightarrow x_1 = 0, x_2 = 0$

$$\boldsymbol{H}^{\otimes 2}|x_1x_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{z=0}^{2^n-1} (-1)^{x_1 \cdot z_1 + x_2 \cdot z_2}\,|z_1z_2\rangle$$

$$\begin{aligned}\boldsymbol{H}^{\otimes 2}|x_1x_2\rangle &= \frac{1}{\sqrt{2^2}}[(-1)^{0\cdot 0+0\cdot 0}|00\rangle + (-1)^{0\cdot 0+0\cdot 1}|01\rangle + (-1)^{0\cdot 1+0\cdot 0}|10\rangle + (-1)^{0\cdot 1+0\cdot 1}|11\rangle] \\ &= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)\end{aligned}$$

5. Bibliography

For create quantum circuit diagrams and export like images. https://quantumcomputing.stackexchange.com/questions/4580/tools-for-creating-quantum-circuit-diagrams

https://algassert.com/quirk#circuit={%22cols%22:[]}