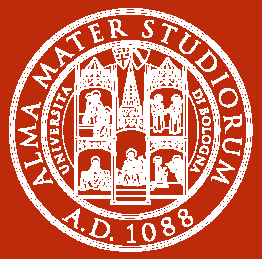


ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

INTRODUCTION TO QUANTUM COMPUTING

Massimo Rudan

DEI & ARCES

massimo.rudan@unibo.it

SUMMARY

- Reduction of calculations to sums.
- Number representations.
- Full adder.
- Logic values and operators.
- Logic functions.
- Complexity of calculations.
- Parallelism.
- Quantum states.
- Qubits, quantum gates, q. circuits.
- Landauer Bound.
- Logical reversibility.
- Implementation of quantum gates.
- Thermodynamic reversibility.
- Bloch sphere, DiVincenzo criteria.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

REDUCTION OF CALCULATIONS TO SUMS

The calculation of **any function** can be reduced to **elementary operations** + − x ÷

$$e^{-x} = 1 - \frac{x}{1} + \frac{x^2}{2\times1} - \frac{x^3}{3\times2\times1} + \cdots$$

In turn, the **elementary operations** can be reduced to **sums**

$$\begin{cases} 12-3 = & 12+(-3) & = 9 \\ 13\times7 = 0+ & \underbrace{13+13+13+13+13+13+13}_{7\text{ times}} & = 91 \\ 20\div5 = & 20\underbrace{-5-5-5-5}_{4\text{ times}} & = 4 \end{cases}$$

That is, given an algorithm able to provide the **sum of two numbers**, the other elementary operations are easily accomplished, and all other calculations ensue.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

NUMBER REPRESENTATION

A number N can be represented as the sum of powers of a given base b, each multiplied by a coefficient with values in the range $0, \ldots, b-1$. For example, N = **nineteen** can be written as:

- Decimal representation (figures 0 … 9): $N = 1 \times 10^1 + 9 \times 10^0 = 19.$
- Septimal representation (figures 0 … 6): $N = 2 \times 7^1 + 5 \times 7^0 = 25.$
- Binary representation (figures 0, 1):

$$N = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 10011.$$

ADDITION OF TWO SINGLE-BIT NUMBERS

Each of the two symbols of the binary system is called ***bit*** (**bi**nary digi**t**). Computers use the binary representation. Therefore, it is necessary to implement the sum of two binary numbers.

0	0	1	1	A
+0	+1	+0	+1	B
0 0	0 1	0 1	1 0	C S

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

The implementation of the above is called ***half adder*** (**S** = sum, **C** = carry). A suitable elaboration on the half adder provides the ***full adder***.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

LOGIC VALUES

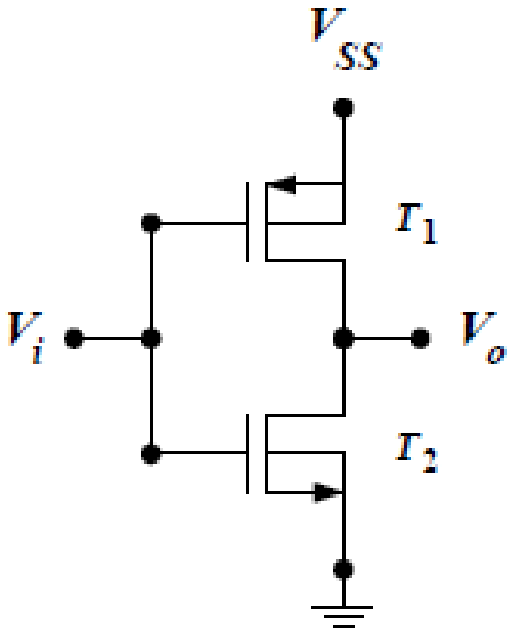
Computers do not use numbers, but voltage levels (high voltage or low voltage). These levels are not related to numbers: they may be viewed as two **logically-opposite conditions**; for them, different equivalent representations are possible:

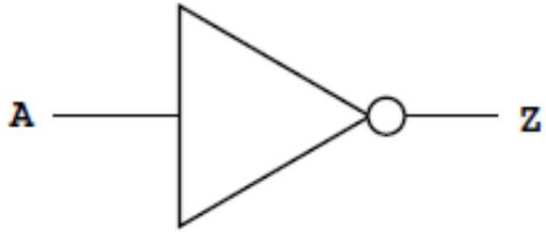
- High voltage – Low voltage.
- Full tank – Empty tank.
- True proposition – False proposition.
- Set A – Complement of A.
- Black – White.
- Symbol "**1**" – Symbol "**0**" (**not numbers!**).

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

EXAMPLE: "NOT" OPERATOR (INVERTER)

The circuit represents a CMOS inverter, that performs the logic operation **NOT**.
The symbol of the **NOT operator** and the corresponding **truth table** are shown.

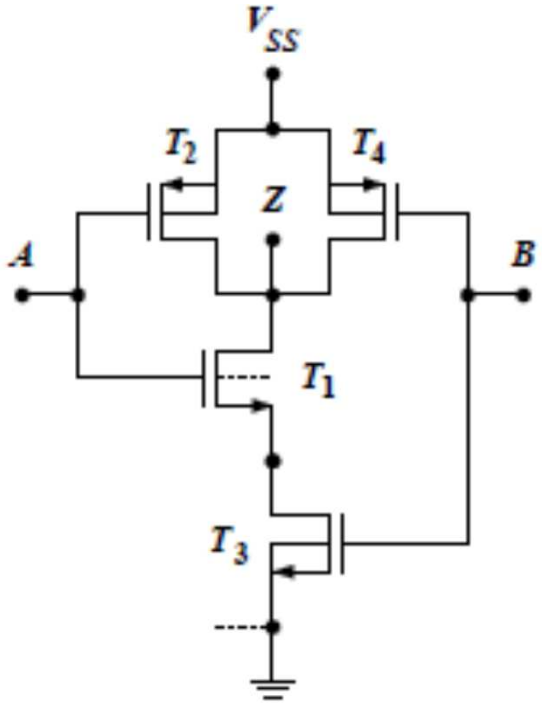


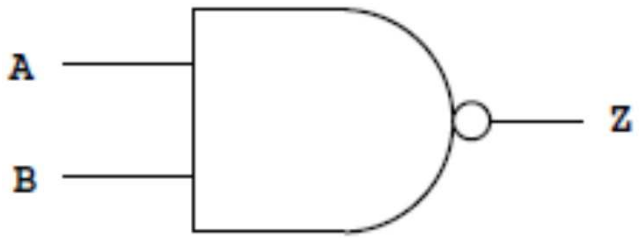


A	Z
1	0
0	1

EXAMPLE: "NAND" OPERATOR

The circuit represents a CMOS implementation of the **NAND** operator. The symbol of the **NAND operator** and the corresponding **truth table** are shown.

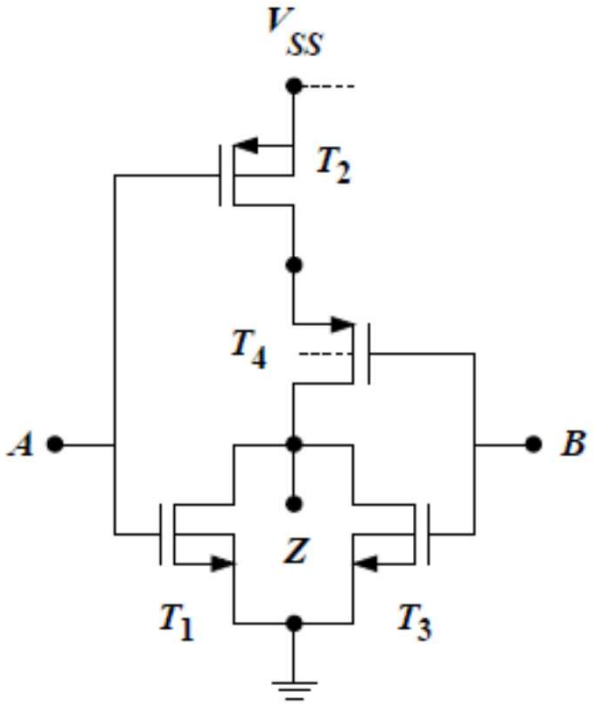


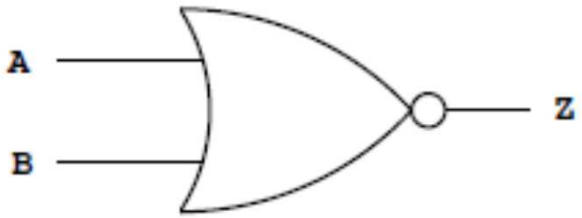


A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

EXAMPLE: "NOR" OPERATOR

The circuit represents a CMOS implementation of the **NOR** operator. The symbol of the **NOR operator** and the corresponding **truth table** are shown.





A	B	Z
0	0	1
0	1	0
1	0	0
1	1	0

LOGIC FUNCTIONS – CANONICAL FORMS – UNIVERSAL OPERATORS

Logic **functions** $F\,(\,A, B, C, \ldots\,, Z\,)$ are combinations of the logic variables $A, B, C, \ldots, Z$.

- **Any** logic function can be expressed as **a sum of products** involving the true or negated form of all variables; alternatively,
- **Any** logic function can be expressed as **a product of sums** involving the true or negated form of all variables.

Thus, the **NOT**, **AND**, and **OR** operators suffice to build-up any logic function. Also,

- Suitable combinations of **NOR (NAND)** operators provide **one or the other** of the NOT, AND, OR operators.

Thus, the **NOR (NAND)** operator suffices to build-up any logic function, **including those necessary to implement the *full adder***.

POLYNOMIAL COMPLEXITY OF A CALCULATION

Problem 1: The interest rate on the **15**-year loans is changed. A bank must recalculate, for each monthly installment of n loans using the French amortization, the part related to the new interest.

The number of calculations necessary is of the order of $K = 15 \times 12 \times n$.

If $n = 1{,}000{,}000$, then $K = 180{,}000{,}000$.

If the size of a problem is n, and the number of calculations necessary to solve it is some function of n that has a polynomial form (like in this case), the complexity of the problem is of the **polynomial (P)** type.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

NON-POLYNOMIAL COMPLEXITY OF A CALCULATION (A)

Problem 2 (the ***Travelling Salesman Problem***): given a list of n cities and the distances between each pair of cities, find the shortest route that visits each city exactly once and returns to the city whence the route started.

The solution is conceptually easy:

- Let K be the total number of routes.
- Calculate the length of routes **1** and **2**, and keep the shorter one.
- Then, calculate the length of route **3**, compare it with the one selected at the previous step, and keep the shorter one.
- Continue down to route K.

NON-POLYNOMIAL COMPLEXITY OF A CALCULATION (B)

Let the number of cities be $n = 26$. It follows $K = 25 \times 24 \times 23 \ldots 3 \times 2 \times 1 \sim 10^{25}$.

If the computer has the size of an atom, and the time necessary to calculate the length of a route equals the time taken by light to cross the atom, the time necessary to calculate all routes is about **116 days**.

In this case the size of the problem is $\exp(n)$, whence the number of calculations necessary to solve it depends on n in a non-polynomial form. The complexity of this problem is of the **non-polynomial (NP)** type.

This and other examples (like, e.g., **weather forecast**) show that a classical computer, that performs the calculations one by one, may not be suited for some types of problems. **The possibility of performing calculations in parallel would be of help**.

EXAMPLE OF PARALLELISM

Problem 3 (extracting global properties): let $x_1 = 0$ and $x_2 = 1$ and assume that the values $f(x_1)$, $f(x_2)$ belong to the same set $\{0,1\}$. One wants to recognize **whether f is constant or not**. The four combinations are:

A	B	C	D
$f(0) = 0$ $f(1) = 0$	$f(0) = 1$ $f(1) = 1$	$f(0) = 0$ $f(1) = 1$	$f(0) = 1$ $f(1) = 0$

To solve the problem with a classical computer one determines $f(0)$ and $f(1)$, that is, a classical computer must be used twice.

It is also obvious that, if the **classical** computer were used **only once**, the outcome would be **insufficient** to determine whether f is constant or not.

Note that the answer sought is just **"the two values of f are equal"**, or the **"two values of f are different"**, i.e., (A, B) or (C, D).

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

EXAMPLE OF PARALLELISM (B)

Assume that we find a **special computer**, not able to determine $f(0)$ and $f(1)$ independently; however it is able, using a **single calculation**, to provide the answer **"equal"** or **"different"** with a **1/2 probability**; the other half of the cases correspond to a non-significant outcome, that indicates that the algorithm has failed and that the computation must be repeated.

This conclusion **seems unsatisfactory**: after all, it is true that the standard computer must perform two calculations instead of one, but, on the other hand, the **"special" computer provides a useful answer for only 50% of cases**; due to this, the time required in the average by the special computer is the same as that of the standard one…

EXAMPLE OF PARALLELISM (C)

Consider, instead, the following example: one must perform a calculation that is **crucial with respect to some decision to be taken**, like, e.g., investing in the stock market on a day-by-day basis; the **decision must be taken within 24 hours**, and a **single calculation of f** takes **almost** 24 hours, because n is very large. Obviously, a standard computer would in this case be useless, whereas a **"special" computer would, at least, provide a sensible answer one day out of two**.

The idea of evaluating a function **as a whole** connects this analysis to **quantum mechanics**. In quantum mechanics we deal with functions (**wave functions**) describing the state of a particle or of a system; their form is:

$$w = c_1 w_1 + c_2 w_2 + c_3 w_3 + \ldots$$

where w_k are complex functions and c_k complex coefficients.

PROPERTIES OF THE WAVE FUNCTION

In the linear combination above, functions w_k are the (mutually-orthogonal) eigenfunctions of an operator $\mathcal{A}$ associated to some dynamic variable A. Assuming that $\| w \| < \infty$, it follows that $|c_k|^2$ is proportional to the probability that a measurement finds the particle or the system in state A_k.

In **computation** it suffices to consider particles with **two** states:

$$w = \alpha w_1 + \beta w_2 \ , \qquad |\alpha|^2 + |\beta|^2 = \| w \|^2$$

Examples of the two states (indicated here with w_1 or w_2) are the **polarization directions** of a **photon** (vertical or horizontal polarization) and the **orientations** of an **electron spin** ("spin up" or "spin down").

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

QUBITS – QUANTUM GATES – QUANTUM CIRCUITS

A qubit is a quantum bit; it is **similar** to a classical bit in that **it can take on 0 or 1 as states**, but it **differs** from a bit in that it can also take on a **continuous range of values** representing a superposition of states, i.e., $w = \alpha w_1 + \beta w_2$

A **quantum logic gate** (or **quantum gate**) is a physical object performing logical operations on a qubit or on a small number of qubits. Connected quantum gates form **quantum circuits**.

When the superposition of states is exploited to carry out calculations, it is also called **quantum parallelism**.

An important point is that the equations of quantum mechanics **are reversible with respect to time**. It follows that, when dealing with quantum gates, one must consider the issue of **logical reversibility** and **thermodynamic reversibility**.

ENERGY CONSUMPTION OF GATES – LANDAUER BOUND

The classical gates consume energy. The **Landauer Bound (L)** or Landauer Limit is the **minimum energy** consumed to erase s bits; it reads

$$L = s\, k_B\, T \log(2)$$

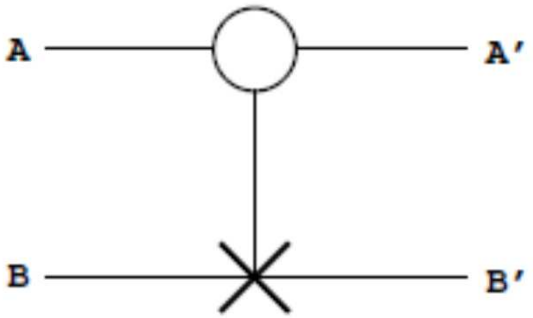
with T the temperature of the heat sink surrounding the device.

Note that the **classical AND, OR, NAND, NOR gates**, and their combinations, erase bits because these gates are **logically irreversible**, that is, some of their output values are such that the input values can not be reconstructed (*)).

One may think of realizing logically-reversible gates **by preventing the bit erasure**, namely, by obtaining gates in which the number of output variables is equal to that of the input variables, and reconstruction of inputs is possible.

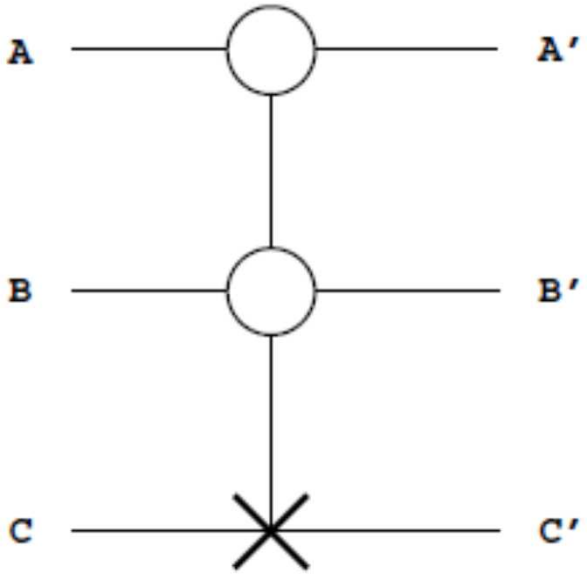
(*) The NOT operator is logically reversible, but its standard implementation consumes energy.

LOGICALLY-REVERSIBLE GATES: CNOT AND CCNOT



A	B	A'	B'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Symbol and truth table of the **Controlled NOT (CNOT)** operator

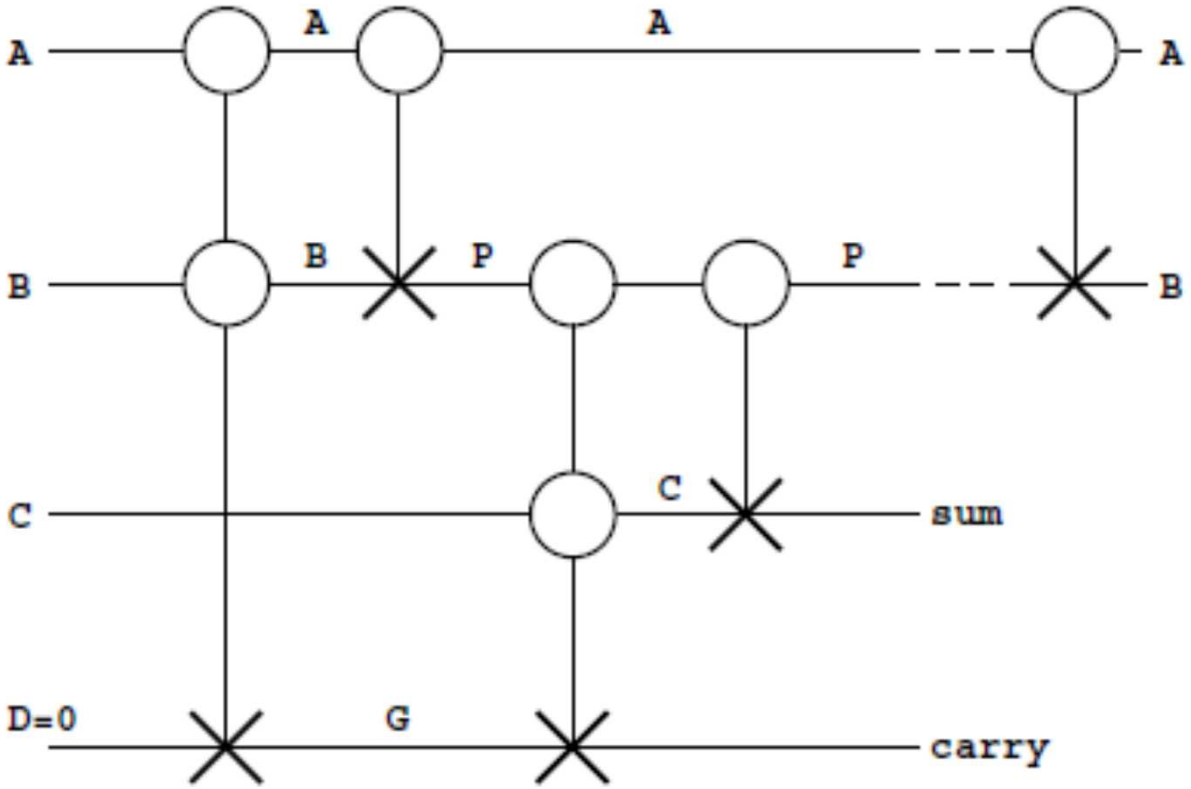


A	B	C	A'	B'	C'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Symbol and truth table of the **Controlled controlled NOT (CCNOT)** operator, also called **Toffoli gate**.

LOGICALLY-REVERSIBLE GATES: FULL ADDER

Implementation of the **full adder** combining the **CNOT** and **CCNOT** operators.



HADAMARD GATE H

It converts each of the states of the basis into a linear combination of them, according to the rules:

$$H\, w_1 = \frac{w_1 + w_2}{\sqrt{2}} \quad , \qquad H\, w_2 = \frac{w_1 - w_2}{\sqrt{2}}$$

where $\| w_1 \| = \| w_2 \| = 1$ is assumed.

When paired with a Hadamard gate, the CCNOT (Toffoli) gate is universal.

QUANTUM GATES: PRACTICAL IMPLEMENTATION

It has been shown so far how, in theory:

- Logically-reversible and universal gates **can be implemented**.
- A suitable combination of such gates provides **the full adder**.

To proceed, the following steps are necessary:

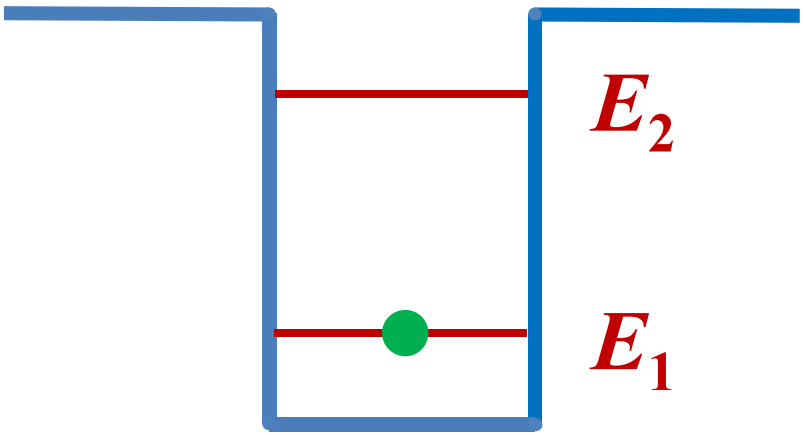
- Associating to each gate one or more **quantum-mechanical operators**.
- Seek for **physical systems** that are described by the same operators.
- Use such systems to **implement** the quantum gates.
- **Identify problems** whose solution is made easier by quantum calculations.

In fact, **different types of implementations** have been investigated so far.

For each implementation, one must check the conditions that would make **thermodynamic reversibility** possible.

THERMODYNAMIC REVERSIBILITY: EXAMPLE

- Consider an electron subjected to a **potential energy** like the one shown in the figure.
- The electron is initially at E_1 (the minimum energy possible in such a system).
- The system is kept at **low temperature**, so that energies like $E_2 - E_1$ or larger are not available.
- In these conditions, it is **impossible** for the electron **to exchange energy** with the environment.
- In quantum terms, the **coherence** of the wave function is kept.



USEFUL REPRESENTATION OF THE QUBIT

In the qubit expression $\boldsymbol{w = \alpha\, w_1 + \beta\, w_2}$ one lets

$$\boldsymbol{\alpha = a\, e^{jp}, \qquad \beta = b\, e^{jq}, \qquad \phi = q - p\,, \qquad 0 \le \phi < 2\pi}$$

whence

$$\boldsymbol{w\, e^{-jp} = a\, w_1 + b\, e^{j\,\phi}\, w_2\,, \qquad \| w \|^2 = a^2 + b^2}$$

due to the orthonormality of $\boldsymbol{w_1}$ and $\boldsymbol{w_2}$. Then, one defines

$$\boldsymbol{\psi = \frac{w}{\| w \|}\, e^{-jp} = \frac{a}{\| w \|}\, w_1 + \frac{b}{\| w \|}\, e^{j\,\phi}\, w_2}$$

and introduces the symbols, with $\boldsymbol{0 \le \theta \le \pi}$,

$$\boldsymbol{\cos\left(\frac{\theta}{2}\right) = \frac{a}{\| w \|}\,, \quad \sin\left(\frac{\theta}{2}\right) = \frac{b}{\| w \|}\,, \qquad |\psi\rangle = \psi, \qquad |0\rangle = w_1\,, \qquad |1\rangle = w_2}$$

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

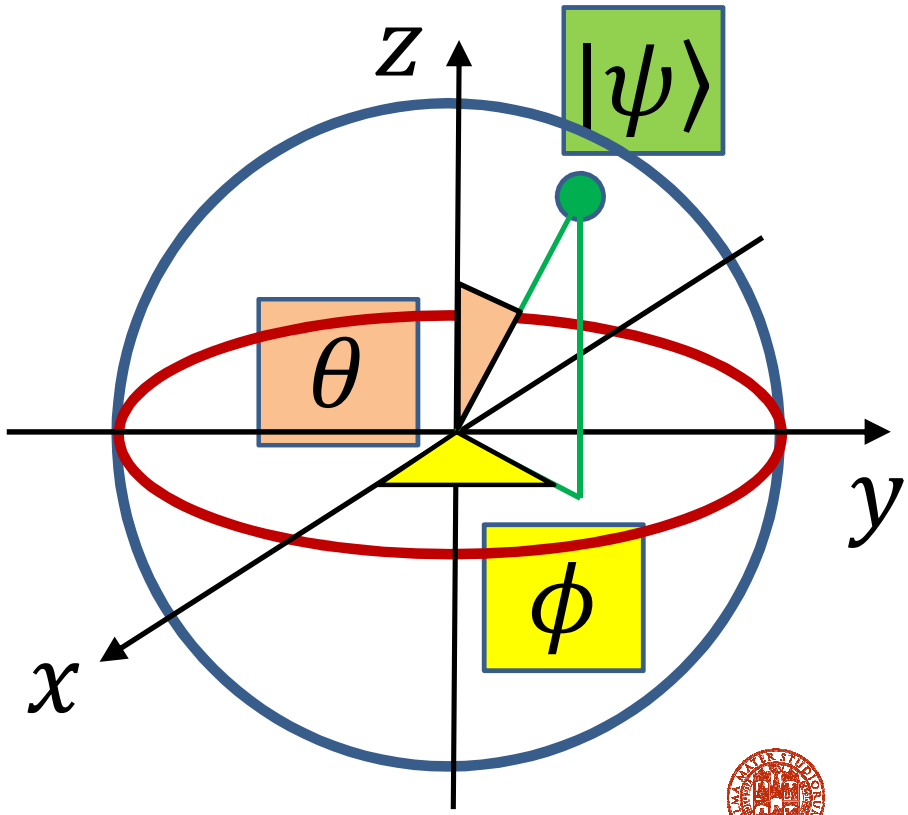
USEFUL REPRESENTATION OF THE QUBIT (B)

In conclusion, the new representation of $\boldsymbol{w = \alpha\, w_1 + \beta\, w_2}$ becomes

$$|\psi\rangle = \cos(\theta/2)\ |0\rangle + \sin(\theta/2)\ e^{j\phi}\ |1\rangle$$

where $|0\rangle$ and $|1\rangle$ are given. It follows that $|\psi\rangle$ is a vector of unit length whose orientation is defined by the θ and ϕ angles. By changing such angles within their full range, the tip of the $|\psi\rangle$ vector spans a sphere called **Bloch sphere.**

The qubit is much **"richer"** than $|0\rangle$ and $|1\rangle$ alone.



DIVINCENZO CRITERIA (*)

The criteria are a formalization of what a quantum computer consists of.

1. A **scalable physical system** with **i)** qubits that are distinct from one another and **ii)** the ability to count exactly how many qubits there are in it.
2. The ability **to initialize the state of any qubit to a definite state** in the computational basis (in the examples above, the comp. basis is $|0\rangle$, $|1\rangle$).
3. The system's qubits **must hold their state: the system must be isolated from the outside world, otherwise the qubits will decohere**. In practice, the qubits must hold their state **long enough** to apply the next operator with assurance that the qubits have not changed state due to outside influences between operations.

(*) [J. D. Hidary, Quantum Computing: An Applied Approach, Springer2019].

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DIVINCENZO CRITERIA (B)

4. The system must be able **to apply a sequence of unitary operators to the qubit states**. The system must also be able to apply **a unitary operator to two qubits at once: this entails entanglement** between those qubits. Let

$$g = s_{11}u_1w_1 + s_{12}u_1w_2 + s_{21}u_2w_1 + s_{22}u_2w_2$$

with $\| g \|^2 = \sum_{ij=1}^{2} |s_{ij}|^2 \ \| u_i \|^2 \ \| w_j \|^2$.

If $s_{11}s_{22} = s_{12}s_{21}$ then $g = (s_{11}u_1 + s_{21}u_2)\,(w_1 + s_{12}\,w_2/s_{11}) = u\,w$,

namely, g **is separable**, otherwise, g **is entangled**. Quoting DiVincenzo "*... **entanglement between different parts of the quantum computer is good; entanglement between the quantum computer and its environment is bad, since it corresponds to decoherence.***"

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

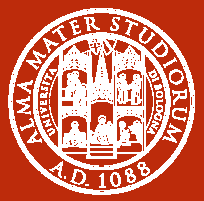
DIVINCENZO CRITERIA (C)

5. The system must be capable of making **"strong" measurements** of each qubit. That is, the measuring technique in the system actually **does measure** the state of the qubit for the property being measured and **leaves the qubit in that state**. E. g., assume that index **1 (2)** means **"spin up (down)"** and that **initially the total spin of a two-electron system equals zero**. Thus,

$$g = s_{12}u_1w_2 + s_{21}u_2w_1$$

Assume that **Alice** (sitting on **Earth**) measures u and finds **spin up**; this is equivalent to forcing $s_{12} = 1$ and $s_{21} = 0$. As a consequence, when **Bob** (sitting on **Anacreon**) measures w, he must necessarily find **spin down**.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Massimo Rudan

DEI & ARCES

massimo.rudan@unibo.it

www.unibo.it