Quantum Computing

Intensive Computation

Annalisa Massini
2022-2023

Lecture 20

References

- *Arithmetic circuits for quantum computing: a software library*
  - L. Raggi - *Master Thesis - University of Turin, Italy – 2020*

- F. Orts, G. Ortega, E.F. Combarro, E.M. Garzon, *A review on reversible quantum adders*, Journal of Network and Computer Applications, 2020

- https://qiskit.org/textbook/what-is-quantum.html

- https://en.wikipedia.org/wiki/Quantum_logic_gate

TELEPORTATION

Teleportation

- We **can not copy** the state of a qubit and give it to somebody
- Anyway, there is a **technique for moving quantum states**, even in the absence of a quantum communications channel linking the sender of the quantum state to the recipient
- This technique goes by the name of **quantum teleportation** and was first published in 1993 by IBM Fellow Charles Bennett *et al*.
- This technique:
  - requires **two classical bits** of information to be transferred via traditional means
  - involves **three qubits**
  - uses **entanglement** as the connection and transference mechanism

Teleportation

The story is the following:

- Alice and Bob met long ago, but now live far apart
- While together **they generated an entangled pair**, each taking one qubit of the pair when they separated
- Many years later, Bob is in hiding, and **Alice's mission** is to **deliver a qubit $|\psi\rangle$ to Bob**
- She does **not know** the state of the qubit, and moreover **can only send classical information** to Bob

- Should Alice accept the mission?

Teleportation

- Intuitively, things look pretty bad for Alice:
  - She does not know the state $|\psi\rangle$ of the qubit she has to send to Bob
  - The laws of quantum mechanics **prevent her from determining the state** when she only has a single copy of $|\psi\rangle$ in her possession
  - Even if she did know the state $|\psi\rangle$, describing it precisely takes an infinite amount of classical information since $|\psi\rangle$ takes values in a continuous space
  - So even if she did know $|\psi\rangle$, it would take forever for Alice to describe the state to Bob

- Fortunately for Alice, **quantum teleportation** is a way of utilizing the **entangled pair** in order to send $|\psi\rangle$ to Bob, with only a **small overhead of classical communication**

Teleportation

- In summary, the steps of the solution are as follows:
  - Alice interacts the qubit $|\psi\rangle$ to deliver to Bob with her half of the entangled pair
  - She then measures her two qubits, obtaining one of four possible classical results: 00, 01, 10, and 11
  - She sends this information to Bob
  - Depending on Alice's classical message, Bob performs one of four operations on his half of the entangled pair
  - By doing this Bob can recover the original state $|\psi\rangle$

Teleportation

- The state to be teleported is $|\psi\rangle = a|0\rangle + b|1\rangle$, where a and b are unknown amplitudes
- We begin with the entanglement of the qubits of Alice and Bob
- We use one of the four Bell states, that is the state:

$$|\Phi\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

- There an infinite number of ways of doing this, and you can use any of them with appropriate changes

- Next, we put the qubit $|\psi\rangle_C$ into the mix and get:
  - $|\psi\rangle_C \otimes |\Phi\rangle_{AB} = (a|0\rangle + b|1\rangle) \otimes \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

Teleportation

$$|\psi\rangle_C \otimes |\Phi\rangle_{AB} = \left(a|0\rangle_C + b|1\rangle_C\right) \otimes \frac{1}{\sqrt{2}}\left(|00\rangle_{AB} + |11\rangle_{AB}\right) =$$

$$= \frac{1}{\sqrt{2}}\left(a|0\rangle_C \otimes |00\rangle_{AB} + a|0\rangle_C \otimes |11\rangle_{AB} + b|1\rangle_C \otimes |00\rangle_{AB} + b|1\rangle_C \otimes |11\rangle_{AB}\right) =$$

$$= \frac{1}{\sqrt{2}}\left(a|000\rangle_{CAB} + a|011\rangle_{CAB} + b|100\rangle_{CAB} + b|111\rangle_{CAB}\right) =$$

$$= \frac{1}{\sqrt{2}}\left(a|00\rangle_{CA} \otimes |0\rangle_B + a|01\rangle_{CA} \otimes |1\rangle_B + b|10\rangle_{CA} \otimes |0\rangle_B + b|11\rangle_{CA} \otimes |1\rangle_B\right)$$

Alice will then make a local measurement in the Bell basis on the two particles in her possession, using the identities:

$$|00\rangle = |0\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}}\left(|\Phi^+\rangle + |\Phi^-\rangle\right) \quad |11\rangle = |1\rangle \otimes |1\rangle = \frac{1}{\sqrt{2}}\left(|\Phi^+\rangle - |\Phi^-\rangle\right)$$

$$|01\rangle = |0\rangle \otimes |1\rangle = \frac{1}{\sqrt{2}}\left(|\Psi^+\rangle + |\Psi^-\rangle\right) \quad |10\rangle = |1\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}}\left(|\Psi^+\rangle - |\Psi^-\rangle\right)$$

Teleportation

$$|\psi\rangle_C \otimes |\Phi\rangle_{AB} =$$
$$= \frac{1}{2}\big(a(|\Phi^+\rangle + |\Phi^-\rangle)\otimes|0\rangle_C + a(|\Psi^+\rangle + |\Psi^-\rangle)\otimes|1\rangle_C$$

Teleportation

The result of Alice's measurement is that the three-particle state would collapse to one of the following four states with equal probability 0.25 of obtaining each:

- $|\Phi^{+}\rangle \otimes (a|0\rangle + b|1\rangle)$
- $|\Phi^{-}\rangle \otimes (a|0\rangle - b|1\rangle)$
- $|\Psi^{+}\rangle \otimes (a|1\rangle + b|0\rangle)$
- $|\Psi^{-}\rangle \otimes (a|1\rangle - b|0\rangle)$

- The measurement does not affect B other than breaking the entanglement
- The two particles are now entangled to each other, in one of the four Bell states and the original quantum state of C is destroyed
- Bob's particle takes on one of the four superposition states above

Teleportation

- The four possible states for Bob's qubit are unitary images of the state to be teleported
- The result of Alice's Bell measurement tells her which of the above four states the system is in
- She can now send her result to Bob through a classical channel using two classical bits

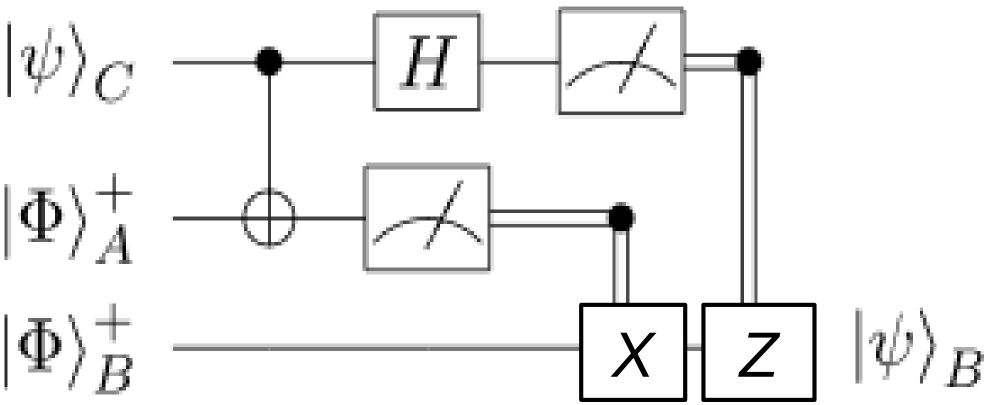
- After Bob receives the message from Alice, he will know which of the four states his particle is in
- Using the information received, he performs a unitary operation on his particle to transform it to the desired state

Teleportation

- If Bob receives code for $|\Phi^+\rangle$ then the quantum state was successfully teleported and Bob has nothing to do
- If Bob receives code for $|\Phi^-\rangle$ the quantum state of Bob has the sign of b wrong, and a Z-gate must be applied to do the phase flip and Bob gets the original state
- If Bob receives code for $|\Psi^+\rangle$ then the quantum state of Bob has a and b reversed, and an X-gate must be applied to do the the bit flip and Bob gets the original state
- If Bob receives code for $|\Psi^-\rangle$ then the quantum state of Bob has a and b reversed with the wrong sign, and an X-gate and then a Z-gate must be applied to gets the original state

Teleportation

Quantum circuit for teleporting a qubit



- The two top lines represent Alice's system, while the bottom line is Bob's system
- The double lines coming out of meters carry classical bits
- After the circuit has run to completion, the value of $|\psi\rangle_C$ will have moved to $|\psi\rangle_B$, and $|\psi\rangle_C$ will have its value set to either $|0\rangle$ or $|1\rangle$, depending on the result from the measurement on that qubit

UNIVERSAL SETS OF QUANTUM GATES

Universal gates

- For the **classical computation**, the **NAND and NOR gates** are **universal gates**, that is any circuit can be designed using only NAND or NOR gates

- It is interesting to understand **which sets of quantum gates can be considered universal** for Quantum computing

- We can observe that the classical world is a subspace of the quantum one
- Therefore, it is possible to implement all classical functions with quantum gates, while the contrary is not possible

Universal gates

- Then there are two aspects to highlight:
  - which set of **quantum gates** allows to implement any kind of **classical function**
  - which set of **quantum gates** is capable to implement any kind of **quantum function**

- We have already verified that the **CCNOT** (Toffoli) gate
  - is **equivalent to a NAND** gate assuming that an ancilla qubit can be initialized to $|1\rangle$
  - and it can be used to **implement any kind of classical function**

Universal gates

- A set of **universal quantum gates** is any set of gates to which any operation possible on a quantum computer can be reduced, that is, **any other unitary operation can be expressed as a finite sequence of gates from the set**
- We have to notice these sets can be said **universal** by implicitly accepting **a given tolerance**
  - This is to say that even in the presence of a set of gates that allow arbitrary rotation around the three axes of the Bloch sphere, *infinitely precise hardware* is required to be able to implement any kind of quantum circuit without errors, and this is clearly *not feasible*

Universal gates

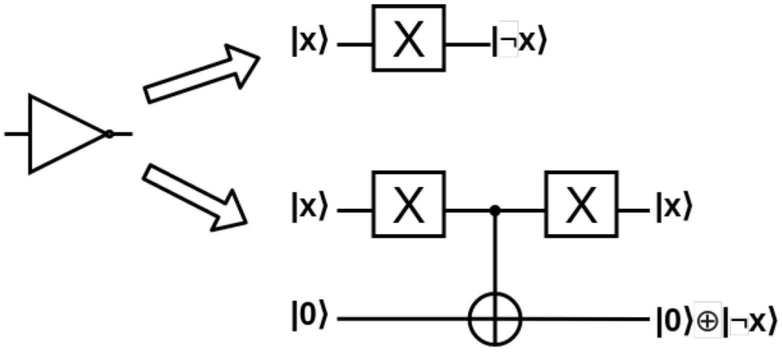
- Some **universal quantum sets** are listed below
  - A common universal gate set is the **Clifford + T gate set**, which is composed of the CNOT, H, S, T
  - The rotation operators $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$, the phase shift gate $P(\phi)$ and CNOT form a widely used universal set of quantum gates
  - Also Toffoli gate + H or Fredkin gate + H are universal sets

REALIZING CLASSICAL GATES AND ARITHMETIC MODULES USING QUANTUM GATES

Realizing classical gates

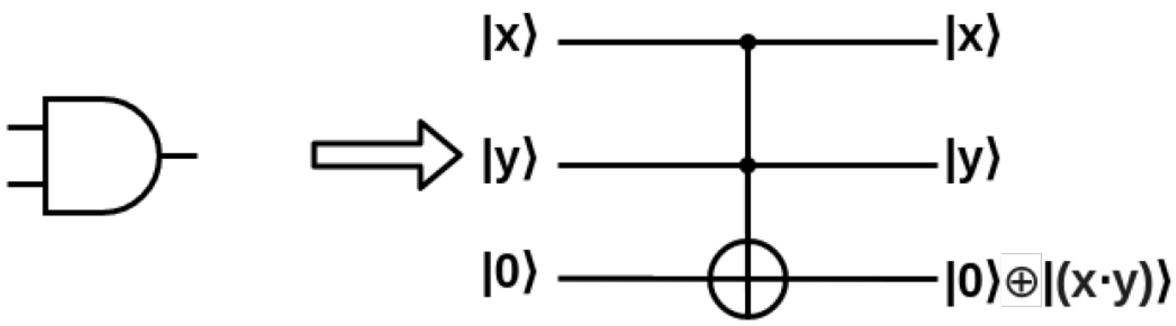
- It can be useful to show how classical gates can be realized using quantum gates
- Some gate implementations may seem redundant, but can be useful for quantum algorithms

- The classical **NOT gate** can be implemented by using one X gate or two X gates combined with a CNOT



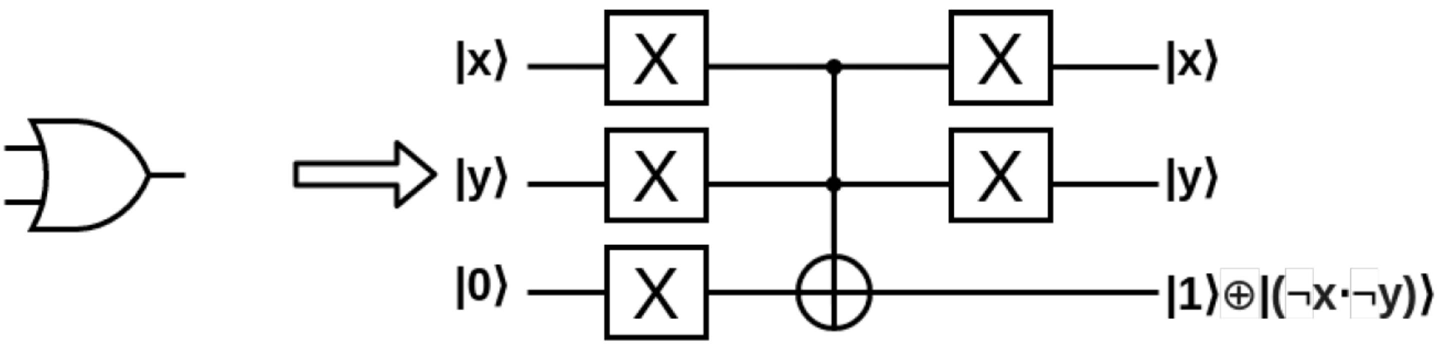
Realizing classical gates

- The classical **AND gate** can be implemented by using the Toffoli (CCNOT) gate setting the ancilla bit to $|0\rangle$



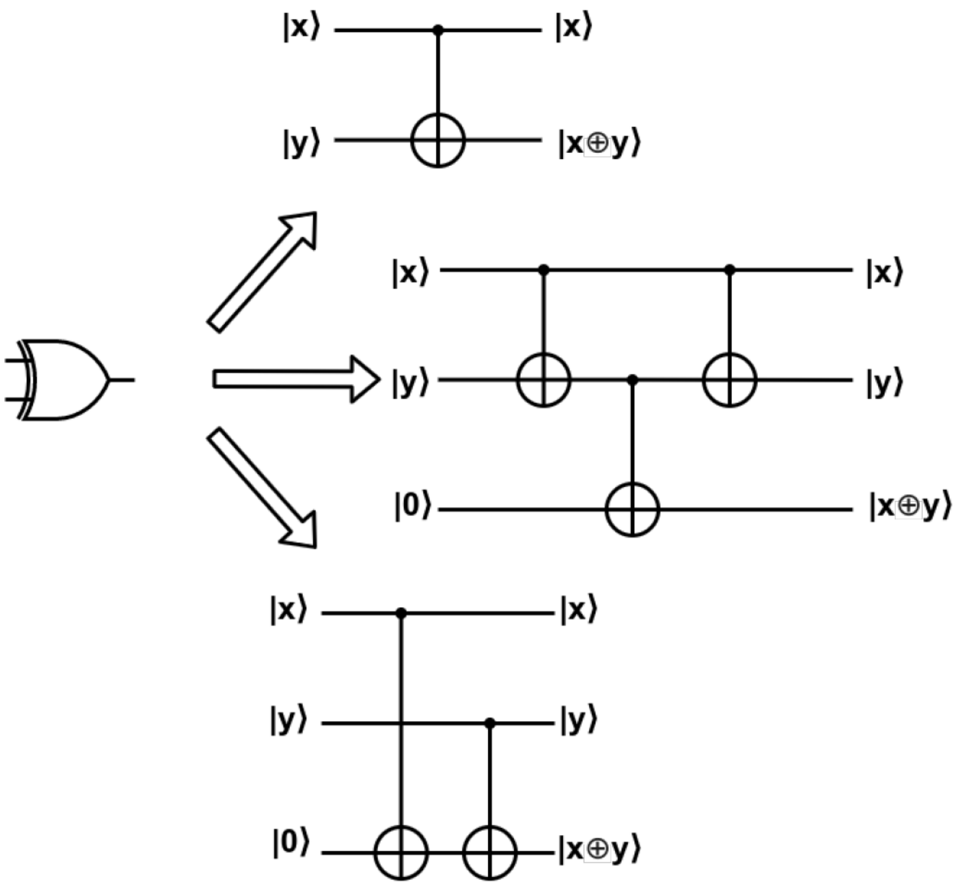
Realizing classical gates

- The classical **OR gate** is implemented using five X gates and one Toffoli setting the ancilla bit to $|0\rangle$



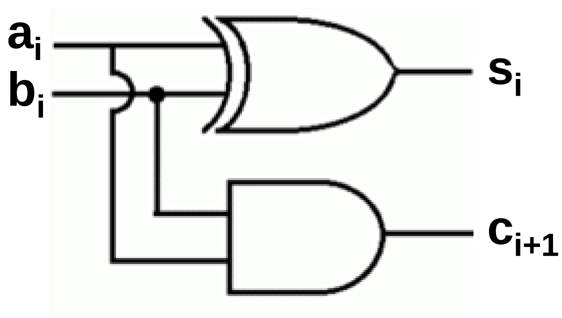
Realizing classical gates

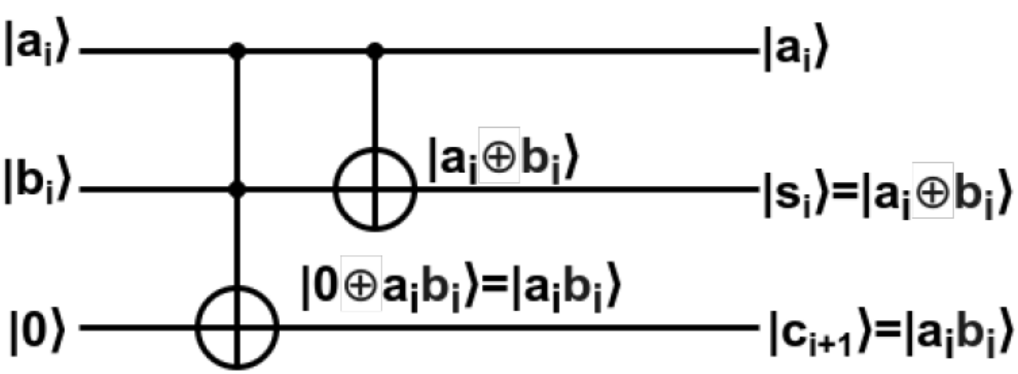
- The classical **XOR gate** can be implemented in different ways



Quantum Half Adder

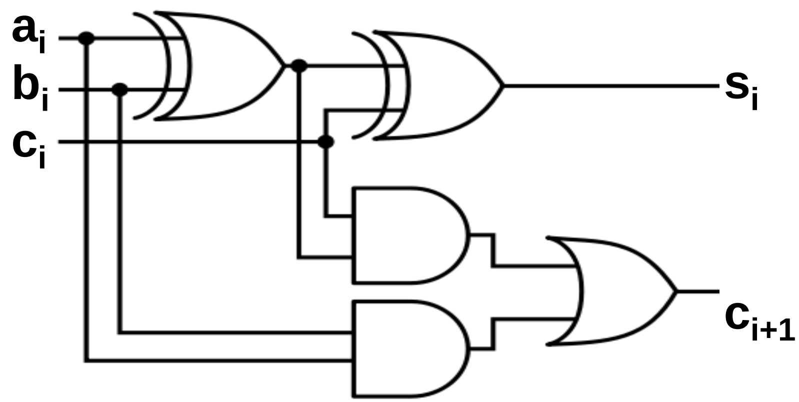
- The quantum realization of the **half adder** can be obtained employing CNOT quantum gate that is equivalent to the XOR gate and a Toffoli gate that is equivalent to the AND gate

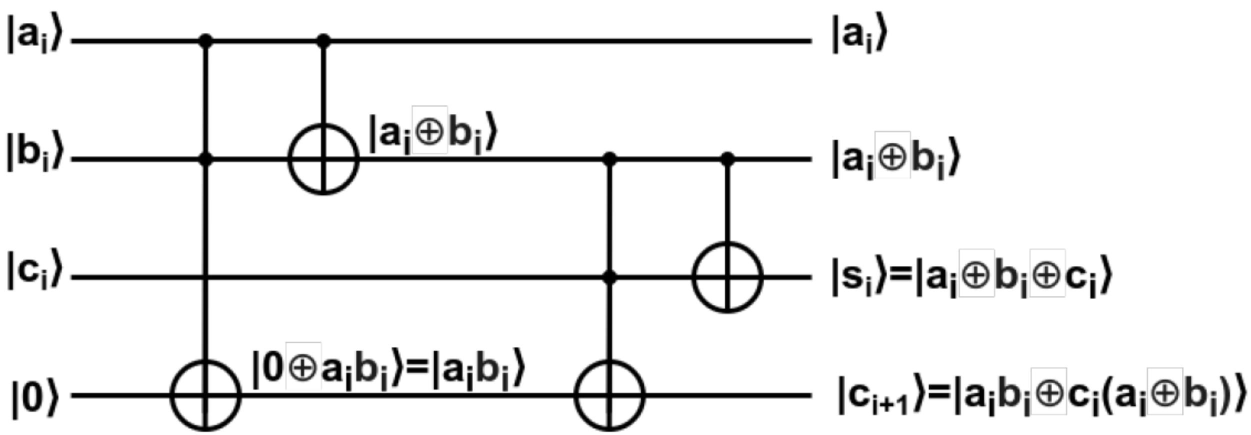




Quantum Full Adder

- The quantum realization of the **full adder** can be obtained employing two CNOT and two CCNOT gates





INTEGER QUANTUM ADDER

Integer quantum adder

An **integer quantum adder** is described in the paper

Cuccaro, Draper, Moulton, Kutin *A new quantum ripple-carry addition circuit,* arXiv: quant-ph/0410184 - 2004

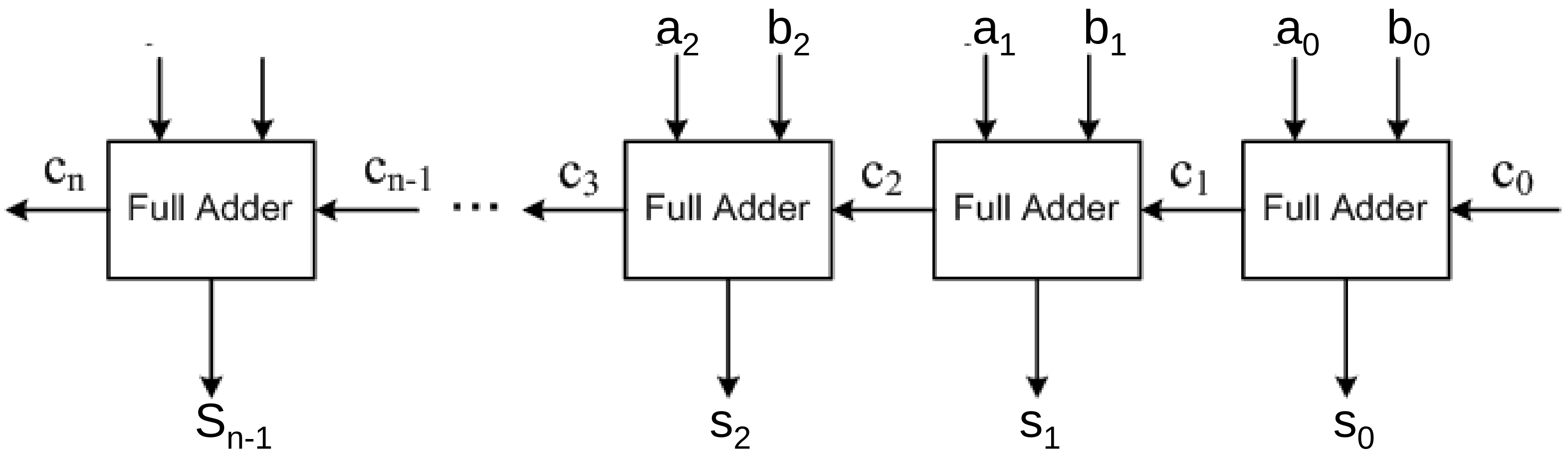
- The Authors propose two versions: **with** and **without carry in**

- The design of the adder is achieved by following two steps:
    1. Development of the basic structure
    2. Optimization by visual inspection of the basic structure

Integer quantum adder

Adder without carry in

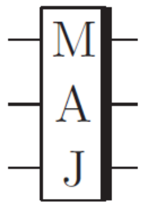
- The structure of the adder makes use of two building blocks: **MAJ** and **UMA**
- The adder ***without*** carry in makes use of an **ancilla qubit**
- **MAJ** and **UMA** are three qubits components that allow to implement a reversible version of the classical ripple carry adder

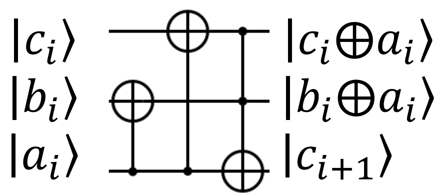


Integer quantum adder

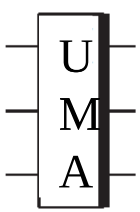
Adder without carry in

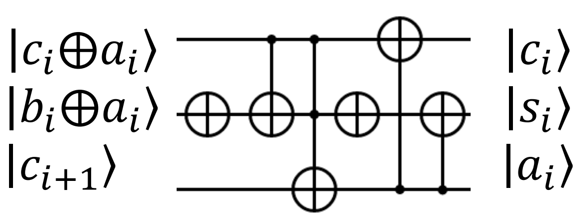
- **MAJ** gate computes the majority of three bits in place

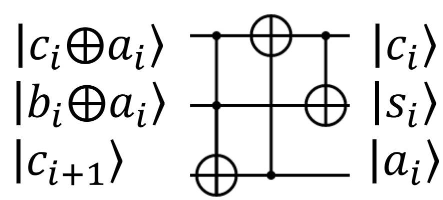




- **UMA** stands for *UnMajority and Add* and is given in two versions: the first (2-CNOT) is conceptually simpler, whereas the second (3-CNOT) admits greater parallelisms



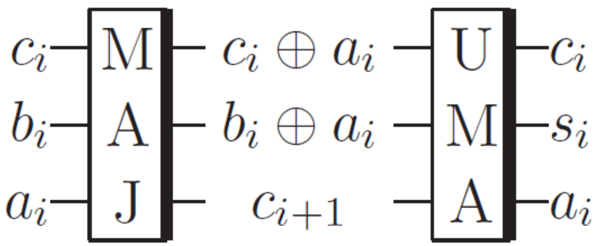




Integer quantum adder

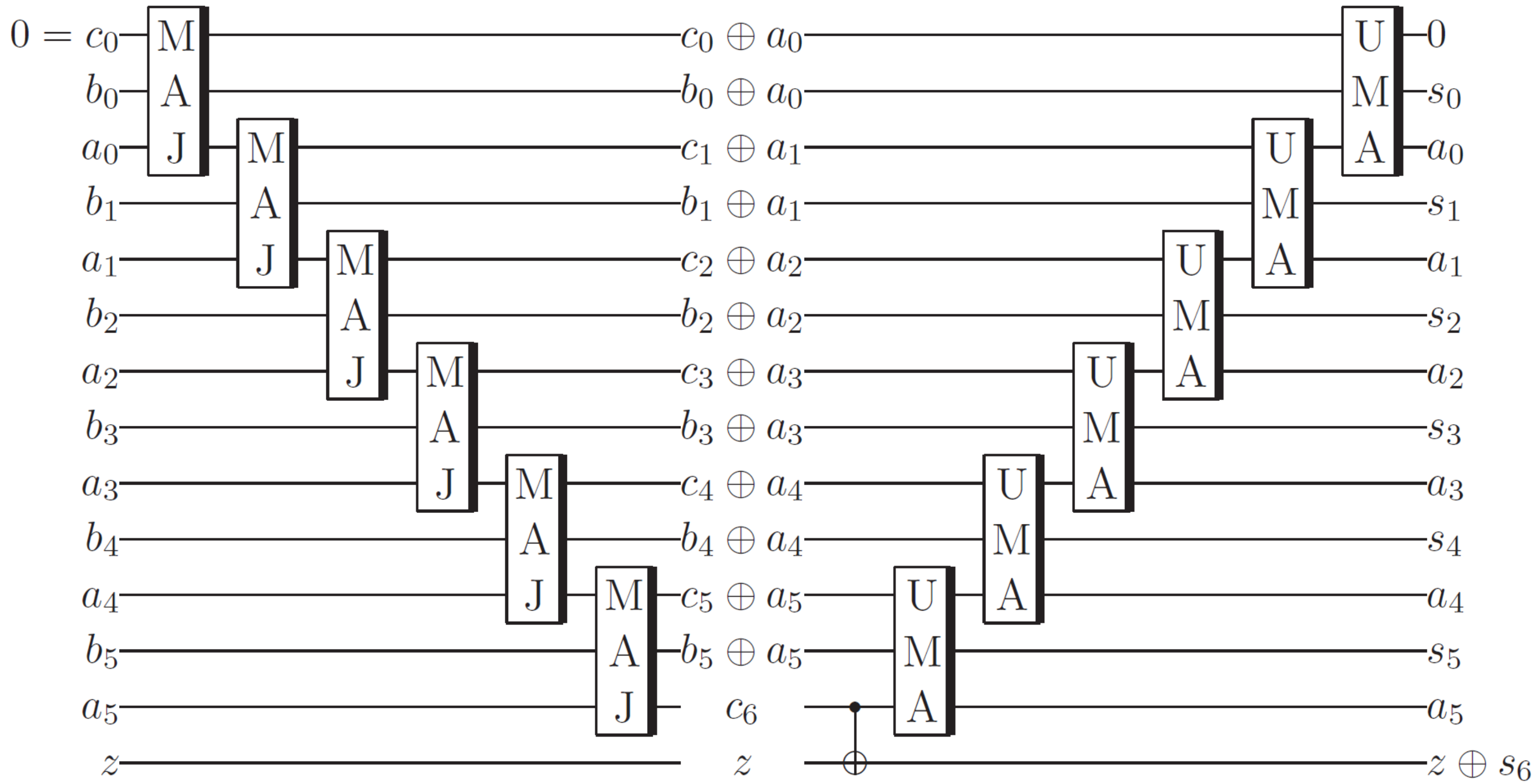
Adder without carry in

- The effect of using MAJ and UMA gates together is shown below
  - Suppose that we have just computed the carry bit c_i
  - We apply the MAJ gate, which writes c_{i+1} into A_i
  - We then continue our computation
  - After we are done using c_{i+1} , we apply the UMA gate, which restores a_i to A_i and c_i to A_{i-1} and writes s_i to B_i



Integer quantum adder

- MAJ and UMA can be cascaded to build a **ripple-carry adder**
- There is an ancilla bit set to 0, containing the initial carry bit c_0
- The output bit contains z when the circuit begins, then $z \oplus s_n$



Improving the circuit

- For the implementation of the adder, we assume that:
  - addends consist of n qubits
  - all qubits are stored within the same quantum register q

- If the number of qubits of each addend is n, considering the two additional qubits for the carry in and the carry out, the total number of qubits for the adders is $2n + 2$

- The quantum register q for all the qubits has size $2n + 2$:
  - **Addend A**: $q[i]$ with i even and $i \in [2, 2n]$
  - **Addend B**: $q[i]$ with i odd and $i \in [1, 2n - 1]$
  - **Carry in**: $q[0]$
  - **Carry out**: $q[2n + 1]$

Improving the circuit

Steps necessary to implement an adder with n -qubit operands:

- **Step 1** Apply n MAJ gates such as:

  MAJ $q[i-1]\ q[i]\ q[i+1]$ with i odd from 1 to $2n-1$

- **Step 2** Apply a CNOT gate:

  CNOT$q[2n]q[2n+1]$

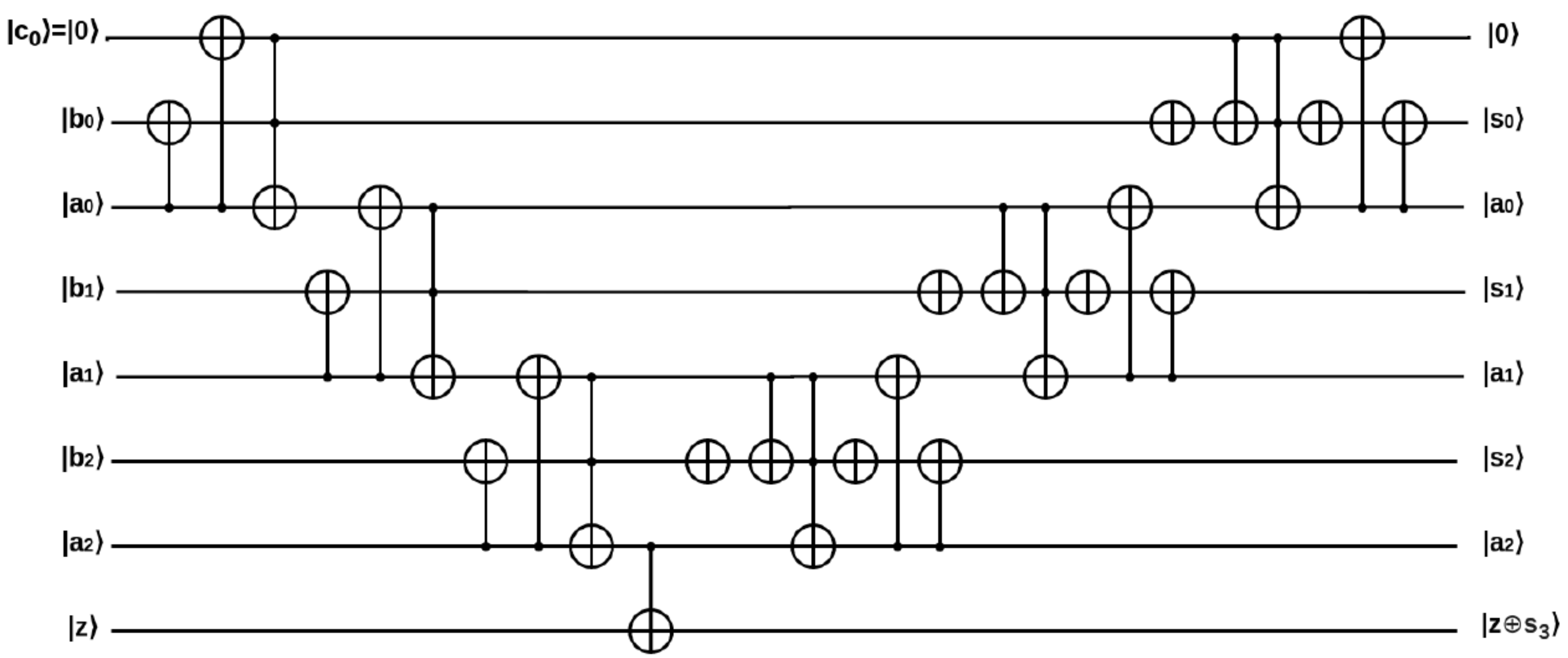
- **Step 3** Apply n UMA gates such as:

  UMA $q[i-1]\ q[i]\ q[i+1]$ with i odd from $2n-1$ to 1

Improving the circuit

We can **reduce the depth and the number of quantum gates** of the basic circuit in several ways

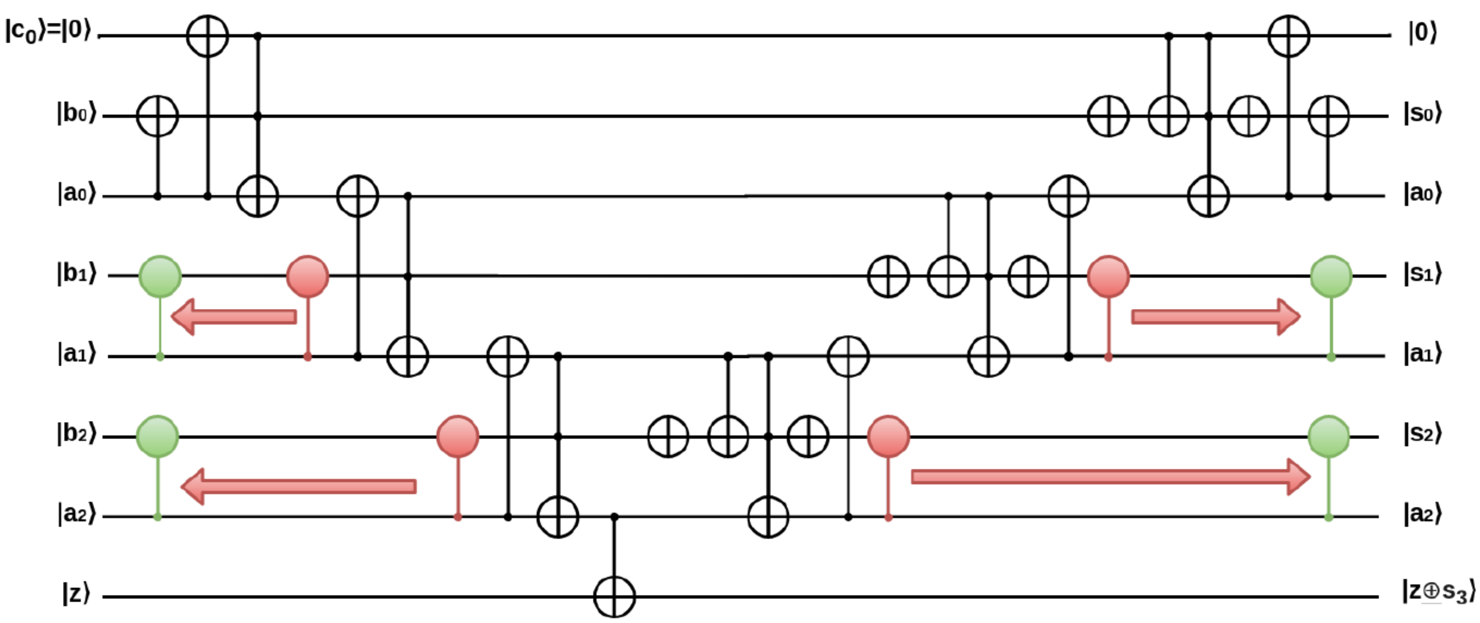
- Let us consider the structure with MAJ and UMA gates replaced with their implementations (using the 3-CX version for the UMA gates)



Improving the circuit

Optimization 1

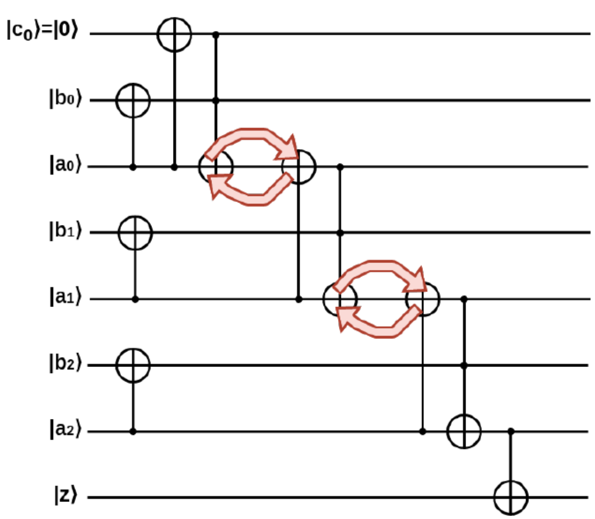
1. The first CNOTs of all the MAJ gates can be performed in a single time-slice at the beginning
2. Similarly, the final CNOTs of all the UMA gates can be performed in a single time-slice at the end

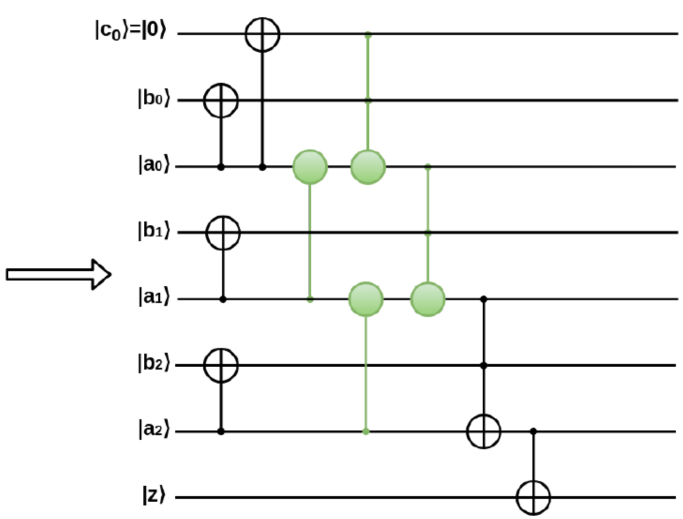


Improving the circuit

Optimization 2

- Consider the first half of the circuit the MAJ ripple: the Toffoli at the end of the i-th MAJ gate can commute with the CNOT of the $(i+1)$-th MAJ gate
- After the swap the Toffoli of the i-th MAJ and CNOT of the $(i+2)$-th MAJ can be done in parallel and the depth decreases

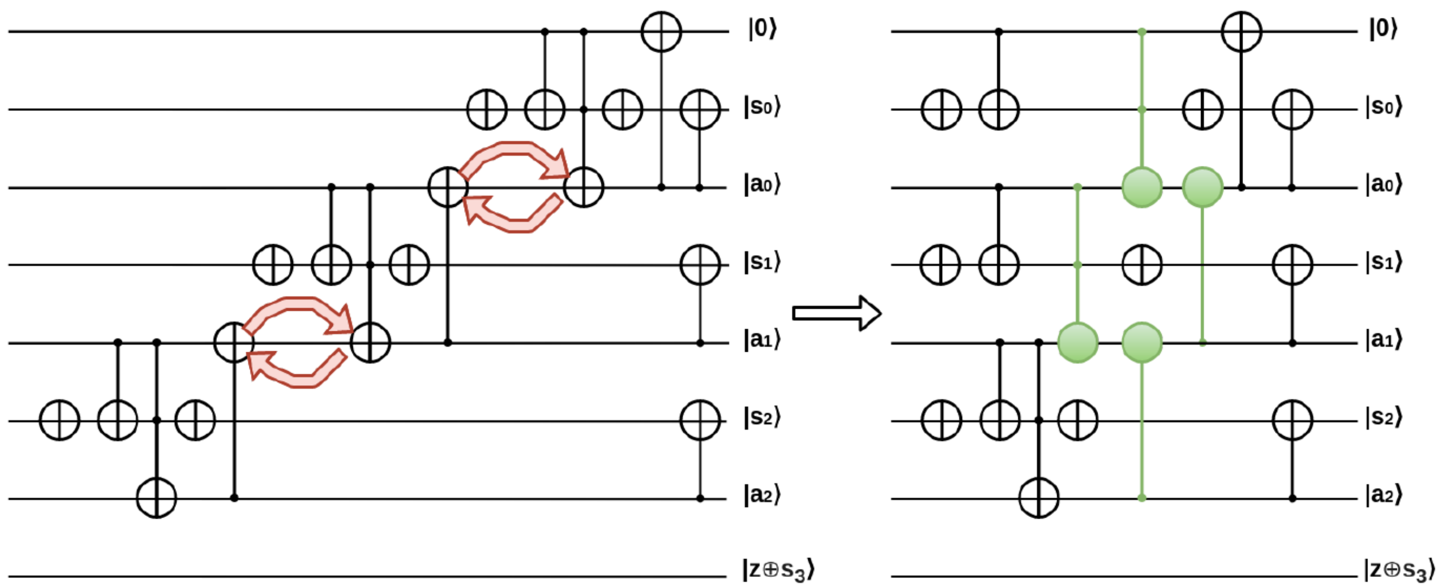




Improving the circuit

Optimization 3

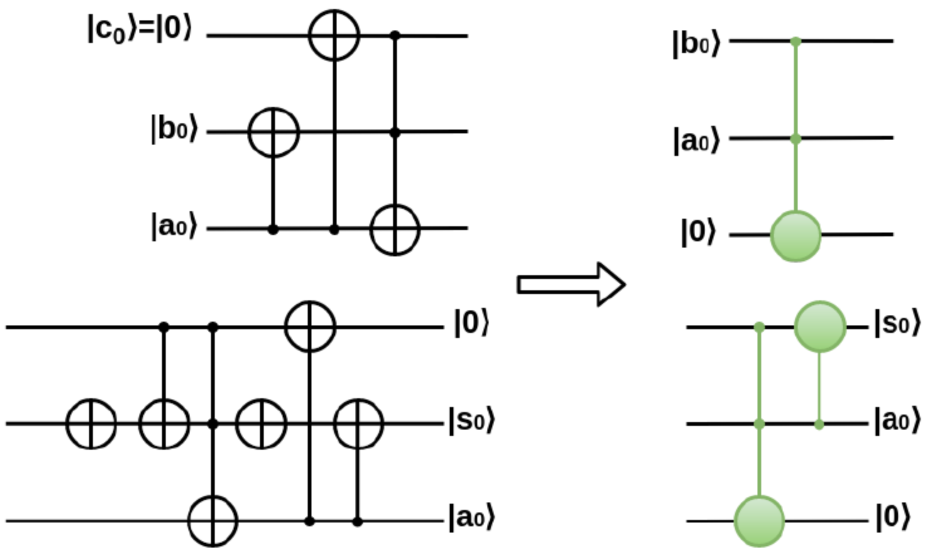
- Similarly, we can swap the Toffoli of the $(i+1)$-th UMA gate with the second CNOT of the i-th UMA gate
- Then, the second CNOT of the i-th UMA can be done in parallel with the Toffoli of the $(i+2)$-th UMA and the depth decreases



Improving the circuit

Optimization 4

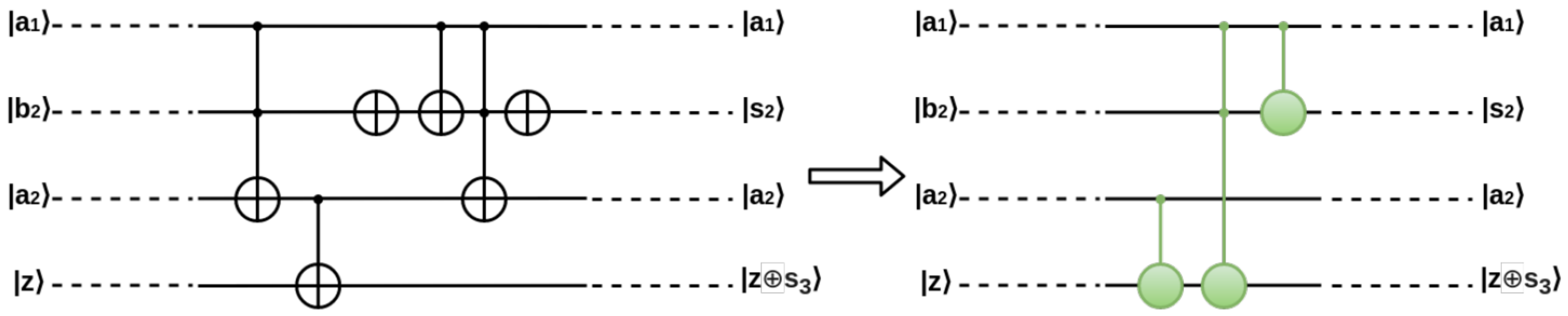
- Since $c_0 = 0$, we do not need a MAJ gate to compute $c_1 = a_0 b_0$
- We can compute c_1 with a single Toffoli and store it in our ancilla
- At the end of the circuit, we undo this same Toffoli, and then set B_0 to s_0 with a single CNOT



Improving the circuit

Optimization 5

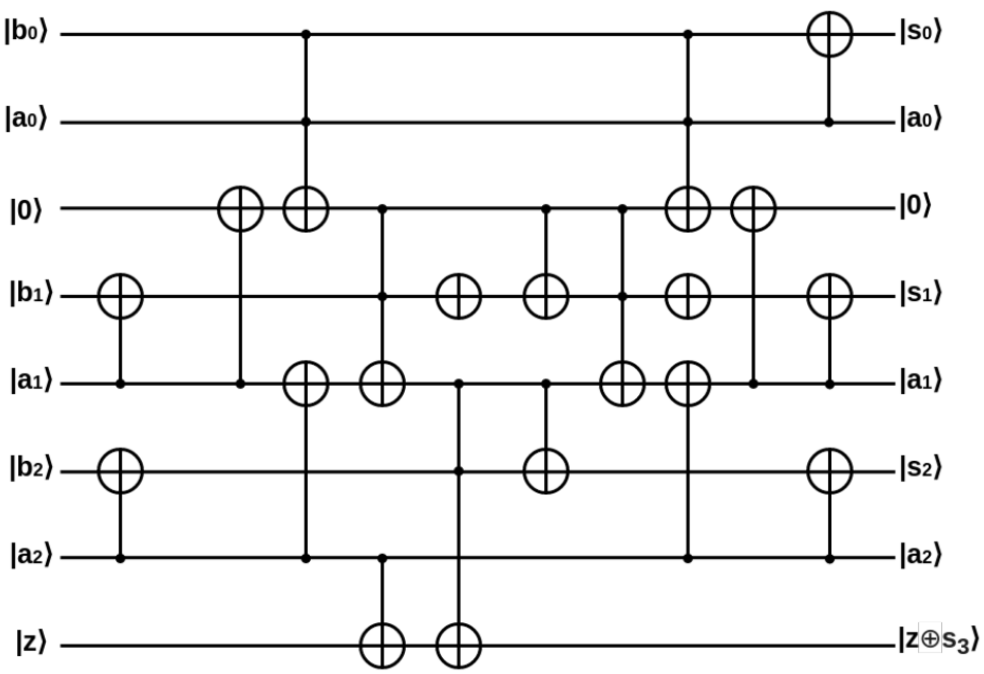
- It is inefficient to write c_n into A_{n-1}, copy it to the output and then erase it, we can instead write directly to the output
- We replace the central piece (two Toffolis, two CNOTs, and two negations) with one Toffoli and two CNOTs
- One of the CNOTs can be done in parallel with other gates



Improving the circuit

Integer adder without carry in optimized

- Assuming $n \geq 2$, the **circuit size** is $2n - 1$ Toffoli gates, $5n - 3$ CNOTs and $2n - 4$ negations
- **Depth** is $2n + 4$: $2n - 1$ Toffoli time-slices and 5 CNOT time-slices



Improving the circuit

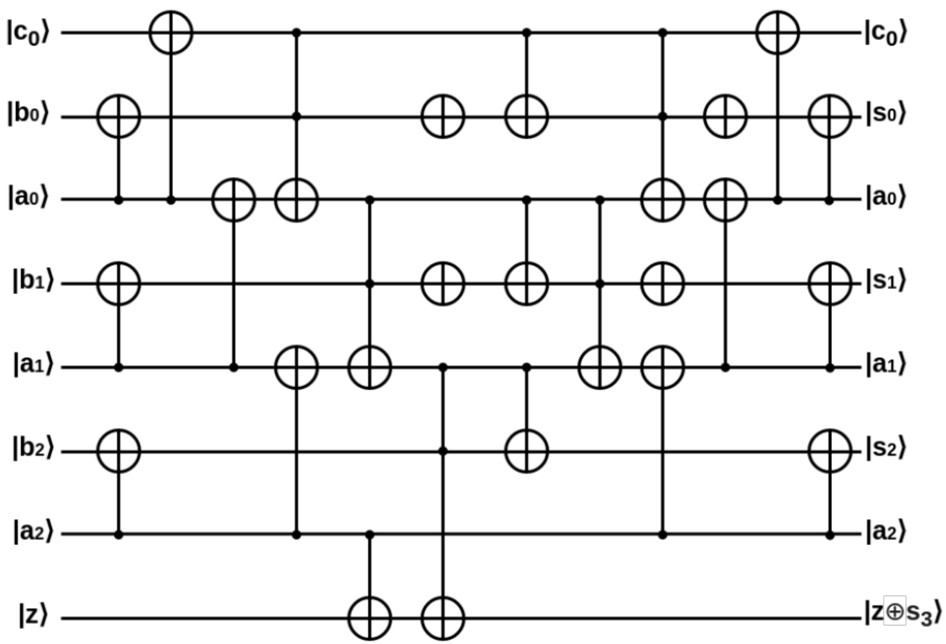
Adder with carry in

- Allowing an incoming carry into the addition circuit implies an additional input bit y, and the operation is $a + b + y$
- The original circuit already works
- Using y instead of the ancilla c_0 the carry c_1 is correctly computed and the ripple continues
- But the **Optimization 4 cannot be used** since we cannot assume the incoming bit is 0
- In this case, assuming $n \geq 2$, the **circuit size** is $2n - 1$ Toffoli gates, $5n + 1$ CNOTs and $2n - 2$ negations
- **Depth** is $2n + 6$: $2n - 1$ Toffoli time-slices and 7 CNOT time-slices

Improving the circuit

Integer adder with carry in optimized

- In this case, assuming $n \geq 2$, the **circuit size** is $2n - 1$ Toffoli gates, $5n + 1$ CNOTs and $2n - 2$ negations
- **Depth** is $2n + 6$: $2n - 1$ Toffoli time-slices and 7 CNOT time-slices



Integer quantum adder

In summary, tables below show the **number of qubits**, the **number of gates** and the **circuit depth**

Carry in version	Operands A and B	Carry in	Carry out	Ancilla	Tot.
No	2n	0	1	1	2n+2
Yes	2n	1	1	0	2n+2

Carry in version	No. X gates	No. CX gates	No. Toffoli gates
No	2n-4	5n-3	2n-1
Yes	2n-2	5n+1	2n-1

Carry in version	CX depth	Toffoli depth	Total depth
No	5	2n-1	$5*CX_{weight}+(2n-1)*CCX_{weight}$
Yes	7	2n-1	$7*CX_{weight}+(2n-1)*CCX_{weight}$

QUANTUM CIRCUIT EVALUATION

Quantum circuit evaluation

- Measuring the complexity of a digital circuit in the classical, non-reversible scenario, is usually straightforward

- A set of universal gates (for instance, AND, OR and NOT or just NAND) is fixed and the circuit complexity can be computed as the number of gates plus a measure of its depth, which captures how many gates can be executed in parallel

- When dealing with reversible circuits, in addition to considering the number of gates and the depth of the circuit, it is also important to take into account other aspects, such as the presence of garbage outputs

Quantum circuit evaluation

- There are a large number of circuits available for quantum computing, in particular for adders
- They all have the common goal to make the addition of two numbers as efficient as possible
- But the **concept of efficiency** often changes among the authors and different authors can measure their circuits using different metrics, taking the ones they consider appropriate or even metrics defined by them
- Comparing circuits, for example adders, becomes difficult if each circuit has been evaluated differently by the authors, in particular if their metrics cannot be directly compared

Quantum cost

- For example, the **quantum cost** of a circuit is usually defined as the **number of gates** which composes a circuit
- According to this, a circuit which consists of 2 Toffoli gates has the same quantum cost than other circuit which consists of 2 CNOT gates
  - Taking into account that a Toffoli gate is composed of 2 CNOT gates and other 3 gates (Nielsen, Chuang), this definition is **imprecise**
- Moreover, an entire circuit built with 5 Toffoli gates could be defined as a novel reversible gate, being its quantum cost 1
  - Comparing this new gate with a circuit which has 2 Toffoli gates would show that the first one has a quantum cost of 1 and the second one a quantum cost of 2

Quantum cost

- Four parameters are used to evaluate reversible circuits (see *On figure of merit in reversible and quantum logic designs*, Mohammadi *et al*., Quant. Inf. Process.2009)
- **Quantum Cost**: the quantum cost of a circuit or a $X \times X$ gate is defined as the number of the 1×1 and 2×2 gates which composes it. The quantum cost of 1×1 and 2×2 gates is 1
  - Note that we are mainly interested in the possibility of using arithmetical reversible circuits in quantum computing and most quantum computers use only 1×1 and 2×2 gates as primitives

Quantum cost

- Four parameters are used to evaluate reversible circuits (see *On figure of merit in reversible and quantum logic designs*, Mohammadi *et al*., Quant. Inf. Process.2009)
- **Delay**: $\triangle$ is the unit of delay defined in and 1×1 and 2×2 gates have a delay of $1\triangle$. The delay of a circuit or a $X \times X$ gate is the number of 1×1 or 2×2 gates computed sequentially
  - If 2 or more gates can be computed in parallel, the delay will be determined by the delay of the slowest gate
  - A higher delay implies that a circuit is slower

Quantum cost

- Four parameters are used to evaluate reversible circuits (see *On figure of merit in reversible and quantum logic designs*, Mohammadi *et al*., Quant. Inf. Process.2009)
- **Number of auxiliary Inputs (Ancilla):** inputs which are set to a constant value (usually 0 or 1) and are used to do auxiliary operations.
- **Garbage Outputs (GO):** outputs which cannot be used at the end of the circuit since they have useless values
  - An output which is uncomputed to its original (and known) value is not considered as a garbage output
  - Uncomputing garbage outputs is especially important if the circuits are to be used in quantum computations, for garbage outputs can prevent the interference that quantum algorithms need to work properly

Quantum circuit evaluation

Furthermore, we can find also:

- **Number of Input:** the number of wires and inputs (it is possible to find the total *Number of Wires* or *Inputs n* of the circuit)
- The increase in the **number of inputs and gates** cost are directly related to the **area/size of the circuit** which will enhance the power dissipation
- **Quantum cost** relates to the number of quantum operation which can be a measure of **delay**
- **Garbage outputs** and **ancilla inputs** are **loss of power** to the environment and **extra consumption** of qubits respectively