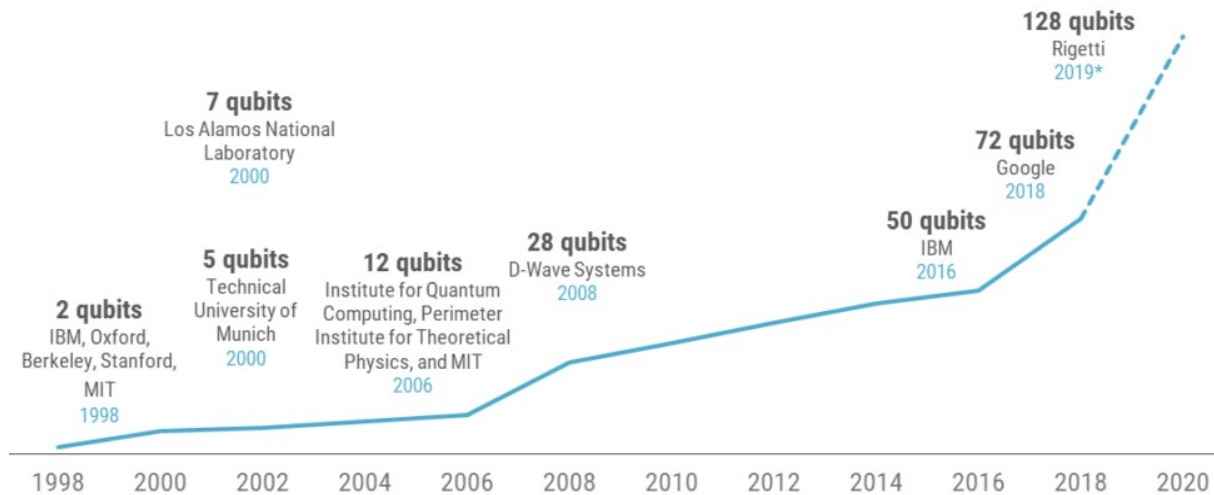


Introduction to Quantum Computers

Quantum computers are getting more powerful

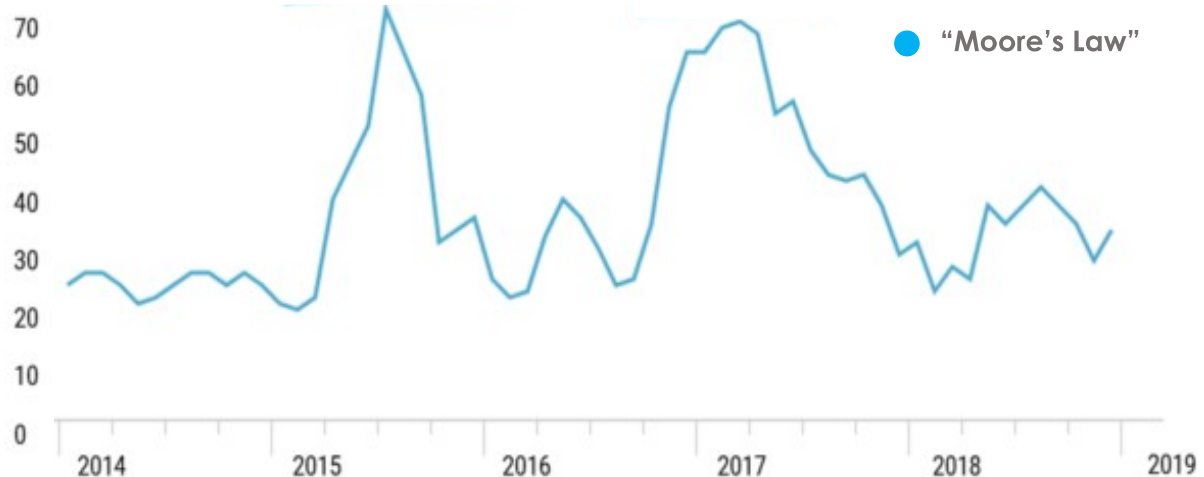


Civilization has advanced as people discovered new ways of exploiting various physical resources such as materials, forces and energies. In the twentieth century information was added to the list when the invention of computers allowed complex info processing to be performed outside human brains. The history of computer technology has involved a sequence of changes from one type of physical realization to another from gears to relays to valves to transistors to integrated circuits. Today's computers are classical, a fact

Which is actually not entirely obvious? A basis of modern computers rests on semiconductor technology. Transistors which are the “neurons” of all computers, work by exploiting properties of semi-conductors. Classical computers are in a certain restricted sense quantum mechanical, because, as far we know today everything is quantum mechanical. No, classical computers, although based on quantum physics, are not fully quantum, because they do not use “quantumness” of matter at the information level, where it really matters.

Need of Quantum Computers

Experts agree Moore's Law is ending



Moore's Law states that the number of transistors on a microchip doubles about every two years, though the cost of computers is halved. Another tenet of Moore's Law says that the growth of micro processors is exponential. In 1965, Gordon E. Moore, the co-founder of Intel, made this observation that became Moore's Law. Continuous decrease in size of Transistor leading it to the size of an atom and it may also possible electrons can pass through it process called quantum tunneling.

In particular, electron tunneling prevents the length of a gate - the part of a transistor that turns the flow of electrons on or off - from being smaller than 5 nm. The other problem hindering smaller transistors is heat extraction. The more transistors there are on a chip, the more heat it produces, and the greater the chance of a malfunction. This is because cooling down the transistors takes more energy than the amount of energy that already passes through the transistors.

Understanding Quantum Mechanics Phenomena

Quantum Computing is the use of the quantum mechanics phenomena such as quantum superposition, entanglement & interference to perform quantum computation.

There are currently two main approaches (Analog and Digital) to physically implement a quantum computing and both approaches uses the qubits.

Analog approach is divided into quantum simulation, annealing and adiabatic. Digital approach uses the quantum logic gates to do quantum computation.

Interference

Quantum interference is a by-product of superposition, is what allows us to bias the measurement of a qubit toward a desired state or set of states. When two or more waves simultaneously and independently travel through the same medium at the same time, their effects are super-positioned. The result of that superposition is called interference.

Interference is further divided into Constructive & Destructive interference. From Interference and Waves Nature we can easily perform Qubits manipulation by using constructive and destructive interference to generate correct and desired output.

Destructive interference occurs when the wave amplitudes oppose each other, resulting in waves of reduced amplitude and Constructive interference occurs when the wave amplitudes reinforce each other, building a wave of even greater amplitude.

Example of interference in real life can be seen in noise cancellation headphone, how they use the interference property to oppose the outside environment sound wave resulting in noise cancellation headphones.

Superposition

One of the properties that sets a qubit apart from a classical bit is that it can be in superposition. Superposition is one of the fundamental principles of quantum mechanics. A typical example visualizing superposition is the double-slit experiment.

Each Qubit utilized could take a superposition of both 0 & 1. Thus, the no. of computation that a quantum computer could undertake is 2 to the power n (2^n), where n is the number of qubits used. When a qubit is in a superposition state of equal weights, a measurement will make it collapse to one of its two basis states $|0\rangle$ and $|1\rangle$ with an equal probability of 50%.

$|0\rangle$ is the state that when measured, and therefore collapsed, will always give the result 0. Similarly, $|1\rangle$ will always convert to 1, whenever you measure its state, you will find that it is indeed either on or off, just like a classical system. But between measurements, a quantum system can be in a superposition of both on and off states at the same time, no matter how counter-intuitive.

Entanglement

It is known that once two quantum systems interact with one another, they become hopelessly entangled partners. From then on, the state of one system will give you precise information about the state of the other system, no matter how far the two are from one another.

Measurements performed on one system seem to be instantaneously influencing other systems entangled with it. But quantum entanglement does not enable the transmission of classical information faster than the speed of light. Quantum entanglement has applications in the emerging technologies of quantum computing and quantum cryptography, and has been used to realize quantum teleportation experimentally.

Understanding Qubits

Qubits represent atoms, ions, photons or electrons and their respective control devices that are working together to act as computer memory and a processor and just as transistors in On/Off states are strung together to form the logic gates that perform classical computations in digital computers, electrons in Up/Down spin states are strung together to form the quantum gates that perform quantum calculations in quantum computers.

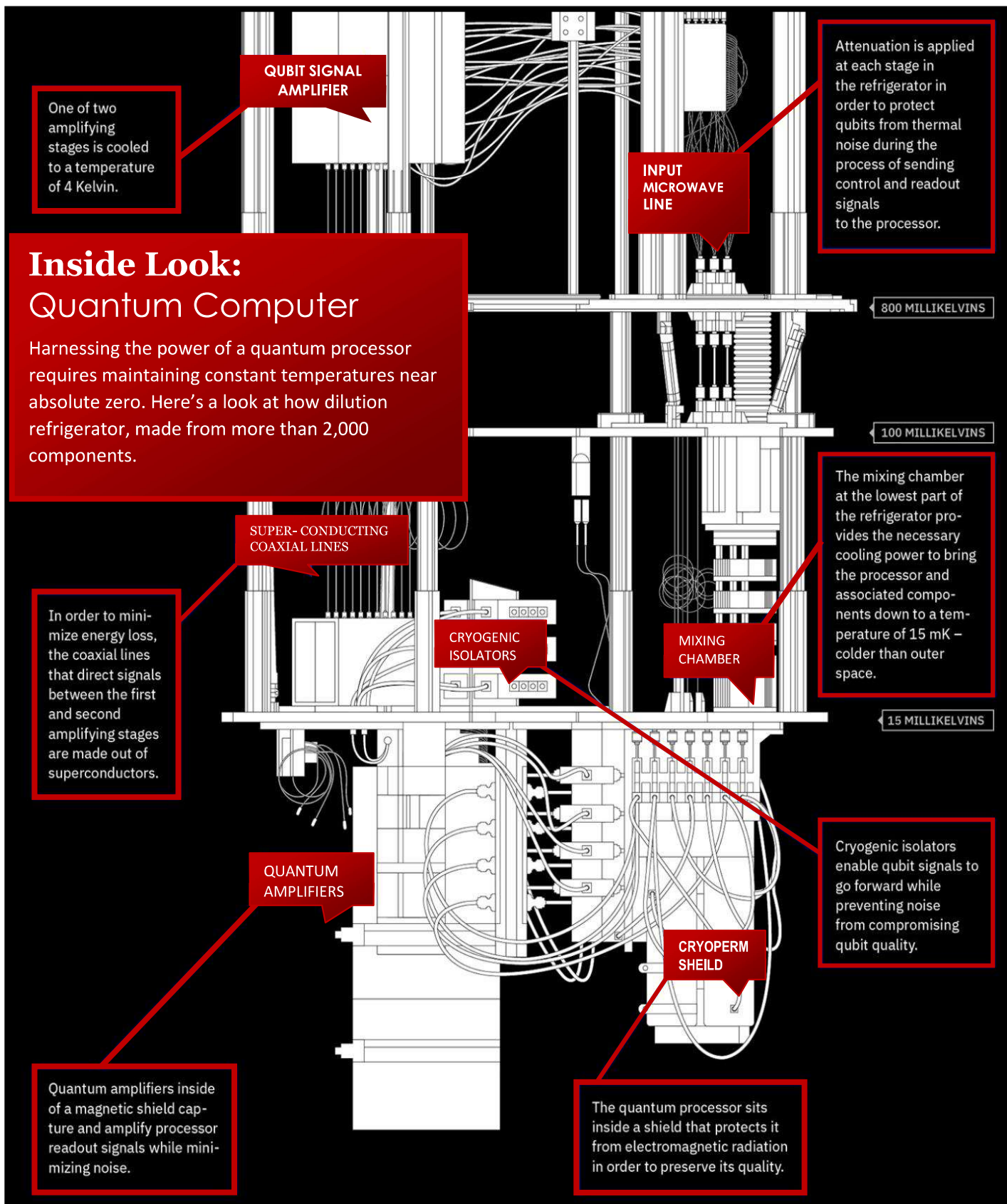
Yet stringing together individual atoms, photon or electrons (while preserving their spin states) is far, far easier said than done. Just as conventional computers are built bit by bit with transistors that are either on or off, quantum computers are built qubits, by qubits with electrons in Spin states that are either up or down.

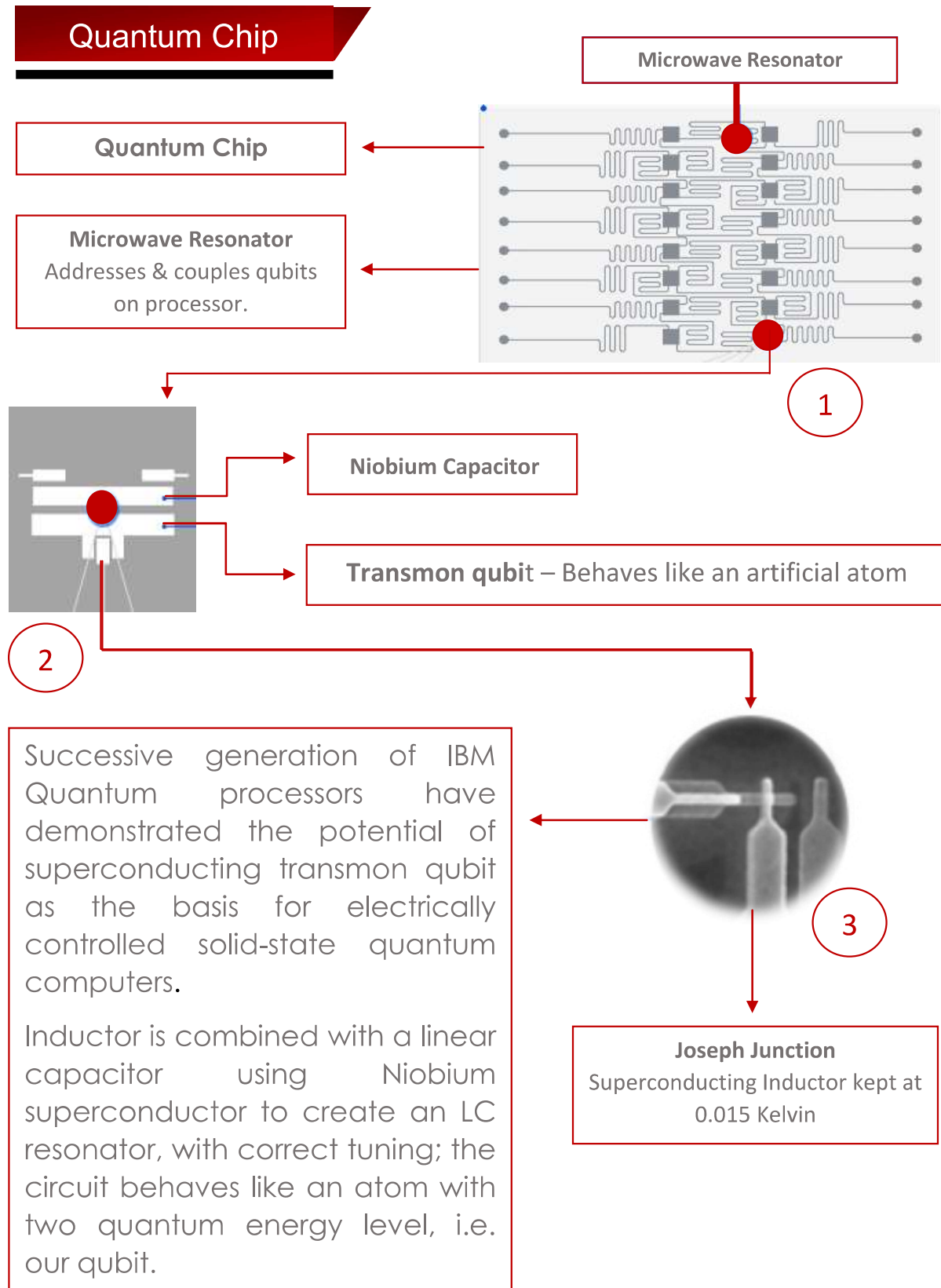
How Qubits are controlled and manipulated?

Computer scientists control the microscopic particles that act as qubits in quantum computers by using **control devices**.

Following are the control devices to control qubits:-

1. **Ion traps** - use optical field or magnetic field to trap ions.
2. **Optical traps** - use light waves to trap & control particles.
3. **Quantum dots** - are made of semiconductor material & are used to contain and manipulate electrons.
4. **Superconducting circuits** - allow electrons to flow with almost no resistance at very low temperatures.
5. **Semiconductor impurities** - contain electrons by using unwanted atoms found in semiconductor material.





Types of Quantum Computers

There are three primary types of quantum computing. Each type differs by the amount of processing power (qubits) needed and number of possible applications, as well as the time required to become commercially viable.

Quantum Annealing

Quantum annealing is best for solving optimization problems. Annealing applies to an array of industry problems for example, Airbus a global aerospace & Defense Corporation known for developing military and commercial aircraft. The company is exploring quantum annealing for digital modeling and materials sciences. While it currently takes engineers years to model the process of air flowing over an aircraft's wing, a quantum computer could take just a few hours to model every single atom of air flowing over a wing at all angles and speeds to determine the optimum or most efficient wing design. Quantum annealing is the least powerful and most narrowly applied form of quantum computing.

Quantum Simulations

In particular, quantum simulators could be used to simulate protein folding one of biochemistry's toughest problems. Misfolded proteins can cause diseases like Alzheimer's and Parkinson's, and researchers testing new treatments must learn which drugs cause reactions for each protein through the use of random computer modeling. It is said that if a protein were to attain its correctly folded configuration by sequentially sampling all the possible drug-induced effects, it would require a time longer than the age of the universe to arrive at its correct natural state. Quantum computers can help compute the vast number of possible protein folding sequences for making more effective medications. In the future, quantum simulations will enable rapid designer drug testing by accounting for every possible protein-to-drug combination.

Universal Quantum Computing

Universal quantum computers are the most powerful and most generally applicable, but also the hardest to build. A truly universal quantum computer would likely make use of over 100,000 qubits; some estimates put it at 1M qubits. Remember that today; the most qubits we can access is not even 128.

The basic idea behind the universal quantum computer is that you could direct the machine at any massively complex computation and get a quick solution. This includes solving the aforementioned annealing equations, simulating quantum phenomena, and more. Researchers have been designing algorithms for years that are only possible on a universal quantum computer. The most well-known algorithms are Shor's algorithm for factoring numbers, and Grover's algorithm for quickly searching unstructured and massive sets of data (to be used for advanced internet search, etc).

In the distant future, universal quantum computers could revolutionize the field of artificial intelligence. Quantum AI could enable machine learning that is faster than that of classical computers. Recent work has produced algorithms that could act as the building blocks of quantum machine learning, but the hardware and software to fully realize quantum artificial intelligence are still as elusive to us as a general quantum computer itself.



Rigetti's 128 qubit quantum chip

Quantum Logic Gates

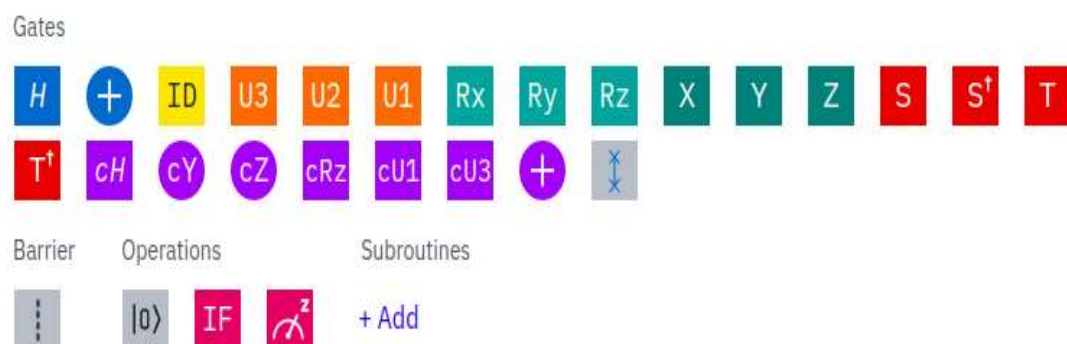
To transform Qubits, we pass them through Quantum Logic Gates that maps the existing state of a Qubit to another state based on the functionality of the gate.

Simple Logic Gates















In order to manipulate fundamental bits, they must be transformed by logic gates that either flips bits to another value or let them remain as they are. Some famous examples of logic gates include AND, OR. They take pairs of bit values and transform them to either a 1 or 0 depending on the functionality of the gate.

Quantum Gates

In quantum computing and specifically the quantum circuit model of computation, a quantum logic gate (or simply quantum gate) is a basic quantum circuit operating on a small number of qubits. They are the analogues for quantum_computers to classical logic gates for conventional digital_computers. Quantum logic gates are reversible, unlike many classical logic gates. Some universal classical logic gates, such as the Toffoli_gate, provide reversibility and can be directly mapped onto quantum logic gates. Quantum logic gates are represented by unitary matrices. The most common quantum gates operate on spaces of one or two qubits. This means that as matrices, quantum gates can be described by 2×2 or 4×4 matrices with orthonormal rows.

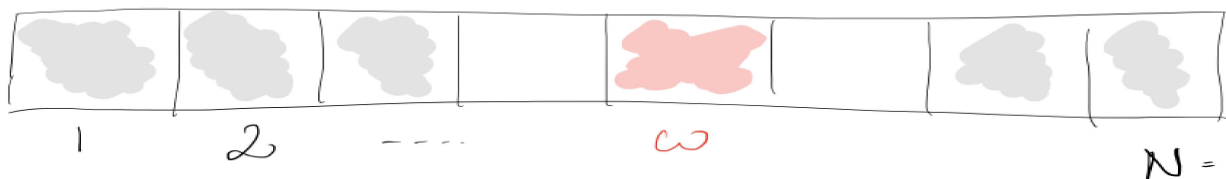


Description of Some Common Quantum Logic Gates:-

 <p>The third physical gate of the Quantum Experience. It is a three-parameter single-qubit gate with duration 2 units of gate time.</p> <div> <div>QASM</div> <div>Matrix</div> </div>	 <p>The identity gate performs an idle operation on the qubit for a time equal to one unit of time.</p> <div> <div>QASM</div> <div>Matrix</div> </div>
 <p>The Pauli Z gate is a π-rotation around the Z axis and has the property that $X \rightarrow -X$, $Z \rightarrow Z$. Also referred to as a phase-flip.</p> <div> <div>QASM</div> <div>Matrix</div> </div>	 <p>The Hadamard gate has the property that it maps $X \rightarrow Z$, and $Z \rightarrow X$. This gate is required to make superpositions.</p> <div> <div>QASM</div> <div>Matrix</div> </div>
 <p>Controlled-NOT gate: a two-qubit gate that flips the target qubit (i.e. applies Pauli X) if the control is in state 1. This gate is required to generate entanglement and is the physical two qubit gate.</p> <div> <div>QASM</div> <div>Matrix</div> </div>	 <p>The Phase gate that is \sqrt{S}, which is a $\pi/4$ rotation around the Z axis. This gate is required for universal control.</p> <div> <div>QASM</div> <div>Matrix</div> </div>
 <p>Measurement in the computational (standard) basis (Z).</p> <div> <div>QASM</div> <div>Matrix</div> </div>	 <p>Conditionally apply quantum operation</p> <div> <div>QASM</div> <div>Matrix</div> </div>
 <p>The Pauli X gate is a π-rotation around the X axis and has the property that $X \rightarrow X$, $Z \rightarrow -Z$. Also referred to as a bit-flip.</p> <div> <div>QASM</div> <div>Matrix</div> </div>	 <p>The Pauli Y gate is a π-rotation around the Y axis and has the property that $X \rightarrow -X$, $Z \rightarrow -Z$. This is both a bit-flip and a phase-flip, and satisfies $Y = XZ$.</p> <div> <div>QASM</div> <div>Matrix</div> </div>
 <p>The Phase gate that is \sqrt{Z} and has the property that it maps $X \rightarrow Y$ and $Z \rightarrow Z$. This gate extends H to make complex superpositions.</p> <div> <div>QASM</div> <div>Matrix</div> </div>	 <p>The Phase gate that is the transposed conjugate of S and has the property that it maps $X \rightarrow -Y$, and $Z \rightarrow Z$.</p> <div> <div>QASM</div> <div>Matrix</div> </div>
 <p>The Phase gate that is the transposed conjugate of T.</p> <div> <div>QASM</div> <div>Matrix</div> </div>	 <p>The barrier prevents transformations across this source line.</p> <div> <div>QASM</div> <div>Matrix</div> </div>

Quantum Algorithms

Grover's algorithm is a quantum algorithm for searching an unsorted database with N entries in $O(\sqrt{N})$ time. It was invented by Lov Grover in 1996. Suppose you are given a large list of N items. Among these items there is one item with a unique property that we wish to locate; we will call this one the winner w . Think of each item in the list as a box of a particular color. Say all items in the list are gray except the winner w , which is pink.



To find the pink box, the marked item using classical computation, one would have to check on average $N/2$ of these boxes, and in the worst case, all N of them. But on a quantum computer we can find the marked item in roughly \sqrt{N} steps with Lov Grover's amplitude amplification trick. A quadratic speed up is indeed a substantial time-saver for finding marked items in long lists.

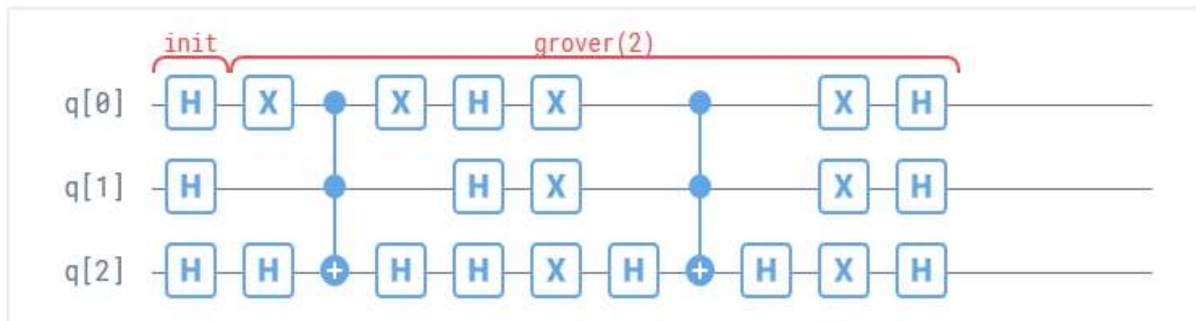
How we can provide the listed items to the quantum computer?

A common way to encode such a list is in terms of a function f which returns $f(x) = 0$ for all unmarked items x and $f(w) = 1$ for the winner. To use a quantum computer for this problem, we must provide the items in superposition to this function, so we encode the function into a unitary matrix called an **oracle**.

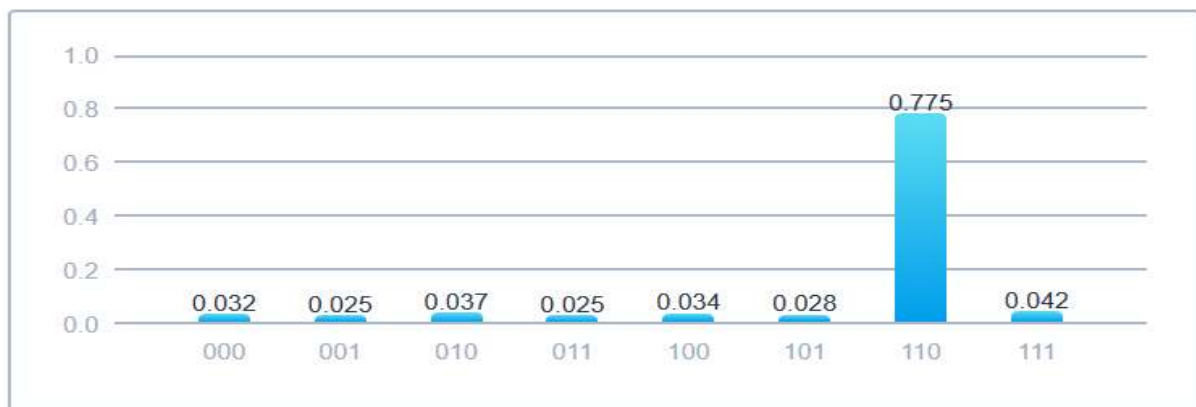
The steps of Grover's algorithm are as follows:

- Initialization of the qubits in the $|0\rangle$ state.
- Creation of a uniform superposition of all basis inputs.
- Execution of the Oracle.
- Application of Grover's diffusion operator.
- Repetitions of step 2 and 3.
- Final measurement.

Grover's algorithm logic gate circuit



Examination of the results



Note that the number of iterations in Grover's algorithm is critical. If you make too much iteration the probability of success decreases again.

Quantum Algorithms

Shor's algorithm is a algorithm for factoring a num N in $O((\log N)^3)$ time and $O(\log N)$ space name after Peter Shor's. The algorithm is significant because it implies that public key cryptography might be easily broken, given a sufficiently large quantum computer.

RSA, for example, uses a public key N which is the product of two large prime numbers. One way to crack RSA encryption is by factoring N , but with classical algorithms, factoring prime numbers becomes increasingly time consuming as N grows large; more specifically, no classical algorithm is known that can factor in time $O((\log N)^k)$ for any k . By contrast, Shor's algorithm can crack RSA in polynomial time.

It has also been extended to attack many other public key crypto systems. Like all quantum computer algorithm, Shor's algorithm is probabilistic it gives the correct answer with high probability, and the probability of failure can be decreased by repeating the algorithm.

The Shor's algorithm uses the following concepts:

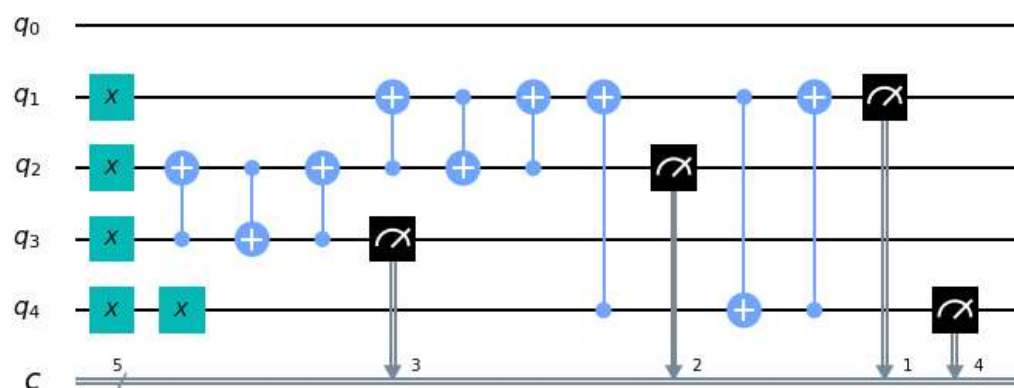
1. Modular Arithmetic
2. Quantum Parallelism
3. Quantum Fourier Transform

To Factor an odd integer N (Let's choose 15)

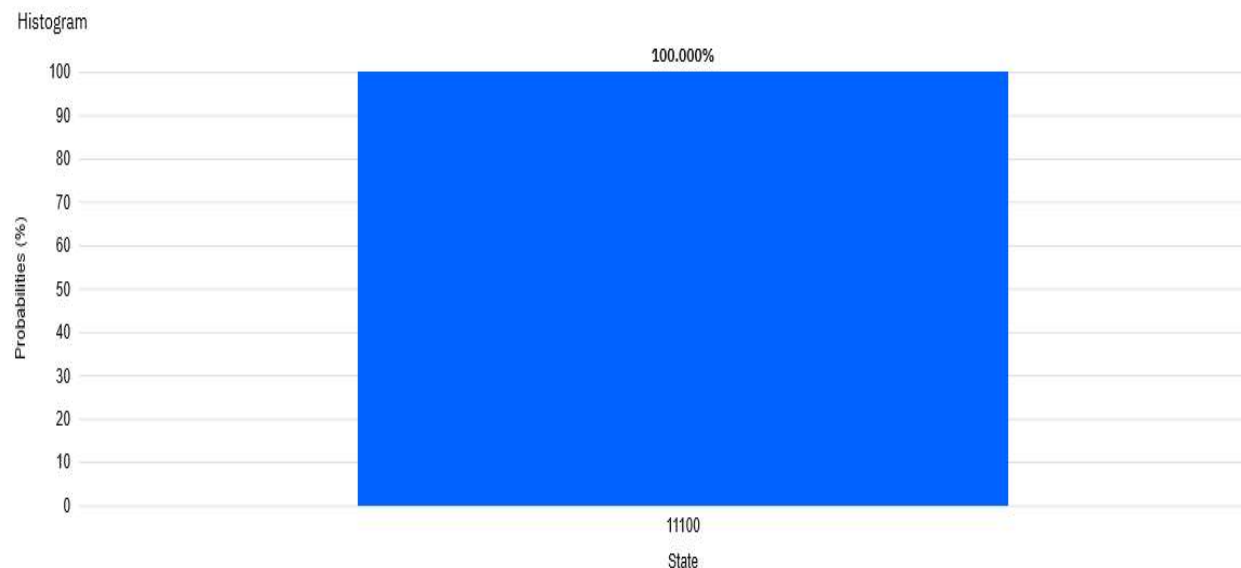
- Choose an integer q such that $n^2 < q < 2n^2$, lets pick 256.
- Choose a random integer x such that $\text{GCD}(x, N) = 1$ let's pick 7.
- Create two quantum registers (these registers must also be entangled so that the collapse of the input register corresponds to the collapse of the output register).
- Input register: must contain enough qubits to represent numbers as large as $q-1$ up to 255, so we need 8 qubits.
- Output register: must contain enough qubits to represent numbers as large as $N-1$ up to 14, so we need 4 qubits.

Shor algorithm was demonstrated in 2001 by a group at IBM, which factored 15 into 3 and 5, using a quantum computer with 7 qubits. An order of twenty to thirty runs are required on a quantum computer in the case of Shor's original algorithm, and with some of the other modifications, in the case of the modification done by David Mc Anally at the University of Queensland an order of only four to eight runs on the quantum computer is required.

Shor's Algorithm Quantum Logic Gate Circuit (Multi7x1Mod15)



Result of Above Logic Gate Circuit



Timeline of Quantum Computing

- 1981** - Quantum computing is first theorized by Paul Benioff.
- 1994** - Discovery of shor's algorithm for integer factorization.
- 1996** - David Di-Vincenzo outlines the five minimal requirements for creating a quantum computer.
- 1998** - First working 2 qubit quantum systems demonstrated.
- 2001** - First time shor's algorithm executed on quantum computer.
- 2005** - First quantum byte created by scientists at Innsbruck (Austria).
- 2007** - Superconducting qubit is designed to have reduced sensitivity to charge noise.
- 2015** - Two qubits silicon logic gate successfully developed.
- 2017** - IBM successfully tests 16-qubit quantum computer.
- 2019** - Google successfully performed a computation in 200 sec on a 53 qubit system and claimed to achieve quantum supremacy.
- 2020** - Honeywell and McAfee started to invest in Quantum computing as well as Indian Government plan to invest ₹8,000 crores in quantum computing research over five years by the country finance minister, Nirmala Sitharaman.

Quantum Computing Applications

Optimization – Improved efficiency in clearing large batches of that transaction that have varying credit, collateral and liquidity constraints.

Artificial Intelligence - AI is based on the principle of learning from experience, becoming more accurate as feedback is given, until the computer program appears to exhibit “intelligence.” This feedback is based on calculating the probabilities for many possible choices.

Weather Forecasting – A quantum computers could help to build better climate models. These models are what we build our estimates of future warming on, and help us determine what steps need to be taken now to prevent disasters.

Quantum Cryptography - It allows the completion of various cryptographic tasks. For example, it is impossible to copy data encoded in a quantum state. If one attempts to read the encoded data, the quantum state will be changed.

Molecular Modeling - Another example is precision modeling of molecular interactions, finding the optimum configurations for chemical reactions because they are quantum in nature as they form highly entangled quantum super-position states.

Financial Services - One potential application for quantum technologies is algorithmic trading the use of complex algorithms to automatically trigger share dealings based on a wide variety of market variables.

Difficulties with Quantum Computing

A Quantum Computer's power depends on more than just adding qubits. If we want to use quantum computers to solve real problems, they will need to explore a large space of quantum states. The number of qubits is important, but so is the error rate. In practical devices, the effective error rate depends on the accuracy of each operation, but also on how many operations it takes to solve a particular problem as well as how the processor performs this operation. Here we introduce a quantity called **Quantum Volume** which accounts for all these things. Think of it as a representation of the problem space these machines can explore.

Below are the problems ahead of quantum computing:-

Error correction - Given the nature of quantum computing error correction is ultra critical & even a single error in a calculation can cause the validity of the entire computation to collapse.

Interference – During the computation phase of a quantum calculation, the slightest disturbance in a quantum system causes the quantum computation to collapse, a process known as decoherence. A quantum computer must be totally isolated from all external interference during the computation phase.

Output observance - With related to the above two, retrieving output data after a quantum calculation is complete risks corrupting the data.

Future Scope

At present, quantum computers and quantum information technology remains in its pioneering stage. At this very moment obstacles are being surmounted that will provide the knowledge needed to thrust quantum computers up to their rightful position as the fastest computational machines in existence. Many governmental agencies, Military and Private Institutions have started using Quantum computing to enhance the level of personal and public security.

Error correction has made promising progress to date, nearing a point now where we may have the tools required to build a computer robust enough to adequately withstand the effects of decoherence. Quantum hardware, on the other hand, remains an emerging field, but the work done thus far suggests that it will only be a matter time before we have devices large enough to test Shor's algorithm and other quantum algorithms.

It is believed that the revolution of Quantum computing is in its initial stages and a lot of transformations are needed to make it handy. Until then the conventional computers will continue to rule over the world. But, in the coming future the demand of Quantum computers will be sure to witness a multifold increase. Thus the idea of merging quantum mechanics and computation is a remarkable venture to explore the potential of quantum computing was spectacle in itself. There is lot more to come from this end of technology. We just have to wait and watch.

Conclusion

In this paper I have reviewed the principles, algorithms, and hardware considerations for quantum computing. Several research groups are investigating qubits and quantum logic circuitry using different resources (i.e., atom, ion, electron, and photon, among others). The realization of a practical quantum computer is expected before we encounter the limit of Moore's law with respect to improvements that may be possible using the classical computer model.

Further research is needed, for example, via simulation, on quantum computers using classical computers. Such a simulator must be able to handle quantum computers that operate on a practically large number of qubits. To this end, we need to employ large-scale parallel processing methods to acquire more meaningful results within a practical time frame. By applying the methods/concepts of classical computers such as hardware abstraction to quantum computers, the research progress may be accelerated. For example, some groups proposed quantum programming languages that allow us to think of quantum computer operations in an abstract manner as we do with a classical computer. Efforts at realization for quantum computers have just begun.

Undoubtedly, we need more intensive research in a physical realization of components of quantum computers. Computer scientists/engineers will need to consider the various architectural solutions for quantum computers as well as the various new (practical) quantum algorithms to advance the state of the art for quantum computers.

References

01. <https://www.dwavesys.com/quantum-computing>
02. <https://www.ibm.com/quantum-computing/learn/what-is-quantum-computing/>
03. <https://research.google/teams/applied-science/quantum/>
04. <https://www.microsoft.com/en-us/quantum>
05. https://en.wikipedia.org/wiki/Quantum_computing
06. <https://hackernoon.com/quantum-computing-explained>
07. <https://www.freecodecamp.org/news/what-is-a-quantum-computer-explained-with-a-simple-example-b8f602035365/>
08. [https://www.slideshare.net/quantum computing](https://www.slideshare.net/quantum-computing)
09. <https://www.qubit.org>
10. <https://computer.howstuffworks.com/quantum-computer1.html>
11. https://www.youtube.com/results?search_query=quantum+computing