# Getting Entangled with Quantum Computing

**By Dr. John Millam**[1]

Quantum computing is set to be the next major technological revolution that will transform our lives. Although still in its infancy, new discoveries in this field are being heralded on a regular basis. Currently quantum computing is largely reserved for scientists, governments, and large companies, yet it is already leading to new scientific discoveries and advancements that are relevant to everyone. But the capabilities that exist now pale in comparison to what we expect will be possible once quantum computing fully matures.

So, what exactly is quantum computing? What can we expect quantum computers to be able to do that regular computers cannot? Where does this field stand right now and who are the major players? And most of all, why should regular individuals care about developments in this highly technical field? All these questions and more will be addressed below.

Before diving in, it is important to emphasize that the field of quantum computing is still in a state of flux. New discoveries are happening at a breakneck pace. What is presented here represents the state of the art as of the time of this writing in the Spring of 2026. Some of the details and conclusions provided below will soon be out of date.

## Classical versus Quantum Computers

A computer is simply a machine that can be programmed to automatically carry out sequences of arithmetic or logical operations (i.e., computation). Modern digital electronic computers can perform generic sets of operations known as computer programs, which enable computers to perform a wide range of tasks.

In our modern culture, we are surrounded by and daily interact with computers. All these computers (e.g., desktop computers, tablets, smartphones, and smart watches) are classified as "classical" computers, where classical simply refers to a pre-quantum understanding of the world. Within the classical framework, all information is represented in the form of bits (binary digits). A bit is the most basic unit of information representing a logical state with one of two possible values. Or more simply, a bit must be either 1 or 0.

Quantum computers can perform computational operations similar to classical computers, but they are special because they leverage the principles of quantum mechanics (e.g., superposition and quantum entanglement) to potentially solve specific problems that are beyond the capabilities of classical computers. Another important difference is that quantum computers operate on qubits (quantum bits). Qubits, unlike classical bits, can represent multiple states simultaneously due to the quantum phenomenon of superposition. This allows quantum computers to perform certain calculations with unprecedented speed and efficiency.

One of the main goals for those developing quantum computers is to achieve quantum supremacy (or quantum advantage) for specific important tasks. Quantum supremacy refers to the point at which a quantum computer is not just faster than classical computers but can solve a

[1] Ph.D. in Theoretical Chemistry from Rice University. 

problem that is infeasible for classical computers to solve in a reasonable amount of time. However, quantum computers are poorly suited for common tasks, such as balancing your checkbook, and so are primarily intended to tackle specific types of problems in which they excel. Therefore, quantum computers will complement, rather than replace, conventional computers.

Quantum computing is already benefiting society, and this impact will surely increase dramatically as this technology matures. It is expected to be a transformative technology that affects every area of our lives, in the same way that cars (1910s), personal computers (1975-1985), cell phones (1990s), internet (1990s), GPS (2000s), smartphones (mid-2000s), and AI (2020s) have already done. Quantum computing will provide us with powerful new tools to tackle some of society's biggest challenges, such drug discovery, lifespan enhancement, nuclear fusion, and climate change. While quantum computing will improve our lives in a myriad of ways, it will create some new problems as well, such breaking current cybersecurity encryption protocols that protect critical information transmitted over the internet.

## A Brief History of Classical Computing

Before exploring quantum computing, let us begin by reviewing the history of classical computing. One of the earliest computing devices is the abacus, a manually operated instrument for basic arithmetic calculations. Another early example is Antikythera mechanism discovered in 1901 near Greece. Constructed around the 2nd century BC, this device could be used to track various astronomical events and was very advanced for its time. A more modern manual calculating tool is the slide rule. Developed in the 17th century, it would play an important role in science and engineering until the 1970s.

With the flourishing of modern science and engineering in the sixteenth and seventeenth centuries, there was a corresponding need to perform ever more complicated and tedious mathematical calculations to test ideas. The people who performed this valuable service were called "computers."[2] Because such work was time-consuming and error-prone, there was a growing need for mechanical devices to help automate such tasks. One early mechanical calculator was developed by Blaise Pascal and Wilhelm Schickard in 1642. Another example is an adding machine developed in 1784 by J. H. Müller, an engineer in the Hessian army.

A major conceptual breakthrough in computing came from British inventor and polymath, Charles Babbage. His inspiration to develop complex mechanical computational devices came from his work checking astronomical tables for astronomer John Herschel. His experience was so tedious that Babbage is said to have cried out, "I wish to God these calculations had been executed by steam!"[3] With this as his motivation, he developed the difference engine in the 1820s. His invention was designed to quickly tabulate polynomial functions for scientific use.

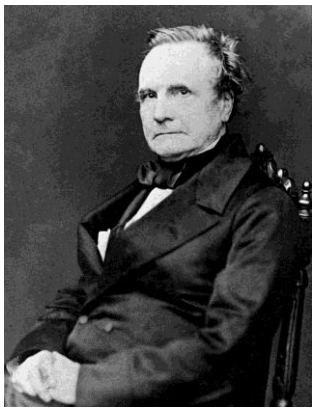

**Figure 1: Charles Babbage.** Credit: Wikipedia Commons.

Although useful, the difference engine was limited to calculating the specific types of values for which it was designed. Babbage then began developing ideas for a generalized version of the difference engine that would be capable of performing any desired sequence of mathematical operations. His proposed analytical engine would have been equipped with a system of punch cards to allow the user to

[2] Today, the term computer is synonymous with modern electronic computers. Prior to their introduction in the 1960s, computer referred to human beings performing mathematical calculations.

[3] Brian Clegg, *Quantum Computing*, p. 1.

input instructions to the device. If it had been built, it would have been the first digital mechanical general-purpose computer. Unfortunately for him, the technology to create all the needed components with the necessary precision did not exist in his day. Nevertheless, Babbage's ideas would pave the way for the development of true computers.

In 1840, Babbage gave a series of talks in the city of Tunis describing his proposed analytical engine. Two years later, Luigi Federico Menabrea wrote up Babbage's talk but in a minor journal in French. There it would have languished in obscurity except Ada King, the Countess of Lovelace in 1843 translated and extended Menabrea's account of Babbage's talk, making it widely available in English. Ada's work introduced the concept of an algorithm, a structured set of instructions for accomplishing a specific task. With the correct algorithm, the analytical engine could be programed to calculate any computable quantity.

While Babbage's work remained largely theoretical, the next major development in computation came in the form of tabulating machines. One major need for them came from the US census that is performed every decade. As censuses collected more and more data, it was taking longer and longer to tabulate the results. The 1880 census took 8 years to compile, so it was expected that without significant automation subsequent censuses might not be completed prior to the beginning of the census that followed.

Inventor Herman Hollerith came to the rescue by developing tabulating machines that were used for the 1890 census. Data was encoded onto punch cards that could then be read by the machines. These cards were chosen to be 3 1/4" by 7 3/8", corresponding to the size of a bank note. (Punch cards would be used for inputting computer programs until the 1960s and 70s before being replaced by magnetic storage.) Hollerith's devices initiated a data processing revolution. However, these machines were limited to sorting and counting, so were not true computers. As a side note, the Hollerith Tabulating Machine Company joined three other companies to become International Business Machines (IBM), a major leader in the computer industry.

Two different devices are credited as being the first true programable computers. First is Colossus that was developed at Bletchley Park in 1943 for the purpose of decoding the German Enigma machine codes during WWII. The second is ENIAC (Electronic Numerical Integrator and Computer), which was developed for artillery calculations in 1945. Both were electronic devices that used vacuum tubes for signal switching. These two computers fulfilled Babbage's dream of constructing an analytical engine.

The British mathematician and logician Alan Turing is the first of two figures who would fundamentally develop the theoretical underpinning of modern computers and thereby found the field of computer science. Among Turing's many contributions is his work building Colossus and then in 1948 he helped construct Manchester Baby, the first stored program electronic computer.

The theoretical basis for these developments came from his seminal 1936 paper, "On Computable Numbers" written to refute a math conjecture from the 1920s.[4] His proof was built around a mathematical model of computation in the form of an imaginary device that would be capable of undertaking any set of mathematical operations. In computer science, this hypothetical device is called a Turing machine in his honor.

**Figure 2: Alan Turing.**
Credit: Wikipedia Commons.

[4] The full title of the paper is "On Computable Numbers, with an Application to the Entscheidungsproblem." In mathematics and computer science, the Entscheidungsproblem (German for "decision problem") is a mathematical conjecture proposed by David Hilbert and Wilhelm Ackermann in 1928. Their conjecture asked whether it was possible to have a mechanism that would be capable of deciding whether any mathematical statement was universally valid given a set of axioms. Such an algorithm was proven to be impossible independently by both Alonzo Church and Alan Turing in 1936.

A Turing machine consists of three principal components. The first is an infinitely long tape that is divided into sections called squares, each of which could be blank or contain a symbol The second part is a read/write head, a device that can move up and down the tape, one square at a time, either reading a symbol, writing a symbol, or erasing a symbol. Finally, there is a controller with a set of rules defining the device's behavior.

The tape would contain sequence of instructions defining the set of mathematical operations to be executed plus any necessary data. By controlling the content of the tape, the Turing machine could be instructed to perform the desired operations without having to manually configure the device itself. The content of the tape thus serves the same function as a modern computer program. The controller, in contrast, would follow a general set of rules instructing the head how to behave depending on what has been read or written and tell it how to start and finish the run. In this sense, these controller rules are what we would now call the operating system.

The American mathematician, physicist, and engineer John von Neumann was the second key figure to make a major contribution to computer science. In addition to his work on the ENIAC computer, his primary contribution was to devise a practical physical architecture for Turing's stored program concept in 1945. The von Neumann architecture, illustrated in **Figure 3**, is used by most computers today. He also replaced Turing's tape with generalized input and output devices. The only major component not described by von Neumann is the system's data bus, namely the set of wires that allow data to flow from one part of the computer to another, or to transfer data to and from the outside world. With von Neumann's contributions, we can see modern computers taking shape.

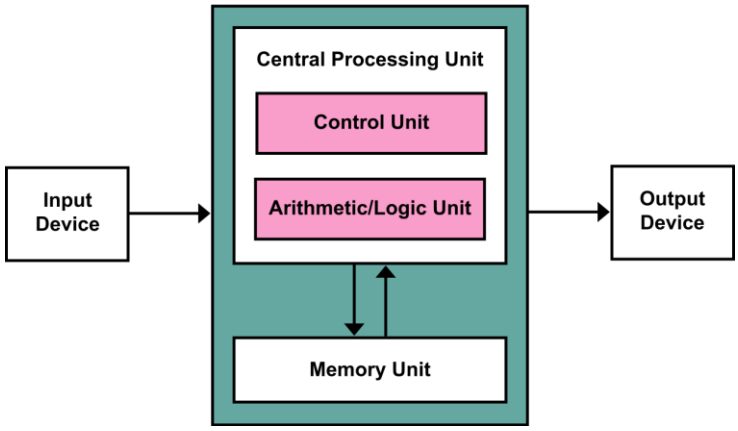


**Figure 3: Von Neumann computer architecture diagram.**
Credit: Wikipedia Commons.

## Computer Evolution

Relying on fickle vacuum tubes, early electronic computers were bulky behemoths and very difficult to maintain. A major step toward miniaturization came with the development of the transistor in 1947 by John Bardeen, Walter Brattain, and William Shockley. Transistors are semiconductor devices that amplify or switch electronic signals that can be made much smaller than vacuum tubes, a major step in shrinking the size of computer components. Further miniaturization and standardization came with the development of the integrated circuit or microchip in 1958-1959. Microchips consist of a compact assembly of electronic circuits formed from various electronic components (e.g., transistors, resistors, and capacitors) and their interconnections, typically etched onto a silicon wafer. This design allows mass production and high reliability.

The incorporation of integrated circuits into computers made them smaller, cheaper, and more reliable. Around 1975-1985, we observe the birth of personal computers, i.e.,

microcomputers that could be owned and operated by individuals. By 1996, personal computers were commonplace at home and in the office. Today, computers are just about everywhere you go. In addition to personal computers, microchips are present in virtually all electronic devices, ranging from cars to home appliances.

In 1965, engineer and businessman Gordon Moore observed that the number of transistors on a single microchip was increasing at an accelerating pace. From this trend, he predicted that the number of transistors in an integrated circuit would double approximately every year for the next decade. In 1975, he revised this trend to doubling every two years. Since then, this bold prediction has continued to hold even fifty years later, becoming popularly known as Moore's law.[5] This trend has significant implications for computers because increasing the number of transistors on a chip directly enhances processing power, operational efficiency, and capacity to perform complex functions. Consequently, the computing power of a single integrated circuit today is roughly 2 billion times what it was in 1960 making today's computers exceptionally fast.

Despite the longevity of Moore's law, it is widely understood that it cannot continue indefinitely. One fundamental challenge to adding more transistors is that they are constructed from atoms, and this makes a single atom the smallest possible computer component. And current technology is already close to hitting this limit. A second major challenge comes from the principles of quantum mechanics. As the components become smaller, quantum effects become more pronounced. In particular, electrons will be able to quantum tunnel to a different location thus randomly switching the state between 0 and 1 or preventing a logic gate from functioning properly. This effect can render classical computing useless.

To proceed further, it is necessary to examine the historical origin of quantum mechanics and explore some of its revolutionary features.

## The Rise of Quantum Mechanics

Long before quantum mechanics, Isaac Newton published his three laws of motion and universal law of gravitation in 1687. These equations represent the first unified mathematical formulation of the laws of physics and are collectively known as Newtonian mechanics. His equations provided scientists with powerful new tools for understanding natural phenomena.

Over the next two centuries, Newton's work helped demystify the world by showing how various phenomena are governed by natural forces and predictable by mathematical calculations. Philosophically, this gave rise to the notion of a clockwork universe, one where everything ran with the predictability and precision of a clock. According to Newtonian mechanics, all behavior is strictly deterministic, that is, every event is entirely determined by previous events leaving no room for choice or freewill. Within this framework, one could theoretically predict a system's past and future with complete accuracy.

A series of discoveries at the beginning of twentieth century exposed critical flaws in Newtonian mechanics. As a result, two more comprehensive laws were created to tackle these problems. The first was quantum mechanics that describes how particles behave at the atomic scale. The second was general relativity, which is needed to describe nature at the cosmic scale and to explain gravity. Together, these new laws are classified as modern physics in contrast to Newtonian mechanics which is now frequently referred to as classical physics.

Of the two theories, we will focus exclusively on quantum mechanics because it is essential for understanding quantum computing. Quantum theory was developed in the early years of the twentieth century with contributions from many different individuals over several decades. A complete history is too long to include here but those details can be found elsewhere.[6] For our

[5] Moore's law is an observation and projection of a historical trend, rather than a true law.

[6] John Millam, "Understanding Quantum Mechanics" (part 1, 2, 3, 4, and 5). Alex Montwill and Ann Breslin, *The*

purposes here, we will focus on the essential individuals and discoveries leading to our current understanding of quantum mechanics.

**Historical Development of Quantum Mechanics**

The first hint heralding the need for quantum mechanics came from an unexpected direction. One of the big unresolved science questions at the end of the nineteenth century was an accurate description of the spectral distribution of black-body radiation, that is, the light (electromagnetic radiation) given off by an object due to its temperature. A typical example is a metal bar glowing red when heated sufficiently. Incandescent light bulbs and the sun giving off white light are two more well-known examples.

Scientists had been working to develop a mathematical formula for the intensity of black-body radiation given off at different frequencies as a function of the object's temperature. Attempts to model this using classical (Newtonian) mechanics, however, failed miserably. In particular, the Rayleigh–Jeans law predicted that even cool bodies should be giving off copious amounts of ultraviolet and x-ray radiation. This prediction was so obviously wrong that it was dubbed the ultraviolet catastrophe.

In 1900, Max Planck published the correct formula for black-body radiation. His critical insight was to look for a way to prevent the growth of high-frequency (short wavelength) light that gave rise to the ultraviolet catastrophe. He accomplished this by introducing the Planck relationship that makes the energy of the emitted photons (E) proportional to its frequency (ν).

$$E = h\nu, \qquad \textbf{Equation 1}$$

where E is the energy, h is Planck's constant, and ν is the frequency of light.

Although Planck introduced this formula as little more than a mathematical trick to get the correct result, the Planck relationship would prove to be his enduring legacy. The associated constant (h) is named Planck's constant in his honor and is now recognized as the universal constant governing quantum behavior.

To complete his derivation of the black-body radiation formula, Planck had to introduce two additional rules:

- Vibrational motion of the atoms was quantized, that is, vibrational motion can only occur at very specific energies, rather than all possible values being allowed.
- These vibrations do not radiate energy continuously, but only in specific "jumps" or quanta. Light is emitted when the vibration jumps from one energy level to another.

Like the Planck relationship, these postulates were very radical at that time but would serve as critical first steps in the development of quantum mechanics.

Planck's insights proved useful in describing another intriguing phenomenon—the photoelectric effect. In the photoelectric effect, shining ultraviolet light on metals causes them to eject electrons from their surface. Albert Einstein used the Planck relationship to explain this phenomenon in 1905. While Planck quantized vibrational motion, Einstein went one step further by quantizing light. In other words, Einstein treated light as a particle rather than a wave and having an energy given by the Planck relationship. He called these particles of light "quanta" but today we call them photons.

The next major development involved attempts to describe the structure of atoms. Previously in 1911, Ernest Ruthford proposed that atoms consisted of positively-charged nuclei surrounded by halos of negatively-charged electrons. The one major deficiency of the Rutheford

---

*Quantum Adventure: Does God Play Dice?* Imperial College Press, 2012. James Kakalios, *The Amazing Story of Quantum Mechanics*, Gotham Books, New York, NY 2010. George Musser, *Spooky Action at a Distance*, Scientific American/Farrar, Straus and Giroux, New York, NY 2015.

model of the atom was that atoms should be unstable and instantly collapse. This is because Maxwell's equations of electromagnetism dictate that the negatively-charged electrons should lose energy by emitting electromagnetic energy and spiral into the nucleus. Everything would then simply collapse into a black hole. Obviously, this does not happen or else we would not exist.

Resolving the structure of the atom required new physics. This was supplied by Niels Bohr who explained the electronic structure of the hydrogen atom based on quantum principles. His model was based on three postulates:

- Electrons in atoms are only allowed in fixed circular orbits and jump directly from one allowed orbit to another.
- The orbital angular momentum of an electron is quantized, i.e., angular momentum = $nh/2\pi$, where n is a positive integer.
- The frequency of light emitted or absorbed is given by the difference in energy levels, i.e., $\Delta E = h\nu$.

Bohr's work represented a major success for quantum theory. His work also explained the spectrum of hydrogen atoms by successfully predicting each of the emitted frequencies of light.

Earlier, Einstein showed that light could be treated like a particle in the photoelectric effect. This raised difficult questions because it ran counter to the common understanding that light behaves like a classical wave. For example, treating light as a wave is fundamental for the field of optics. In 1923, Arthur Compton provided a second example of this particle-like behavior for light. In his Compton scattering experiments, when light scattered off electrons it behaved like classical particles rather than waves. With Compton's contribution, this unexpected behavior of light could no longer be ignored or treated as an exception.

How can light behave like a wave under most circumstances but act like a particle in others? Louis de Broglie resolved this tension by proposing wave-particle duality in 1924. In short, he argued that wave behavior and particle behavior are simply two sides of the same coin rather than separate phenomena. He then took this idea one step further by recognizing this duality was a two-way street. If light waves could be treated like particles, then fundamental particles could also be treated as waves. The logical consequence of his work is stunning. An electron, previously viewed as a point-like object, could now be thought of as a matter wave. Most scientists at the time found this hard to accept.

De Broglie's work provided the critical missing piece needed to inspire Erwin Schrödinger in 1926 to develop an equation describing the behavior and evolution of quantum systems. Known as Schrödinger's equation, it was a monumental achievement because it provided a general model that could be applied to any system. Schrödinger's equation fully explained the previous *ad hoc* rules proposed by Planck, Einstein, and Bohr that had been introduced to match specific observations.

While Schrödinger's equation resolved many questions, it resulted in many more. Solving this equation for a given system yields its wavefunction. The wavefunction then provides a mathematical description of what can be known about the quantum state. One of the greatest difficulties for the pioneers of quantum mechanics was understanding the meaning of the wavefunction, because it had no correspondence in classical mechanics to act as a guide.

Initially, Schrödinger understood the wavefunction as a matter wave describing the particle itself. More study, however, showed that this interpretation simply did not work. Max Born in 1926 laid out the now-accepted interpretation—the square of the wavefunction gives the probability density. So, instead of describing the particle's location, the wavefunction simply provides a probability of where we might find the particle.

The probabilistic behavior identified by Born is an inherent feature of quantum mechanics. This revelation disturbed many physicists at the time because it meant that quantum behavior had

an intrinsic random and unpredictable element in contrast to the strict deterministic behavior of classical mechanics. Einstein was so bothered by this notion that he famously quipped that God "does not play dice."[7] On a philosophical level, this probabilistic behavior undercuts the argument against free will coming from classical determinism.

Another unexpected consequence of quantum mechanics was discovered by Werner Heisenberg in 1927. Based on Schrödinger's equation, he developed his uncertainty principle that applies to all quantum systems. It states that certain pairs of physical properties, like position and momentum, cannot be simultaneously measured with arbitrary precision. The more accurately you measure one property, the less precisely you know the other. In summary, the uncertainty principle demonstrates that contrary to classical physics, there is a fundamental limit on how accurately we can measure physical properties.

Two years after Schrödinger published his equation, Paul Dirac extended it to satisfy special relativity. Solutions to the Dirac equation predicted two new phenomena:

- **Spin.** Spin is a form of angular momentum intrinsic to all elementary particles. It is a fundamental property like mass and charge.
- **Antimatter.** According to the Dirac equation, every fundamental particle has an antimatter twin that is identical except the twin has the opposite charge. For example, a negatively charged electron has a positively charged antimatter partner called a positron.

The predicted existence of antimatter particles was a major surprise for physicists at the time because no one had been expecting it. In 1931, positrons (antielectrons) were experimentally discovered thus confirming the existence of antimatter. Dirac's prediction of antimatter was a major triumph for theoretical physics.

Additional details can be found in **Appendix: Key Founders of Quantum Mechanics**.

**Quantum Weirdness**

Quantum mechanics is not merely a refinement of classical (Newtonian) mechanics; it represents a major paradigm shift in how we understand the world. The behavior of subatomic world governed by quantum mechanics is strikingly different when compared to what we experience in our daily lives. At the quantum level, objects behave in unfamiliar, counter-intuitive, and bizarre ways with a whimsical quality reminiscent of *Alice in Wonderland*. To begin, let us quickly summarize some of the strange quantum behaviors we have already encountered:

- **Wave-particle duality.** Light can behave like a particle, and particles can behave like waves.
- **Reality is quantized.** Measurement of a given property is restricted to specific values.
- **Uncertainty principle.** Some properties cannot be simultaneously known to complete precision and are not fully defined until measured.
- **Quantum outcomes are probabilistic.** Unlike classical mechanics, quantum outcomes cannot be perfectly predicted. Instead, quantum mechanics can only predict the probability of an event occurring.
- **Measuring inherently changes the system.** In quantum mechanics, it is impossible to measure the system without altering it.

If these five general behaviors are not weird enough, there are some specific behaviors that completely rocked the world when they were discovered. (See **Appendix: Key Quantum**

[7] In December 1926, Albert Einstein wrote to Max Born that "[quantum] theory produces a good deal but hardly brings us closer to the secret of the Old One. I am at all events convinced that He does not play dice."

**Concepts**.) Four of these very non-classical behaviors are foundational for understanding what makes quantum computers so useful. Let us briefly overview these special phenomena:

- **Quantum Tunneling.** A particle may cross a barrier even when classically it lacks enough energy to do so. (Link to fuller discussion of quantum tunneling.)
- **Superposition.** According to this principle, a quantum particle or system can represent multiple values simultaneously. (Link to fuller discussion of superposition.)
- **Quantum Entanglement.** When entangled, multiple quantum particles become correlated in ways that classical probability does not allow. In this state, affecting one qubit can instantly influence the state of another, regardless of the distance between them. (Link to fuller discussion of quantum entanglement.)
- **Interference.** This phenomenon occurs when quantum states interact, leading to the amplification or cancellation of probabilities. (Link to fuller discussion of interference.)

A fifth behavior, decoherence, is not weird yet is critical for understanding quantum computing:

- **Decoherence.** This is the process by which quantum particles lose their quantum state and transition into classical states due to interactions with their environment.

Because these quantum behaviors are what grant quantum computers their computational power, let us explore each of these behaviors in more depth and examine their implications.

**Quantum Tunneling**

After the publication of Schrödinger's equation, various groups began finding solutions for very simple cases. In 1927, Friedrich Hund and Lothar Nordheim independently found puzzling cases where a particle could get over a barrier despite lacking sufficient kinetic energy to do so. This surprising result is like walking at a wall and finding yourself on the other side. This behavior is impossible according to classical (Newtonian) mechanics and everyday experience, yet quantum mechanics predicted a finite chance of it happening. Scientist refer to this behavior as quantum tunneling because the particle seems to tunnel through the barrier rather than going over it.

Quantum tunneling was first shown to occur in alpha decay, a form of nuclear fission, in 1928. Elements undergoing alpha decay emit alpha particles (equivalent to Helium nuclei) even though the particles do not have enough energy to overcome the Coulomb barrier in order to escape.[8] In addition to fission, quantum tunneling finds application in a wide variety of unexpected places.

Many components of modern classical computers make use of quantum tunneling, e.g., flash memory, cold emission, tunnel junction, quantum-dot cellular automata, tunnel diode, and tunnel field-effect transistors. An unfortunate consequence of tunneling in classical computers is that it causes a significant current leakage in computer microchips resulting in substantial power loss and heating effects that can cripple such devices. In this way, tunneling establishes an effective lower bound on how small microelectronic device elements can be and still function effectively.

**Implications for quantum computing.** Quantum tunneling is essential for manipulating qubits. In superconducting circuits, electrons can tunnel through barriers, enabling qubits to occupy multiple states at once, a property that is crucial for quantum computation. Tunneling is also utilized in the design of quantum gates, which are the building blocks of quantum circuits. These gates manipulate the quantum states of particles, allowing for complex computations that are not feasible with classical computers.

[8] The Coulomb barrier is the electromagnetic repulsion between the positively charged nucleus and the alpha particle.

**Superposition and Schrödinger's Cat.**

Solving Schrödinger's equation for a given system yields a set of allowed outcomes. We will call them "pure" solutions because they correspond to the values that can potentially be measured.[9] However, when the system is not being measured, it is free to evolve to a more general solution corresponding to a superposition (a complex mixture) of the pure states.

To illustrate this situation, let us consider a "quantum coin" representing the simplest possible quantum system.[10] When we flip an ordinary coin, it will land either heads or tails. In the same way, a quantum coin has two pure states: heads and tails. When measuring a quantum coin, only these two possible outcomes are observed. Outside of measurement, however, the quantum coin can exist in a continuum of superposition states. For example, the quantum coin could be in the superposition state of 50% heads and 50% tails. Or it could be 70% heads and 30% tails. And so on. Given that we never observe this behavior in real coins, how are we to understand the physical meaning of such a state?

To help highlight the absurdity of the situation, Erwin Schrödinger proposed his Schrödinger's cat paradox thought experiment in 1935. The setup is fiendishly simple (see **Figure 4**). Start by placing a cat (A) in a sealed box with a radioactive atom (B) having a 50% chance of decaying during the period of study. If the atom decays, it will trigger a Geiger counter (C) that would cause a hammer (D) to smash a flask of poison (E) and kill the cat. But if the atom does not decay, the cat remains alive. According to the Copenhagen interpretation of quantum mechanics, the cat ends up in a superposition state of 50% alive and 50% dead until we look in the box to observe it. How could a cat be alive and dead at the same time?

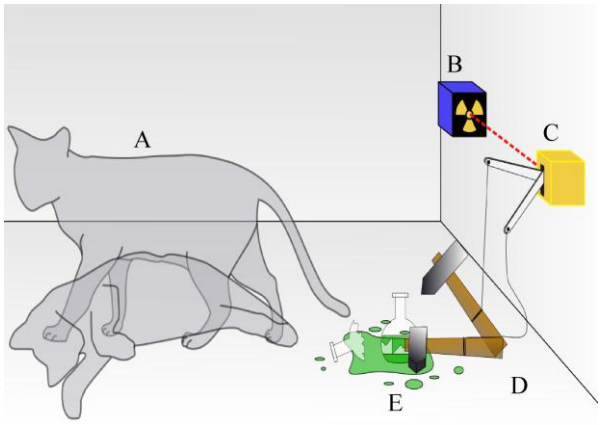


**Figure 4: Schrödinger's cat experiment.**
Credit: Wikipedia commons.

Superposition is only known to occur in the quantum realm, so we cannot directly observe this phenomenon. When a particle in a superposition state is measured, its wavefunction somehow "collapses" to yield a pure state. But how and when does that happen? This conundrum is known as the measurement problem which remains unresolved. While early physicists struggled with the notion of superposition, it has been consistently demonstrated to occur and is now widely embraced.

**Implications for quantum computing.** Superposition plays a central role in allowing quantum computers to perform many tasks faster than possible with classical computers. With classical computers, you can only test one binary value at a time but with quantum computers exploiting superposition, you can potentially test all the combinations at the same time. The more qubits used the greater the potential speed up.

---

[9] Math note. Solving Schrödinger's equation yields a set of eigensvalue/eigenvector pairs. The eigenvalues represent the values for the quantity that you can physically measure, and the eigenvectors are the wavefunctions associated with those values. In this paper, a "pure" state is simply a particular eigenvalue/eigenvector pair. Because Schrödinger's equation is linear, any linear combination of these eigenstates is also a solution to the equation. These are superposition states and are considered indeterminate because they do not correspond to a specific measurable outcome.

[10] A quantum coin is a good stand in for the spin of an electron or the polarization state of a photon. In both cases, there are only two possible measurable states (that is, pure states).

**Quantum Entanglement and the EPR Experiments**

Quantum entanglement is by far the most mysterious yet most intriguing aspect of quantum mechanics. Early physicists realized based on Schrödinger's equation that when particles interact in certain ways, their properties are no longer fully independent. When one of the particles is then measured, its entangled partners are simultaneously affected regardless of the distance between them.

Consider two electrons sharing the same orbital in an atom, such as the two electrons in a helium atom. Because of the Pauli exclusion principle, two electrons occupying the same orbital must have opposite spins. That is, one electron must be spin up, and the other spin down. This seemingly straightforward situation is complicated by the fact that electrons are indistinguishable particles. Indistinguishability means that one cannot identify which of the electrons is spin up because that would allow you to distinguish between them. Instead, the pair of electrons are in a superposition state that preserves their indistinguishability. Labeling the electrons as A and B respectively, we can describe the entangled state as consisting of 50% of the state with A as spin up and B as spin down and 50% of the state with A as spin down and B as spin up. These electrons are entangled in the sense that neither particle has a well-defined spin. Instead, we can only know the total spin of the combined pair. However, if we were to measure the spin of one of the electrons, it would be forced to take on a definite state—either spin up or spin down. Once the spin of one is known, the other must now possess the opposite spin.

To illustrate this, let us consider a pair of electrons entangled as previously described, but not associated with an atom (see **Figure 5)**.[11]

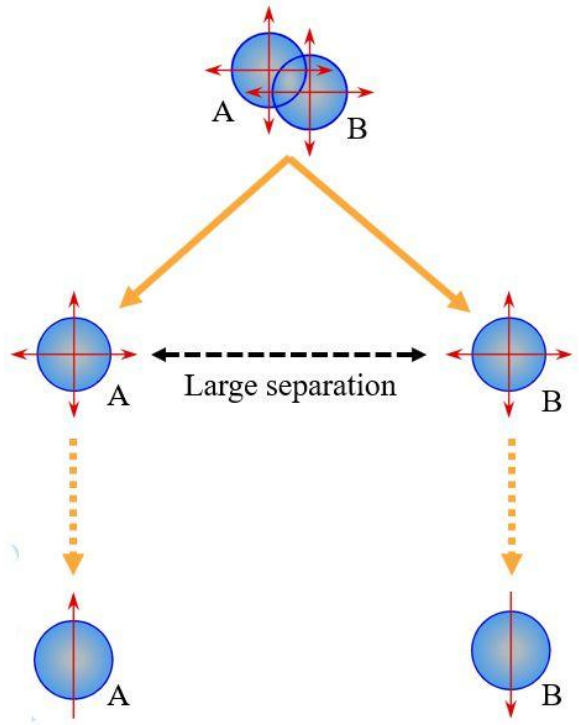


**Figure 5: Effect of measurement on entanglement.**
Credit: John Millam based on footnote 11.

1) Take two electrons entangled to have opposite spins.

2) Now, spatially separate them until they are very far apart. They remain entangled even though they can no longer interact with each other.

3) If electron A is measured to be spin up, then electron B must be spin down. The change in B occurs instantly.

The big mystery of quantum entanglement is explaining how measuring A can instantly affect electron B despite the distance between them. If the electrons were close to each other, then one could imagine a hypothetical force transmitting the effect from one to the other. This would be like gluing the faces of two coins together, so they are forcefully constrained, such that if one

[11] Summarized from https://cosmosmagazine.com/physics/einstein-bohr-and-the-origins-of-entanglement/.

lands heads, the other must be tails and vice versa. While some unknown force could potentially explain the behavior for cases when the particles are located close together, it cannot explain the case when they are far apart. Effects from quantum entanglement are instantaneous, regardless of distance, whereas known forces cannot act faster than the speed of light.

This phenomenon deeply troubled Albert Einstein because it *seemed* to allow one particle to influence its partner faster than the speed of light in violation of his theory of relativity. He famously dismissed entanglement as "spooky action at a distance."[12] To express his concern, Einstein and his research students, Boris Podolsky and Nathan Rosen, published a paper in 1935.[13] The EPR paper (based on the author's initials) presented a novel thought experiment designed to highlight their objections to the idea of entanglement (although the term entanglement would not coined until two years later). The paper concluded that quantum mechanics must be incomplete and suggested local hidden variables as an alternative to quantum entanglement.

The EPR paradox thought experiment could not be experimentally carried out at that time, so the validity of quantum entanglement remained purely theoretical until John Stewart Bell in 1964 developed a mathematical criterion for testing the validity of local hidden variables. Bell's theorem was a mathematical inequality that placed an upper limit on the degree of correlation that local hidden variables could produce. Experiments in 1970s and 80s ultimately vindicated quantum entanglement. Entanglement is now widely accepted by scientists.

**Implications for quantum computing.** Entanglement plays a central role in enabling quantum computers to perform complex calculations efficiently. It allows qubits to represent and process multiple values simultaneously, increasing computational parallelism allowing quantum computers to outperform their classical counterparts for certain tasks. Entangled qubits are also used to detect and correct errors in quantum systems, maintaining coherence over longer periods. Multi-qubit quantum gates depend on entanglement to manipulate data across qubits. In essence, without entanglement, a quantum computer would be little more than a probabilistic classical computer. It is entanglement that allows qubits to function together as a system, unlocking the true power of quantum computation.

**Quantum Interference**

Imagine visiting a lake on a calm day where the water is as smooth as glass. Now imagine throwing a rock into the water and watching the undulating waves radiate outward from the impact site. This water wave is a good visual representation of the quantum wavefunction. One key aspect of waves is that the crests of the waves extend above the general water level and their troughs go below. Mathematically, we can view the crests as representing a positive value and the troughs as negative.

Revisiting our imaginary scenario, consider the case where we simultaneously toss two rocks into the water at adjacent points. The resulting waves will then overlap and interfere with each other. In places where two peaks or two troughs meet, they add together reinforcing each other. This is known as *constructive interference*. In contrast, when a peak and a trough meet, they partially or completely cancel each other. We call this situation *destructive interference*.

---

[12] In a letter to Max Born in 1947 Einstein said of the statistical approach to quantum mechanics, which he attributed to Born, "I cannot seriously believe in it because the theory cannot be reconciled with the idea that physics should represent a reality in time and space, free from spooky action at a distance" (the actual phrase used by Einstein was German, "*spukhafte Fernwirkung*." Spooky, or ghostly, is a reasonable translation, although "spooky" was not in common usage in English in 1947).

[13] Albert Einstein, Boris Podolsky, and Nathan Rosen, "Can Quantum-Mechanical Description of Physical Reality be Considered Complete?" *Phys. Rev.* 47, 777-780 (May 15, 1935).

According to quantum mechanics, the square of a particle's wavefunction gives the probability of finding it at a given location. That means constructive interference increases the probability of the particle being found in those areas while destructive interference reduces it.

**Implications for quantum computing.** In quantum computing, interference is harnessed to manipulate and control quantum states, particularly qubits. By applying quantum gates that create superpositions of qubits and controlling the relative phases of these states, quantum algorithms can leverage interference to amplify desired outcomes and suppress unwanted ones. This capability is essential for achieving significant speedups in computation compared to classical algorithms. Interference also plays a role in quantum error correction, which is vital for protecting quantum information from decoherence and other errors. By measuring interference patterns among qubits, quantum computers can detect and correct errors that may occur during computation.

### Quantum Decoherence

One of the biggest obstacles to developing effective quantum computers is the problem of decoherence. Quantum computing requires getting qubits in complex quantum states involving superposition and entanglement. The problem is that these states can be very fragile. Interactions with the environment can cause qubits to decohere, that is, to lose their quantum characteristics and behave in a more classical manner. Decoherence renders the qubit useless for computation leading to errors in the results.

Coherence time is the length of time a qubit retains its superposition state before decohering. The longer this time, the more useful computation can be done. Some qubit types are naturally more resistant to decoherence and have longer coherence times, but all qubit types must address this challenge. Understanding and mitigating decoherence is essential for reliable quantum computing. One common way to reduce (but not completely eliminate) decoherence is to isolate the quantum system from its environment using very cold temperatures (around 4 K or less) and/or maintaining a hard vacuum. (For perspective 4 K is equivalent to -269 ºC.) To deal with any remaining errors, quantum error correction must be applied as will be discussed later in this paper.

## From Quantum Mechanics to Quantum Computing

The theory of quantum mechanics was largely complete by 1930 while the first true electronic computers appeared in the mid-1940s. The first inkling of how to marry these two very different technologies came from theoretical physicist Richard Feynman. In his 1959 lecture, "There's Plenty of Room at the Bottom", Feynman predicted that people would eventually be able to manipulate matter at the atomic scale. With such technology we could inscribe the entire contents of the *Encyclopedia Britanica* on the head of a pin. What Feynman envisioned in his talk is being accomplished today within the emerging field of nanotechnology. These ideas are also important to many developing fields, including miniaturizing transistors for computer chips.

In his presentation, Feynman recognized that as we miniaturize components close to the atomic scale, we need to consider new possibilities granted by quantum mechanics. Instead of thinking of a single atom as the smallest possible object, he suggested we could manipulate atomic energy levels or use the quantized spin of electrons. Thus, he foresaw the outlines of what would become qubits for quantum computing. Today, much of Feynman's dream has been accomplished setting the stage for the birth of quantum computers.

### Three Types of Quantum Computers

When we hear the words quantum computer, we generally think of quantum mechanical versions of conventional computers. We refer to these as gate-based general-purpose quantum computers.

- **Gate-based quantum computers.** These computers use gates to manipulate qubits in a way that is similar to how classical computers use logic gates to process bits. However, the quantum nature of the qubits allows for more complex and powerful operations.

While this type of quantum computer is the ultimate end goal of quantum computing, there are two additional specialized types of quantum computers to briefly consider:

- **Quantum simulators.** Specialized systems designed to mimic the behavior of complex quantum phenomena that are hard or impossible to simulate with classical computers.
- **Quantum annealers.** A type of quantum computer that is designed to solve optimization problems in which the goal is to find the maximum or minimum value of a certain function.

While not as general as gate-based quantum computers, these two types have great utility at solving specific types of problems.

Let us consider each of these three options and how they are being pursued today.

### Gate-based Quantum Computers

The holy grail of quantum computing is gate-based general-purpose quantum computers. These machines would be capable of carrying out any set of operations that classical computers can perform, while also benefitting from quantum effects to speed up operations. In 1980, Paul Benioff laid out the foundational theory behind quantum computers much like Turing did for classical computers. In his seminal paper, Benioff extended the Turing machine concept to apply to quantum computers.[14] In particular, he showed that the deterministic elements of classical computing could be replaced by nondeterministic quantum versions.

Building on Benioff's work, physicist David Deutsch pointed out in a 1985 paper that because the universal computer described by Turing relied on classical physics, it would be unable to simulate a quantum computer.[15] He then reformulated Turing's work using quantum mechanics to devise a "universal quantum computer," which is capable of simulating any physical process. This could be done by developing quantum gates that would function in a similar fashion to traditional digital computing binary logic gates.

Before discussing quantum gates, we must first understand some basic details about the qubits upon which they operate.

### Qubits

Generically, a qubit consists of a quantum system with two distinct states denoted $|1\rangle$ and $|0\rangle$. These states are analogous to the binary states of 1 and 0 of classical bits.[16] While it is popularly stated that qubits can be both 1 and 0 at the same time, the real story is a bit more complicated. These states can be mixed together into different superposition states. It takes two complex numbers to describe the full range of ways the states can be combined. To help visualize these combinations, Felix Bloch developed his Bloch sphere representation (see **Figure 6**). Each point on the surface of the sphere corresponds to a unique superposition state. This is helpful for visualizing that action of

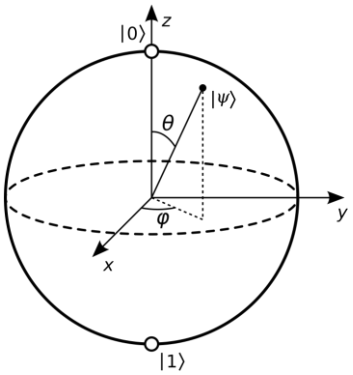


**Figure 6. Bloch sphere representation of a qubit.**
Credit: Wikipedia commons.

[14] Paul Benioff, "The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines," *Journal of Statistical Physics* 22 (5): 563–591 (1980).

[15] David Deutsch, "Quantum theory, the Church-Turing principle and the universal quantum computer," *Proceedings of the Royal Society,* 400 (1818): 97–117 (July 1985).

[16] Individual quantum states are often written using bra-ket notation (or Dirac notation), such as $|1\rangle$ and $|0\rangle$. This

single qubit operators, such as Hadamard (H), Pauli-X, Y, and Z gates, that shift the qubit state from one point on the Bloch sphere to another.

**Quantum Gates**

To understand how gate-based quantum computers operate, we must first understand some basic details about the inner workings of classical computers.[17] To begin, all data is represented as strings of bits (binary digits). Individual bits represent the state of either 1 or 0 and are physically instantiated by the presence or absence of an electrical current. Calculations are performed using a series of logic gates to manipulate the bits. Typical gates take one or two bits as input and return one bit as output. There are seven basic logic gates: NOT, OR, NOR, AND, NAND, XOR, and XNOR. Of these, NOR and NAND are considered universal logic gates in that each individually can be used to reproduce the functions of all the other logic gates.

Quantum computers instead use quantum logic gates. While similar in spirit to classical gates, many are unique to quantum computers because qubits are not limited to being just 1 or 0. The main single qubit gates are Hadamard (H), Pauli-X, Y, and Z gates. The principle two qubit gate is the CNOT (controlled NOT) gate. It is classified as a universal quantum logic gate because any quantum circuit can be simulated to an arbitrary degree of accuracy using a combination of CNOT gates and single qubit rotations. This means that a critical step in creating a quantum computer using a particular type of qubit is to design a CNOT gate for it. For more information, see **Appendix: Quantum Circuits**.

For efficient computing, it is desirable to only incorporate gates that are reversible, such as the CNOT gate. A gate is classified as reversible if applying it twice restores the qubits to their original state. The importance of reversible gates is because most of the energy expended in a calculation is not a result of manipulating qubits but rather the deleting of information at the end. Physicist Rolf Landauer showed in 1961 that information and heat are closely related, therefore, you cannot delete data without releasing waste heat to the outside world. Reversible gates avoid this expensive deletion by being able to walk the calculation back to the start.

**The Power of Quantum Computing**

With all this as background, we can now address the central question—what makes gate-based quantum computers special? Their computational power lies in the massive parallel processing arising from superposition and entanglement. Superposition allows a qubit's state to be in a combination of both $|1\rangle$ and $|0\rangle$ at the same time, whereas a single classical bit can only be 1 or 0. This makes a qubit superior because it can test both possibilities at the same time. Entanglement pushes this further by allowing the qubit states to be linked together. This allows the qubits to simultaneously occupy all possible combinations of both states across the entire collection at the same time. Taken together, this means $n$ fully entangled qubits can test all $2^n$ combinations at once whereas a classical computer can only explore one at time. When we have fully capable quantum computers, they'll operate exponentially faster than classical computers for specific problems that quantum computers are good at.

To better understand this, consider a quantum computer with 256 qubits.[18] It would be able to find the right answer to problems with $2^{256}$ possible answers in one go. If we used a classical

compact notation was created by Paul Dirac in 1939 to simplify complex quantum mechanical expressions.

[17] For a more detailed discussion of gates and their usage in classical computing, see Brian Clegg, Quantum Computing, p. 22-31 and Chris Bernhardt, *Quantum Computing For Everyone*, p. 89-111.

[18] Because qubit states are fragile and error prone, reliable calculations some form of quantum error correction as will be discussed later in this paper. Error correction methods, however, require the use of additional qubits. For a calculation using 256 logical qubits, it might require many times that many physical qubits to support error correction.

computer that takes 1 second to come up with a single answer to a problem that has $2^{256}$ possible answers. It will then take another second with the next answer, and so on. You would have to run the problem through the classical computer $2^{256}$ times to generate all the possible answers—longer than the age of the universe.

**Quantum Simulators**

When studying complex systems, there are two complementary approaches. The first involves modeling the system using computers to solve complex mathematical equations and the second is to test a physical model using real world physics. As an example, consider engineers studying the aerodynamics of an aircraft. Using the first approach, computer models of aerodynamic flow around the aircraft can yield great insight but often require various approximations and simplifications to keep the computational workload manageable. The second approach involves building a model plane and testing it in a wind tunnel. Both approaches are valuable with different strengths and limitations.

For systems of quantum particles, direct computer modeling for many systems of interest is infeasible because the computational complexity grows rapidly with the number of interacting particles. For example, systems involving low-temperature physics and many-body physics remain poorly understood because the underlying quantum mechanics is vastly complex. Conventional computers, including supercomputers, are inadequate for simulating many quantum systems with as few as 30 particles because the complexity of the calculation grows exponentially with number of particles. Gate-based quantum computers may eventually become powerful enough to computationally model some of these systems, but they are currently unable to do so.

To study these and many other types of systems, Richard Feynman championed developing quantum simulators.[19] He laid out much of the theory and motivation in 1981 in his seminal lecture, "Simulating Physics with Computers."[20] The basic idea is to simulate one quantum system using another, more controllable one. One profound reason to do this is a quantum computer would be able to simulate physics *precisely*, not just approximately.[21] No matter how good the approximations involved in simulating the Universe in a classical computer may be, they can never be perfect because the Universe is not classical. Or as Feynman so elegantly summarized this point:

> "Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical…" – Richard Feynman[22]

Based on these principles, quantum simulators are set to revolutionize our understanding and study of complex chemical systems, including nitrogen fixation, photosynthesis, and superconductivity.

When Feynman gave his talk, no one had any idea how to build a quantum simulator because the technology simply did not exist at that time. This situation is slowly changing with much progress being made in recent years. For example, IBM announced in 2017 that it was able to simulate the three-atom molecule beryllium hydride ($BeH_2$) on a quantum simulator using 7 qubits.[23] Not very impressive because this simple system can be simulated by a classical computer.

---

[19] Quantum simulators can also refer to classical computers used to mathematically model the behavior of quantum systems. These simulators run on classical processors and emulate qubits through mathematical models and are primarily used for testing and debugging quantum algorithms before deploying them on real hardware.

[20] Richard Feynman, "Simulating Physics with Computers," *International Journal of Theoretical Physics*, volume 21, 1982, p. 467-488.

[21] John Gribbin, *Computing with Quantum Cats*, p. 210.

[22] Richard Feynman, "Simulating Physics with Computers," p. 486.

[23] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta, "Hardware-efficient Variational Quantum Eigensolver for Small Molecules and Quantum Magnets,"

However, as quantum simulators harness more qubits, they will eventually be able to tackle problems that are well beyond even the most powerful classical computers.

**Quantum Annealers**

One common real-world challenge is finding an optimal mathematical solution to a problem subject to various constraints. Or more simply to find the minimum (or maximum) value of a complicated function. These types of problems are important in many fields, including finance, industrial operations, and logistics. Typical industrial applications include finding the lowest cost for production, the smallest amount of material required, the shortest time to completion, the highest quality, and the greatest profit.[24] Traditionally, these types of optimization problems have been handled by classical computers but the ability to solve them is limited by the available computing power. In 1989, Bikas K Chakarbarti proposed quantum annealing as the means of solving them much faster than possible using classical computers. This would allow quantum annealers to tackle more complex problems than are currently solvable.

The term annealing comes from the ancient practice of ironworking. Most iron naturally has a somewhat random internal structure, which tends to make it stiff and brittle. To make it softer and more easily worked, the iron is put through the process of annealing. Historically, this was done by heating it until its internal structure dissolves and holding it there for some period of time. If the iron is then allowed to slowly cool, its internal structure crystalizes in a more orderly way. This brings the metal's internal structure closer to its equilibrium state, eliminating internal stress within the metal.

A more general way to describe this process is that annealing drives the system to a lower energy configuration, i.e., toward a minimum value. In the same way, quantum annealing leverages the physical changes in the energy levels of qubits to find an optimal mathematical solution. Quantum annealing offers one more critical benefit. While traditional (i.e., non-quantum) approaches tend to get stuck in local minima, rather than finding the global minimum (i.e., the best solution overall). Quantum annealing benefits from the fact that qubits can undergo quantum tunneling to escape local minimum and move toward the desired global minimum.

Quantum annealers are more limited compared to gate-based quantum computers in the types of problems they can solve but offer significant advantages for solving optimization problems because this technology is more mature having a several year's headstart over general-purpose computers. Currently, D-Wave is the only company offering this kind of hardware. Their first commercially available quantum computer utilizing 128 qubits was released in 2011. As a sign of progress, they recently started selling computers with 5,000 plus qubits.[25] Companies using D-Wave computers include Volkswagen, Google, and Lockheed Martin.[26]

---

IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA, Oct 16, 2017, arXiv:quant-ph/1704.05018.

[24] Whuley and Floyd Smith, *Quantum Computing for Dummies*, p. 164.

[25] D-Wave's quantum annealers can utilize more than 5,000 qubits, which is dramatically more than the 100 qubits used in current gate-based quantum computers. This larger number of qubits does not, however, translate to greater computational power for several reasons. First, these qubits are limited in how much they can be entangled; and the underlying technology does not maximize the value of every qubit the way gate-based quantum computing can. Second, no quantum computer so far has robust error correction, so all of them incur undetected errors and often deliver approximate, or simply wrong, results. D-Wave tries to overcome this limitation by using more qubits. (Whuley and Floyd Smith, *Quantum Computing for Dummies,* p. 165.)

[26] Chris Bernhardt, *Quantum Computing For Everyone*, p. 186.

## Quantum Algorithms

An algorithm is a structured set of instructions for accomplishing a specific task (like to a recipe). Just as classical computers need algorithms to solve problems, quantum computers need quantum algorithms. The main difference is that quantum algorithms are designed to take advantage of the quantum mechanical nature of the computer. One critical consideration in evaluating quantum algorithms is their computational efficiency or more specifically how their speed compares to that of the corresponding classical algorithms. This is important because achieving quantum supremacy for important tasks is the primary motivator behind the development of quantum computers.

### Computational Complexity

Before discussing specific algorithms, we need a means of comparing their performance to their corresponding classical counterparts. The main way to do that is to consider their computational complexity, the amount of resources (e.g., time, memory, and disk space) required to run the algorithm. For this work, we will focus exclusively on time complexity, that is, the number of operations required as a function of the problem size, $n$. As the problem size increases, we generally expect the solution to require more work, but how much more work depends on the chosen algorithm. The time complexity reflects the general trend, rather than the exact cost, making it useful for evaluating the overall behavior and for comparison to other approaches. The time complexity is commonly expressed using big O notation, such as $O(n)$, $O(n^2)$, $O(2^n)$, etc., where $n$ is the problem size.

To help understand the concept of time complexity and how it scales with problem size, let us consider the well-known wheat on a chessboard story. According to legend, the inventor of chess taught the game to a rich ruler. The ruler was so impressed that he allowed the inventor to name his reward. Rather than requesting a large quantity of gold as expected, the inventor simply asked for a single grain of wheat to be placed on the first chess square. This was then to be followed by two on the next square, four on the next, and then doubling with each square until the entire chess board was filled. The ruler laughed it off as a meager prize for such a brilliant invention, only to have court treasurers report the unexpectedly huge number of wheat grains would outstrip the ruler's vast resources. Versions of the story differ as to whether the inventor became a high-ranking advisor to the ruler or was executed.

Starting with one grain and doubling each square to fill all 64 chess squares is an example of *exponential scaling*. With $n = 64$, The total number of grains can be shown to be $2^{64} - 1$ or more than a staggering eighteen quintillion (or $1.8*10^{19}$)! This demonstrates how rapidly exponential scaling balloons in value with problem size. As a contrast, imagine if instead the number of grains stayed fixed instead of doubling (i.e., one grain per square). That would result in a mere 64 grains of wheat. This illustrates *linear scaling*. As another example, consider the case where the number of grains simply increases by 1 each time giving the sequence: 1, 2, 3, 4, … The total number of grains would be 2080.[27] This behavior shows *quadratic scaling*. These simple examples illustrate widely different behavior for three separate complexity classes.

Another way to look at scaling is to do it comparatively. That is, to determine the relative change with respect to the problem size, $n$. For example, consider how things would change if we used two chess boards ($n = 128$) instead of one. For linear scaling, the total number of grains of wheat would merely double to 128. With quadratic scaling, the number would instead grow by a factor of 4. The exponential case would rise so rapidly that it would reach even more unimaginable values (more than 324 quintillion quintillion)!

[27] The exact number of grains of wheat for this case is given by $n*(n+1)/2$ and equals 2080 for the $n = 64$ case. The measure of complexity only considers the highest order term, so this example is considered quadratic scaling, $O(n^2)$.

There are many classes of time complexity. For this work, we will only consider two broad categories: polynomial and superpolynomial scaling.

**Polynomial scaling (specific examples):**

- Linear scaling. $O(n)$
- Quadratic scaling. $O(n^2)$
- Cubic scaling. $O(n^3)$

**Superpolynomial scaling (specific examples):**

- Exponential scaling. $O(2^n)$
- Factorial scaling. $O(n!)$

There are many other classes, but we will restrict our discussion here to just these examples.

A major goal in developing quantum algorithms is to find ones with a time complexity that is lower than the equivalent classical version. For these cases, this will allow quantum computers to tackle much larger and more complicated problems than are currently possible. For example, many important classical algorithms have superpolynomial scaling making such problems intractable for larger problem sizes. If the corresponding quantum algorithm has only polynomial scaling, then a sufficiently powerful quantum computer could tackle problems that were previously unsolvable. With that in mind, let us consider the most important quantum algorithms.

**Deutsch-Jozsa Algorithm**

In 1985, David Deutsch published an algorithm to solve a toy problem that was specifically designed to be easy for a quantum computer to solve but hard for any classical computer.[28] For this problem, a classical computer requires two evaluations, whereas a quantum computer requires only one. Although it had no real-world utility, the Deutsch algorithm did serve as proof that a quantum computer could outperform a classical one.[29]

Deutsch, with the help of Richard Jozsa, generalized the algorithm in 1992 to handle any number of bits. The Deutsch-Jozsa algorithm was later improved by Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca in 1998 and then experimentally executed in 2003. The Deutsch-Jozsa algorithm is one of the first examples of a quantum algorithm that is exponentially faster than any possible deterministic classical algorithm. This algorithm does have application to cryptography where it is important to distinguish between secure and insecure encryption algorithms.[30]

**Shor's Algorithm**

While quantum computing may seem like a quiet academic subject, the next major quantum algorithm sent shockwaves around the world and made quantum computing a national security issue. This reaction was the result of Peter Shor publishing a quantum algorithm in 1994 for finding the prime number factors of very large numbers. By taking advantage of quantum mechanics, Shor's algorithm can do this exponentially faster than possible on classical computers. This has enormous implications because most current cybersecurity encryption protocols, such as RSA, rely on the fact that factoring very large numbers is effectively impossible on classical computers.

---

[28] David Deutsch, "Quantum theory, the Church-Turing principle and the universal quantum computer," *Proceedings of the Royal Society,* 400 (1818): 97–117 (July 1985).

[29] Chris Bernhardt, *Quantum Computing For Everyone*, p. 145.

[30] Whuley and Floyd Smith, *Quantum Computing for Dummies,* p. 265.

Encryption is vital to private, corporate, and governmental interests because it allows one to securely store and communicate data. When sending information over the internet, generally the data is encrypted by the sender such that if someone was able to intercept the data it would appear to be gibberish. Only the recipient with the correct key may unencrypt the data and successfully read it. Without encryption, sensitive information—from personal information such as medical or criminal records, to financial information such as bank account and credit card numbers, to cutting-edge commercial research and development, to classified national security information—could be accessed by hostile actors. In the same way, blockchain that is at the heart of cryptocurrency may eventually be vulnerable to manipulation using quantum computers.

Shor's algorithm is one of the few known quantum algorithms with compelling potential applications and strong evidence of superpolynomial speedup compared to the best-known classical algorithms. In plain English, that means that a sufficiently powerful quantum computer utilizing this algorithm could crack current encryption schemes while it would take classical machines millions of years to do the same. Fortunately, such capability is likely still 10-20 years away. Researchers and developers are already working on post-quantum cryptography that will be resistant to even quantum computers.

**Grover's Algorithm**

Another area where quantum computers are looking to revolutionize the world is the task of searching for a specific item in a large, unsorted database. In 1996, Lov Grover proposed a quantum algorithm to do this much faster than possible on classical computers. To understand the utility of Grover's algorithm, imagine trying to find a particular book in a large collection of $n$ books arranged in no particular order. A simple search will require an effort required scales linearly with the number of books, that is $O(n)$. This simply means searching through twice as many books would require approximately twice the effort. Using Grover's algorithm instead reduces the computational effort to the order of the square root of $n$ or simply $O(\text{sqrt}\{n\})$. For a bookshelf with only a few books ($n = 100$), Grover's algorithm would be about 10 times faster. But when searching through an enormous library ($n = 1{,}000{,}000$), there would be a 1,000-fold speed up. Searching through even large collections would yield an even more dramatic improvement.

Not surprisingly, Grover's algorithm is of great interest to companies like Google who perform internet searches. Rather than scanning through a billion plus websites for each user request, Google maintains an enormous index that summarizes everything on the internet and is constantly being updated. But even performing the search on just the index is a daunting task. It is estimated that Google's index is a mindboggling 100,000 terabytes ($10^{17}$ bytes) in size.[31] Using Grover's algorithm on future quantum computers would dramatically speed up these searches. For this reason, Google started working on quantum computing in 2006 and remains a major player in this endeavor.

**Other Algorithms**

In addition to the featured algorithms, there are a plethora of other quantum algorithms, although most are only of interest to scientists and big companies. Many of these algorithms are primarily used as components of other algorithms.

- **Simon's Algorithm.** A predecessor to Shor's algorithm, it has applications for cryptanalysis and codebreaking. It can also be used to solve the discrete logarithm problem.

[31] Brian Clegg, *Quantum Computing*, p. 8. For comparison, modern hard drives can typically hold about 1 terabyte of data, so the Google database is equivalent in size to about 100,000 of those hard drives.

- **Quantum Fourier Transform (QFT).** Core of many algorithms including Shor's. Used in solving problems like period finding, signal processing, and phase estimation.
- **Quantum Phase Estimation.** Estimates eigenvalues—key in many physics and chemistry applications. Also, a building block of other quantum algorithms.
- **Quantum Approximate Optimization Algorithm (QAOA)**. A quantum algorithm designed to solve combinatorial optimization problems.
- **Variational Quantum Eigensolver (VQE).** Useful for solving quantum chemistry and optimization problems.
- **Quantum Machine Learning (QML) Algorithms.** Powerful tools for classification, clustering, and data analysis. Quantum machine learning seeks to enhance classical machine learning models using quantum techniques.

A complete list of known quantum algorithms with descriptions can be found in the quantum algorithm zoo.

## Quantum Hardware

Quantum algorithms are only useful if the necessary hardware exists to carry them out. Having covered much of the theory behind quantum computers, we now turn to the engineering side. How can we turn theory into reality? Just as it took visionaries like Turing and von Neumann to develop working electronic computers, so too it will take the efforts of a great many individuals to overcome the innumerable remaining engineering challenges needed to develop viable quantum computers.

In 1996 and again in 2000, David DiVincenzo at IBM published a paper outlining the five conditions he believed were necessary for creating a quantum computer.[32,33] This has since become known as the DiVincenzo criteria and has influenced much of the experimental research into developing a working quantum computer.

- **A scalable physical system with well-characterized qubits.** For qubits to be useful, they must be well-defined with properties that can be accurately manipulated. Second, the system needs to be scalable, that is, able to use a sufficiently large number of the qubits to perform calculations.
- **The ability to initialize the state of the qubits to a simple fiducial state.** Setting qubits to a well-defined initial state (e.g., set to 0) is an essential first step in calculations. In addition, many error correction schemes require a constant supply of initialized qubits. This can be accomplished using a "qubit conveyer belt" to relocate certain qubits, reinitialize them, and then return them to service.
- **Long relevant decoherence times, much longer than the gate-operation time.** One of the major challenges for quantum computing is that qubits quickly "decohere," i.e. lose their quantum properties thus rendering them useless for calculation. The decoherence time may be short (e.g., millionth of a second) but can still be feasible as long as it is significantly longer than the operation time of the quantum gates (e.g., a million millionth

[32] David P. DiVincenzo, "Topics in Quantum Computers" (version 2), Dec 15, 1996, arXiv:cond-mat/9612126v2.

[33] David P. DiVincenzo, "The Physical Implementation of Quantum Computation" (version 3), April 13, 2000, arXiv:quant-ph/0002077v3.

of a second). Decoherence times depend upon the type of qubit involved and how well it can be isolated from the surrounding environment.

- **A "universal" set of quantum gates.** Quantum gates perform computational operations in much the same way that classical gates do in classical computers. A 2-qubit CNOT (controlled NOT) gate is the minimum requirement for constructing a gate-based quantum computer.[34] These gates need to be reversible (i.e., if applied twice in a row the qubits are returned to their original state). One additional challenge is the ability to turn gates on and off when required.
- **A qubit-specific measurement capability.** The final state of the computation is to measure the qubits to read the answer. This must be done with high fidelity to ensure accuracy. Because this act of measuring is not 100% accurate, the calculation may need to be performed multiple times to compensate.

DiVincenzo also described two additional criteria involving quantum communication (the act of transmitting intact qubits from place to place) but they will not be covered here.

**Types of Qubits**

Qubits are the functional elements of quantum computers. They can be created by manipulating and measuring quantum particles, such as photons, electrons, trapped ions, and atoms. Of the many types of qubits that are currently being tested, there are some early leaders, but situation remains fluid with no clear favorite. This situation is akin to the early stages of a horse race—there is still lots of time for one horse to pull out front or for another to fall behind.

What practical considerations can we use to evaluate the merits of a particular choice for qubits? In evaluating the proliferation of potential qubit candidates, there are at least six major engineering tradeoffs that must be considered.[35] (Three of these—coherence time, gate speed, and scale—were previously mentioned as part of the DiVincenzo criteria.)

- **Fidelity.** How close is the final state of the qubit to the ideal state that a perfect qubit would be in, after the same initialization process and running the same gates? Because error rates compound across qubits, having high fidelities—well above 99 percent—for each qubit is important.
- **Coherence time.** Using (relatively) mainstream supporting technologies, how long can the qubit be kept coherent? The longer that is, the more operations can be accomplished. Coherence times range from a fraction of a thousandths of a second to several seconds.
- **Gate speed.** How long does it take for the qubit to complete a single processing step, or gate? Faster gate speeds are better, and gate speed is related to coherence time. A qubit with a shorter coherence time must have a very fast gate speed, or you'll never be able to run a program with many gates (steps) in it.
- **Scale.** How many qubits can be created and used together in a quantum computer of a given type (while maintaining isolation, control, measurement, and communication mechanisms)? The more qubits, the more room for error correction and the greater the capability to solve useful problems. In the future, how easy will it be to scale to thousands of qubits?
- **Manufacturability.** Is there a way to manufacture processing units of a given modality in volume and at reasonable cost, while maintaining the required mechanisms and meeting

[34] John Gribbin, *Computing with Quantum Cats*, p. 215-216.

[35] Whuley and Floyd Smith, *Quantum Computing for Dummies*, p. 183.

any special requirements, such as supercooling? Some qubits are not single particles and for those cases one must consider for construction variance.

- **Entangleability.** Ideally, all the qubits in a quantum computer can be entangled with each other, allowing the system to achieve the maximum power from the qubits. Some qubit types are easier than others to configure in ways that allow for greater entangleability. Although this capability is often not included with others in the list, entangleability is of great importance.

Of these, fidelity (to ensure meaningful outcomes) and scale (to tackle larger problems) stand out as the most critical.

Currently, there are several major types of qubits that are currently being utilized with no clear winner for best qubit. Let us explore each of these qubits along with their strengths and weaknesses.

**Trapped Ions**

Trapped ions were proposed by Ignacio Cirac and Peter Zoller in 1995 making them one of the earliest qubits to be developed.[36] Simultaneously, they are currently the most used type. Ions (charged atomic particles) can be confined and suspended in free space using special traps—typically a linear Paul trap that confines ions in a line using oscillating electric fields. Qubits are stored in stable electronic states of each ion, and quantum information can be transferred through the collective quantized motion of the ions in a shared trap (interacting through the electromagnetic force). Lasers are applied to induce coupling between the qubit states (for single qubit operations) or coupling between the internal qubit states and the external motional states (for entanglement between qubits).

One strength of trapped ions is that it allows for all-to-all entangleability, that is, it is easy to entangle all the qubits in the same trap to each other. This high level of entanglement maximizes the processing power of the qubits. The main downside is that a single trap is effectively limited regarding how many ions it may hold because the ions naturally repel each other. To increase the total number of qubits means spreading them over multiple ion traps, but coordinating interactions between ions in different traps is challenging.

Trapped ions represent one of only a few qubit technologies which have fulfilled all DiVincenzo's original criteria with high fidelity.

**Advantages:**

- **High Fidelity.** Trapped ion systems have demonstrated exceptionally low error rates in qubit operations, with single-qubit fidelities exceeding 99.99% and two-qubit gate fidelities surpassing 99%.
- **Long Coherence Times.** The coherence times of trapped ions can be significantly longer than other qubit types, allowing for more complex computations without losing quantum information.
- **All-to-All Entangleability.** All ions in the same trap can easily be entangled to each other.
- **Uniformity.** All ions of the same species are identical qubits—no fabrication variations.
- **Room Temperature Operation.** The ions need to be kept in a vacuum and laser-cooled, but the setup is in a room-temperature environment.

[36] J. I. Cirac and P. Zoller, "Quantum computations with cold trapped ions," *Physical Review Letters*, 74 (20), 4091-4094 (1995).

**Challenges:**

- **Scaling Difficulties.** Trapped ion qubits face challenges related to scaling, such as high heating rates and complex control systems required for larger ion chains. Chief among these is increasing the number of simultaneously trapped ions while maintaining the ability to control and measure them individually with high fidelity.
- **Slow Gate Times.** Trapped ions suffer long gate operation times, which limits how much work can be done before decoherence.
- **Hard Vacuum.** These qubits generally utilize cold (about 4 K) but not supercold temperatures, however, they do require a strong vacuum.

**Companies:**

- IonQ, founded in 2015, is the current leader for this technology and favor using ytterbium ($^{171}Yb^{+}$) for their ion source.
- Other major leaders include Quantinuum, Universal Quantum, Alpine Quantum Technologies, Quantum Factory, Oxford Ionics, and Eleqtron.

**Superconducting Qubits**

Superconducting qubits are essentially microscopic circuits made of a superconducting metal that behave like artificial atoms.[37] They were first proposed as qubits in 1985 and consist primarily of tiny Josephson junctions (loops of superconducting material separated by a thin insulating barrier).

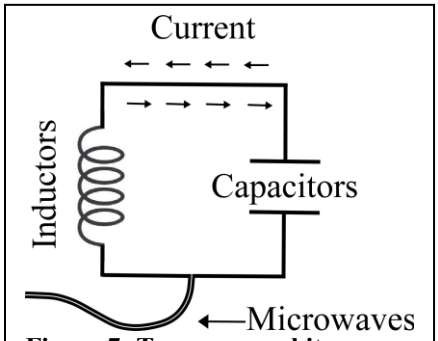

**Figure 7: Transmon qubit.**
Credit: John Millam

There are three main types of superconducting qubits based on which quantity they measure: charge, phase, and flux qubits. Multiple variations of these three types exist today, each with their own strengths and weaknesses.[38] The transmon (transmission line shunted plasma oscillation) qubit, introduced in 2007, is the current favorite because it is less sensitive to environmental noise.

In contrast to trapped ions, entanglement among superconducting qubits is generally restricted to just neighboring qubits. This tends to limit the effective computational power of these qubits often requiring more total qubits to compensate for the reduced entanglement.

**Advantages:**

- **High Fidelity.** Superconducting qubits have very high fidelities (around 99.5%), second only to trapped ions.
- **Manufacturability.** Relatively easy to manufacture using standard techniques.
- **Fast Gate Operations.** Superconducting qubits are known for their fast gate operations.
- **High Scalability Potential.** Because superconducting qubits are built using standard microfabrication techniques (lithography, thin-film deposition, etching), they are inherently scalable using existing chip-based manufacturing.

---

[37] Superconductivity is a phenomenon where certain materials exhibit zero electrical resistance when cooled below a specific temperature.

[38] The main types of superconducting qubits include: phase qubit (utilizes the phase difference across a Josephson junction), charge qubit (based on the charge state of a superconducting island), flux qubit (relies on the magnetic flux through a superconducting loop), transmon qubit (a modified charge qubit that is less sensitive to charge noise), fluxonium qubit (a hybrid of flux and transmon qubits), xmon qubit (an advanced version of the transmon qubit designed for scalability), and quantronium qubit (a type of charge qubit that incorporates a flux degree of freedom).

**Challenges:**

- **Qubit Variability.** These qubits are not entirely uniform and therefore must be properly calibrated before use.
- **Cryogenic Cooling Requirement.** Must be supercooled to around 10 millikelvin requiring complex cooling devices.
- **Short Coherence Times.** A significant problem is the tremendously limited coherence time (tens of microseconds) of early superconducting qubits. To minimize this problem, superconducting qubits typically need to operate at millikelvin temperatures. This requires the use of large cryostats filled with liquid helium for refrigeration.
- **Error Rates.** Quantum operations on superconducting qubits are unreliable, with error rates around 0.1% in earlier stages. This requires the execution of quantum error correction techniques (discussed in the next section), which use multiple physical qubits to compose more reliable logical qubits.

**Companies:**

- IBM Quantum is a major player with roadmap goals.
- D-Wave with quantum annealers.
- Rigetti, Oxford Quantum Circuits (OQC), Google, Baidu, and Amazon Bracket.

**Topological Qubits**

Topological qubits are interesting in that they utilize quasiparticles (the collective motion of many particles that behave like a single particle) rather than elementary particles or atoms. The theoretical basis for these qubits is the Majorana fermion, a hypothetical fermion that acts as its own antiparticle. Although proposed by Ettore Majorana in 1937, no elementary particles are currently classified as Majorana fermions. Instead, there are quasiparticles known as Majorana zero modes (MZMs) that can serve as qubits for quantum computing. Unlike conventional qubits that rely on local properties of particles (like spin or charge), topological qubits utilize global properties, making them inherently more stable against environmental noise and disturbances. Topological qubits are still in early experimental stages but could be a game-changer once fully developed.

**Advantages:**

- **High Fault Tolerance.** Topological qubits are less sensitive to noise and errors, making them more robust for quantum computations compared to other qubits.
- **Stability.** The information stored in topological qubits is protected by the topological properties of the system, which are not easily altered by local disturbances.
- **Scalability.** The unique properties of topological qubits may allow for the development of larger and more complex quantum systems that can perform computations more efficiently.
- **Lower Operational Overhead.** Traditional qubits require thousands of physical qubits to form a single logical qubit with error correction. Topological qubits could reduce this overhead, making large-scale quantum computing more practical.

**Challenges:**

- **Creating Stable Majoranas.** Majorana modes require special quantum materials that are hard to fabricate and maintain at extremely low temperatures.
- **Precise Braiding Control.** Developing nanotechnologies capable of moving quasiparticles with extreme accuracy is very difficult.
- **Verification Complexity.** Measuring and verifying topological states without disturbing them remains a major challenge.

**Companies:**

- Microsoft is at the forefront of developing topological qubits with its Majorana 1 quantum processor.
- Nokia Bell Labs.

**Photons**

Photons are an obvious choice for qubits. They are tiny, massless, and chargeless particles that always move at the speed of light. Best of all, they interact weakly with the environment giving them very long coherence times without needing for ultracold temperatures or requiring a vacuum. And as an added bonus, integrated photonic chips, similar to those used in modern telecommunications, allow for packing thousands or even millions of optical components onto a single wafer. This scalability advantage makes photonics potentially ideal for both industrial and commercial quantum computing.

Photonic quantum computing also goes hand in hand with existing telecommunications technology. The same fiber-optic cables that carry internet data can also transmit photonic qubits, allowing quantum processors to be linked across cities, or even continents, without the need for entirely new infrastructure. This compatibility minimizes the cost and time required for large-scale deployment. Telecom-grade optical components such as wavelength multiplexers, splitters, and amplifiers are already widely available and manufactured at scale, meaning photonic quantum devices can benefit from decades of advancements in the telecom sector.

Despite some incredible advantages, photonic qubits face several major challenges. First, photons are "flying" (rather than stationary) qubits meaning they are always in motion and cannot be stored. A second major difficulty is getting photons to interact in the form of gates operating on two or more photonic qubits. There are many additional technological challenges, such as high photon loss (which corresponds to losing quantum information), the difficulty of generating individual photons on demand, and the need for highly efficient photon detectors.

**Advantages:**

- **Long Coherence Times.** Photons interact weakly with the environment, which naturally preserves and protects quantum states.
- **Long-Distance Transmission.** Ideal for quantum communication and distributed quantum computing.
- **Compatibility with Existing Infrastructure.** Can leverage advanced photonic technologies.
- **Scalability Potential.** Possibility of large-scale integration using photonic circuits and native networkability.
- **Room Temperature Operation.** Photonic systems can operate at room temperature, reducing the need for complex cooling systems.

**Challenges:**

- **Photon Loss.** High rates of photon loss necessitate quantum error correction to ensure reliable measurement results.
- **Deterministic Single Photon Sources.** Generating on-demand, indistinguishable single photons is difficult.
- **Efficient Detectors.** Developing high-efficiency, low-noise single-photon detectors is important.
- **Two-Qubit Gates.** Implementing deterministic two-qubit operations is challenging.

**Companies:**

- Leading companies: PsiQuantum, Xanadu, Quandela, Quantum Source, Orca, and Quix Quantum.

## Table 1: Best Qubits According to Functionality[39]

| | Qubit Type (Metric) |
|---|---|
| **Fidelity** | Trapped ions (99.9%), superconducting qubits (99.5%). |
| **Coherence Time** | Photons (long) and trapped ions (seconds). |
| **Gate Speed** | Superconducting qubits (fast) and photons (fast). |
| **Manufacturability** | Superconducting qubits (high). |
| **Entangleability** | Trapped ions (all-to-all) and photons (all-to-all). |
| **Qubit Count** | Superconducting qubits (low thousands for quantum annealing or low hundreds) and trapped ions (dozens). |

### Other Qubits

There are several additional types of qubits that are being experimented with.[40]

- **Neutral/cold atoms.** Similar to trapped ions, neutral (aka cold) atoms are trapped by laser beams rather than magnetic fields. Single outer-shell electrons are excited by laser to do their quantum part. These qubits are easily entangled and, unlike ions (which naturally repel each other), can be packed tightly into an array, forming multi-qubit gates. Infleqtion, Pasqual, and QuEra use this technology.
- **Silicon spin.** Silicon spin qubits are conceptually similar to transmon superconducting qubits, in that they act like artificial atoms. But instead of chunks of superconducting metal, they use a single atom of a semiconducting element (such as silicon). A single electron, controlled by microwaves, is the quantum mechanical actor. The qubit must be placed in a quantum dot, enhancing manufacturability because lots of quantum dots are already out there, such as in every pixel of many TV screens. This type of qubit is backed by Intel, Quantum Motion, and Silicon Quantum Computing. As the Intel connection might suggest, silicon spin qubits are meant to be highly manufacturable using existing chip technology, but error rates are currently high.
- **NMR qubits.** Matter kept as a liquid in a test tube can be used for quantum computing by picking out a few atoms on each molecule and using nuclear magnetic resonance (NMR) to manipulate their spins as the basis for qubits. This is how the first quantum computers were achieved in the 1990s. They are not currently being used in commercially funded work.
- **BEC qubits.** Qubits can be made from a superconducting gas in the form a Bose-Einstein condensate (BEC). This has been done experimentally but not yet taken up for commercial use. BECs would be somewhat similar to superconducting qubits, in that each qubit would be made up of a huge number of atoms.
- **Defect-based qubits.** Invisible flaws in a diamond can be controlled with good fidelity at room temperature. Successfully using these flaws to form qubits could make diamonds everyone's best friend. Other kinds of flaws, such as a missing qubit in an array of other kinds of qubits, can also be used as defects as part of this approach.

With more research, several of these have potential to be competitive with the main qubit types.

[39] Based on table in Whuley and Floyd Smith, *Quantum Computing for Dummies*, p. 180. Note, this information represents the state of the art at the time of the book's publication and may not reflect current technology.

[40] Whuley and Floyd Smith, *Quantum Computing for Dummies*, p. 199-202.

**Error Correction**

Before quantum computing can be truly useful, it is essential to address the challenge of managing errors that inherently arise during processing. Qubits are significantly more delicate than classical bits, which makes them susceptible to errors caused by decoherence. There are two main sources of error that must be addressed:

- **Interactions with the Environment.** Most qubits consist of single quantum particles and thus are very vulnerable to corruption when interacting with the environment. Using ultra cold temperatures and/or a strong vacuum can limit but not eliminate this problem.
- **Performing Quantum Operations.** As qubits are forced to interact through quantum gates, this process can induce errors. Even high-fidelity qubit operations can still have error rates of about 1 in 1,000. That is very high compared to compared to classical bits, which can achieve error rates as low as 1 in a million million. Even worse, errors caused by quantum gates accumulate with each successive operation.

An accumulation of errors will cause quantum computers to deliver results that are either approximate or simply wrong. And to make matters worse, this problem will grow as quantum computers harness more and more qubits.

The first step in dealing with this problem is error mitigation involving strategies that help prevent errors from occurring. To tackle environmental interference, one can either use fault-tolerant qubits (e.g., photonic or topological) or isolate the system from the environment using ultracold temperatures and/or a vacuum. Limiting errors in quantum operations requires developing high-fidelity gates, such as already existing for trapped ions and superconducting qubits.

While error mitigation is important, it must be combined with some form of error correction to detect and correct errors that will inevitably crop up during the calculation.[41] Both error mitigation and robust error correction are necessary to ensure that the results of the calculation can be trusted with a high level of confidence. Error correction is widely utilized in the context of classical computing, but these methods cannot be straightforwardly applied to quantum computing for several reasons:

- **No Copying of Qubits.** A common classical strategy is to duplicate some of the input bits and then compare their computed values to their twins with differences indicating errors. Unfortunately, this type of strategy will not work with qubits because the quantum no cloning theorem forbids the copying of quantum data.[42]
- **Quantum Entanglement.** Error correction in quantum systems is more complicated than for classical systems because the qubits are entangled and therefore affecting one qubit simultaneously affects the others.
- **Measuring Affects Qubits.** Another classical error correction strategy involves testing bits at different stages looking for errors to correct. For quantum computers, this strategy is not only costly, it is problematic because measuring the qubits can potentially change their value thus spoiling the results.

[41] Joschka Roffe, "Quantum Error Correction: An Introductory Guide," Jul 25, 2019, arXiv:quant-ph/1907.11157.

[42] The quantum no cloning theorem (1978) is a fundamental principle in quantum mechanics that states it is impossible to create an independent and identical copy of an arbitrary unknown quantum state. This theorem has significant implications in the field of quantum computing and quantum information science. Partnered with the quantum no deleting theorem (2000), it tells us that information cannot be created or destroyed.

Given these fundamental challenges, it seemed early on that quantum error correction was impossible. And without it, there would be no way to make quantum computing reliable enough to be useful.

A breakthrough came in 1995 when Peter Shor and Andrew Steane proposed the first practical scheme for quantum error correction. Instead of copying qubits (which is forbidden), they showed how their state information could be spread over multiple mutually entangled qubits. (This is the quantum version of don't put all your eggs in one basket.) This scheme was built on previous efforts from 1985 by Asher Peres, Shor, and Steane that showed how to use this redundancy to defeat errors and recover from them. The first practical demonstration of this error correction strategy came in 1998.[43]

The Shor-Steane approach involves encoding logical qubits onto multiple physical qubits. This redundancy allows the system to detect and correct errors without losing the encoded information. The downside of this strategy is that it increases the resource requirements for quantum systems, as more physical qubits are needed to encode fewer logical qubits. According to some early estimates, error correction might require 100 to 1,000 physical qubits to make one logical qubit for failsafe computing. Currently, 90% of the time spent in quantum calculations is the result of error correction.[44] In addition, this overhead can complicate the design and scalability of quantum computers.

Once it was demonstrated that quantum error correction was feasible, it spurred the development of more sophisticated approaches. A detailed description of these methods is beyond the scope of this paper, but there are two main approaches.

- **Stabilizer Codes.** These are a class of quantum error-correcting codes that use a set of commuting operators to define codewords. The most well-known stabilizer code is the surface code, which is widely pursued for its practical implementation in quantum computing.
- **Real-Time Error Detection.** Effective quantum error correction requires real-time monitoring and correction of errors. This involves integrating fast measurement and correction cycles to maintain logical qubit fidelity during computations.

Recent developments in quantum error correction (QEC) have shown promising results, such as Google's demonstration in 2024 of a quantum processor that successfully implemented QEC at scale. This landmark achievement provides evidence that QEC can be effectively applied to larger quantum devices, paving the way for practical quantum computing applications.

## Milestones of Progress

Having laid out the theory behind quantum computers, we can now consider the progress that has been made in building and using them for real world applications. Recall that the earliest classical computers were slow and clunky but swiftly transformed into the computational powerhouses that have transformed our lives. In the same way, quantum computers started very simple but have already made enormous strides. Let us consider all that has happened in just the last 30 years. (Discussion summarized from Forbes timeline of 27 milestones and Live Science's summary of 12 key moments.)

**1998: First Demonstration of a Quantum Algorithm.** To create the first proof of concept quantum computers, researchers turned to Nuclear Magnetic Resonance (NMR), a standard

[43] D. G. Cory, W. Mass, M. Price, E. Knill, R. Laflamme, W. H. Zurek, T. F. Havel, S. S. Somaroo, "Experimental Quantum Error Correction," Feb 6, 1998, arXiv:quant-ph/9802018.

[44] Brian Clegg, *Quantum Computing*, p. 215.

chemistry tool for studying the spin on atomic nuclei.[45] This allowed them to cobble together simple quantum computers using well-understood existing technology. In 1998, two physicists at Oxford University successfully ran Deutsch's algorithm on a simple 2-qubit NMR computer. Another group that year executed Grover's algorithm on a different 2-qubit NMR computer. The crowning achievement for the NMR qubit approach came in 2001 when researchers at IBM Almaden Research Center factored 15 using Shor's algorithm. Their set up used 7-qubits corresponding to 5 fluorine atoms and 2 carbon atoms of a certain fluorocarbon molecule.[46] While this may seem a trivial task, it was one of the first demonstrations that quantum principles could be translated to a realizable system for computation beyond classical systems. This achievement was a significant step forward in the development of quantum computing technology and its potential applications.

**1999: The Birth of the Superconducting Quantum Computer.** The fundamental building blocks of a quantum computer, known as qubits, can be implemented using a wide range of physical systems. But in 1999, physicists at Japanese technology company NEC hit upon an approach that would go on to become the most popular approach to quantum computing today. In a paper in *Nature*, they showed that they could use superconducting circuits to create qubits, and that they could control these qubits electronically.[47] Superconducting qubits are now used by many of the leading quantum computing companies, including Google and IBM.

**2011: First Commercial Quantum Computer Released.** Despite considerable progress, quantum computing remained primarily an academic discipline. The launch of the first commercially available quantum computer by Canadian company D-Wave in May 2011 heralded the start of the quantum computing industry. The start-up's D-Wave One featured 128 superconducting qubits and cost roughly $10 million. However, the device wasn't a universal quantum computer. It used an approach known as quantum annealing to solve a specific kind of optimization problem, and there was little evidence it provided any speed boost compared to classical approaches.

**2012: First Dedicated Quantum Computing Software Company is Founded.** 1QB Information Technologies (1QBit) is a pioneering quantum computing software company based in Vancouver, British Columbia. It is recognized as the first dedicated quantum computing software company and focuses on developing algorithms for various applications, including computational finance, materials science, and quantum chemistry. 1QBit has established partnerships with major companies like Microsoft, IBM, and D-Wave Systems, and has raised significant funding to support its innovations in quantum computing technology.

**2016: IBM Makes Quantum Computer Available Over the Cloud.** While several large technology companies were developing universal quantum computers in-house, most academics and aspiring quantum developers had no way to experiment with the technology. In May 2016, IBM made its five-qubit processor available over the cloud for the first time, allowing people from outside the company to run quantum computing jobs on its hardware. Within two weeks more than 17,000 people had registered for the company's IBM Quantum Experience service, giving many their first hands-on experience with a quantum computer.

**2019: Google Claims "Quantum Supremacy."** Despite theoretical promises of massive "speedup," nobody had yet demonstrated that a quantum processor could solve a problem faster

---

[45] NMR is closely related to the more familiar magnetic resonance imagining (MRI) that is commonly used in healthcare to generate images of what is happening inside the human body.

[46] Lieven M.K. Vandersypen, Matthias Steffen, Gregory Breyta, Costantino S. Yannoni, Mark H. Sherwood, Isaac L. Chuang (2001), "Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance," *Nature*, 414 (6866): 883–887, Dec 27, 2001, arXiv:quant-ph/0112176.

[47] Y. Nakamura, Yu. A. Pashkin & J. S. Tsai, "Coherent control of macroscopic quantum states in a single-Cooper-pair box," *Nature*, volume 398, pages 786–788, Apr 1, 1999, arXiv:cond-mat/9904003.

than a classical computer. But in September 2019, news emerged that Google had used 53 qubits to perform a calculation in 200 seconds that it claimed would take a supercomputer roughly 10,000 years to complete. The problem in question had no practical use: Google's processor simply performed random operations and then researchers calculated how long it would take to simulate this on a classical computer. But the result was hailed as the first example of "quantum supremacy," now more commonly referred to as "quantum advantage."

**2022: A Classical Algorithm Punctures Supremacy Claim.** Google's claim of quantum supremacy was met with skepticism from some corners, in particular from arch-rival IBM, which claimed the speedup was overstated. A group from the Chinese Academy of Sciences and other institutions eventually showed that this was the case, by devising a classical algorithm that could simulate Google's quantum operations in just 15 hours on 512 GPU chips. They claimed that with access to one of the world's largest supercomputers, they could have done it in seconds. The news was a reminder that classical computing still has plenty of room for improvement, so quantum advantage is likely to remain a moving target.

**2023: QuEra Smashes Record for Most Logical Qubits.** One of the biggest barriers for today's quantum computers is that the underlying hardware is highly error-prone. Due to the quirks of quantum mechanics, fixing those errors is tricky and it has long been known that it will take many physical qubits to create so-called "logical qubits" that are immune from errors and able to carry out operations reliably. In December 2023, Harvard researchers working with start-up QuEra smashed records by generating 48 logical qubits at once, more than ten times more than anyone had previously achieved. The team was able to run algorithms on these logical qubits, marking a major milestone on the road to fault-tolerant quantum computing.

## Transformative Technology

As quantum computing is slowly transforming from a dream into a concrete reality, the next big question is, "So, what can we do with it?" Simply put, it will allow us to tackle some of the most pressing problems facing mankind. Here are some of the most important examples:[48]

**Genetics.** All living creatures carry within themselves stands of DNA that encode the instructions needed for building and maintaining the organism. This DNA can be divided into individual genes that code for specific proteins that are needed for the biological processes. Much work over the last few decades has gone into mapping the entire genome of humans and many other creatures. Subsequent work has helped determine the function of each of these genes. The next step is to modify and improve specific genes leading to new therapies and creating new tools to treat incurable diseases. Quantum computers will be able to help with this analysis.

**Greening the World.** Photosynthesis in plants converts water and carbon dioxide into sugars and oxygen using the power of visible light. Quantum mechanics plays a key role in making this process nearly 100% energy efficient, however, many details about photosynthesis are not fully understood. Quantum computers can help us unravel these mysteries, allowing us to create artificial leaves powered by sunlight. These could allow us to (1) efficiently generate power from sunlight, (2) remove carbon dioxide from the atmosphere to help reduce global warming, and (3) generate biofuels and other useful organic compounds.

**Feeding the Planet.** Growing enough food to feed the rapidly expanding human population rests heavily on the use of fertilizer. Almost all the ammonia used in fertilizer is generated from the breakdown of nitrogen gas in the atmosphere. This is done on an industrial scale using the Haber-Bosch process, but this is extremely energy intensive consuming 2% of the world's energy output. Certain bacteria, however, have special enzymes that can break down

[48] Discussion summarized from Michio Kaku, *Quantum Supremacy: How the Quantum Computer Revolution Will Change Everything*, Double Day, New York, NY 2023.

nitrogen gas (a process known as nitrogen fixation). Quantum computers could help us to better understand this process, thus providing us with a cleaner and much more energy efficient way to generate ammonia for fertilizer.

**Energizing the World.** The last few decades have seen a shift away from using gasoline to power our cars to electricity. Central to this effort is creating better batteries. Unfortunately, battery technology has remained largely static over the last 200 years. Quantum computers offer the chance to dramatically improve batteries by (1) improving energy storage, (2) increasing lifespan, (3) reducing charging times, and (4) avoiding the use of toxic materials (such as cobalt).

**Medicine.** The development of antibiotics has saved millions of lives, but we are faced with the growing challenge of drug-resistant germs. Developing new antibiotics by trail-and-error is costly and time consuming. Quantum computers will be able to help us understand the underlying mechanisms of diseases and quickly develop new medicines to treat them. Another challenge we face is the growing threat of pandemics as illustrated by the recent spread of Covid-19. The difficulty is that by the time a new disease is detected in patients it may be too late to stop its spread. There are several possible early warning systems for detecting a potential pandemic in its early stages: (1) sensors in sewers combined with rapid antigen tests, (2) thermometers or other sensors that are connected to the internet, and (3) social media. All of these require the power of quantum computers to identify unusual trends that hint at an emerging threat.

**Gene Editing and Curing Cancer.** Cancer remains a leading cause of death and is very difficult to treat because there are many variants caused by different gene mutations. One powerful way of fighting cancer is to detect it early before it manifests symptoms. Liquid biopsies based on analyzing blood or urine samples could non-invasively detect key cancer markers long before any symptoms appear. For this to work, however, requires sensitivity to extremely small concentrations of markers and must be reasonably inexpensive. Another strategy to fight cancer is immunotherapy, where the immune system is trained to detect and fight the cancer. Aside from cancer, there are 10,000 known diseases caused by a single genetic mutation. Gene editing using CRISPR may cure these and other maladies, such as sickle cell anemia, AIDS, cystic fibrosis, and Huntington's disease. All these approaches would benefit from the power of quantum computing.

**AI and Quantum Computers.** Artificial intelligence (AI) is already a game-changing technology. As an example, AI has already solved protein folding, which is critical for the fields of medicine and biology. This involves mapping a given protein sequence to its correct 3D structure. After mapping all known proteins and determining their function, the next logical step is creating new proteins and medicines. This has important medical applications because many conditions may involve misfolded proteins, such as prions, ALS, Parkinson's disease, and other incurable maladies. To accomplish this, we need to extend the processing power of AI using quantum computing.

**Immortality.** Aging is a major issue facing all people. Currently, the maximum human lifespan is about 120 years but what if we could live much longer? This could be possible if we had a better understanding of the causes of aging and attacked it at the molecular level. Already several proteins have been identified as playing critical roles but much more work is required. One pathway forward is sequencing the genome of a large percentage of the population and then using quantum computers to comb through the data to isolate key genes related to long life spans and then fix it using gene editing.

**Global Warming.** Since the beginning of the Industrial Revolution 100 years ago, there has been a sharp increase in temperatures worldwide. Much of this change is a result of increased carbon dioxide in the atmosphere. All of this has serious implications for the world, so what can we do about it? Some possible solutions include (1) carbon sequestration, (2) weather modification, (3) algae blooms, (4) rain clouds, (5) planting trees, and (6) calculating virtual weather. Understanding global weather using complex simulations will be vital in guiding us forward and for that we need powerful quantum computers.

**The Sun in a Bottle.** Nuclear fusion, the process that powers the sun, offers humanity the potential for limitless clean energy. Practical fusion has been studied for seventy years yet remains so expensive and complex that commercialization of this technology is likely still decades away. Many attempts at practical fusion are currently active but involve very different designs. Quantum computing combined with AI can help develop better reactor designs as well as designing better superconducting magnets for use in the reactors.

**Simulating the Universe.** Quantum computers will find many applications in astronomy. (1) The universe is vast, containing some 10 billion trillion ($10^{22}$) stars, making it difficult for even supercomputers to model. (2) The nature of exoplanets (planets orbiting stars other than the sun) and the related question of extraterrestrial life are questions of great scientific interest, but we are still in the early stages of exploring. (3) Much needs to be done to monitor and understand potential threats to life on Earth, such as killer asteroids and coronal mass ejections (CME) from the sun. (4) There are also a great number of other astronomical phenomena that are still not fully understood, such as stellar evolution, gamma ray bursts, black holes, and the nature of dark matter.

## Conclusion

We stand at the cusp of the quantum computing revolution. Quantum computers are no longer theoretical constructs but are a concrete reality. Dozens of companies are currently working on quantum computers with multiple types of qubits being explored. Many quantum algorithms to tackle important problems have been developed along with the software needed to implement them. While owning and maintaining quantum computers is still largely reserved for governments, universities, and large companies, cloud-based access makes them available to even individuals and small businesses. Anyone, regardless of experience, can today experiment with running a program on a quantum computer for free.[49]

Quantum computers have come a long way, but they have not yet fully arrived. The situation today resembles where classical computers stood in the 1950s and 60s. Quantum computers are useful yet are not the game-changing revolutionary technology they have the potential to become. Sabine Hossenfelder has identified two major challenges facing today's quantum computers:

- **Lack of clear demonstrable quantum supremacy.** It has been difficult to find genuine examples of quantum supremacy. Google's claim in 2019 was more hype than substance. It was a toy problem, and the task could be done by a classical computer much faster than originally had been asserted. Some proposed that modeling a key enzyme in nitrogen fixation was too difficult for classical computers thus requiring quantum computers. This claim, however, was refuted when the enzyme was recently modeled to chemical accuracy using a classical computer. Alternatively, the well-known traveling salesman problem with O(n!) scaling is frequently cited as a problem where quantum computers might demonstrate a clear quantum advantage. Recent research, however, shows that a purely quantum approach might not be very helpful here (although a hybrid quantum-classical computing strategy may yield a significant speed up).
- **Major energy and hardware requirements.** Quantum computers require enormous amounts of energy to operate. Much of that comes from the requirement of maintaining extremely low temperatures and/or a hard vacuum. And this cost should only grow as more qubits are incorporated. This makes the energy costs comparable to those of

[49] For details on how to access and use a quantum computer, see Whuley and Floyd Smith, *Quantum Computing for Dummies*, chapter 13.

supercomputer clusters meaning that quantum and classical computers will be competing for the same peak power.

Sabine's analysis is a sobering reality check regarding the current status of quantum computers. But it should be emphasized that her analysis is unfairly pessimistic because the field is very much in its infancy and much progress has been made in recent years. And there is an additional challenge not mentioned by Sabine:

- **Quantum resistant problems.** Some types of problems are classified as quantum resistant, meaning that even quantum computers won't be able to solve them effectively.

On the plus side, quantum resistant problems can serve as a basis for post-quantum encryption. That will be essential once quantum computers become powerful enough to break current encryption methods.

While we wait for the arrival of truly powerful quantum computers, there is one near-term strategy to consider: hybrid quantum-classical computing. This involves alternately using quantum computers and classical computers to tackle computational problems, allowing users to get the best of both worlds. The quantum processor would be used for tasks that benefit from quantum parallelism while relying on classical computation for iterative refinement and optimization. Several important quantum algorithms (e.g., variational quantum eigensolver, quantum approximate optimization algorithm, and quantum phase estimation) are naturally suited for this approach.

And now for the final question: how long will it be before we have full-scale machines capable of running general quantum computational algorithms? Nobody truly knows but estimates range from an optimistic 10 years to a more pessimistic 30-year time frame. Nevertheless, there is good reason to lean more toward the optimistic side. For example, IBM has published their roadmap for what they have and hope to achieve with their quantum computers (see this 2022 summary of the roadmap). By the end of the decade, their Starling processors are expected to possess 200 error-corrected qubits with 100 million gates.

In the end, only time will tell where things land. Likely, this paper will need to be updated every few years as new developments occur.

## Resources on Quantum Computing

**For Beginners:**


Brian Clegg, *Quantum Computing: The Transformative Technology of the Qubit Revolution*, Icon Books, London, UK 2021.

Whuley and Floyd Smith, *Quantum Computing for Dummies*, John Wiley & Sons, Inc., Hoboken, NJ 2024.

Michio Kaku, *Quantum Supremacy: How the Quantum Computer Revolution Will Change Everything*, Double Day, New York, NY 2023.

John Gribbin, *Computing with Quantum Cats: From Colossus to Qubits*, Prometheus Books, Amherst, NY 2014.

Mark Jackson, "The Race for Quantum Computing," Thinking Beyond (video), April 2024.

Dominic Walliman, "The Map of Quantum Computing - Quantum Computing Explained," Domain of Science (video), December 2021.


**Advanced Resources:**


Chris Bernhard, *Quantum Computing For Everyone*, The MIT Press, Cambridge, MA 2019.

Andrew Glassner, *Quantum Computing: From Concepts to Code*, No Starch Press, Inc., San Francisco, CA 2025.

## Appendix: Key Founders of Quantum Mechanics

### Founders/Developers of Quantum Mechanics

- **Max Plank.** Quantization of vibrations in black body radiation (1900).
- **Albert Einstein.** Quantization of light in the photoelectric effect (1905).
- **Niels Bohr.** Explained atomic structure based on quantum ideas (1913).
- **Louis de Broglie.** Wave-particle duality. Matter acts as a wave (1924).
- **Wolfgang Pauli.** Pauli exclusion principle (1925).
- **Erwin Schrödinger.** Schrödinger's equation (1926).
- **Max Born.** Probabilistic interpretation of quantum mechanics (1926).
- **Werner Heisenberg.** Uncertainty principle (1927).
- **Paul Dirac.** Dirac's equation (1928).
- **John von Neumann.** Rigorous mathematical framework for quantum mechanics (1932).
- **Richard Feynman.** Path integral formulation of quantum mechanics, the theory of quantum electrodynamics, and development of Feynman diagrams (1947-1948).
- **David Bohm.** Pilot wave interpretation of quantum mechanics (1952).

## Appendix: Key Quantum Concepts

- **Quantum Tunneling.** A particle may cross a barrier even when classically it lacks enough energy to do so.
- **Superposition.** This principle allows a quantum particle or system to represent multiple possibilities simultaneously. A qubit can be in a state of 0, 1, or both 0 and 1 at the same time.
- **Quantum Entanglement.** This is a process where multiple quantum particles become correlated in ways that classical probability does not allow. When qubits are entangled, the state of one qubit can instantly influence the state of another, regardless of the distance between them.
- **Interference.** This phenomenon occurs when quantum states interact, leading to the amplification or cancellation of probabilities.
- **Decoherence.** This is the process by which quantum particles lose their quantum state and transition into classical states due to interactions with their environment.

## Appendix: Quantum Circuits

What do classical and quantum computers look like "under the hood?" How do they do the things they do? Thankfully, we do not need to understand all the intricate details but having a basic understanding of the main principles behind how these machines work can be very useful.

For classical computers, each computational function, such as adding two binary numbers, must be expressed as a series of logical operations performed using logical gates and associated connections. These operations can be depicted as a circuit diagram. **Figure 8** shows a very simple circuit diagram. This circuit manipulates three input bits (A, B, and Cin) and returns two output bits (S and Cout). Using standardized symbols for the logical gates, this particular example uses two XOR gates, two AND gates, and one OR gate.

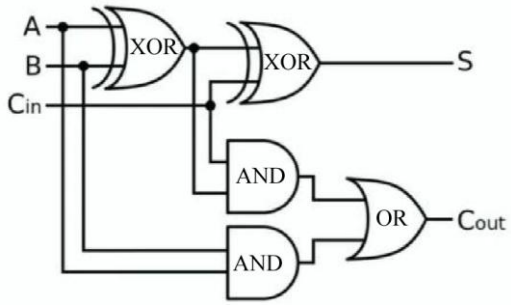


**Figure 8: Example classical computer circuit diagram.**
Credit: ElectronicsAndYou.com (labels added).

In the same way, engineers for quantum computers use quantum circuit diagrams to lay out all the logic and hardware needed to implement a particular function. Ultimately, these diagrams guide engineers in designing the physical hardware. While these circuit diagrams are qubit independent, details of the implementation depend heavily on the nature of qubit being utilized.

Quantum circuits differ considerably from their classical counterparts in both depiction and organization. Consider the example quantum circuit diagram shown in **Figure 9**. This example utilizes three qubits denoted, $\mathbf{q_0}$, $\mathbf{q_1}$, and $\mathbf{q_2}$. The order of operations in quantum circuits is read from left to right. Horizontal lines follow the behavior of individual qubits. Squares with letters designate specific gates with the letter identifying the specific type of gate. For example, H is a Hadamard gate and X is a Pauli-X gate. The final dials represent measurement where the qubit is converted into a definite state that can be read as a classical bit.

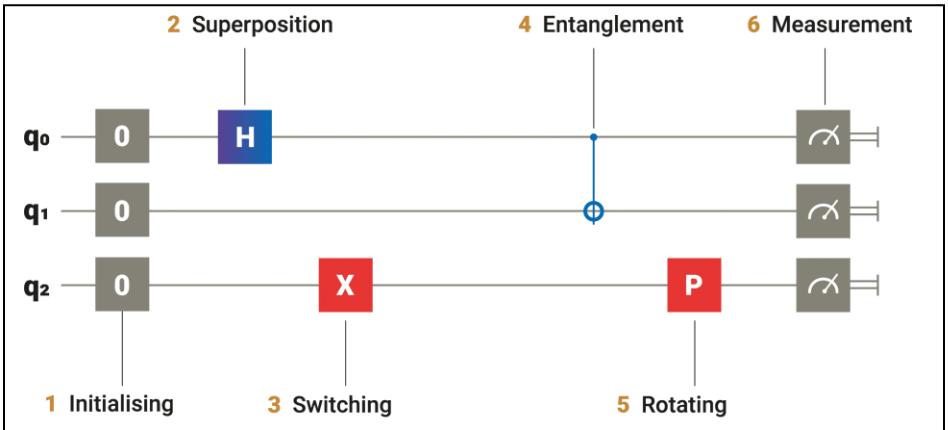


**Figure 9: Example quantum circuit diagram.**
Credit: TUDelft.

The example quantum circuit can be broken down into six distinct steps:

1) **Initializing.** All three qubits are initialized to $|0\rangle$.
2) **Superposition.** The Hadamard gate (H) puts qubit $\mathbf{q_0}$ in superposition of $|0\rangle$ and $|1\rangle$.

3) **Switching.** A Pauli-X gate (or simply X-gate) is used to flip the state of a qubit from $|0\rangle$ to $|1\rangle$ or $|1\rangle$ to $|0\rangle$. Since $\mathbf{q_2}$ is initially in the $|0\rangle$ state, it is switched to $|1\rangle$.

4) **Entanglement.** The vertical line represents a CNOT-gate, which is conditional. If qubit $\mathbf{q_0}$'s state equals $|1\rangle$, it applies the same transformation as the X-gate to qubit $\mathbf{q_1}$. Because qubit $\mathbf{q_0}$ is in a superposition state, this causes qubits $\mathbf{q_0}$ and $\mathbf{q_1}$ to be entangled.

5) **Rotating.** This gate, known as the phase gate (or simply P-gate), rotates the vector representing a qubit's state around the z-axis (of the Bloch sphere shown in **Figure 6**) by a specified amount.

6) **Measurement.** Measuring a qubit makes it collapse into one defined state (known as an eigenstate). The measured value we get reflects this state.

Once the qubits have been measured, they can be read by a classical computer as ordinary bits.

In our discussion on quantum algorithms, we discussed the Deutsch algorithm, the earliest and simplest quantum algorithm.[50] **Figure 10** shows the Deutsch algorithm in the form of a quantum circuit diagram. This algorithm is designed to determine if a 1-bit oracle function should be classified as either balanced or complete.

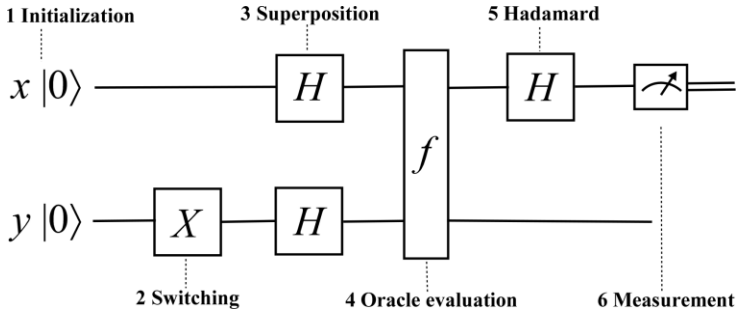


**Figure 10: Quantum circuit diagram for the Deutsch algorithm.**
Credit: John Millam.

For this algorithm, we start with two qubits denoted *x* and *y* and proceed through six steps.

1) **Initialization.** Both qubits are given an initial state of $|0\rangle$.

2) **Switching.** The Pauli-X gate flips the state of a qubit *y* from $|0\rangle$ to $|1\rangle$.

3) **Superposition.** The Hadamard gates transform each qubit into a superposition state.

4) **Oracle evaluation.** The "*f*" gate represents the unknown function whose nature is to be determined. It is evaluated only once in contrast to classical computers that require two evaluations.

5) **Hadamard.** A Hadamard gate is applied to qubit *x*.

6) **Measurement.** The *x* qubit is measured thus forcing it into a definite state that can be translated into a classical bit. Note, the *y* qubit does not need to be measured.

If the returned result is 0, then one concludes that the function should be classified as constant and if 1 then it is balanced.

[50] For an advanced discussion of quantum algorithms, quantum circuits, and the related mathematics, read Andrew Glassner, *Quantum Computing*. This example is drawn from chapter 8. See also, Chris Bernhardt, *Quantum Computing For Everyone*, p. 145-152.