

1. Introduction:

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

Today's advanced lithographic techniques can squeeze fraction of micron wide logic gates and wires onto the surface of silicon chips. Soon they will yield even smaller parts and inevitably reach a point where logic gates are so small that they are made out of only a handful of atoms. On the atomic scale matter obeys the rules of quantum mechanics, which are quite different from the classical rules that determine the properties of conventional logic gates. So if computers are to become smaller in the future, new, *quantum* technology must replace or supplement what we have now. The point is, however, that quantum technology can offer much more than cramming more and more bits to silicon and multiplying the clock-speed of microprocessors. It can support entirely new kind of computation with qualitatively new algorithms based on quantum principles.

Quantum computation is an extremely exciting and rapidly growing field of investigation. An increasing number of researchers with a whole spectrum of different backgrounds, ranging from physics, via computing sciences and information theory to mathematics and philosophy, are involved in researching properties of quantum-based computation. Interplay between mathematics and physics of course has always been beneficial to both types of human activities.

The story of quantum computation started as early as 1982, when the physicist Richard Feynman considered simulation of quantum-mechanical objects by other quantum systems. However, the unusual power of quantum computation was not really anticipated until the 1985 when David Deutsch of the University of Oxford published a crucial theoretical paper in which he described a universal quantum computer. After the Deutsch paper, the hunt was on for something interesting for quantum computers to do. At the time all that could be found were a few rather contrived mathematical problems and the whole issue of quantum computation seemed little more than an academic curiosity. It all changed rather suddenly in 1994 when Peter Shor from AT&T's Bell Laboratories in New Jersey devised the first quantum algorithm that, in principle, can perform efficient factorization. This became a 'killer application' --- something very useful that only a quantum computer could do. Difficulty of factorization underpins security of many common

methods of encryption; for example, RSA --- the most popular public key cryptosystem which is often used to protect electronic bank accounts gets its security from the difficulty of factoring large numbers. Potential use of quantum computation for code-breaking purposes has raised an obvious question --- what about building a quantum computer.

Today's computers are classical, a fact which is actually not entirely obvious. A basis of modern computers rests on semiconductor technology. Transistors, which are the "neurons" of all computers, work by exploiting properties of semiconductors. However, the explanation of how semiconductors function is entirely quantum mechanical in nature: it simply cannot be understood classically. Are we thus to conclude that classical physics cannot explain how classical computers work?! Or are we to say that classical computers are, in fact, quantum computers! The answer to both these questions is yes and no. Yes, classical computers are in a certain, restricted, sense quantum mechanical, because, as far as we understand today, everything is quantum mechanical. No, classical computers, although based on quantum physics, are not fully quantum, because they do not use "quantumness" of matter at the information-theoretical level, where it really matters.

Gordon Moore proposed Moore's law in 1965, which originally stated that processor power and speed would double in size every eighteen months (this was later revised to two years). This law still holds but is starting to falter, and components are getting smaller. Soon they will be so small, being made up of a few atoms that quantum effects will become unavoidable, possibly ending Moore's law. There are ways in which we can use quantum effects to our advantage in a classical sense, but by fully utilizing those effects we can achieve much more. This approach is the basis for quantum computing.

2. The future of computing: classical or quantum?

Computers increasingly pervade our society. This increasing influence is enabled by their ever increasing power, which has roughly doubled every 18 months for the last half-century. The increase in power, in turn, is primarily due to the continuing miniaturization of the elements of which computers are made, resulting in more and more elementary gates with higher and higher clock pulse per unit of silicon, accompanied by less and less energy dissipation per elementary computing event. Roughly, a linear increase in clock speed is accompanied by square increase in elements per silicon unit--so if all elements

compute all of the time, then the dissipated energy per time unit rises cubically (linear times square) in absence of energy decrease per elementary event. The continuing dramatic decrease in dissipated energy per elementary event is what has made Moore's law possible. But there is a foreseeable end to this. There is a minimum quantum of energy dissipation associated with elementary events. This puts a fundamental limit on how far we can go with miniaturization, or does it?

It turns out that only irreversible elementary events (like erasing information) by the laws of thermodynamics necessarily dissipate energy; there is no physics law that requires reversible events (like negation) to dissipate energy. But so far the development of computation machinery is mostly based on the principles of classical physics and irreversible components. At the basic level, however, matter is governed by quantum mechanics, which is reversible. Further miniaturization will very soon reach scales where quantum mechanical effects take over and classical laws cease to apply accurately. The mismatch of computing organization and reality will express itself in friction. Computers will generate gigantic (megawatts) of energy unless their mode of operation becomes quantum mechanical (and thus reversible). That is, harnessing quantum mechanical effects is essential for further miniaturization and hence acceleration of classical computing methods.

There is an added bonus. Once we get involved in quantum effects, it appears we can go further than just miniaturizing classical computers to the quantum scale. Quantum mechanics may actually spawn a *qualitatively new* kind of computing. A kind which profits from quantum effects to boost computation to such an extent that things are achieved that would forever be out of reach of classical computers, even if these could be miniaturized to the same level. The area of quantum computing has a great economical and societal potential.

3. Quantum Mechanics

Quantum mechanics is generally about the novel behaviour of very small things. At this scale matter becomes *quantized*, this means that it can be subdivided no more. Quantum mechanics has never been wrong, it explains why the stars shine, how matter is structured, the periodic table, and countless other phenomena. One day scientists hope to use quantum mechanics to explain everything, but at present the theory remains incomplete as it has not been successfully combined with classical theories of gravity. Some strange effects happen at the quantum scale.

The following are main parts of quantum mechanics that are important for quantum computing:

- Superposition and interference
- Uncertainty
- Entanglement
- Linear algebra
- Dirac notation
- Representing information

3.1 Superposition

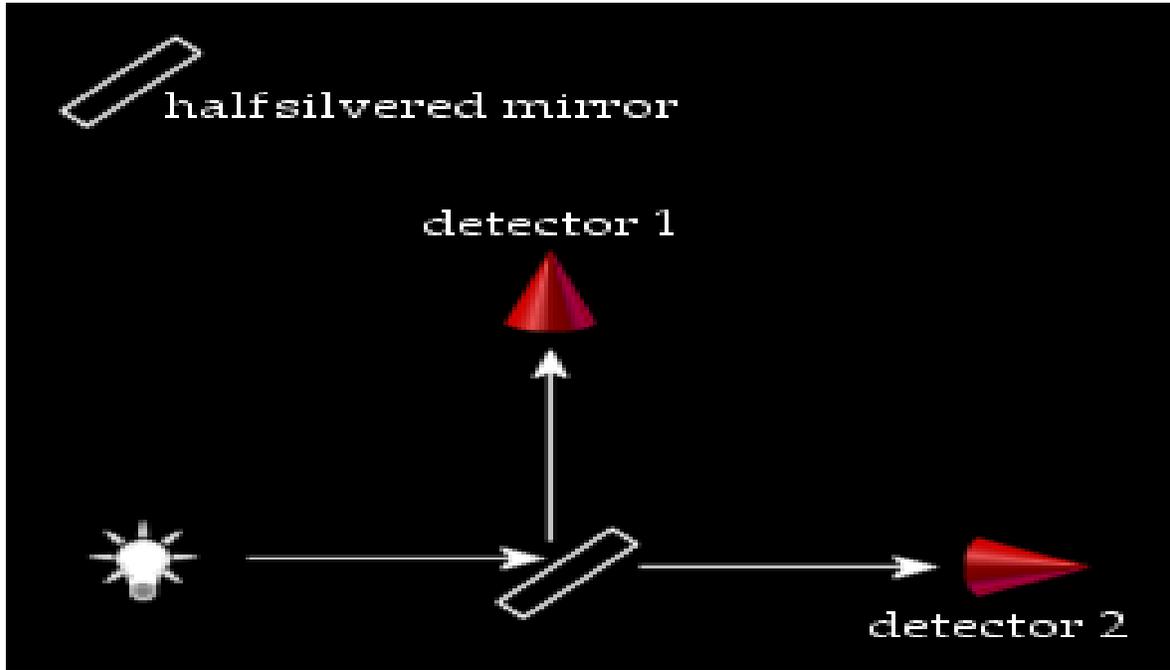
Superposition means a system can be in two or more of its states simultaneously. For example a single particle can be traveling along two different paths at once. This implies that the particle has wave-like properties, which can mean that the waves from the different paths can *interfere* with each other. Interference can cause the particle to act in ways that are impossible to explain without these wave-like properties.

The ability for the particle to be in a superposition is where we get the parallel nature of quantum computing: If each of the states corresponds to a different value then, if we have a superposition of such states and act on the system, we effectively act on all the states simultaneously.

An Example with Silvered Mirrors

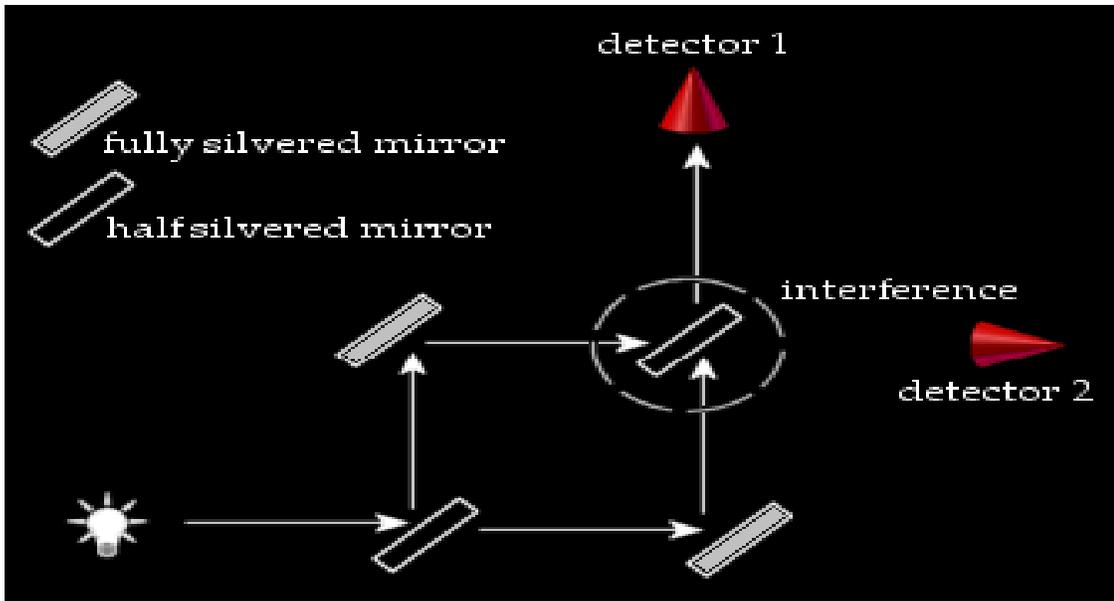
Superposition can be explained by way of a simple example using silvered and half silvered mirrors.

A half silvered mirror reflects half of the light that hits it and transmits the other half of the light through it (figure 3.1). If we send a single photon through this system then this gives us a 50% chance of the light hitting detector 1 and a 50% chance of hitting detector 2. It is tempting to think that the light takes one or the other path, but in fact it takes both! It's just that the photo detector that *measures* the photon first breaks the superposition, so it's the detectors that cause the randomness, not the half silvered mirror.

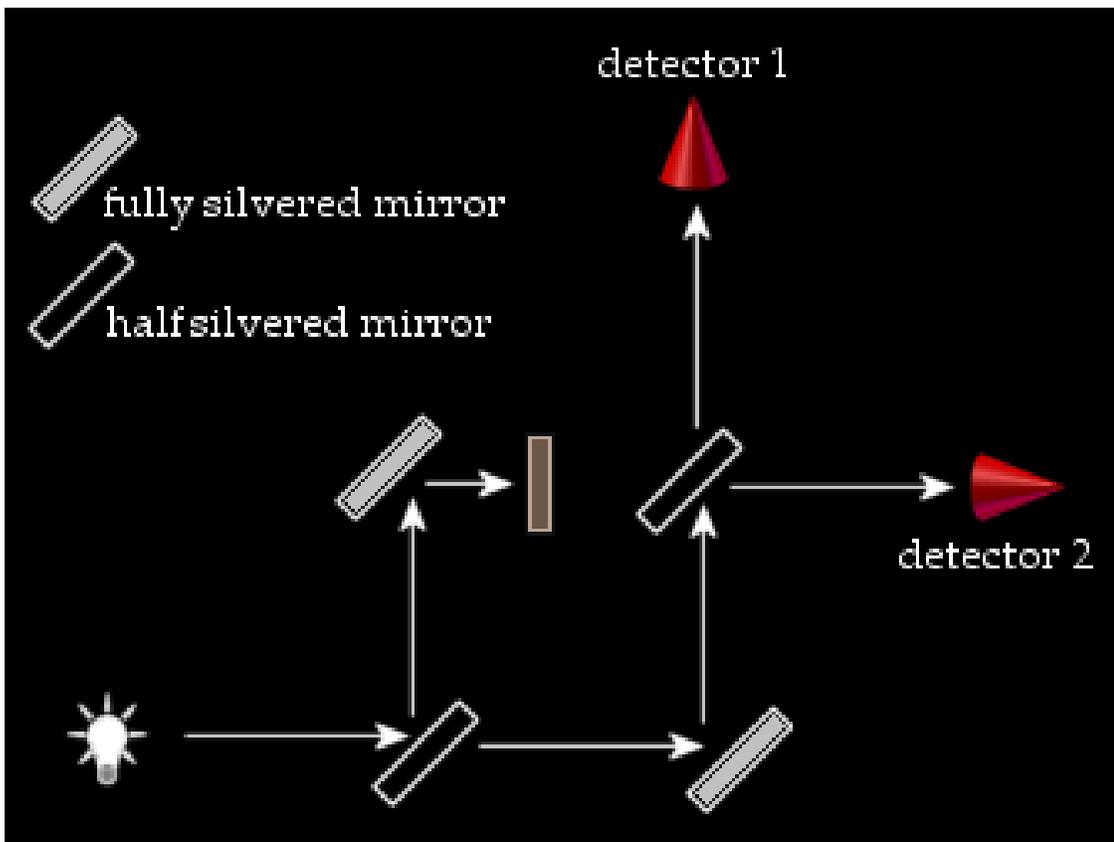


(Fig 3.1)

This can be demonstrated by adding in some fully silvered mirrors and bouncing both parts of the superposed photon (which is at this point is in two places at once) so that they meet and interfere with each other at their meeting point. If another half silvered mirror (figure 3.2) is placed at this meeting point and if light was just particle like. We would expect that the light would behave as before (going either way with 50% probability), but the interference (like wave interference when two stones are thrown into a pond near each other simultaneously) causes the photon to always be detected by detector 1. A third example (figure 3.3) shows clearly that the photons travel both paths because blocking one path will break the superposition and stop the interference.



(Fig 3.2)



(Fig 3.3)

3.2 Uncertainty

The quantum world is irreducibly small so it's impossible to measure a quantum system without having an effect on that system as our measurement device is also quantum mechanical. As a result there is no way of accurately predicting all of the properties of a particle. There is a trade off - the properties occur in complementary pairs (like position and momentum, or vertical spin and horizontal spin) and if we know one property with a high degree of certainty then we must know almost nothing about the other property.

That unknown property's behaviour is essentially random. An example of this is a particle's position and velocity: if we know exactly where it is then we know nothing about how fast it is going. This indeterminacy is exploited in quantum cryptography. It has been postulated (and currently accepted) that particles in fact DO NOT have defined values for unknown properties until they are measured. This is like saying that something does not exist until it is looked at.

3.3 Entanglement

In 1935 Einstein (along with colleagues Podolski and Rosen) demonstrated a paradox (named *EPR* after them) in an attempt to refute the undefined nature of quantum systems. The results of their experiment seemed to show that quantum systems were defined, having *local state* BEFORE measurement. Although the original hypothesis was later proven wrong (i.e. it was proven that quantum systems do not have local state before measurement). The effect they demonstrated was still important, and later became known as *entanglement*.

Entanglement is the ability for pairs of particles to *interact* over any distance instantaneously. Particles don't exactly communicate, but there is a statistical *correlation* between results of measurements on each particle that is hard to understand using classical physics. To become entangled, two particles are allowed to interact; they then separate and, on measuring say, the velocity of one of them (regardless of the distance between them), we can be sure of the value of velocity of the other one (before it is measured). The reason we say that they communicate instantaneously is because they store no local state and only have well defined state once they are measured. Because of this limitation particles can't be used to transmit classical messages faster than the speed of light as we only know the states upon measurement. Entanglement has applications in a wide variety of quantum algorithms and machinery.

3.4 Linear Algebra

Quantum mechanics leans heavily on linear algebra. Some of the concepts of quantum mechanics come from the mathematical formalism, not thought experiments, that's what can give rise to counter intuitive conclusions.

3.5 Dirac Notation

Dirac notation is used for quantum computing. We can represent the states of a quantum system as kets. For example, an electron's spin can be represented as $|0\rangle$ spin up and $|1\rangle$ as spin down. The electron can be thought of as a little magnet, the effect of a charged particle spinning on its axis. When we pass a horizontally traveling electron through an inhomogeneous magnetic field, in say, the vertical direction, the electron either goes up or down. If we then repeat this with the up electron it goes up, with the down electron it goes down. We say the up electron after the first measurement is in the state $|0\rangle$ and the down electron is in state $|1\rangle$.

But, if we take the up electron and pass it through a horizontal field it comes out on one side 50% of the time and on the other side 50% of the time. If we represent these two states as $|+\rangle$ and $|-\rangle$ we can say that the up spin electron was in a superposition of the two states $|+\rangle$ and $|-\rangle$:

$$|0\rangle = \frac{1}{\sqrt{2}}|+\rangle + \frac{1}{\sqrt{2}}|-\rangle$$

such that, when we make a measurement with the field horizontal we project the electron into one or the other of the two states, with equal probabilities 1/2 (given by the square of the amplitudes).

3.6 Representing Information

Quantum mechanical information can be physically realised in many ways. To have something analogous to a classical bit we need a quantum mechanical system with two states only, when measured.

Methods for representing binary information in a way that is capable of exhibiting quantum effects (e.g. entanglement and superposition) are: electron spin, photon direction, polarisation of photons and nuclear spins.

4. Elements of Quantum Computing

Generally we'll think of a quantum computer as a classical computer with a quantum circuit attached to it with some kind of interface between conventional and quantum logic. Since there are only a few things a quantum computer does better than a classical computer it makes sense to do the bulk of the processing on the classical machine.

4.1 Bits and Qubits

These are the "nuts and bolts" of quantum computing. It describes qubits, gates, and circuits. Quantum computers perform operations on qubits which are analogous to conventional bits but they have an additional property in that they can be in a superposition.

A quantum register with 3 qubits can store 8 numbers in superposition simultaneously, and a 250 qubit register holds more numbers (superposed) than there are atoms in the universe. The amount of information stored during the "computational phase" is essentially infinite - it's just that we can't get at it. The inaccessibility of the information is related to quantum measurement: When we attempt to readout a superposition state holding many values the state collapses and we get only one value (the rest get lost). This is tantalising but, in some cases, can be made to work to our computational advantage.

4.1.1 Single Qubits

Classical computers use two discrete states (e.g. states of charging of a capacitor) to represent a unit of information, this state is called a binary digit (or bit for short). A bit has the following two values:

0 and 1.

There is no intermediate state between them, i.e. the value of the bit cannot be in a superposition.

Quantum bits, or *qubits*, can on the other hand be in a state "between" 0 and 1, but only during the computational phase of a quantum operation. When measured, a qubit can become either:

$|0\rangle$ or $|1\rangle$

i.e. we readout 0 or 1. This is the same as saying a spin particle can be in a superposition state but, when measured, it shows only one value. The $| \rangle$ symbolic notation is part of the Dirac notation.

In terms of the above it essentially means the same thing as 0 and 1, just like a classical bit. Generally, a qubit's state during the computational phase is represented by a linear combination of states otherwise called a superposition state.

$$\alpha|0\rangle + \beta|1\rangle.$$

Here α and β are the probability amplitudes. They can be used to calculate the probabilities of the system jumping into $|0\rangle$ or $|1\rangle$ following a measurement or readout operation. There may be, say a 25% chance a 0 is measured and a 75% chance a 1 is measured. The percentages must add to 100%. In terms of their representation qubits must satisfy:

$$|\alpha|^2 + |\beta|^2 = 1.$$

This the same thing as saying the probabilities adds to 100%.

Once the qubit is measured it will remain in that state if the same measurement is repeated provided the system remains closed between measurements. The probability that the qubit's state, when in a superposition, will collapse to states $|0\rangle$ or $|1\rangle$ is

$$|\alpha|^2 \text{ for } |0\rangle$$

and

$$|\beta|^2 \text{ for } |1\rangle.$$

$|0\rangle$ and $|1\rangle$ are actually vectors, they are called the computational basis states that form an orthonormal basis for the vector space \mathbb{C}^2 .

The state vector $|\Psi\rangle$ of a quantum system describes the state at any point in time of the entire system. Our state vector in the case of one qubit is:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle.$$

The α and β might vary with time as the state evolves during the computation but the sum of the squares of α and β must always be equal to 1.

Quantum computing also commonly uses

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \text{ and } \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

basis for \mathbb{C}^2 , which is often shortened to just $|+\rangle$, and $|-\rangle$. These bases are sometimes represented with arrows which are described below, and are referred to as *rectilinear* and *diagonal* which can say refer to the polarisation of a photon. You may find these notational conventions being used:

$$|0\rangle = |\rightarrow\rangle.$$

$$|1\rangle = |\uparrow\rangle.$$

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle = |\nearrow\rangle.$$

$$\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle = |\nwarrow\rangle.$$

The sign in the middle of the two values can change, which affects the internal evolution of the qubit, not the outcome of a measurement. When measuring in the basis $\{|0\rangle, |1\rangle\}$ the sign is actually the *relative phase* of the qubit. So,

$$\alpha|0\rangle + \beta|1\rangle$$

and

$$\alpha|0\rangle - \beta|1\rangle$$

have the same output values and probabilities but behave differently during the computational phase. Formally we say they differ by a relative phase factor. So in the case of the qubits above they differ by a phase factor of -1. It is called a phase factor because it always has magnitude 1 and so its value, as a complex number, is determined entirely by the phase.

4.1.2 The Ket $| \rangle$

Part of Dirac's notation is the ket ($| \rangle$). The ket is just a notation for a vector. The state of a single qubit is a unit vector in \mathbb{C}^2 . So,

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

is a vector, and is written as:

$$\alpha|0\rangle + \beta|1\rangle$$

with

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and

$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

4.1.3 Multiple Qubits

The potential amount of information available during the computational phase grows exponentially with the size of the system, i.e. the number of qubits. This is because if we have n qubits the number of basis states is 2^n . E.g. if we have two qubits, forming a quantum register then there are four ($=2^2$) computational basis states: forming,

$$|00\rangle, |01\rangle, |10\rangle, \text{ and } |11\rangle.$$

Here $|01\rangle$ means that qubit 1 is in state $|0\rangle$ and qubit 2 is in state $|1\rangle$, etc. We actually have $|01\rangle = |0\rangle \otimes |1\rangle$, where \otimes is the tensor product.

Like a single qubit, the two qubit register can exist in a superposition of the four states (below we change the notation for the complex coefficients, i.e. probability amplitudes):

$$|\Psi\rangle = \alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle.$$

All of the probabilities must sum to 1, formally for the general case of n qubits this can be written as:

$$\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1.$$

Tensor Products

A decomposition into single qubits of a multi-qubit system can be represented by a tensor product, \otimes

$$\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle).$$

A tensor product can also be used to combine different qubits.

$$(\alpha_0|0\rangle + \alpha_1|1\rangle) \otimes (\beta_0|0\rangle + \beta_1|1\rangle) = \alpha_0\beta_0|00\rangle + \alpha_0\beta_1|01\rangle + \alpha_1\beta_0|10\rangle + \alpha_1\beta_1|11\rangle.$$

5. Entangled States

Subatomic particles can be entangled; this means that they are connected, regardless of distance. Their effect on each other upon measurement is instantaneous. This can be useful for computational purposes.

Consider the following state (which is not entangled):

$$\frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$$

it can be expanded to:

$$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|01\rangle + 0|10\rangle + 0|11\rangle.$$

Upon measuring the first qubit (a partial measurement) we get 0 100% of the time and the state of the second qubit becomes:

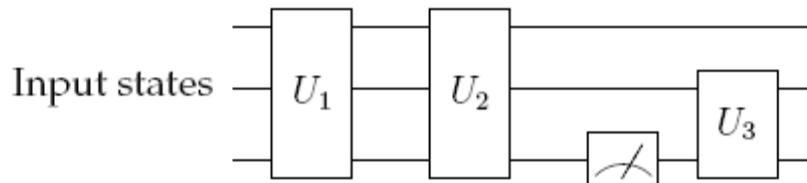
$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

giving us equal probability for a 0 or a 1.

If we try this on an entangled state we find that the results for the qubits are correlated. This type of correlation can be used in a variety of ways in application to the first or second qubit to give us correlations that are strongly statistically connected. This is a distinct advantage over classical computation. Measuring entangled states accounts for the correlations between them.

6. Quantum Circuits

If we take a quantum state, representing one or more qubits, and apply a sequence of unitary operators (quantum gates). The result is a quantum circuit. We now take a register and let gates act on qubits, in analogy to a conventional Circuit



This gives us a simple form of quantum circuit (above) which is a series of operations and measurements on the state of n -qubits. Each operation is unitary and can be described by an $2^n \times 2^n$ matrix. Each of the lines is an abstract *wire*, the boxes containing U_n are *quantum logic gates* (or a series of gates) and the meter symbol is a measurement. Together, the gates, wires, input, and output mechanisms implement quantum algorithms.

Unlike classical circuits which can contain loops, quantum circuits are "one shot circuits" that just run once from left to right (and are special purpose: i.e. we have a different circuit for each algorithm). It is always possible to rearrange quantum circuits so that all the measurements are done at the end of the circuit.

Important Properties of Quantum Circuits

Quantum circuit diagrams have the following constraints which make them different from classical diagrams.

1. They are acyclic (no loops).
2. No FANIN, as FANIN implies that the circuit is NOT reversible, and therefore not unitary.
3. No FANOUT, as we can't copy a qubit's state during the computational phase because of the no-cloning theorem.

7. Quantum Gates

7.1 Single Qubit Gates

Just as a single qubit can be represented by a column vector, a gate acting on the qubit can be represented by a 2 x 2 matrix. The quantum equivalent of a NOT gate, for example, has the following form:

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

The only constraint these gates have to satisfy (as required by quantum mechanics) is that they have to be unitary, where a unitary matrix is one that satisfies the condition underneath. This allows for a lot of potential gates.

$$U^\dagger U = I.$$

The matrix acts as a quantum operator on a qubit. The operator's matrix must be unitary because the resultant values must satisfy the normalisation condition. Unitarity implies that the probability amplitudes must still sum to 1. If (before the gate is applied)

$$|\alpha|^2 + |\beta|^2 = 1$$

then, after the gate is applied:

$$|\alpha'|^2 + |\beta'|^2 = 1$$

where α' and β' are the values for the probability amplitudes for the qubit after the operation has been applied.

Pauli I Gate

This is the identity gate.

$$\sigma_0 = I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

which gives us the following:

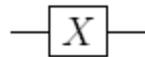
$$|0\rangle \rightarrow I \rightarrow |0\rangle,$$

$$|1\rangle \rightarrow I \rightarrow |1\rangle,$$

$$\alpha|0\rangle + \beta|1\rangle \rightarrow I \rightarrow \alpha|0\rangle + \beta|1\rangle.$$

Pauli X Gate

The Pauli X gate is a quantum NOT gate.



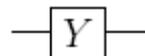
$$\sigma_1 = \sigma_X = X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

which gives us the following:

$$|0\rangle \rightarrow X \rightarrow |1\rangle,$$

$$|1\rangle \rightarrow X \rightarrow |0\rangle,$$

$$\alpha|0\rangle + \beta|1\rangle \rightarrow X \rightarrow \beta|0\rangle + \alpha|1\rangle.$$

Pauli Y Gate

$$\sigma_2 = \sigma_Y = Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

which gives us the following:

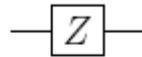
$$|0\rangle \rightarrow Y \rightarrow i|1\rangle,$$

$$|1\rangle \rightarrow Y \rightarrow -i|0\rangle,$$

$$\alpha|0\rangle + \beta|1\rangle \rightarrow Y \rightarrow -\beta i|0\rangle + \alpha i|1\rangle.$$

Pauli Z Gate

This gate flips a qubit's sign, i.e. changes the relative phase by a factor of -1.



$$\sigma_3 = \sigma_Y = Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

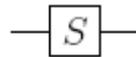
which gives us the following:

$$|0\rangle \rightarrow Z \rightarrow |0\rangle,$$

$$|1\rangle \rightarrow Z \rightarrow -|1\rangle,$$

$$\alpha|0\rangle + \beta|1\rangle \rightarrow Z \rightarrow \alpha|0\rangle - \beta|1\rangle.$$

Phase Gate



$$S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

which gives us the following:

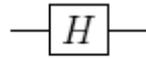
$$|0\rangle \rightarrow S \rightarrow |0\rangle,$$

$$|1\rangle \rightarrow S \rightarrow i|1\rangle,$$

$$\alpha|0\rangle + \beta|1\rangle \rightarrow S \rightarrow \alpha|0\rangle + \beta i|1\rangle.$$

Hadamard Gate

Sometimes called the *square root of NOT gate*, it turns a $|0\rangle$ or a $|1\rangle$ into a superposition (note the different sign). This gate is one of the most important in quantum computing.



$$H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

which gives us the following:

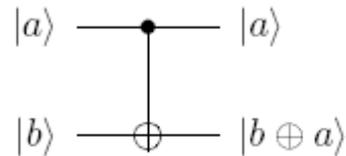
$$|0\rangle \rightarrow H \rightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle),$$

$$|1\rangle \rightarrow H \rightarrow \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle),$$

$$\alpha|0\rangle + \beta|1\rangle \rightarrow H \rightarrow \alpha \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) + \beta \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right).$$

7.2 Multi Qubit Gates

A true quantum gate must be reversible, this requires that multi qubit gates use a control line, where the control line is unaffected by the unitary transformation.



In the case of the CNOT gate, the classical XOR with the input on the b line and the control line a . Because it is a two qubit gate it is represented by a 4 x 4 matrix:

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

which gives the following:

$$|00\rangle \rightarrow \text{CNOT} \rightarrow |00\rangle,$$

$$|01\rangle \rightarrow \text{CNOT} \rightarrow |01\rangle,$$

$$|10\rangle \rightarrow \text{CNOT} \rightarrow |11\rangle,$$

$$|11\rangle \rightarrow \text{CNOT} \rightarrow |10\rangle,$$

$$(\alpha|0\rangle + \beta|1\rangle)|1\rangle \rightarrow \text{CNOT} \rightarrow \alpha|00\rangle + \beta|10\rangle),$$

$$|0\rangle(\alpha|0\rangle + \beta|1\rangle) \rightarrow \text{CNOT} \rightarrow \alpha|00\rangle + \beta|01\rangle),$$

$$|1\rangle(\alpha|0\rangle + \beta|1\rangle) \rightarrow \text{CNOT} \rightarrow \alpha|11\rangle + \beta|10\rangle).$$

Qubit Two NOT Gate

As distinct from the CNOT gate we have a NOT₂ gate, which just does NOT on qubit two and has the following matrix representation:

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

which gives the following:

$$|00\rangle \rightarrow \text{NOT}_2 \rightarrow |01\rangle,$$

$$|01\rangle \rightarrow \text{NOT}_2 \rightarrow |00\rangle,$$

$$|10\rangle \rightarrow \text{NOT}_2 \rightarrow |11\rangle,$$

$$|11\rangle \rightarrow \text{NOT}_2 \rightarrow |10\rangle.$$

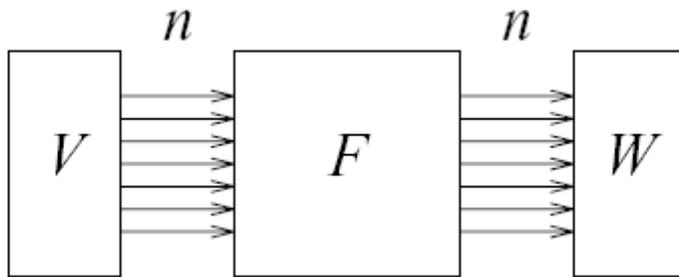
$$\begin{aligned} \text{NOT}_2 &= I \otimes X = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ &= \begin{pmatrix} 1 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & 0 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ 0 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & 1 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{pmatrix}. \end{aligned}$$

So, as well as using the NOT₂ notation we can use the tensor product of Pauli gates on qubits one and two, shown below:

$$\begin{aligned}
|00\rangle &\rightarrow I \otimes X \rightarrow |01\rangle, \\
|01\rangle &\rightarrow I \otimes X \rightarrow |00\rangle, \\
|10\rangle &\rightarrow I \otimes X \rightarrow |11\rangle, \\
|11\rangle &\rightarrow I \otimes X \rightarrow |10\rangle.
\end{aligned}$$

8. Quantum Computer

A quantum computer looks like this, taking n input qubits, the register V , and producing n output qubits, the register W :



The input register can be prepared as a superposition of states, e.g. an equal superposition of *all* integers from 0 to 2^n :

$$V = \sum_i^{2^n} 1/\sqrt{2} (|0_i\rangle + |1_i\rangle)$$

The computer then calculates in parallel the function applied to all 2^n integers simultaneously. From QMP (Quantum Measurement Postulate), when we measure W , it will choose a Boolean for each bit of the output register according to the resulting entangled wave function of the output qubits. Design F so that it maximizes the probability that the output we measure is the answer we want.

Measuring the output collapses the wave function: get Boolean values for all the qubits in W . The result is one of the possible outputs.

Imagine that F is (integer) square root $W = \sqrt{V}$. Prepare V as the superposition of all integers from 0 to 2^n , run the computer, then measure W . Result will square root of *some* number between 0 and 2^n . The square root of *any* such number, with equal probability. F calculates the square roots of all the integers in parallel,

but QMP says we can only find out about one. For real problems, arrange F so the probability amplitudes of the output state strongly favor the desired output from F .

Quantum computers are like huge multidimensional arrays of slits that generate interference patterns in the wave functions. Design the array right, and the pattern solves your problem.

A quantum computer is *probabilistic*: we may need to run it multiple times before we get the answer we want.

8.1 What quantum computers can do:

The biggest success so far -- and the event which ignited the current explosive growth of the field of quantum computing -- was Peter Shor's 1994 discovery of an efficient quantum algorithm for finding the prime factors (factoring) of large integers.

By making clever use of superpositions, interference, quantum parallelism, and some classical number theory, Shor's algorithm finds a factor of a number N in time roughly the square of the length of the input (which is $\log N$ bits). In contrast, every known classical algorithm requires exponential time to factor. Since factoring is one of the most elementary aspects of number theory, the oldest mathematical discipline, and centuries of efforts by the greatest mathematicians have not yielded better methods, it is widely believed that such better methods either do not exist or are prohibitively difficult to find.

In fact, this belief underlies most of current public-key cryptography, notably the RSA system, ubiquitously used on the Internet and in the financial world. Such crypto-systems can be broken if one can factor large numbers fast. Accordingly, the advent of quantum computing compromises all such systems: if a quantum computer can be built, then most of current cryptography becomes totally insecure, and, for example, electronic money can be forged.

What quantum computing takes away with one hand (classical public-key crypto), it gives back in another form with the other (quantum secret-key crypto). In 1984, Bennett and Brassard found a scheme which allowed two distant parties to obtain a shared secret key via quantum mechanical communication. Their scheme was always believed to be fully secure against any type of spy or eavesdropper, and recently this has indeed been formally proven. On the other hand, some other parts of electronic transactions, like unforgeable signatures, appear to be beyond the power of quantum methods.

A third application is Grover's 1996 algorithm for searching databases. Consider finding some specific record in a large unordered database of N items. classically, there is no smarter method than just to go through all records

sequentially, which will require expected $N / 2$ time steps for a record in general position. Grover's algorithm, however, uses quantum superpositions to examine all records "at the same time", and finds the desired record in roughly \sqrt{N} steps.

Examining a 10^{12} records with unit microsecond probes, this is the difference between about two months of computing and one second of computing! His algorithm also allows to solve the widespread and notoriously hard NP-complete problems (such as the traveling salesman problem) quadratically faster than known classical methods--reducing say exponential time with exponent N to exponential time with exponent $N / 2$.

A fourth application was initially conceived and primarily developed in collaboration with the CWI (Centrum voor Wiskunde en Informatica, University of Amsterdam) group. It deals with the setting where two separated parties, Alice and Bob, want to compute some function $f(x,y)$ depending on x (only known to Alice) and y (only known to Bob).

A simple scheme would be for Alice to send her x to Bob and then let Bob do all the work by himself, but this may take a lot of bits of communication and often there are much more clever schemes requiring less communication. The field of *communication complexity* examines the optimal number of bits that have to be communicated in order to compute the function at hand. What happens if we generalize this setting to the quantum world and allow Alice and Bob the use of quantum computers and qubit-communication?

It turns out that some tasks can be solved with significantly less communication if we allow such quantization. We have obtained similar advantages by sticking to classical communication, but allowing Alice and Bob the use of pre-established "entangled" qubits. Both approaches beat the limits provable for just classical communication.

The above developments suggested the vision that *all* computation can be enormously speeded up by quantum computers. But not so! CWI's researchers obtained strong and general *limitations* of quantum computers as well. Grover's algorithm is quadratically faster than classical search algorithms. It was already known that such a quadratic speed-up is the best quantum computers can achieve for searching a database, so exponential speed-ups cannot be obtained for this problem.

CWI-researchers recently showed that the same holds for *all* problems in the database-setting of Grover's algorithm: for all such problems, quantum computers can be at most polynomially faster than classical computers.

Limiting results like the above, of course, do not preclude exponential speed-ups in different settings, like Shor's, or a clever future setting as yet unknown. Exploring this potential of quantum computation remains an exciting and important task for computer scientists and physicists alike.

8.2 How Quantum Computers Do It:

The above results are very promising, but so far mostly theory. How about actually building quantum computers which can run the fast algorithms like Shor's, Grover's, or CWI's? To date only very small quantum algorithms (and slightly bigger quantum crypto devices) have been implemented, but the physical realization of quantum computers is still in its infancy.

The main problem is that quantum superpositions are extremely vulnerable and any interactions with its environment will quickly cause errors, which degrade the performance of the computer. Quantum versions of error-correcting codes have been developed recently which to a large extent solve this problem in theory, but not yet in the brittle practice of the physical lab (let alone the brittle practice of our desktops).

This is related to development of Quantum Information Theory--the quantum extension of classical information theory. CWI's group has contributed to this research, and to related notions of the information in individual quantum states: Quantum Kolmogorov Complexity.

Building large quantum computers presents formidable problems to experimental physicists reminiscent of the initial barriers to classical computing: unreliable components, physically large components, memory, organization, communication, programming. The theory of quantum mechanics is currently extended, partially by CWI research, in particular with respect to the algebraic analysis of "quantum entanglement"--a vital notion in many quantum algorithms, apparently not yet thoroughly investigated in quantum theory.

9. Conclusion and Future Prospects

The laws of quantum mechanics imply a different kind of information processing to the traditional one based on the laws of classical physics. The central difference, as we emphasised, was in the fact that quantum mechanics allows physical systems to be in an entangled state, a phenomenon non-existent in classical physics. This leads to a quantum computer being able to solve certain tasks faster than its classical counterpart.

More importantly, factorization of natural numbers into primes can be performed efficiently on a quantum computer using Shor's algorithm, whereas it is at present considered to be intractable on a classical computer. However, to realize a quantum computer (or indeed any other computer) we have to have a physical medium in which to store and manipulate information.

It is here that quantum information becomes very fragile and it turns out that the task of its storage and manipulation requires a lot of experimental ingenuity.

Linear ion trap, one of the more promising proposals for a physically realizable quantum computer. Here information is stored into electronic states of ions, which are in turn confined to a linear trap and cooled to their ground state of motion. Laser light is then used to manipulate information in the form of different electronic transitions. However, the uncontrollable interactions of ions with their environment induce various errors known as decoherence (such as e.g. spontaneous emission in ions) and thus severely limit the power of computation.

Quantum error correction leads to the notion of fault tolerant quantum computation, which is a method of performing reliable quantum computation using unreliable basic components (e.g. gates) providing that the error rate in this components is below a certain allowed limit. Much theoretical work has been undertaken in this area at the moment and there is now a good understanding of its powers and limitations. The main task is now with the experimentalists to try to build the first fully functional quantum computer, although it should be noted that none of the present implementations appear to allow long or large scale quantum computations and a breakthrough in technology might be needed.

Despite the fact that at present large computational tasks seem to lie in the remote future, there is a lot of interesting and fundamental physics that can be done almost immediately with the present technology. A number of practical information transfer protocols use methods of quantum computation. One example is teleportation involving two parties usually referred to as Alice and Bob. Initially Alice and Bob share an entangled state of two qubits (each one having a single qubit) Alice then receives a qubit in a certain (to her unknown) state which she wants to “transmit” this state to Bob without actually sending the particle to him. She can do this by performing a simple quantum computation on her side, and communicating its result to Bob. Bob then performs the appropriate quantum computation on his side, after which his qubit assumes the state of the Alice’s qubit and the teleportation is achieved.

Since entangled states are non-existent in classical physics this kind of protocol is impossible, leading to another advantage of quantum information transfer over its classical analogue. An extension of this idea leads to more than two users, say N , this time sharing entangled states of N qubits. If each of the users does a particular quantum computation locally, and then they all communicate their results to each other, then more can be achieved than if they did not share any entanglement in the first place. This idea is known as distributed quantum computation and is currently being developed. There is a number of other interesting protocols and applications of quantum computation that have either been achieved or are within experimental reach.

We can hope that quantum factorization and other large and important quantum

computations will be realized eventually. Fortunately, there is a vast amount of effort and ingenuity being applied to these problems, and the future possibility of a fully functioning quantum computer still remains very much alive.

En route to this realization, we will discover a great deal of new physics involving entanglement, decoherence and the preservation of quantum superpositions.

References

<http://www.consciousness.arizona.edu/quantum/Library/qmlecture1.htm>

<http://www.cse.iitd.ernet.in/~suban/quantum/lectures/lecture1.pdf>

<http://www.Qubit.org/library/intros/comp/comp.html>

<http://www.sra.itc.it/people/serafini/quantum-computing/20001006.ps>

<http://www.cs.ualberta.ca/~bulitko/qc/schedule/qcss-notes.pdf>

<http://www.cl.cam.ac.uk/Teaching/current/QuantComp/>

<http://www.indiqosim.com/tutorials/communication/t3s2.htm>

<http://www.consciousness.arizona.edu/Quantum/>

<http://pages.pomona.edu/~jbm04747/courses/fall2001/cs10/lectures/>

<http://www.qinfo.org/people/nielsen/qicss.html>

<http://xxx.lanl.gov/archive/quant-ph>