Logical AND

$$00 \rightarrow 0 \; 01 \rightarrow 0 \; 10 \rightarrow 0 \; 11 \rightarrow 1$$

Logical XOR

$$00 \rightarrow 0 \; 01 \rightarrow 1 \; 10 \rightarrow 1 \; 11 \rightarrow 0$$

$$|00\rangle \rightarrow |00\rangle \; |01\rangle \rightarrow |00\rangle \; |10\rangle \rightarrow |00\rangle \; |11\rangle \rightarrow |01\rangle$$

Using the classical gate analog doesn't work, since there are three equivalent output states with different input states. Therefore there is no unitary matrix that would represent this transformation.

$$|00\rangle \rightarrow |a0\rangle \; |01\rangle \rightarrow |b0\rangle \; |10\rangle \rightarrow |c0\rangle \; |11\rangle \rightarrow |01\rangle$$

Unitary transformation preserves orthogonality.

Even varying the first bit (we cannot modify the bit that holds our computation) does not help since there are no three single qubits that are orthogonal (and all the input states are).

Unitary transformations do not destroy information.

Quantum gates must be *reversible*.

The problem with these approaches has been that unitary transformations are reversible, the computation can be backtracked to the original state. In other words, unitary transformations do not destroy information.

So to create a working quantum gate, the gate has to be reversible. Reversible computing has been proposed for classical computers too, but quantum gates require it.

$$U_{AND}|x_1, x_2, y\rangle = |x_1, x_2, y \oplus (x_1 \wedge x_2)\rangle$$

$$\begin{array}{llll} |000\rangle \rightarrow |000\rangle & |010\rangle \rightarrow |010\rangle & |100\rangle \rightarrow |100\rangle & |110\rangle \rightarrow |111\rangle \\ |001\rangle \rightarrow |001\rangle & |011\rangle \rightarrow |011\rangle & |101\rangle \rightarrow |101\rangle & |111\rangle \rightarrow |110\rangle \end{array}$$

So we will have to add a third bit. The two first bits will stay the same and third will hold the result of the computation.

However since the quantum gate needs to be an unitary transformation, defining the third output bit simply as the AND of first two bits won't work, we would again get multiple equivelant output states from orthogonal input states.

But defining the third output as an XOR of the AND and third input bit works. It is not a problem from a practical point either since this bit isn't data and we can get it from a sink of preset qubits.

Unitary transformations are linear:

$$U_{AND}(\alpha|s\rangle + \beta|t\rangle) = \alpha(U_{AND}|s\rangle) + \beta(U_{AND}|t\rangle)$$

The final step to define our quantum version of classical AND is to ensure linearity; unitary transformations should work for other states than our predefined orthogonal basis too.

Photons

Ions

Superconducting loops

Knowing the mathematical theory of quantum gates doesn't actually help us much in implementing a quantum computer. Several attempts have been made, none of which is yet determined to be any better than others.

One is simply using photons as qubits. The transformations are done by changing the polarization of the photons. Second could be using a matrix of ionized atoms. The transformations can be implemented by a combination of laser beams, hitting multiple ions at the same time.

Third and more exotic example is using superconductive loops. A group of researchers found in 2005 that they can implement a CNOT gate by putting two distinct superconducting loops next to each other. $|0\rangle$ can be indicated as a clockwise current and $|1\rangle$ as counterclockwise. The team found that modifying the magnetic field surrounding the first loop can effect currents in both directions at the same time, in other words the loop is in a superposition. The superposition also carried over to the other loop that partially changed it direction of current.

Why bother?

The engineering problems in implementing a quantum computer are considerable, and even now research concentrates on single or few qubit system, a far cry from complete circuits. So why bother with quantum versions of logic gates?

Avoids von Neumann - Landauer limit.

The first answer lies in the inherent reversible property of quantum gates, reversibility. von Neumann hypothesized and Landauer formulated an important corollary from Newton's thermodynamics: Destroying information - increasing entropy - costs energy. Therefore there is an effiency limit for systems that lose data.

However the concept of reversibility is older than that of a quantum gate, so reversible computing is not limited to quantum computers. Also the VNL is still far from limits imposed by materials and techniques available.

Hadamard gate

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

There is more in quantum computing, however. First let's introduce a new quantum gate operating on a single qubit, the Hadamard gate.

$$H_A \otimes H_B \otimes I_C(|0\rangle_A|0\rangle_B|0\rangle_C)$$
$$= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle$$
$$= \frac{1}{2}(|000\rangle + |010\rangle + |100\rangle + |110\rangle$$

Let's take a unitary transformation combining a couple Hadamard gates and an identity to get us a bit more interesting combined state of three qubits.

$$U_{AND}\frac{1}{2}(|000\rangle \otimes |010\rangle \otimes |100\rangle \otimes |110\rangle)$$
$$= \frac{1}{2}(|000\rangle + |010\rangle + |100\rangle + |111\rangle)$$

Now what happens when we put this three qubit system through an AND gate is that the transformation operates on all of the possible states of the superposition at the same time.

Superpositions for parallel computation.

It could be said that the quantum computer computed the output for all the possible inputs at one operation, utilizing quantum superpositions for massive parallelism.

Life isn't as sweet as it sounds at first, though, since it is only possible to read one of these states, and we cannot even select which one - the measurement takes a random state.

$$\psi \to U_N U_{N-1} \dots U_1 \psi = U_\sigma \psi$$

We don't have to settle for simple operations, however. By combining simple quantum gates we can formulate an unitary transformation operating on arbitarily large amounts of qubits. Even if output has randomness, this opens completely new venues for algorith design.

Search from an unsorted database:

Classical computing: $O(N)$

Quantum computer: $O(\sqrt{N})$

We will here more about possible algorithms next week, but let's look into a search from an unsorted database as an example. Of course using a classical computer, finding one particular entry from the database of N entries will take on average $\frac{N}{2}$ steps. There are several algorithms for quantum computers that can do better, Grover's can do $O(\sqrt{N})$. With M quantum computers working in parallel, this could be reduced to $O(\sqrt{\frac{N}{M}})$.

A clear advantage!

"All computing machines operating with the laws of [given] realm of physics are equivalent."

-Gui Lu Long

The reason why these sorts of algorithms are possible is that quantum algorithm operates on a different realm of physics than classical computers. Classical computers can be simulated be simulated on an universal turing machine with a maximum of polynomial slow down. But UTM cannot efficiently simulate quantum systems, while at the same time quantum computers can do everything that classical computers can.

Babbage's engine, Intel Core 2 Duo

equivalent

Quantum computers (ions, photons, . . .)

equivalent

The actualy implementation of the computer doesn't really matter. In a sense, all our computing machines from Babbage's differential engine to the latest microprocessors are the same.

In same sense all quantum computers are the same, no matter how you implement them.

Note that we are speaking in a theoretical sense. It could be argued that if the noise and other problems of analog computers were lesser, they couldn't be simulated by digital computers operating in the realm of integers. However in theory, a digital computer with unlimited memory can simulate an analog computer.

Where is particle wave duality?

The quantum gates described earlier utilize the phenonemom of superpositions to do magic that's impossible in classical physics. That isn't all that quantum mechanics has to offer. An important aspect of QM is that everything can be described as an particle and also as a wave, at the same time.

So where does the duality come into picture?

Classical computers

Quantum particle computer

Duality quantum computer

In fact quantum mechanics does not expand our realm of computing into just one, but two new categories: Quantum computers using quantum bits, qubits, as particles, but also a duality computer understanding and using also the wave properties of every particle. We are not dealing with single, descrete qubits anymore, but with duality enabled dubits.

New gates for dubits:

Wave dividers / splitters and combiners.

Duality computer is a relatively new concept. One of the first to talk about it with full speed ups of algorithms was Gui Lu Long on 2005.

The main concept introduced to duality computing is general quantum interferece.

To demonstrate, concider the double slit experiment of light. A laser shone into a plate with two slits, wave fronts are formed at both slits. The interference from these fronts can be seen as a pattern on a shade behind the slits.

Now, the same thing happens also with just one photon. This seems illogical, the photon is a particle and must pass through only one slit. However, if we try to measure through which slit the photon passed, the interference pattern dissappears and the photon acts like a particle. If we don't measure it, the pattern is there again. Same phenomenom can be found with other fundamental particles like electrons, but also with ensembles like ions and even molecules.

General quantum interference can be useful. As long as we don't try to measure the path and let the qubit act as a qubit, we can control the interference. A duality computer does exactly that: in addition to 'normal' quantum gates, it can also divide a quantum wave into subwaves and combine them later into one.

$$\begin{tikzcd}[row sep=tiny]
 & & p_1\psi \arrow[r] & p_1U_1\psi \arrow[dr] & \\
\psi \arrow[r] & {} \arrow[ur] \arrow[dr] & & & (p_1U_1+p_2U_2)\psi \\
 & & p_2\psi \arrow[r] & p_2U_2\psi \arrow[ur] &
\end{tikzcd}$$

$$\psi \to (\textstyle\sum p_i U_i)\psi$$

Dividing and combining in itself wouldn't be so useful, but this allows us to use different unitary transformations for the different subpaths. Where a quantum particle computer can compute only an arbitary unitary transform, the duality computer can use an linear combination of any unitary transformations.

The combination isn't an unitary transformation any more, and as a side effect duality computer isn't, and perhaps even cannot be reversible.

Photons - nonlinear quantum optics
Giant molecules

For now there hasn't been attempts to actually build a duality computer, even of few gates. Nevertheless, several possible ways to proceed have been proposed. One is, again, using photons. Splitting the photons into subwaves can be done somewhat easily with dichromic beam splitters and parametric conversion crystals. Dichromic beam splitter is a device to divide and combine beams of different frequency and parametric conversion crystal divide (down conversion) and combine (up conversion) one photon of a higher frequency to and from two photons of lesser frequency.

Another possibility would be using giant molecules, which relates to ionic particle computer. A molecule would consist of several ions, each ion representing a dubit. The ions need to be in the same molecule so that they would act as a combined quantum wave.

Search from an unsorted database:
Classical computing: $O(N)$
Quantum *particle* computer: $O(\sqrt{N})$
Duality quantum computer: Single query

The theoretical research of a duality computer is in it's infancy, but it has been proposed that while a quantum particle computer can handle a search from an unsorted database in $O(\sqrt{N})$, a duality computer could do the search in a single query.

Questions?

Thank you