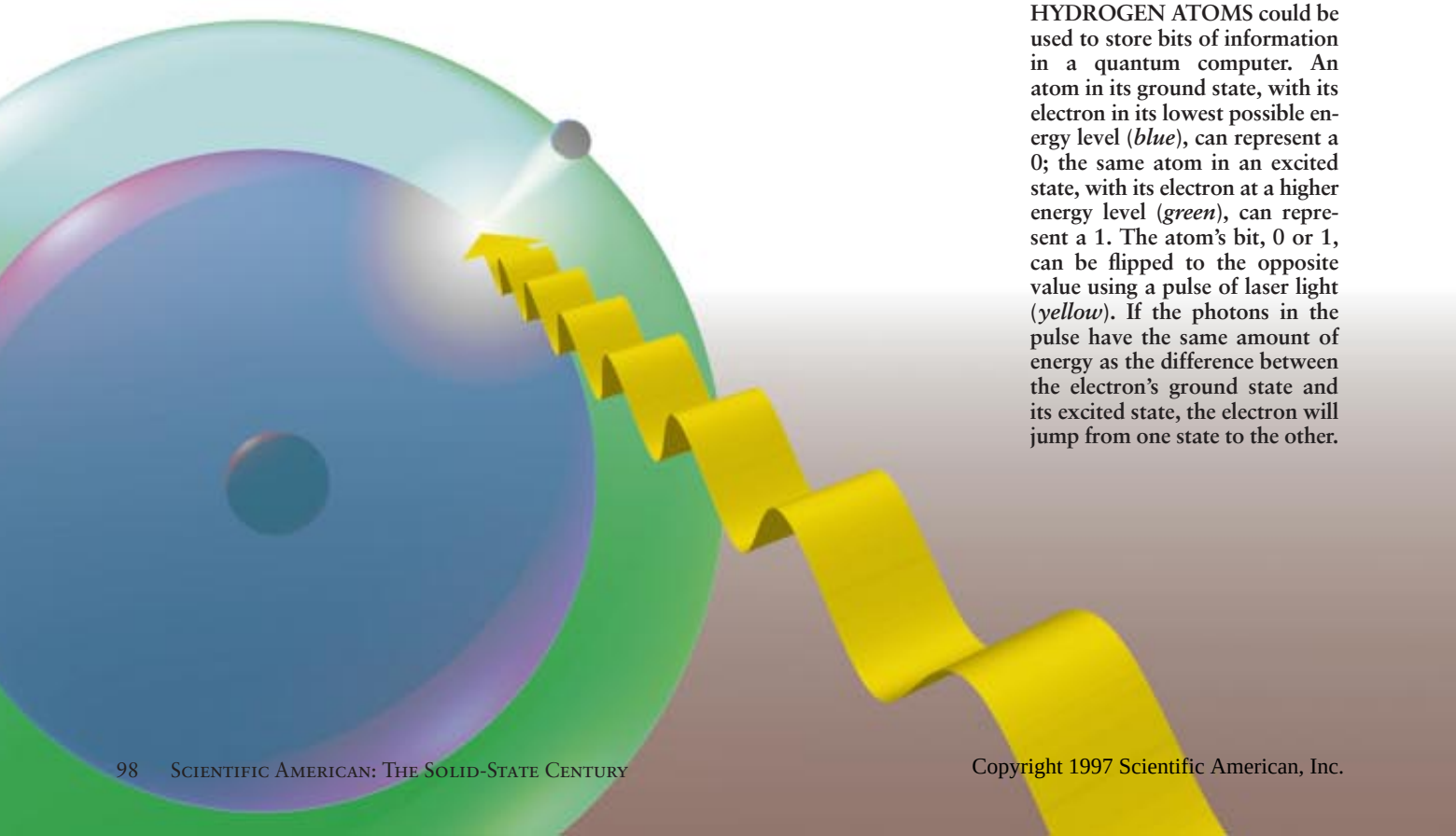
Quantum-mechanical computers, if they can be constructed, will do things no ordinary computer can

Quantum-Mechanical Computers

by Seth Lloyd

Every two years for the past 50, computers have become twice as fast while their components have become half as big. Circuits now contain wires and transistors that measure only one hundredth of a human hair in width. Because of this explosive progress, today's machines are millions of times more powerful than their crude ancestors. But explosions do eventually dissipate, and integrated-circuit technology is running up against its limits.

Advanced lithographic techniques can yield parts $^1/_{100}$ the size of what is currently available. But at this scale—where bulk matter reveals itself as a crowd of individual atoms—integrated circuits barely function. A tenth the size again, the individuals assert their identity, and a single defect can wreak havoc. So if computers are to become much smaller in the future, new technology must replace or supplement what we now have.



HYDROGEN ATOMS could be used to store bits of information in a quantum computer. An atom in its ground state, with its electron in its lowest possible energy level (*blue*), can represent a 0; the same atom in an excited state, with its electron at a higher energy level (*green*), can represent a 1. The atom's bit, 0 or 1, can be flipped to the opposite value using a pulse of laser light (*yellow*). If the photons in the pulse have the same amount of energy as the difference between the electron's ground state and its excited state, the electron will jump from one state to the other.

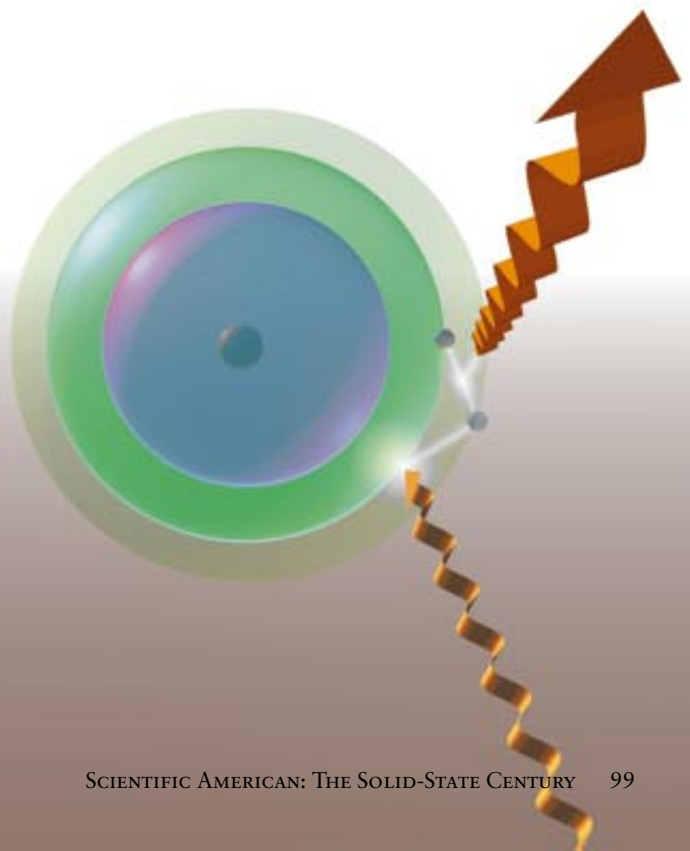
Several decades ago pioneers such as Rolf Landauer and Charles H. Bennett, both at the IBM Thomas J. Watson Research Center, began investigating the physics of information-processing circuits, asking questions about where miniaturization might lead: How small can the components of circuits be made? How much energy must be used up in the course of computation? Because computers are physical devices, their basic operation is described by physics. One physical fact of life is that as the components of computer circuits become very small, their description must be given by quantum mechanics.

In the early 1980s Paul Benioff of Argonne National Laboratory built on Landauer and Bennett's earlier results to show that a computer could in principle function in a purely quantum-mechanical fashion. Soon after, David Deutsch of the Mathematical Institute at the University of Oxford and other scientists in the U.S. and Israel began to model quantum-mechanical computers to find out how they might differ from classical ones. In particular, they wondered whether quantum-mechanical effects might be exploited to speed computations or to perform calculations in novel ways.

By the middle of the decade, the field languished for several reasons. First, all these researchers had considered quantum computers in the abstract instead of studying actual physical systems—an approach that Landauer faulted on many counts. It also became evident that a quantum-mechanical computer might be prone to errors and have trouble correcting them. And apart from one suggestion, made by Richard Feynman of the California Institute of Technology, that quantum computers might be useful for simulating other quantum systems (such as new or unobserved forms of matter), it was unclear that they could solve mathematical problems any faster than their classical cousins.

In the past few years, the picture has changed. In 1993 I described a large class of familiar physical systems that might act as quantum computers in ways that avoid some of Landauer's objections. Peter W. Shor of AT&T Bell Laboratories has demonstrated that a quantum computer could be used to factor large numbers—a task that can foil the most powerful of conventional machines. And in 1995, workshops at the Institute for Scientific Interchange in Turin, Italy, spawned many designs for constructing quantum circuitry. More recently, H. Jeff Kimble's group at Caltech and David J. Wineland's team at the National Institute of Standards and Technology have built some of these prototype parts, whereas David Cory of the Massachusetts Institute of Technology and Isaac Chuang of Los Alamos National Laboratory have demonstrated simple versions of my 1993 design. This article explains how quantum computers might be assembled and describes some of the astounding things they could do that digital computers cannot.

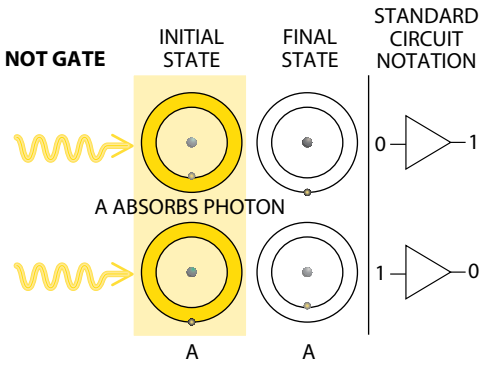
BORIS STAROSTA



READING the bit an atom stores is done using a laser pulse having the same amount of energy as the difference between the atom's excited state, call it E_1, and an even higher, less stable state, E_2. If the atom is in its ground state, representing a 0, this pulse has no effect. But if it is in E_1, representing a 1, the pulse pushes it to E_2. The atom will then return to E_1, emitting a telltale photon.

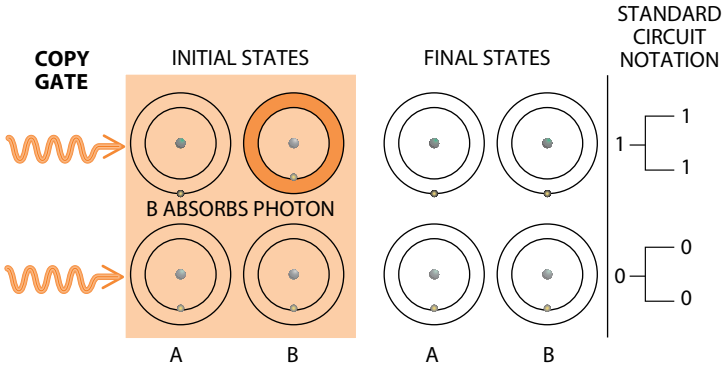
Quantum Logic Gates

Logic gates are devices that perform elementary operations on bits of information. The Irish logician George Boole showed in the 19th century that any complex logical or arithmetic task could be accomplished using combinations of three simple operations: NOT, COPY and AND. In fact, atoms, or any other quantum system, can perform these operations. —*S.L.*



NOT involves nothing more than bit flipping, as the notation above shows: if *A* is 0, make it a 1, and vice versa. With atoms, this can be done by applying a pulse whose energy equals the difference between *A*'s ground state (its electron is in its lowest energy level, shown as the inner ring) and its excited state (shown as the outer ring). Unlike conventional NOT gates, quantum ones can also flip bits only halfway.

COPY, in the quantum world, relies on the interaction between two different atoms. Imagine one atom, *A,* storing either a 0 or 1, sitting next to another atom, *B,* in its ground state. The difference in energy between the states of *B* will be a certain value if *A* is 0, and another value if *A* is 1. Now apply a pulse of light whose photons have an energy equal to the latter amount. If the pulse is of the right intensity and duration and if *A* is 1, *B* will absorb a photon and flip (*top row*); if *A* is 0, *B* cannot absorb a photon from the pulse and stays unchanged (*bottom row*). So, as in the diagram below, if *A* is 1, *B* becomes 1; if *A* is 0, *B* remains 0.



MICHAEL GOODMAN

Let's face it, quantum mechanics is weird. Niels Bohr, the Danish physicist who helped to invent the field, said, "Anyone who can contemplate quantum mechanics without getting dizzy hasn't properly understood it." For better or worse, quantum mechanics predicts a number of counterintuitive effects that have been verified experimentally again and again. To appreciate the weirdness of which quantum computers are capable, we need accept only a single strange fact called wave-particle duality.

Wave-particle duality means that things we think of as solid particles, such as basketballs and atoms, behave under some circumstances like waves and that things we normally describe as waves, such as sound and light, occasionally behave like particles. In essence, quantum-mechanical theory sets forth what kind of waves are associated with what kind of particles, and vice versa.

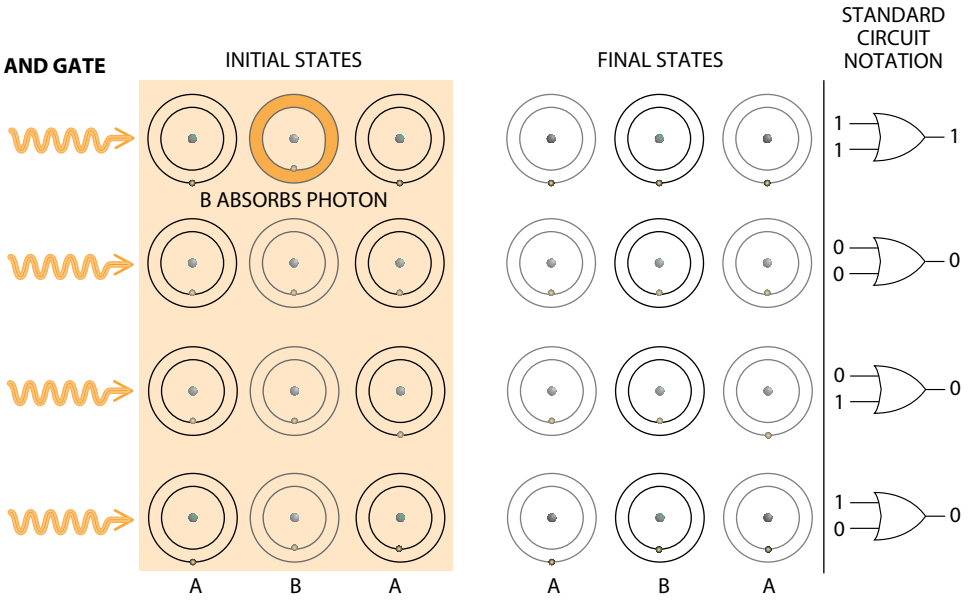
The first strange implication of wave-particle duality is that small systems such as atoms can exist only in discrete energy states. So when an atom moves from one energy state to another, it absorbs and emits energy in exact amounts, or "chunks," called photons, which might be considered the particles that make up light waves.

A second consequence is that quantum-mechanical waves, like water waves, can be superposed, or added together. Taken individually, these waves offer a rough description of a given particle's position. When two or more such waves are combined, though, the particle's position becomes unclear. In some weird quantum sense, then, an electron can sometimes be both here and there at the same time. Such an electron's location will remain unknown until some interaction (such as a photon bouncing off the electron) reveals it to be either here or there but not both.

When two superposed quantum waves behave like one wave, they are said to be coherent; the process by which two coherent waves regain their individual identities is called decoherence. For an electron in a superposition of two different energy states (or, roughly, two different positions within an atom), decoherence can take a long time. Days can pass before a photon, say, will collide with an object as small as an electron, exposing its true position. In principle, basketballs could be both here and there at once as well (even in the absence of Michael Jordan). In practice, however, the time it takes for a photon to bounce off a ball is too brief for the eye or any instrument to detect. The ball is simply too big for its exact location to go undetected for any perceivable amount of time. Consequently, as a rule only small, subtle things exhibit quantum weirdness.

Quantum Information

Information comes in discrete chunks, as do atomic energy levels in quantum mechanics. The quantum of information is the bit. A bit of information is a simple distinction between two alternatives—no or yes, 0 or 1, false or true. In digital computers, the voltage between the plates in a capacitor represents a bit of information: a charged capacitor registers a 1 and an uncharged capacitor, a 0. A quantum computer functions by matching the familiar discrete character of digital information processing to the strange discrete character of quantum mechanics.



AND also depends on atomic interactions. Imagine three atoms, *A*, *B* and *A*, sitting next to one another. The difference in energy between the ground and excited states of *B* is a function of the states of the two *A*'s. Suppose *B* is in its ground state. Now apply a pulse whose energy equals the difference between the two states of *B* only when the atom's neighboring *A*'s are both 1. If, in fact, both *A*'s are 1, this pulse will flip *B* (*top row*); otherwise it will leave *B* unchanged (*all other rows*).

Indeed, a string of hydrogen atoms can hold bits as well as a string of capacitors. An atom in its electronic ground state could encode a 0 and in an excited state, a 1. For any such quantum system to work as a computer, though, it must be capable of more than storing bits. An operator must be able to load information onto the system, to process that information by way of simple logical manipulations and to unload it. That is, quantum systems must be capable of reading, writing and arithmetic.

Isidor Isaac Rabi, who was awarded the Nobel Prize for Physics in 1944, first showed how to write information on a quantum system. Applied to hydrogen atoms, his method works as follows. Imagine a hydrogen atom in its ground state, having an amount of energy equal to E_0. To write a 0 bit on this atom, do nothing. To write a 1, excite the atom to a higher energy level, E_1. We can do so by bathing it in laser light made up of photons having an amount of energy equal to the difference between E_1 and E_0. If the laser beam has the proper intensity and is applied for the right length of time, the atom will gradually move from the ground state to the excited state, as its electron absorbs a photon. If the atom is already in the excited state, the same pulse will cause it to emit a photon and go to the ground state. In terms of information storage, the pulse tells the atom to flip its bit.

What is meant here by gradually? An oscillating electrical field such as laser light drives an electron in an atom from a lower energy state to a higher one in the same way that an adult pushes a child on a swing higher and higher. Each time the oscillating wave comes around, it gives the electron a little push. When the photons in the field have the same energy as the difference between E_0 and E_1, these pushes coincide with the electron's "swinging" motion and gradually convert the wave corresponding to the electron into a superposition of waves having different energies.

The amplitude of the wave associated with the electron's ground state will continuously diminish as the amplitude of the wave associated with the excited state builds. In the process, the bit registered by the atom "flips" from the ground state to the excited state. When the photons have the wrong frequency, their pushes are out of sync with the electron, and nothing happens.

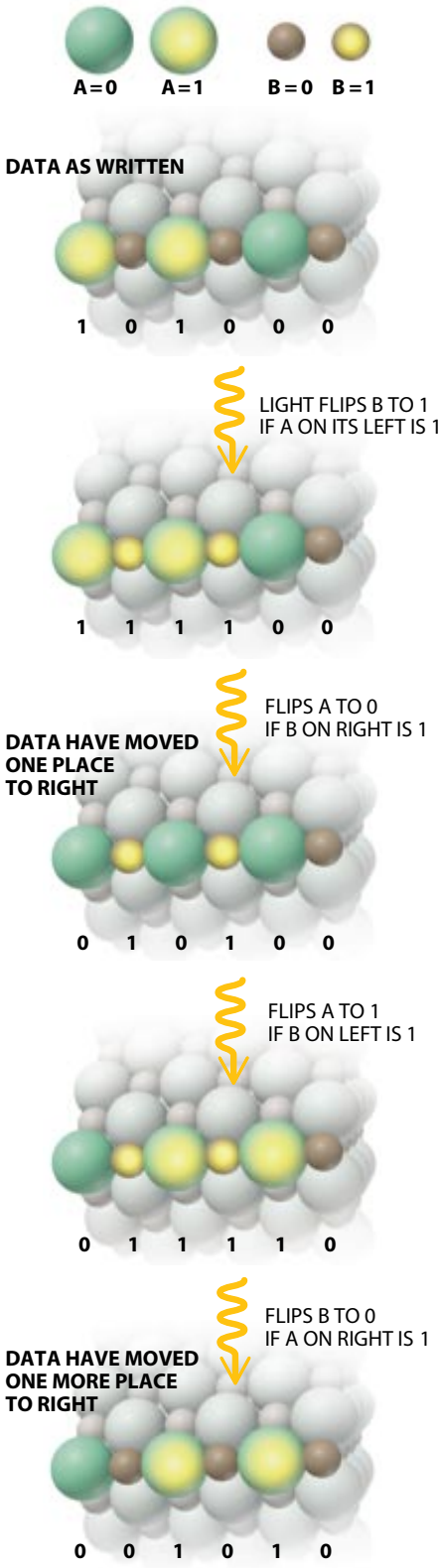
If the right light is applied for half the time it takes to flip the atom from 0 to 1, the atom is in a state equal to a superposition of the wave corresponding to 0 and the wave corresponding to 1, each having the same amplitudes. Such a quantum bit, or qubit, is then flipped only halfway. In contrast, a classical bit will always read either 0 or 1. A half-charged capacitor in a conventional computer causes errors, but a half-flipped qubit opens the way to new kinds of computation.

Reading bits from a quantum system is similar to flipping them. Push the atom to an even higher, less stable energy state, call it E_2. Do so by subjecting the atom to light having an energy equal to the difference between E_1 and E_2: if the atom is in E_1, it will be excited to E_2 but decay rapidly back to E_1, emitting a photon. If the atom is already in the ground state, nothing happens. If it is in the "half-flipped" state, it has an equal chance of emitting a photon and revealing itself to be a 1 or of not emitting a photon, indicating that it is a 0. From writing and reading information in a quantum system, it is only a short step to computing.

Quantum Computation

Electronic circuits are made from linear elements (such as wires, resistors and capacitors) and nonlinear elements (such as diodes and transistors) that manipulate bits in different ways. Linear devices alter input signals individually. Nonlinear devices, on the other hand, make the input signals passing through them interact. If your stereo did not contain nonlinear transistors, for example, you could not change the bass in the music it plays. To do so requires some coordination of the information coming from your compact disc and the information coming from the adjustment knob on the stereo.

Circuits perform computations by way of repeating a few simple linear and nonlinear tasks over and over at great speed. One such task is flipping a bit, which is equivalent to the logical operation called NOT: true becomes false, and false be-



comes true. Another is COPY, which makes the value of a second bit the same as the first. Both those operations are linear, because in both the output reflects the value of a single input. Taking the AND of two bits—another useful task—is a nonlinear operation: if two input bits are both 1, make a third bit equal to 1 as well; otherwise make the third bit a 0. Here the output depends on some interaction between the inputs.

The devices that execute these operations are called logic gates. If a digital computer has linear logic gates, such as NOT and COPY gates, and nonlinear ones as well, such as AND gates, it can complete any logical or arithmetic task. The same requirements hold for quantum computers. Artur Ekert, working with Deutsch and Adriano Barenco at Oxford, and I have shown independently that almost any nonlinear interaction between quantum bits will do. Indeed, provided a quantum computer can flip bits, any nonlinear quantum interaction enables it to perform any computation. Hence, a variety of physical phenomena might be exploited to construct a quantum computer.

In fact, all-purpose quantum logic gates have been around almost as long as the transistor! In the late 1950s, researchers managed to perform simple two-bit quantum logic operations using particle spins. These spins—which are simply the orientation of a particle's rotation with respect to some magnetic field—are, like energy levels, quantized. So a spin in one direction can represent a 1 and in the other, a 0. The researchers took advantage of the interaction between the spin of the electron and the spin of the proton in a hydrogen atom; they set up a system in which they flipped the proton's spin only if the electron's spin represented a 1. Because these workers were not thinking about quantum logic, they called the effect double resonance. And yet they used double resonance to carry out linear NOT and COPY operations.

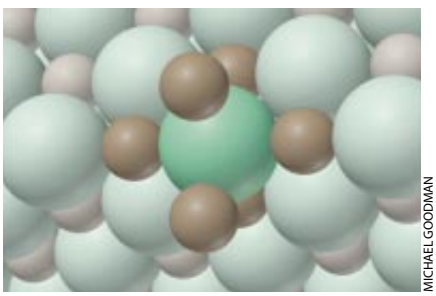
Since then, Barenco, David DiVincenzo of IBM, Tycho Sleator of New York University and Harald Weinfurter of the University of Innsbruck have demonstrated how, by flipping proton and electron spins only partway, double resonance can be used to create an AND gate as well. Such quantum logic gates, wired together, could make a quantum computer.

A number of groups have recently constructed quantum logic gates and proposed schemes for wiring them together. A particularly promising development has come from Caltech: by concentrating photons together with a single atom in a minute volume, Kimble's group has enhanced the usually tiny nonlinear interaction between photons. The result is a quantum logic gate: one photon bit can be flipped partway when another photon is in a state signifying 1. Quantum "wires" can be constructed by having single photons pass through optical fibers or through the air, in order to ferry bits of information from one gate to another.

An alternative design for a quantum logic circuit has been proposed by J. Ignacio Cirac of the University of Castilla-La Mancha in Spain and Peter Zoller of the University of Innsbruck. Their scheme isolates qubits in an ion trap, effectively insulating them from any unwanted external influences. Before a bit were processed, it would be transferred to a common register, or "bus." Specifically, the information it contained would be represented by a rocking motion involving all the ions in the trap. Wineland's group at NIST has taken the first step in realizing such a quantum computer, performing both linear and nonlinear operations on bits encoded by ions and by the rocking motion.

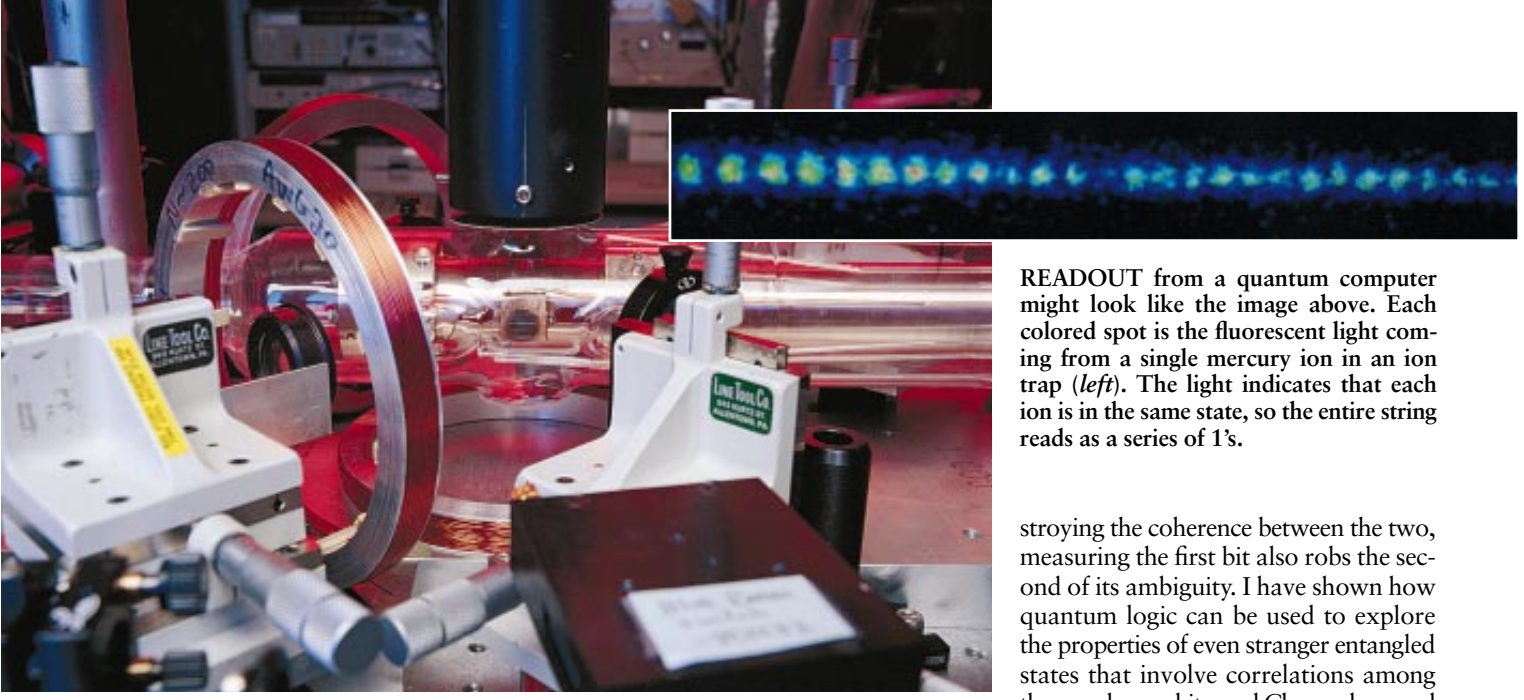
In an exciting theoretical development under experimental investigation at Caltech, Cirac, Zoller, Kimble and Hideo Mabuchi have shown how the photon and ion-trap schemes for quantum computing might be combined to create a "quantum Internet" in which photons are used to shuttle qubits coherently back and forth between distant ion traps.

Although their methods can in princi-



MICHAEL GOODMAN

SALT CRYSTAL could be made to compute by acting on pairs of neighboring ions. Flip the bit held by each *B* if the *A* on its left stores a 1; then flip each *A* if the *B* on its right is 1. This moves the information from each *A* to the *B* on its right. Now, using the same tactics, move the information from each *B* to the *A* on its right. The process allows a line of atoms to act as a quantum "wire." Because a crystal can carry out these "double resonance" operations simultaneously in all directions with every neighboring ion (*bottom, right*), the crystal can mimic the dynamics of any system and so serves as a general-purpose quantum analog computer.



READOUT from a quantum computer might look like the image above. Each colored spot is the fluorescent light coming from a single mercury ion in an ion trap (*left*). The light indicates that each ion is in the same state, so the entire string reads as a series of 1's.

GEOFFREY WHEELER (*left*); NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (*right*)

ple be scaled up to tens or hundreds of quantum bits, the Caltech and NIST groups have performed quantum logic operations on just two bits (leading some wags to comment that a two-bit microprocessor is just a two-bit microprocessor). In 1997, however, Neil A. Gershenfeld of M.I.T., together with Chuang of Los Alamos, showed that my 1993 method for performing quantum computing using the double resonance methods described above could be realized using nuclear spins at room temperature. The same result was obtained independently by M.I.T.'s Cory, working with Amr Fahmy and Timothy F. Havel of Harvard Medical School. With conventional magnets of the kind used to perform magnetic resonance imaging, Chuang and Cory both succeeded in performing quantum logic operations on three bits, with the prospect of constructing 10-bit quantum microprocessors in the near future.

Thus, as it stands, scientists can control quantum logic operations on a few bits, and in the near future, they might well do quantum computations using a few tens or hundreds of bits. How can this possibly represent an improvement over classical computers that routinely handle billions of bits? In fact, even with one bit, a quantum computer can do things no classical computer can. Consider the following. Take an atom in a superposition of 0 and 1. Now find out whether the bit is a 1 or a 0 by making it fluoresce. Half of the time, the atom emits a photon, and the bit is a 1. The other half of the time, no photon is emitted, and the bit is a 0. That is, the bit is a random bit—something a classical computer cannot create. The random-number programs in digital computers actually generate pseudorandom numbers, using a function whose output is so irregular that it seems to produce bits by chance.

Multiparticle Quantum States

Imagine what a quantum computer can do with two bits. Copying works by putting together two bits, one with a value to be copied and one with an original value of 0; an applied pulse flips the second bit to 1 only if the first bit is also 1. But if the value of the first bit is a superposition of 0 and 1, then the applied pulse creates a superposition involving both bits, such that both are 1 or both are 0. Notice that the final value of the first bit is no longer the same as it was originally—the superposition has changed.

In each component of this superposition, the second bit is the same as the first, but neither is the same as the original bit. Copying a superposition state results in a so-called entangled state, in which the original information no longer resides in a single quantum bit but is stored instead in the correlations between qubits. Albert Einstein noted that such states would violate all classical intuition about causality. In such a superposition, neither bit is in a definite state, yet if you measure one bit, thereby putting it in a definite state, the other bit also enters into a definite state. The change in the first bit does not *cause* the change in the second. But by virtue of destroying the coherence between the two, measuring the first bit also robs the second of its ambiguity. I have shown how quantum logic can be used to explore the properties of even stranger entangled states that involve correlations among three and more bits, and Chuang has used magnetic resonance to investigate such states experimentally.

Our intuition for quantum mechanics is spoiled early on in life. A one-year-old playing peekaboo knows that a face is there even when she cannot see it. Intuition is built up by manipulating objects over and over again; quantum mechanics seems counterintuitive because we grow up playing with classical toys. One of the best uses of quantum logic is to expand our intuition by allowing us to manipulate quantum objects and play with quantum toys such as photons and electrons.

The more bits one can manipulate, the more fascinating the phenomena one can create. I have shown that with more bits, a quantum computer could be used to simulate the behavior of any quantum system. When properly programmed, the computer's dynamics would become exactly the same as the dynamics of some postulated system, including that system's interaction with its environment. And the number of steps the computer would need to chart the evolution of this system over time would be directly proportional to the size of the system.

Even more remarkable, if a quantum computer had a parallel architecture, which could be realized through the exploitation of the double resonance between neighboring pairs of spins in the atoms of a crystal, it could mimic any quantum system in real time, regardless of its size. This kind of parallel quantum computation, if possible, would give a huge speedup over conventional methods. As Feynman noted, to simulate a quantum system on a classical computer generally requires a number of steps

that rises exponentially both with the size of the system and with the amount of time over which the system's behavior is tracked. In fact, a 40-bit quantum computer could re-create in little more than, say, 100 steps, a quantum system that would take a classical computer, having a trillion bits, years to simulate.

What can a quantum computer do with many logical operations on many qubits? Start by putting all the input bits in an equal superposition of 0 and 1, each having the same magnitude. The computer then is in an equal superposition of of all possible inputs. Run this input through a logic circuit that carries out a particular computation. The result is a superposition of all the possible outputs of that computation. In some weird quantum sense, the computer performs all possible computations at once. Deutsch has called this effect "quantum parallelism."

Quantum parallelism may seem odd, but consider how waves work in general. If quantum-mechanical waves were sound waves, those corresponding to 0 and 1—each oscillating at a single frequency—would be pure tones. A wave corresponding to a superposition of 0 and 1 would then be a chord. Just as a musical chord sounds qualitatively different from the individual tones it includes, a superposition of 0 and 1 differs from 0 and 1 taken alone: in both cases, the combined waves interfere with each other.

A quantum computer carrying out an ordinary computation, in which no bits are superposed, generates a sequence of waves analogous to the sound of "change ringing" from an English church tower, in which the bells are never struck simultaneously and the sequence of sounds follows mathematical rules. A computation in quantum-parallel mode is like a symphony: its "sound" is that of many waves interfering with one another.

Factoring could be an easy task for a quantum computer.

Shor of Bell Labs has shown that the symphonic effect of quantum parallelism might be used to factor large numbers very quickly—something classical computers and even supercomputers cannot always accomplish. Shor demonstrated that a quantum-parallel computation can be orchestrated so that potential factors will stand out in the superposition the same way that a melody played on violas, cellos and violins an octave apart will stand out over the sound of the surrounding instruments in a symphony. Indeed, his algorithm would make factoring an easy task for a quantum computer, if one could be built. Because most public-key encryption systems—such as those protecting electronic bank accounts—rely on the fact that classical computers cannot find factors having more than, say, 100 digits, quantum-computer hackers would give many people reason to worry.

Whether or not quantum computers (and quantum hackers) will come about is a hotly debated question. Recall that the quantum nature of a superposition prevails only so long as the environment refrains from somehow revealing the state of the system. Because quantum computers might still consist of thousands or millions of atoms, only one of which need be disturbed to damage quantum coherence, it is not clear how long interacting quantum systems can last in a true quantum superposition. In addition, the various quantum systems that might be used to register and process information are susceptible to noise, which can flip bits at random.

Shor and Andrew Steane of Oxford have shown that quantum logic operations can be used to construct error-correcting routines that protect the quantum computation against decoherence and errors. Further analyses by Wojciech Zurek's group at Los Alamos and by John Preskill's group at Caltech have shown that quantum computers can perform arbitrarily complex computations as long as only one bit in 100,000 is decohered or flipped at each time step.

It remains to be seen whether the experimental precision required to perform arbitrarily long quantum computations can be attained. To surpass the factoring ability of current supercomputers, quantum computers using Shor's algorithm might need to follow thousands of bits over billions of steps. Even with the error correction, because of the technical problems described by Landauer, it will most likely prove rather difficult to build a computer to perform such a computation. To surpass classical simulations of quantum systems, however, would require only tens of bits followed for tens of steps, a more attainable goal. And to use quantum logic to create strange, multiparticle quantum states and to explore their properties is a goal that lies in our current grasp. SA

The Author

SETH LLOYD is the Finmeccanica Career Development Professor in the mechanical engineering department at the Massachusetts Institute of Technology. He received his first graduate degree in philosophy from the University of Cambridge in 1984 and his Ph.D. in physics from the Rockefeller University in 1988. He has held postdoctoral positions at the California Institute of Technology and at Los Alamos National Laboratory, and since 1989 he has been an adjunct assistant professor at the Santa Fe Institute in New Mexico.

Further Reading

Quantum-Mechanical Models of Turing Machines That Dissipate No Energy. Paul Benioff in *Physical Review Letters,* Vol. 48, No. 23, pages 1581–1585; June 7, 1982.

Quantum Theory: The Church-Turing Principle and the Universal Quantum Computer. David Deutsch in *Proceedings of the Royal Society of London,* Series A, Vol. 400, No. 1818, pages 97–117; 1985.

A Potentially Realizable Quantum Computer. Seth Lloyd in *Science,* Vol. 261, pages 1569–1571; September 17, 1993.

Algorithms for Quantum Computation: Discrete Logarithms and Factoring. Peter W. Shor in *35th Annual Symposium on Foundations of Computer Science: Proceedings*. Edited by Shafi Goldwasser. IEEE Computer Society Press, 1994.

Quantum Computations with Cold Trapped Ions. J. I. Cirac and P. Zoller in *Physical Review Letters,* Vol. 74, No. 20, pages 4091–4094; May 15, 1995.