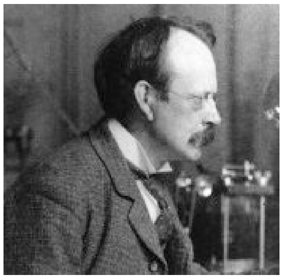
From Counting to Qubits

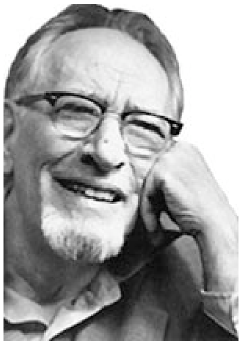
Computers as we know them, are a modern development, evolving from the 1940's to present day. The basic components of an electronic computer are electronic switches, used to represent the various states of a finite automata, or Turing machine. Computers can be classified into generations based on the technology used for these switches. The older electro-mechanical computers, like the Harvard based Mark 1 (1939-1944), used relays, but the first electronic computers used vacuum tubes, which signified the beginning of the contemporary computing age.

Edison's light bulb, in essence, is a vacuum tube, utilizing an evacuated tube of glass and a filament to coax light energy from charged electrons. Joseph John Thompson[1] (1856-1940), an English physicist, developed the vacuum tube to study the nature of cathode rays. He showed that the cathode rays were really made up of particles, or "corpuscles" as Thomson called them, which were contained in all material. Thomson had discovered the electron, for which he received the Nobel Prize for in 1909.



JJ Thompson

The most defining characteristic of a light bulb is that it can only be in the *on* or *off* state and can alternate between these states given some outside adjustment (like pulling a lever). This is precisely what makes a vacuum tube such a great candidate for a switch. Connecting many vacuum tubes in series and creating transitions rules between them would constitute a Turing machine. Such a machine was brought to life when the Ballistics Research Laboratory (BRL), responsible for providing calibration information to soldiers in the field, was falling behind. The BRL heard about the work of John Mauchly[2] at the Moore School in Pennsylvania. In 1942, he had suggested using vacuum tubes to speed computer calculations.



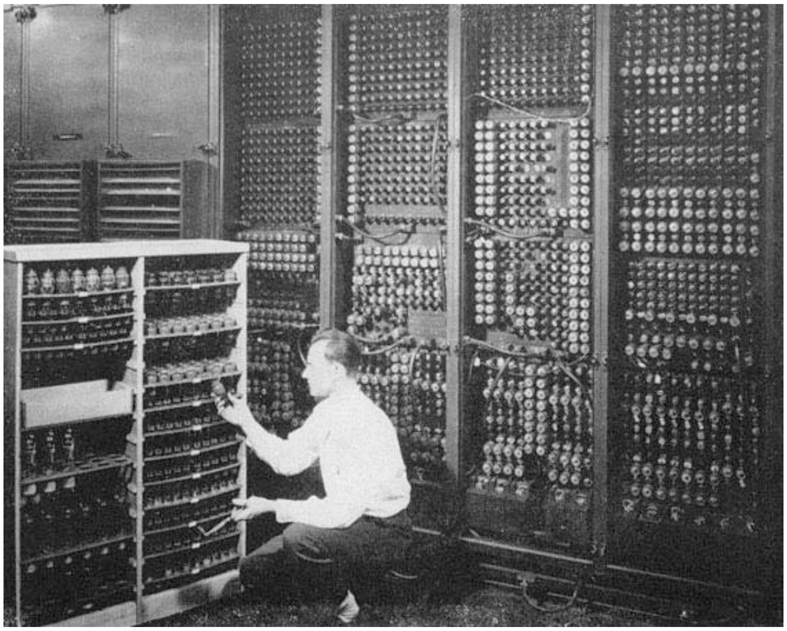
John Mauchly

The BRL commissioned work on this new high-speed computer in 1943. The computer nicknamed ENIAC short for "Electronic Numerical Integrator and Computer" took about a year to design and eighteen months to build. When finished the ENIAC weighed in at thirty tons, with 17,468 vacuum tubes, 70,000 resistors, 10,000 capacitors, 1,500 relays, and 6,000 manual switches, which occupied a 1800 square foot room. Housed at the Moore School of Electrical Engineering at the University Of Pennsylvania, the ENIAC required so much energy that the city of Philadelphia reportedly experienced brown outs during the computers operation. By the time the computer was finished in November 1945, the war had been over for 3 months. The project was also 200% over budget, costing over half a million dollars. But it had achieved what it set out to do. A calculation like the cube root of 2589 to the 16th power could be done in a fraction of a second. In a given second the ENIAC could do 5000 additions, 357 multiplications, and 38 divisions.

[1] Picture and information provided: http://www.chemheritage.org/EducationalServices/chemach/ans/jjt.html

[2] Picture and information provided: http://www.pbs.org/wgbh/aso/databank/entries/dt45en.html

The ENIAC



A thousand times faster then its predecessors the ENIAC proved very useful during the cold war where it was used to do calculations for the design of the hydrogen bomb. Unfortunately ENIAC’s main drawback was that programming it was a nightmare. In that sense it was not a general use computer. To change its program meant essentially rewiring it, with punch cards and switches in wiring plug boards. It could take a team two days to reprogram the machine.

However, despite its flaws the lessons learned from designing the ENIAC helped improve the next generation of computers, including the EDVAC, WHIRLWIND and the UNIVAC all of which improved on programmability and memory storage. In a sense, ENIAC’s greatest feat was showing the potential of what could be done.

Vacuum tubes, as fantastic as they seemed, had a lot of drawbacks. Like light bulbs, they emitted an incredible amount of heat. Not only was this a tremendous waste of energy, the heat also had a tendency of melting the metal filament housed inside the vacuum tube, causing the tube to leak and otherwise completely malfunction. Not to mention the sheer size of a vacuum tube required at least a 100 square meters of space to build a computer that could do anything useful. It became clear in the late 40’s that the age of vacuum tubes was coming to an end.

Instead of using electrons in a vacuum, scientists began to consider how one might control electrons in solid materials, like metals and semiconductors. Already in the 1920's, scientists understood how to make a two terminal device by making a point contact between a sharp metal tip and a piece of semiconductor crystal. These point-contact diodes were used to rectify signals (change oscillating signals to steady signals), and make simple AM radio receivers (crystal radios). However, it took many years before a solid-state device - the transistor - was discovered.

William B. Shockley, Walter Houser Brattain, and John Bardeen

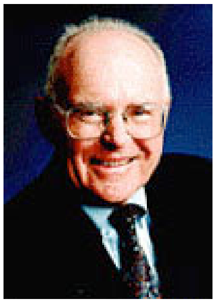
John Bardeen, William B. Shockley and Walter Brattain, were all employees Bell laboratories. Realizing that vacuum tubes were no longer up to par with current computer needs, all three transferred to Bell's superconductor laboratory to work on a replacement. They realized that by making two point contacts very close to one another, they could make a three terminal device - the first "point contact" transistor.

The age of the transistor marks the second generation of computers (1959-1963). The solid-state device functioned as a switch, hindering or amplifying a given electrical signal. Initially made from Germanium transistors could withstand a great deal of heat, and on top of this were a great deal smaller then the now outdated vacuum tube.

This meant that second generation computers were much smaller and more reliable than first generation computer. Eventually the cheaper material Silicon that could also withstand much higher heats replaced Germanium. The first such chip was made in San Francisco and thus, "Silicon Valley" was born.

The transistor didn't really make an impact on computing until the realization of integrated circuits. An integrated circuit is a solid-state device on which an entire circuit – transistors and the connections between them – can be created (etched). This meant that a single integrated circuit chip, not much bigger than early transistors, could replace entire circuit boards containing many transistors, again reducing the size of computers.

Naturally, scientists tried to cram as many transistors onto these integrated circuits. This is typically referred to as VLSI or very large-scale integration. Currently it is possible to place many millions of transistors and the accompanying circuitry on a single integrated circuit chip. By the mid 70's, it was possible to put the complete circuitry for the processor of a simple computer on a single chip, called a microprocessor, and the microcomputer was born in 1977.



Gordon Moore

Gordon Moore[3] made the famous observation in 1965 that computer speeds double and computer size halves every 18 months. To this day Moore's law is holding or being surpassed. Unfortunately, it seems that even if this trend holds computers will never be fast enough to solve some fairly simple mathematical problems…

Just beyond the narrow passage separating two mesas where Otowi Bridge spans the Rio Grande, New Mexico State Road 502, begins its steep ascent up Pajrito Plateau home, home of Los Alamos National Laboratory, which for a brief period of time during World War II housed the worlds first nuclear bomb.

[3] Picture and information provided by: http://www.intel.com/research/silicon/mooreslaw.htm

Though no longer able to build and test nuclear bombs, the Los Alamos labs are still actively studying nuclear explosions. To do this Los Alamos employs Blue Mountain, one of the most powerful computers in the world. Blue Mountain consists of 384 towering cabinets each containing 16 high-speed processors that are only found in the most top of the line desktop machines. The result is a machine in which a total of 6.144 processors united in parallel to collaborate on a horrendously complex problem: *simulating* a nuclear explosion.

Blue Mountain

The computers keepers boast its formidable specs. Thousands of processors are laced together with some 500 miles of fibre-optics cable, wire that carries impulses of light. The machinery consumes 1.6 megawatts of electrical power requiring 520 tons of cooling capacity. The result is a 3-teraops, computer, meaning it performs 3 trillion mathematical operations a second.

Now suppose that you can perform a single calculator operation in one second. There are about 3 billion seconds in a century, so it would take you a thousand centuries to do what blue mountain does in one second. To put that in perspective you would have to go back to the late Middle Pleistocene epoch when *Homo sapiens* were just emerging. Of course you would have to repeat the process once you finished to compensate for the next second of Blue Mountain calculation.

Not far from Blue Mountain, in a laboratory is another computer recently installed in a 303,000 square-foot building. This computer boasting twice as many processors as Blue Mountain, each running fives times faster, results in a 30-teraops machine. It is named Q after the dimension hoping character from Star Trek, as well as the James Bond character. It would take a fast fingered button pusher a million years to match Q's one second of computation.

It comes somewhat as a shock that these computers, seemingly limitless in computation power cannot solve one of the most basic number theory problems in the book. Breaking a composite number into its prime factors. The record to date, set in 2002 by 292 computers in the Netherlands, Canada, the United Kingdom, France, Australia, and the United States took a little more than five months to find the two primes factors of a 155-digit number. Not too impressive considering the numbers used in RSA cryptography to encode messages use composites much larger then 155-digits.

However there is a small hope that one day we will be able to factor these composites in less than a second. I mean "small hope" quite literally. Take a short walk from the massive expanse of a building housing the Q supercomputer and you will find two young physicists, Manny Knill and Raymond Laflamme, who have been taking a quieter approach to super computing. In a nondescript brown stucco building on the outskirts of the main laboratory complex Knill and Laflamme (who I am happy to say now works at Waterloo University) are programming a computer so small that you can't even see it with a microscope: a single molecule strung together form a dozen atoms.

Raymond Laflamme

The resulting computer is what we call a Quantum Computer. Without delving into the conceptual horror that is Quantum mechanics let us try to understand the basics of a quantum computer.

Today's conventional computers all boil down to a set of states and transitions, in other words Blue Mountain only differs from the ENIAC by number of components and the speed at which it can flick a switch back and forth. The simplest form of this construct is a Turing machine, which is again a set of states (ON and OFF) and transition rules. Any past, present and future computers based on this principle are basically just really big Turing machines.

It turns out that a Quantum Computer is also a Turing machine with one major difference. A "switch" on a Quantum Computer can be both ON and OFF, which is already a fairly abstract notion. The consequences of this idea though are fairly revolutionary.

Consider the bit (binary digit) representation of numbers in a conventional computer. Typically the computer will group these bits in eights, called a register, so by putting the numbers 1 and 0 into the spaces: _ _ _ _ _ _ _ _ we can represent the numbers 0 to 256 individually. In a Quantum Computer using the same number of spaces we can represent the numbers 0 to 256 simultaneously, since every space will have the value 0 *and* 1, (this entity is called a qubit). What this means if I were to do an operation like square root on a number in a quantum registry I am in fact doing 256 calculations at the same time.

Want to know the sin of every number between 0 and 1000? In a quantum computer you would just have to load ten atoms into a quantum register and perform a single calculation to get them. No one has yet pulled off such a delicate feat, but noting in the laws of physics seems to prevent it.

With the addition of every atom you double the amount of calculations you can do in one sweep. With 13 atoms, you have a device that can do 2 to the 13th power or 8,192 parallel calculations, surpassing Blue Mountain's mere 6,144. To match the computing power of Q, just add one more atom and double 8,192 to get 16,384 calculations, all of which can be carried out tandem.

Optimists say that our development of quantum computing is analogous to the state of quantum field theory in the 1940's, one can only dream of the day that the theory is made a reality. Perhaps a Q-ENIAC is just beyond the horizon, until then we remain adrift in a vast sea of quantum laws.

Main Sources

“Computer Science Using JAVA” :: David Hughes

“A Shortcut Through Time : The Path To The Quantum Computer” :: George Johnson