PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

Computing in the 21st century: nanocircuitry, defect tolerance and quantum logic

R. Stanley Williams

Phil. Trans. R. Soc. Lond. A 1998 **356**, 1783-1791
doi: 10.1098/rsta.1998.0249

Computing in the 21st century: nanocircuitry, defect tolerance and quantum logic

By R. Stanley Williams

Hewlett-Packard Laboratories, 3500 Deer Creek Road, Bldg 26U,
Palo Alto, CA 94304-1392, USA

The geometrical scaling era of microelectronics technology will end around the year 2010, if current extrapolations of physical and economic issues are valid. Computers built then should be 256 times as capable as the current generation, according to industry projections. However, physical laws suggest that it should be possible to compute non-reversibly at least 10^9 times present speeds with the expenditure of only 1 W of electrical power. The challenges faced by those who intend to build affordable appliances with capabilities far beyond those of microelectronic circuits are to invent new computer architectures suitable for nanometre-scale devices and techniques to fabricate and assemble vast numbers of such devices inexpensively. These circuits will operate according to quantum mechanical principles, and will necessarily be very different from those based on present technology.

Keywords: information age; disruptive technology; solid state devices; non-reversible computing; Teramac; nanocircuitry

1. Introduction

The year 1997 marks a number of notable anniversaries for the field of electronics: the 100th of the discovery of the electron, the 90th of the patenting of the first amplifying vacuum tube, the 50th of the invention of the transistor, and the 25th of the introduction of the microprocessor. The realization that the electron was a discrete particle with mass and charge provided scientists and engineers with a fresh understanding that allowed them to harness it for performing useful work. Over the past century, the pervasiveness of electronic tools, and the efficiency with which they operate, have increased dramatically. The 'Information Age' is being created because we have learned how to use the tremendous computational capacity of electrons. In this paper I present a brief history of electronic computation leading up to the present time and speculate about how we will be computing in the next century.

The first stored-program electronic computer, ENIAC, was commisioned in 1946. This machine was a major triumph of vacuum tube technology. ENIAC could add 5000 numbers in 1 s, which enabled it to calculate the trajectory of an artillery shell in only 30 s, while an expert human with a mechanical calculator would require about 40 h for the same task. It was also large and expensive: ENIAC contained 17 468 vacuum tubes, weighed 60 000 pounds, occupied 16 200 cubic feet and consumed 174 kW (233 horsepower). The amount of energy ENIAC expended to compute a single shell trajectory was about 5.25 MJ, which is comparable to the explosive discharge required to actually fire the shell. It was still the fastest computer on Earth nine years later, when ENIAC was turned off because the US Army could no longer

TEX Paper

justify the expense of operating and maintaining it. However, even in the early days of ENIAC, technologists dreamed of smaller and faster computers.

An article in the March 1949 *Popular Mechanics* confidently predicted that some day, a computer as powerful as ENIAC would contain only 1500 vacuum tubes, weigh 3000 pounds, and require 10 kW of power to operate. Such a machine would be about the size and weight of an automobile. This bold extrapolation seems quaint to us now in the era of Mega-Flops and palmtop computers. The 90 MHz processor in the computer on my desk, which is nearly a generation out of date, can add 13 million numbers per second with a power expenditure of 3.5 W. Measured in terms of the energy required for an addition, my computer can calculate 10^8 times more efficiently than ENIAC. What caused such a dramatic underestimate of the capability of computers in 1949, and how much more improvement can we now expect in the future? The pundits of *Popular Mechanics* probably extrapolated correctly for the possible advancement of computers built with vacuum tubes. However, they overlooked the transistor, which had already been invented and represented a disruptive technology.

Solid-state devices represented an opportunity to build electronic circuits that were smaller, faster, cheaper and more reliable, but another decade was required for the transistor to significantly challenge the vacuum tube. After 40 years of development, vacuum tube technology was mature and the associated manufacturing infrastructure was enormous. Even though transistors as discrete devices had significant advantages over vacuum tubes and progress was steady during the 1950s, the directors of many large electronics companies believed that the vacuum tube held an unassailable competitive position. Their firms were eventually eclipsed by those that invested heavily in research and development of transistor technology and were poised to exploit new advancements.

The era of explosive growth in the capability of electronic circuits began with the 1958 invention of the integrated circuit or chip, in which transistors, wires and capacitors were all fabricated on the same substrate as a single unit. The first computer-on-a-chip or microprocessor, the Intel 4004, was introduced in 1972. It was able to perform 5000 binary coded decimal additions per second, which was roughly the same computing capacity as ENIAC but with a total power consumption (including companion chips) of only about 10 W. An interesting metric to use in comparing the efficiency of different computers is the energy cost of adding two real numbers, which was about 35 J for ENIAC, 2×10^{-3} J for the Intel 4004 and 3×10^{-7} J for a Pentium, or, in mechanical equivalents, enough work to lift a 1 g mass 3.6 km in 1947, 2 m in 1972, and 30 μm in 1997, respectively.

Early in the integrated circuit era, Gordon Moore observed that the number of transistors that could be integrated onto a single chip was increasing exponentially with time. Although the rate constant has changed somewhat since 1965 (the growth during the microprocessor era has been a factor of four every generation, or 3.4 years), the basic nature of the growth has been so consistent that the observation is now known as 'Moore's law' (Schaller 1997). This exponential growth in chip functionality is closely tied to the exponential growth in the chip market, which has been approximately doubling every five years. This dramatic climb has fuelled the fortunes of several major companies that either make or use chips, and has also been a significant factor in the growth in the gross national product of several nations. Thus, any factor that might slow or halt the exponential growth of chip functionality needs to be examined seriously.

At the present time, there are two recognized factors that could bring Moore's law scaling to an end. The first, according to Moore himself, is economic (Schaller 1997). The cost of building fabrication facilities to manufacture chips has also been increasing exponentially, about a factor of two every chip generation (this is sometimes known as Moore's second law). Thus, the cost of manufacturing chips is increasing significantly faster than the market is expanding, and at some point a saturation effect should slow the exponential growth to yield a classic S-curve for expanding populations. In 1995, a single fabrication facility (FAB) cost about US\$$10^9$ to build, or *ca.* 1% of the entire annual chip market; by the year 2010 a FAB could cost as much as US\$$10^{11}$, or *ca.* 10% of the total annual market at that time if Moore's second law continues to hold.

The second issue impacting on chip manufacture is that the transistors and wires in integrated circuits are starting to approach some fundamental physical limitations, in contrast with the engineering difficulties that have continually haunted the industry during the past two decades. There are several such limits, but perhaps the most startling is that transistors are starting to run out of electrons. By the year 2010, if the field effect transistors (FET) in integrated circuits have scaled according to Moore's law, only 10 electrons will be required to switch the FET from off to on. The statistics of small numbers will become significant, and the ability to distinguish between zero and one in a digital circuit will be severely compromised. It is possible to design smaller FETs that would use a constant number of electrons as they shrink geometrically, but then the electrical power required by an entire integrated circuit of such devices would increase exponentially and soon exceed reasonable limits. Thus, it is not likely that conventional geometrical scaling of integrated circuits will continue beyond the year 2010, which is the end of the 1994 National Technology Roadmap for Semiconductors issued by the Semiconductor Industry Association. However, that projection calls for chips that are 256 times more capable than current designs, with no increase in power dissipation. If this goal is attained, then the silicon-based integrated circuit will have accomplished an improvement in performance of more than six orders of magnitude, using energy as a metric, with a single manufacturing paradigm. Compared to the advances experienced in most human endeavours, this increase is extraordinary.

2. Nanocircuitry

The question of the fundamental limits to computation has already been a subject of great interest for decades. There are several different answers to that question, all of which are correct, but some of which are more practical than others. After all, if computational technology will be close to any fundamental limit with the integrated circuits of 2010, it does not make sense to invest in a huge research and development effort to replace a mature technology with one that is only marginally superior.

As with many other issues, Richard Feynman provided an exceptionally illuminating overview of the fundamentals of computing (Feynman 1995). Landauer (1961) showed that a non-reversible computer performing Boolean logic operations requires a minimum energy for a bit operation:

$$E_0 = kT \ln 2, \tag{2.1}$$

where k is Boltzmann's constant and T is the operating temperature of the system. This is the energy cost of throwing away one bit of information and increasing the entropy of the surroundings. At room temperature, this equation predicts that it is possible to perform 3.5×10^{20} bit operations per second with the expenditure of 1 W of power (obviously, there would have to be a huge number of processes operating in parallel in any real system). Further thermodynamic considerations (Feynman 1995) show that the amount of energy required (as opposed to that now dissipated in a resistive network) to transport a bit irreversibly from device to device in a computational system is

$$E_{\mathrm{t}} = kT\lambda\nu/c, \tag{2.2}$$

where λ is the transmission distance, ν is the operating frequency and c is the speed of light. This equation is in accord with the usual understanding of non-reversible processes, which cost more energy the faster they occur, but it also shows that smaller systems will expend less energy. For realizable operating frequencies, this energy is large compared to that determined from equation (2.1) for bit destruction.

A crude estimate of the energy required to add two 10 digit numbers using an ideal non-reversible computer is $1000kT \ln 2$, which implies that 3×10^{17} additions J^{-1} can be performed at room temperature, a factor of 10^9 times the estimated upper limit of Si-integrated circuit technology in 2010. Thus, even if the thermodynamic limit of efficiency is never achieved, the fact that there is such a huge improvement possible means that the search for new alternatives to the present technology is both prudent and potentially very rewarding. Such efficiency increases would allow either greater computational speed at constant power dissipation or smaller appliance size for constant computational throughput. To achieve these incredible advances we will require a totally different type of computational machinery. The requirement for inventing a new technology paradigm, coupled with the economic rewards that would follow from such a development, has created exciting research opportunities for mathematicians and scientists of many disciplines, as well as for electrical engineers. In fact, much of the current interest in interdisciplinary research areas such as nanofabrication, self-assembly, molecular electronics, etc., is being driven by this search for a new computer archetype.

Thus, it appears as though the age of computation has not yet even begun, since even on a logarithmic scale there is further to go into the future than we have come from ENIAC. The implementation of some reversibility in a machine would provide even greater capability. A number of alternatives to Si-based FETs have been proposed, including single-electron transistors (Chen *et al*. 1996), quantum cellular automata (Tougaw & Lent 1994), neural networks (Mead 1990; Hopfield & Tank 1986), molecular logic devices (Aviram & Ratner 1974; Petty *et al*. 1995) and others. A common theme that underlies many of these schemes is the push to fabricate logic devices on the nanometre length scale, which will therefore be dominated by quantum mechanical effects. Such dimensions are more commonly associated with molecules than integrated circuits, and it is not surprising that chemically assembled configurations, rather than artificially drawn structures, are expected to play an increasingly important role in the manufacture of devices.

One very significant constraint that trying to manufacture the nanocircuitry of the future will involve is expense. Given Moore's second law, it is very unlikely that systems with feature sizes of a few nm will be made using traditional lithographic and subtractive processes, since scaling rules indicate that the cost of a facility for doing so would be nearly the gross national product of the Earth. Instead, at some point the cost advantage of using chemical assembly procedures to fabricate nanocircuitry will outweigh the disadvantages. At present, chemical assembly processes can produce nanocrystals as small as 10 nm directly on a surface (Kamins *et al*. 1997) and just a few nm in size in solution growth. One approach to fabricating useful circuitry may be to use the best available lithography to define the outlines of the circuit and to use chemically grown nanocrystals as the critical components of the device (Kamins & Williams 1997).

Assume for the moment that it were possible to chemically synthesize various electronic components and connect them together to form a relatively ordered configuration. Several problems will arise when one attempts to use this assembly for some computational task. The fabrication efficiency of operational discrete devices will not be unity, but rather will reflect the statistical yields of chemical syntheses. In addition, the system will suffer an inevitable amount of uncertainty in the connectivity of the devices. Given this, how does one communicate with the system from the outside world in a reliable and predictable fashion and be assured that it is performing error-free computations? Furthermore, since one goal of nanoscale technology is to provide a huge number (e.g. a mole) of devices for a system, how does one impose an organization that allows the entire ensemble to operate efficiently?

3. Defect tolerance

The fact that economic considerations will be a significant constraint on the future of nanoelectronics means that it is reasonable to examine issues of circuit architecture at this early stage, before settling on a device type that will later be too expensive to fabricate. If nanocircuits are fabricated using chemical assembly, then they must be able to tolerate a significant number of defects that are introduced during fabrication because of statistical mechanical fluctuations in small systems. Even if the defect rate is only one per billion components, which is the current best practice in chip FABs, that still results in a million defects in a system that contains 10^{15} components. The largest defect-tolerant computer built to date is the Hewlett-Packard Teramac (Culbertson *et al*. 1997), which is an existence proof that a large system with a significant fraction of defects can still provide a huge amount of computational power.

Although Teramac was constructed using conventional technology, many of the problems associated with this machine are similar to the challenges that are faced by scientists exploring nanoscale paradigms. Teramac was built from a large number of components that had a significant defect probability. To keep the construction costs reasonable, the builders knowingly used components that were defective, and inexpensive but error-prone techniques were used to connect all the components together. However, because of the physical architecture chosen to implement powerful software algorithms, Teramac could be configured into many different types of extremely capable supercomputer, even in the presence of the defects. The architecture of Teramac and its implementation of defect tolerance are relevant to any computational nanotechnology.

Teramac is a multi-architecture computer (MAC) with 10^6 gates that operate at 1 MHz (or a total of 1 THz bit operation frequency). Teramac is based on field programmable gate arrays (FPGAs), which are essentially look-up tables (LUTs) connected by a huge number of wires and switches arranged in full cross bars. In principle, FPGAs substitute memory for logic whenever possible. As the number of resources available in a computer increases, it makes more sense to store as many intermediate results as possible and just look them up when needed.

Teramac contains 864 separate FPGA chips, but only 256 of them are used for their look-up tables. The remaining FPGAs are used only for their cross-bar switches to provide the massive interconnectivity that defines the six-level hierarchy of the architecture. Thus, Teramac is an example of the philosophy of the rich man; if resources are cheap, one can afford to waste them. For nanocircuits, one can afford to disregard huge numbers of non-functional devices as long as they are cheap and the functioning ones provide more computational power than a competing technology.

Perhaps the most amazing fact about Teramac is that it was comatose at birth. Three quarters of the FPGAs contain defects that would be fatal for an isolated chip (and in fact, the manufacturer provided those chips to the creators of Teramac for free, charging only for the perfectly functioning ones). Teramac contains a total of 220 000 wiring and gate defects, for a total of 3% of all of its resources. The cost of building Teramac was substantially reduced by using inexpensive (or free) components and less than rigorous assembly techniques. For the first 24 h of its existence, Teramac was connected to a workstation that performed a series of tests to find out where the defective resources were. These locations were then written to a configuration table as being 'in use', to insure that the defective components would not be accessed by a running program. Because of the very high degree of connectivity in Teramac, it was possible to access nearly all of the good components in the system while insuring that none of the bad ones were used.

Once the configuration table has been prepared to omit defects, the job of the compiler for Teramac is to map a logical machine onto the physical structure of the computer. Teramac requires a 300 Mbit configuration word to set all the switches, which essentially defines the type of computer it will be for a particular operation. It was made generally available to users for architecture and algorithm development, and most of those users were completely unaware of the fact that it was so highly defective. However, it can perform most calculations about 100 times faster than a top-end workstation, which means that the defective components did not drastically degrade the performance of the 256 chips actually used as processors.

An architecture similar to Teramac can also be the basis for highly efficient reversible computing. A system can be envisioned in which bits are never created or destroyed, but are stored in look-up tables and transported from place to place as needed. This would avoid the necessity of storing large amounts of redundant bits generated by reversible logic operations. For a nanotechnology in which there are more than 10^{15} resources in a system, the need for logic may actually be small for most applications.

4. Quantum logic

We have seen that there are tremendous advances possible for computing, even if quantum logic never becomes a reality. Many people question the need for another 12 orders of magnitude of computing power beyond what we have today in an appliance, especially when that represents six orders of magnitude more computing power than that of a human brain. However, for some applications the reversibility and inherent parallel nature of quantum logic represent a leap far beyond what ideal non-reversible computing can offer, perhaps by another nine orders of magnitude or more.

What would future quantum logic processors look like? At present, there are three significant experimental realizations of a qubit: an ion in a trap, a photon in a resonant cavity and a nuclear spin. The first two resemble the technology of an atomic clock and the third is represented by a nuclear magnetic resonance instrument. These are machines that are quite familiar today, and a great deal of work is going into making them smaller, easier to use, and more robust. Thus, it is at least conceivable that even such exotic implementations of quantum logic that are pursued today could wind up in desktop computers of the next century. It may even be the case that quantum logic processors are the only types of logic devices in use, if non-quantum computations can be handled with look-up tables.

5. Conclusions

What will happen after 2010? No one can say for certain, but it is possible that a new device paradigm will be invented to replace semiconductor FETs. Discovering another disruptive technology is the object of significant research at a large number of laboratories all over the world. Breakthroughs will require significant advances in the understanding of fundamental questions, and will undoubtedly act as the foundation for new mathematical and scientific disciplines. The economic prize for those who succeed is access to one of the world's most important markets. The social implications are also enormous, since even by 2010, appliances will have nearly the processing capacity of a human brain.

The important lessons of Teramac for nanotechnology are that a system does not have to be perfect to be very powerful and the more defects a system can tolerate the cheaper it will be to build. Thus, perhaps the search for enabling nanostructured devices should concentrate on wires and switches, since these are the components that will allow highly defect tolerant systems to be built.

It is possible that history is about to repeat itself, with the introduction of a new disruptive technology for computation in the 21st century. In 1937, there was a mature technology, the vacuum tube, which was still a decade away from its ultimate accomplishment. However, there was already a significant search for something that would be better, a solid-state switch. The development of that switch required a great deal of basic research in both materials purification and in device concepts. Today, we have the silicon FET, but we speculate that a quantum-state switch could be better. We are now engaged in basic research in the fabrication of materials into arbitrary shapes and sizes, and looking for the device concept that will lead to a new technology.

My collaborators in the search for a new type of computational machinery and the means to fabricate it are T. I. Kamins, A. M. Bratkovski, G. Medeiros-Ribeiro, P. Kuekes, C. A. Flory and D. A. A. Ohlberg, all of Hewlett-Packard Laboratories, and J. R. Heath of UCLA. We have benefited greatly from our interactions with L. S. Cutler of H.-P. Labs, A. Ekert of Oxford University, and many others too numerous to mention.

References

Aviram, A. & Ratner, M. 1974 *Chem. Phys. Lett.* **29**, 277.

Chen, R. H., Korotov, A. N. & Likharev, K. K. 1996 *Appl. Phys. Lett.* **68**, 1954.

Culbertson, W. B., Amerson, R., Carter, R. J., Kuekes, P. & Snider, G. 1997 *Proc. 1997 IEEE Symp. on FPGAs for Custom Computing Machines.*

Feynman, R. P. 1995 *Feynman lectures on computation* (ed. A. J. G. Hey & R. W. Allen), pp. 137–184. Menlo Park, CA: Addison-Wesley.

Hopfield, J. J. & Tank, D. W. 1986 *Science* **233**, 625.

Kamins, T. I. & Williams, R. S. 1997 *Appl. Phys. Lett.* **71**, 1201.

Kamins, T. I., Carr, E. C., Williams, R. S. & Rosner, S. J. 1997 *J. Appl. Phys.* **81**, 211.

Landauer, R. 1961 *IBM Jl Res. Devel.* **5**, 183.

Mead, C. 1990 *Proc. IEEE* **78**, 1629.

Petty, M. C., Bryce, M. R. & Bloor, D. (eds) 1995 *Introduction to molecular electronics.* London: Edward Arnold.

Schaller, R. R. 1997 *IEEE Spectrum* **34**, 53.

Tougaw, P. D. & Lent, C. S. 1994 *J. Appl. Phys.* **75**, 181.

Discussion

B. Christianson (*Computer Science Department, University of Hertfordshire, Hatfield, UK*). The energy required to carry out an operation reversibly decreases with the rate at which the operation must be carried out. This seems to be an argument for parallelizing algorithms and building large, slow hardware: 1000 processing elements, each carrying out one reversible operation per second, would use much less power in total to complete a computation than would be required to complete the same computation in the same elapsed time using a single processor element carrying out 1000 operations per second. In the first case, a lower potential gradient suffices to drive the computation forward, so the power consumption per reversible operation is less (in theory by a factor of a 1000), and only the very final result need be latched.

R. Stanley Williams. This is a very good point, and it is relevant to non-reversible as well as reversible machines. To be able to use the number of bit operations that will be available with 1 W of power, even for totally non-reversible computation, will require massive parallelism (at least 10^6 subprocessors) in the architecture of the machine. Certainly, there are known algorithms that will benefit greatly from such a system, but they make up a relatively small proportion of the tasks that currently occupy computers. The major current challenge for the information science community is to develop parallel algorithms for problems that currently appear to be inherently serial in nature.

Th. Beth (*University of Karlsruhe, Germany*). What is the physical reason why Ge atoms form a self-organized quantum-dot pattern along the edges of the trapezoid of silicon?

From AFM-imaging of other surfaces (e.g. Au), it is known that there are strong random fluctuations after short time intervals. How long does the proposed Ge quantum-dot pattern live?

Does the quantum device proposed use the concept of entangled states?

R. Stanley Williams. The reason for the self-organization of Ge dots on Si surfaces is still a matter of considerable discussion and controversy. For our view of the matter and for a general set of references, see Kamins & Williams (1997) and Medeiros-Ribeiro *et al*. (1998).

For Ge islands on Si surfaces, we have annealed samples at temperatures from 550 to 650 °C for times up to 10 h. At the lower temperatures, the islands are quite stable for several hours, but at higher temperatures Si from the substrate diffuses up into the islands to form larger islands of Si–Ge alloy. We have submitted manuscripts describing these experimental observations. At room temperature, the islands are stable for at least two years, which is the age of our oldest samples.

My quantum device does not use the concept of entangled states. The point of my presentation was to see what could be done with computational systems based on quantum mechanics but using standard Boolean logic, rather than the logic associated with entangled quantum states.

Additional references

Medeiros-Ribeiro, G., Bratkovski, A. M., Kamius, T. I., Ohlberg, D. A. A. & Williams, R. S. 1998 *Science* **279**, 353.