COMPUTERS THEN AND NOW
(Turing Award Lecture by Maurice V. Wilkes)

MARTIN BRÜGGEMANN
Ortolfstr. 11a, 12524 Berlin

This paper tries to put Maurice V. Wilkes' Turing Award Lecture "Computers Then And Now" into a historical context and, by doing so, bring it closer to the reader. It was tried to put more interest on what was "then" than on what was "now", to underline the importance and impact of inventions made in the past for today's development in the field of computer science.

Abbreviations: ENIAC: Electronic Numerical Integrator And Computer; EDSAC: Electronic Delay Storage Automatic Computer; EDVAC: Electronic Discrete Variable Automatic Computer; bit: binary digit; RAM: random-access memory; byte: 8 bit

Introduction

Wilkes starts his lecture in 1946 with a chapter he named "Pioneering Days". The author did not follow this example, since the real pioneering days of modern electronic computing are seen hundreds of years earlier, before there was even thought of a computing device as elaborate as the ENIAC or its successors, computing machines such as Wilkes' EDSAC or the EDVAC.

Mechanical Era

Leibniz was born into a Europe that was in its ruins due to the 30-year-war, which was just over. It was a complete chaos, and still his parents enabled him to become a lawyer. But Leibniz was restless. He applied himself to various topics, as for instance philosophy, logic, mathematics – to name only a few.

As a philosopher, he spent a lot of time on thinking about topics such as the Dyadik (i.e., in modern words: binary notation). To him, it was peculiar how all natural numbers could be constructed out of nothing (binary 0) and an element (binary 1). In his papers, Leibniz used the word "algorithm" to describe and define a static set of instructions.

The word algorithm derives from the name of an Arabic mathematician, Al-Chwarizmi, who wrote several books about algebra. In his books, he defined instructions for operations such as adding, subtracting or multiplying. As time passed, his books and algebraic instructions more and more became a synonym with his name, and therefore we use the word algorithm today in the way Leibniz understood it.

Very early, Leibniz designed a four-operation calculating machine, but no one wanted to or was able to built his machine. On one of his trips to Paris he finally found someone who offered to do so, but Leibniz's machine was not very reliable, due to the technical restrictions of that time.

He died in 1716, but only nine years later a producing machine with stored program (as we might call it today) was designed – a loom, controlled by a punched-hole-strip of paper. This was done by Basile Bouchon from Lyon, and only three years later, in 1728, Falcon (also from Lyon) carried on to the closely related punched-hole card. This card-controlled loom underwent various improvements, the most famous of them in 1801 by Joseph Maria Jacquard.

In Jacquard's times, the handling of a carry was a problem. One might say that this is a minor problem – but it might have been the reason why machines for automatic production arrived earlier than ones for "automatic thinking" –i.e., like the universal computers we have so broadly in use today.

Jacquard was the first one to commercialize automatic looms – looms which are controlled by punched-hole cards. He was very successful, and in 1812 approximately 11,000 units of that kind were in use. And the idea of using little holes punched into cardboard was

not only applicable to automating looms: it was multi-purpose. It was used for programming the first computers and remained in use as long as until the 1960s.
To give an example of the capabilities of such a storage system: a carpet with a high degree of detail would have needed about 24,000 cards with 1,050 holes each (1 hole equals 1 bit). That sums to 25 million bit or 3.125 megabyte.
At the time Charles Babbage - a man we could surely look at as the inventor of the computer, born one hundred years early - started thinking about constructing an automatic calculating machine, thousands of automatic looms, controlled by punched-hole cards, were well in use all over Europe. So it is obvious why he used this technology for his own machines.
He invented two kinds of machines. His first project was called the "Difference Engine" and became operational in 1822, but it was more or less a simple calculating machine – not an automatic or general-purpose computing device.
Which cannot be said of his second project, the "Analytical Engine" (completed 1833). Babbage and his ideas have long been forgotten, but then scientists became aware that Babbage had, theoretically, invented every part a modern computer consists of, and he was well aware of the enormous power offered by such machines:
The Analytical Engine was equipped with an arithmetical unit (AU) - he called it "Mill" – which was controlled by punched-hole cards. The machine was capable of storing 1,000 words with 50 decimal digits each in its 167,000 bit wide storage. Input was implemented by use of the same system, by punched-hole cards. His machine was fully programmable and could perform very complex operations completely automatical.
It is believed that the Engine was something like an attraction for 19th-century aristocracy in England. So come that his landlord's daughter, Ada Lovelace, learned about it and became very fascinated in Babbage's projects. She was the first to write any piece of software and wrote very precise descriptions of the machine. It is only because of her that English engineers were able to reconstruct the Analytical Engine during the 1990's, and because of the descriptions being so accurate, it was operational almost instantly.

Konrad Zuse was a civil engineer, and in being so, he had endless tables of data to analyse. Since the calculations he performed were always following the same scheme, he wondered if and how they can be done by an automatic computing machine.
So he started design of his first computer, the Zuse Z1. But this machine was nothing more than a calculator – since it was not able to do any operation automatically. The first Zuse-built automatic computer that really deserved this name was the Z3, which became operational in 1941. He was not government funded and therefore short of money, so he had to use second-hand equipment. Plus, he had to re-invent Babbage's ideas. This fact makes the technical data of that machine appear even more ahead of its time:
The Z3 was the first device to use relays for switching instead of gears, this change made Zuse's device more precise than anything available at that time, on account of the energy that did not have to be wasted to friction anymore. It had a memory of 64 words, each 22 bit wide and used floating-point arithmetic.
But Zuse was not only ahead of his time in the field of hardware design. To program his machine, he developed the world's first programming language, which he called the "Plankalkül".

Pioneering Days

For someone not knowing him, John von Neumann might have appeared as a very strange person. For instance, he always wore the formal suit and tie of a banker, no matter what weather or temperature it was. He had been following his own rules for years. He owned a photographic memory that held the complete texts of works of literature. Yet he would phone home to ask his wife to help him remember an appointment. He loved to throw parties - and sometimes would steal away to work in his office while his guests remained downstairs. Among his friends he was nearly as well known for his traffic accidents as for his accomplishments in mathematics. A strong supporter of the military, he was fond of attending nuclearweapons tests. He died of cancer at the age of fifty-three.
Through it all, he was one of the century's most creative and productive mathematicians, lifting his intellectual scepter across a host of technical fields. Mostly he worked with pencil and paper, but in the years after 1945, for the first time in his life, he set himself the task of managing the design and construction of a piece of equipment. This was the Institute for Advanced Study computer, and it set the pattern for subsequent computers as we know them today.

What distinguished this IAS machine was programmability. It embodied von Neumann's insistence that computers must not be built as glorified adding machines, with all their operations specified in advance. Rather, he declared, they should be built as generalpurpose logic machines, built to execute programs of wide variety. Such machines would be highly flexible, readily shifted from one task to another. They could react intelligently to the results of their calculations, could choose among alternatives, and could even play checkers or chess.

This represented something unheard of: a machine with built-in intelligence, able to execute programs of wide variety. Such operate on internal instructions. Before, even the most complex mechanisms had always been controlled from the outside, as by setting dials or knobs. Von Neumann did not invent the computer, but what he introduced was equally significant: computing by use of computer programs, the way we do it today.

The roots of this invention lay not in electronics but in the higher reaches of mathematics, in a problem that tantalized specialists in mathematical logic during the early decades of the 20th century: the challenge of establishing basic foundations for math. These would take the form of an explicit set of definitions and axioms, or fundamental statements, from which all known results might be derived.

Everyone expected that such foundations could be constructed if people were only clever enough. David Hilbert of Göttingen University, widely regarded as the world's leading mathematician, summarized this viewpoint in a 1900 address: "Every mathematical problem can be solved. We are all convinced of that. Afler all, one of the things that attracts us most when we apply ourselves to a mathematical problem is precisely that within us we always hear the call: here is the probiem, search for the solution; you can find it by pure thought, for in mathematics there is no ignorabimus [we will not know]."

In fact, however, a powerful ignorabimus lay at the center of the problem of mathematical foundations. The man who demonstrated this was Kurt Gödel, a logician at the University of Vienna. He was a smallish man with an earnest expression and a thick pair of glasses; he appeared even smaller than he was because of his reluctance to eat. Psychological depressions and other illnesses dogged him throughout much of his life, made more serious at times by his distrust of doctors. In contrast with the gregarious and hearty von Neumann, Gödel was solitary in his habits, but he did form a few close relationships. One was his lifelong marriage to Adele Nimbursky, a former cabaret dancer. Another was a warm friendship with Albert Einstein.

In two epochal papers, published in 1931, when he was twenty-five, Gödel showed that no foundations could be constructed. More particularly, he showed that if anyone tried to set forth such foundations, it would be possible to devise mathematical statements that were "formally undecidable"— incapable of being proved or disproved using the proposed foundations.

In particular, this work offered two major results for the eventual development of computer science. To prove his theorems, Gödel introduced a notation whereby statements in mathematical logic were encoded as numbers. Every such statement could be expressed as an integer, usually a very large one, and every integer corresponded to a statement in logic. This introduced a concept that would be key to the later advent of computer programming: that not only numerical data but also logic statements - and by extension, programming instructions could be expressed in a common notation. Further, Gödel's work showed that this notational commonality could give results of the deepest significance in mathematics.

Among the mathematicians who soon took up the study of these matters was Alan Turing, of Cambridge University. Turing was a vigorous man, fond of running and cycling, and sometimes eccentric. Issued a gas mask he wore it to prevent hay fever. Fearing that British currency would be worthless in World War II, he withdrew his savings and purchased two ingots of silver, buried them in his yard-and then failed to draw a suitable treasure map that would permit him to find them. And when his bicycle developed the habit of having its chain come loose, he refused to take it in for repairs. Instead he trained himself to estimate when this was about to happen so he could make timely preventive fixes by himself.

Turing was a 25-year-old undergraduate when he made his major contribution to computer science. It came in a 1937 paper, "On Computable Numbers With An Application To The Entscheidungsproblem" in which he specifically dealt with an imaginary version of the computer. This idealized machine was to follow coded instructions, equivalent to computer programs. It was to deal with a long paper tap that would be marked off in squares, each square either black or white and thus representing one bit of information. On this tape, in response to the coded commands, the machine would execute a highly limited set of operations: reading, erasing, or marking a particular square and moving the tape.

Analyzing this idealized computer, Turing proved that it offered properties closely related to Gödel’s concept of formal undecidability. What was important for computer science, however, was another realization: that with sufficiently lengthy coded instructions this simple machine would be able to carry out any computation that could be executed in a finite number of steps. Here, in its essential form, was the concept of a general-purpose programmable computer.The basic idea of a calculating machine was not new (see above). What was new and pathbreaking in Turing’s work was that for the first time he gave a clear concept of what a computer should be: a machine that carries out a few simple operations under the direction of a program that can be as intricate as one may wish.

These developments were very interesting to John von Neumann. As a student in Germany (he was born in Hungary in 1903), he had worked closely with Hilbert himself, plunging deeply into the search for mathematical foundations. He had shared Hilbert’s belief that such foundations could in fact be constructed, had written a paper that contributed some mathematical bricks to the intellectual masonry-and was surprised and chagrined by Gödel’s proofs. He had not thought that formal undecidability might exist, and he came away with the feeling that Gödel had scooped him.

He had plenty of reasons to feel confident, however. The son of a Budapest banker who had received a minor title of nobility, the source of his “von”, he had shown himself very early to be a what one might call a “Wunderkind”, dividing eight-digit numbers in his head at age six and talking with his father in ancient Greek. By age eight he was doing calculus and demonstrating a photographic memory: he would read a page of the Budapest phone directory and recite it back with his eyes closed. His father’s wealth made it easy for him to attend the University of Budapest, from which he traveled widely: to the University of Berlin, to Zurich and its equally famous university, and to Göttingen, the world’s center of mathematics. At age 22 he received his Ph.D.

Von Neumann had made his reputation during the 1920s, establishing himself as clearly one of the world’s outstanding mathematicians. Particularly significant was his work in developing a rigorous mathematical basis for quantum mechanics. That brought him an invitation to Princeton University, which he joined in 1930, when he was 26. Then in 1936 Turing came to Princeton to do his graduate work; he was 24. Von Neumann, who had moved to the Institute for Advanced Study in 1933, was quite interested in Turing’s work and offered him a position as his assistant after he received his doctorate, but Turing chose to return to Cambridge.

Meanwhile, von Neumann was doing much more than reading his colleague’s papers. During the early 1940s he began to work extensively on problems of fluid flow. These problems were widely regarded as nightmares, marked by tangles of impenetrable equations. To von Neumann that meant they were interesting; understanding them could lead to such consequences as accurate weather prediction, and because such problems posed intractable difficulties, they were worthy of his attention.

Then came the war and the Manhattan project. Von Neumann’s expertise in fluid flow now took on the highest national importance. As the work at Los Alamos advanced, he became responsible for solving a problem that was essential to building the plutonium bomb.

Even von Neumann’s brilliance was inadequate for this. He had hoped that ingenuity and insight would enable him to simplify the pertinent equations to a form both solvable and sufficiently accurate. His collaborator Stanislaw Ulam insisted that it would be necessary to face their full complexity and calculate them, in an age when there were no computers, using methods that would later be programmed to run on computers. Fortunately, the Los Alamos lab was due to receive a shipment of IBM calculating machines. Stanley Frankel, another Los Alamos man, set up a lengthy sequence of steps that these machines could carry out, with Army enlistees running them. It amounted to a very slow computer with human beings rather than electronic devices as the active elements, but it worked. Von Neumann got the solutions he needed, and he proceeded to design the high-explosive charges for Fat Man, the bomb dropped on Nagasaki.

Meanwhile, at the University of Pennsylvania, another effort as secret as the Manhattan Project was under way: the construction of the first electronic computer. This was ENIAC, an Army-sponsored project intended for use in calculating the trajectories of artillery shells. Its employment of vacuum tubes rather than people as active elements represented a decided advance, but while the potential value of such tubes for high-speed computing was widely appreciate, the tubes of the day were not particularly reliable. That did not matter when only a few were needed, as in radar or radio, but it would matter greatly in a computer, where a single failed tube could vitiate a lengthy calculation. (Because of this, Harvard’s Howard Aiken had gone to work on a computer that would use the eletromechanical switches of telephone circuitry. They were far slower than vacuum tubes, but still much faster than human beings, and they were reliable.)

The ENIAC project leaders, John W. Mauchly and J. Presper Eckert, Jr., solved the reliability problem in a simple way. They were working with tubes whose manufacturers had guaranteed a service life of twenty-five hundred hours. With 17,468 tubes in ENIAC, that meant one could be expected to fail, on the average, every eight minutes - and with major computations requiring weeks of operation, this was quite unacceptable. Eckert, however, simply “unloaded” the tubes, arranging it so that they would handle no more than one-half of their rated voltage and one-fourth of their rated current. This reduced the failure rate from one every eight minutes to about one every two days, which was sufficient for practical operation.

The Army’s representative on the project was Lt. Herman H. Goldstine, who had taught mathematics at the University of Michigan. He was working out of the Aberdeen Proving Grounds in Maryland, where von Neumann was a consultant.

ENIAC was a large air-conditioned room whose walls were covered with cabinets containing electronic circuitry - three thousand cubic feet of it. It weighed thirty tons and drew 174 kilowatts of power. Its computational speed and capability would fail to match the hand-held programmable calculators of the mid- 1970s, but even so, it was such an advance over all previous attempts at automatic computation as to stand in a class by itself. Still, it was not without its faults, as its builders were well aware. Its main memory (RAM) could hold only a thousand bits of information - the equivalent of about three lines of text. And it was completely lacking in any arrangements for computer programming.

You did not program ENIAC; rather, you set it up, like many other complex systems. Although it was a general purpose computer, able to solve any problem, it relied on physical interconnections. You prepared for a particular problem by running patch cords between jacks and other plugs, with cabling up to eighty feet long. The task could take two days or longer. In a 1943 report the builders admitted that “no attempt has been made to make provision for setting up a problem automatically,” adding that “it is anticipated that the ENIAC will be used primarily for problems of a type in which one setup will be used many times before another problem is placed on the machine.”

By the summer of 1944, however, Eckert, Mauchly, and their colleagues were already beginning to think seriously about ENIAC’s successor. This would have the name EDVAC. As early as January of that year Eckert had described a computer in which an “important feature” was that “operating instructions and function tables would be stored in exactly the same sort of memory device as that used for numbers.” Eckert was also inventing an appropriate memory device: a “delay line”, or long tube filled with mercury in which bits of data would take the form of pressure pulses traversing the tube at high speed. And in October 1944, at Goldstine’s urging, the Army awarded a $105,600 contract for work on the EDVAC concept.

Into this stimulating environment stepped von Neumann. He joined the ENIAC group as a consultant, with special interest in ideas for EDVAC. He helped secure the EDVAC contract and spent long hours in discussions with Mauchly and Eckert. Von Neumann’s particular strength was the logical structure of a computer, the details of its logic operations. His leadership made the EDVAC discussions more systematic.

In late June 1945, working at Los Alamos, von Neumann completed a 101-page document titled “First Draft of a Report on the EDVAC”. In his clear and penetrating way, he set forth an overview of the design of a digital computer that would feature stored-program operation. Von Neumann boldly drew comparisons between his electronic circuits and the brain’s neurons, emphasizing that just as the brain relies on its memory, so the computer would depend on its programs. Goldstine soon was distributing copies to interested scientists. In time the “First Draft” would become one of the most influential papers in computer science.

Goldstine circulated the draft with only von Neumann’s name on the title page. In a later patent dispute, von Neumann declined to share credit for his ideas with Mauchly, Eckert, or anyone else. So the “First Draft” spawned the legend that von Neumann invented the stored-program computer. He did not, though he made contributions of great importance. But by writing the “First Draft” and subsequent reports, he gave a clear direction to the field. The prestige of his name ensured that he would be followed. “The new ideas were too revolutionary for some, powerful voices were being raised to say that to mix instructions and numbers in the same memory was to go against nature.” Maurice Wilkes once said. Von Neumann stilled such doubts.

As 1945 proceeded, he became convinced that he should not merely write about stored-program computers but should take the lead in another way: by building one.

Then the ENIAC group broke up. The source of this was a new director of research at the University of Pennsylvania, where the computer had been built, Irven Travis. Travis had spent his war in the Navy and proposed to run a tight ship now that he was back in the civilian world. He soon was quarreling with Eckert and Mauchly over the issue of patents.

The two ENIAC inventors saw great commercial prospects in computers and had a letter from the university president that agreed they could hold patents on ENIAC. Travis, however, insisted that they must sign patent releases. He made no bones about it; in one meeting with Mauchly he stated “If you want to continue to work here at the university, you must sign these agreements.” Mauchly and Eckert refused and were soon out on their own as independent entrepreneurs.

By the summer of 1946, then, three groups were seeking to build a stored-program computer along the lines of the “First Draft.” Eckert and Mauchly had by far the most experience in this area but were out in the cold with little money, few contacts, and slight business experience. The remnants of the ENIAC group, at the University of Pennsylvania had few good people but were committed by contract to build an EDVAC, and build it they would, however slowly. Von Neumann had the overall vision, the charismatic reputation, the genius, and the acquiescence of the IAS. What he lacked was experience in project management.

Of these deficiencies, von Neumann’s was the most easily remedied. He had technical support from RCA, which had built a lab in Princeton. He had Herman Goldstine, who left the Army to join him. And at Norbert Wiener’s recommendation he hired Wiener’s wartime assistant, Julian Bigelow, who had worked on radar-guided fire control of antiaircraft guns and who knew how to build electronic systems of a very demanding character.

The computer was to be built in the boiler room of Fuld Hall, the main building at the IAS. As Bigelow describes the work “Von Neumann would put half-finished ideas on the blackboard and Goldstine would take them back down and digest them and make them into something for the machine. On the other hand, von Neumann often had only the foggiest ideas about how we should achieve something technically. He would discuss things with me and leave them completely wide open, and I would think them over and come back with an experimental circuit, and then my group would test it out.”

Good Language And Bad

When completed - in a building of its own, well across the IAS campus - the computer had only twenty-three hundred vacuum tubes, considerably fewer than ENIAC’S almost eighteen thousand. It was fully automatic, digital, and general-purpose, but like other programmable computers of its generation, it was built years in advance of programming languages such as Fortran or Pascal. Its commands instead were written directly in machine language, long strings of ones and zeros. An expression such as “A+B”, for instance, might be rendered as something like 01101101 10110110 01110011; a significant program would feature many pages written in such notation. This was practiced mostly due to performance reasons, since the technology of automatic programming was only about to mature at that time.

In Bigelow’s words, there were “none of the tricks that we now have. This was a case where von Neumann was so clever technically that he had no problem with it. And he couldn’t imagine anyone else working with a computer who couldn’t program in machine code.”

How significant was this IAS computer? The science historian Joel Shurkin, who has sought to assess fairly the claims of various inventors as to priority, writes, “Von Neumann’s technical contributions are manifest and beyond controversy. The machine he designed would be faster than anything else .... While all the other computer makers were generally heading in the same direction, von Neumann’s genius clarified and described the paths better than anyone else in the world could. Moreover, many of the developments in programming and in machine architecture at the Institute profoundly influenced future computer development .... While others were using crude digital instructions for their machines, von Neumann and his team were developing instructions (what scientists call codes) that would last, with modification, through most of the computer age.”

The machine received its baptism with the nation again at war, in Korea, and with the hydrogen bomb now a matter of highest priority. Von Neumam, who had maintained his leadership in nuclear-weapons work, arranged to run a problem dealing with H-bomb physics. It would be the most extensive computation ever carried out. “It was computed in the summer of 1950,” says Bigelow, “... while the machine had clip leads on it. We had engineers there to keep it running and it ran for sixty days, day and night, with very few errors. It did a nice job.” The way was open, then, for the computer to sweep all before it.

In 1953, Wilkes took part in a controversy about the use of higher level programming languages. The main disadvantage of automatic programming is it’s lack of efficiency: A program’s running time increases and one

needs more memory, thus, a bigger computer for the same problem.
But it's main advantage is that it makes the art of programming computers accessible to nearly anybody, without the need to deal with machine-readable binary code. To Wilkes, the arrival of automatic programming only made expansion of the field of computing possible.

The Higher Syntax

The world's first compiler-compiler was built by Brooker and Morris, and it was a great achievement in compiler construction. Now it was possible to design multi-use programming languages, which does not mean that something like an omnibus language will always be available.
But the mobility of languages between computers increases, since the possibility to create a compiler for every platform was finally given. This makes data machine-independent, and Wilkes envisions a "future" that provides such a high degree of connectivity between devices that, when buying a computer, one would be able to choose the manufacturer of the computer's processor. A "future" which has become very real today.

Design And Assembly

With computers becoming more complex, bigger and faster, a new interest in the field of computer graphics began to arise. Up to then, computers were only experimental and not applied to any practical use at all. To change that, there was the need to develop new ways to communicate with the machines, to make the interface more natural and, therefore, more intuitive, more human. The target group for such improvements on computing machinery were engineers; they were used to communicate by and about data which is visualized, by charts and tables – something computers were not able to do then. But Wilkes does not stop here – he sees a need in computer-controlled machines for object manipulation, and he already envisions fully-automated factories, pure fiction in times of the first computers.

The Next Breakthrough

After substantial technical advances in the field of computer science, for instance, higher-level programming, the invention of the transistor and integrated circuits, Wilkes in his lecture saw the next breakthrough close at hand. By using integrated circuits, computers can be made highly parallel, i.e., they are able to compute several problems at the same time. Computers as elaborate as this can be used for 2-D pattern recognition, or OCR (optical character recognition).

Summary

As if computation carried with it some dreadful incubus, a number of its pioneers would die amid tragedy. Alan Turing was the first, in 1954, at age forty-one. Convicted in England of soliciting sexual favors from a teenage boy, he was given a choice of prison or hormone treatments. He chose the hormones and soon found his breasts growing. Driven to despair, he made up a batch of cyanide in a home laboratory and died an apparrent suicide.
For von Neumann it was even worse. In the summer of 1955 he was diagnosed with bone cancer, which soon brought on excruciating pain. Early in 1957 he too was gone.
For Kurt Gödel it was his own personal daemons that would drive him to death. In an epic escape from Nazi-occupied Austria he and his wife had crossed the Soviet Union and then the Pacific to reach the United States. From 1940 to 1976 he was himself a member of the IAS. After his wife underwent surgery and was placed in a nursing home, in 1977, Gödel refused to take any food. He starved himself to death. When he died early the next year, the death certificate of this great logician stated that the cause was "malnutrition and inanition caused by personality disturbance."

These great men might fade away of memories, but all around the world are today's computers, which still follow the directions von Neumann set forth in his "First Draft" and subsequent writings and that he demonstrated in his project at the IAS. These computers, rather than plaques or busts cast in bronze, are among the true monuments to the cheerful and highly creative man who was John von Neumann.

References

- Computers Then And Now, Wilkes, 1967
- How von Neumann showed the way, Heppenheimer, T.A., 1990
- Die Ahnen des Computers, Kaufmann, 1974
- Computer Organization and Design, Patterson/Hennessy, 1997