Konrad Zuse: Reflections on the 80th Birthday of the German Computing Pioneer

Wolfgang K. Giloi

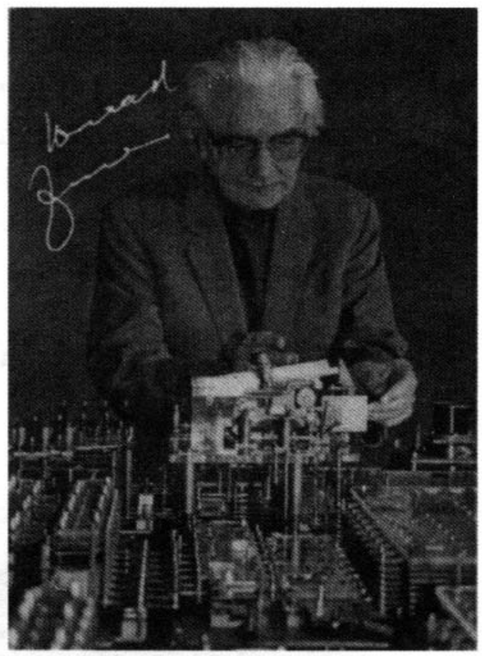
GMD Research Institute for Computer Architecture and Software Technology

Relatively little known outside of Germany, Konrad Zuse realized the first functioning computer in 1941 and also developed with his so-called Plankalkül in 1943-1945 the first high-level programming language. However, his inventions were overshadowed by World War II, and his work was overtaken in the public mind by the works of Aiken and von Neumann. This essay, a shortened and revised version of a eulogy by the author in December 1996 at the celebration of Zuse's 80th birthday at his former alma mater, the Technical University of Berlin, gives an overview of Zuse's work and puts his contributions in context.

1. The Computer Engineer

Zuse started developing program-controlled binary calculators in 1936 [1]. In 1941, he completed the first fully operational digital computer. His machine, called Z1, was the size of a pool table and featured mechanical binary logic and flip-flop memory. Its arithmetic unit worked with a "semi-logarithmic" number representation invented by Zuse—years later this became known as *floating-point numbers*. The original Z1 did not survive the air raids on Berlin. By initiative of the German National Research Center for Information Technology (GMD) and with contributions from the industry, a replica was built in 1988-1989 by two students and a mechanic under Zuse's supervision, using his original documentation. The replica has been installed in 1989 at the Museum of Technology at Berlin. It proves that the civil engineer Konrad Zuse was, among his other talents, an ingenious designer of mechanical hardware.

FIGURE 1. Konrad Zuse with replica of the Z1



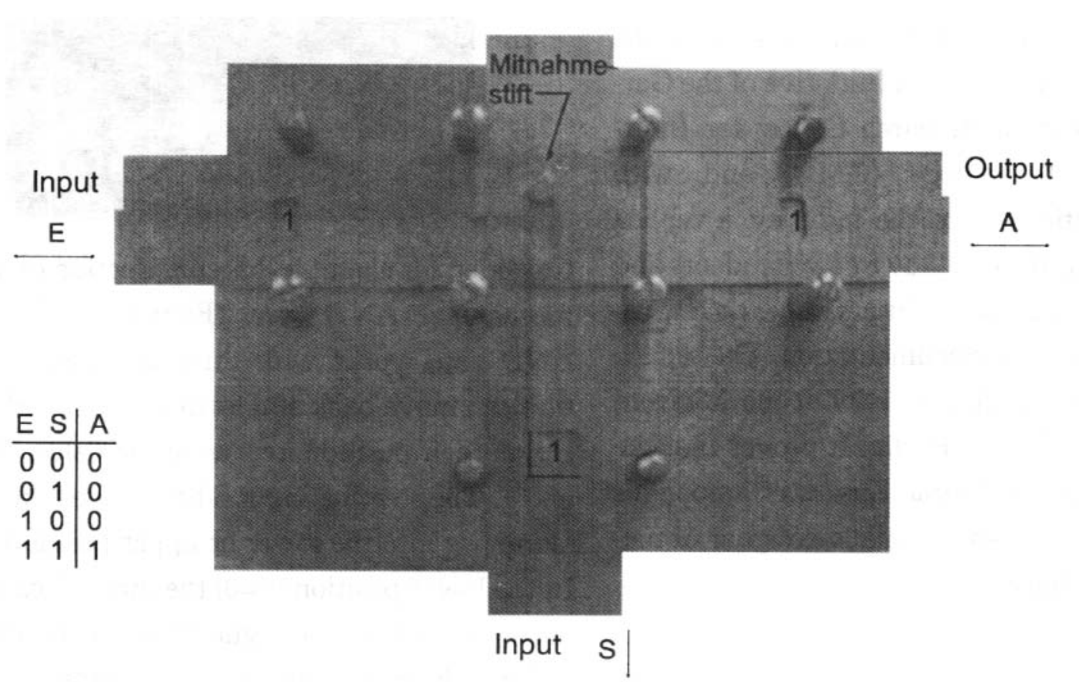
Consider, for example, his construction of a mechanical AND gate (Figure 2). The AND gate works with three metal strips that can move back and forth and, depending on their position, represent the values 0 and 1. The steering input S brings the take-along peg into the lower or upper position. In the lower position (S=0) the input E can move left (E=0) or right (E=1) without "taking along" the output A. In contrast, in the upper position (S=1) the output A follows the input E.

The mechanical ALU of the Z1 had some problems: under certain conditions it would get jammed and had to be reset. While struggling to make his mechanical design work, Zuse started in 1938 to design a second machine, the Z2. The ALU of the Z2 employed telephone relays instead of mechanical switches, while the memory still consisted of mechanical flip-flops.

Zuse's work on his digital computers was temporarily interrupted by the outbreak of the war, when he was drafted into the army. Later he was exempted from military service in order to work at the Henschel aircraft company. There he designed the special-purpose computers S1 and S2, relay machines that were used to compute the wing corrections needed to make the unmanned airplane HS293 (a kind of buzz bomb) fly. The parts for these drones were-manufactured by using inexpensive metal manufacturing techniques and required individual corrections on the wings to render them capable of flight. Ironically, at the time Zuse's computers were not considered important for the German war effort.

The Z3, the successor of the Z2 built by Zuse from 1939 to 1941, was completely a relay machine. It was actually the first fully operational, programmable digital computer in history. Immediately after completion of the Z3, Zuse started designing an upgraded version, the Z4. He had the good luck that, toward the end of the war, both he and the Z4 were evacuated to a safe place in Bavaria, before the Red Army closed in on Berlin. In 1951 Zuse

FIGURE 2. Mechanical AND gate ("Mitnahmestift" is German for "carry along peg")



leased the Z4 to the Technical University of Zurich (ETH), where it was in operation until 1953. A detailed description of the architecture of the Z1 and Z3 is provided in an article by Raúl Rojas [2]. After the war, Zuse formed a computer company that, in the late fifties and early sixties, became the main supplier of German universities with an affordable electronic digital computer, the drum machine Z22.

Through Brian Randell's book [3], an English translation of Zuse's first patent application of 1936 has become more widely known. The application proves that Zuse had already developed all major concepts of the digital computer years before Burks, Goldstine, and von Neumann wrote their famous report *Preliminary Discussion of the Logical Design of an Electronic Computing Instrument* [4]. Indeed, Zuse's vision of digital machines and their potential capabilities went beyond the purely sequential computer that is known to this day under the name "von Neumann machine." For example, the possibility of array processing and even of parallel processing is already mentioned in his patent application of 1936 [3].

2. The Plankalkül

Zuse was far more than a tinkerer. Inspired by the *Dyadik* of G. W. Leibniz (1646-1716), he recognized early that the funtioning of a computer based on binary numbers could be described mathematically by propositional calculus. Propositional calculus (also called Boolean algebra) became the foundation for the design of his computers. Zuse was also interested in predicate logic, and that led him in 1945 to invent the first higher programming language, the so-called Plankalkül (PK), a short expression in German for "calculus for computing plans (PK)." PK was intended to provide a formal description of a computing procedure, for both combinatorial and numerical problems. Zuse writes about his motivations [5, p. 25]: "I was not concerned with programs for numerical calculations when I developed the PK. I did not expect any difficulties in this field then, and consequently I concentrated my efforts mainly on the logical problems beyond the common numerical calculations." Chess was one of Zuse's motivating combinatorial challenges. His initial description of the Plankalkül dedicates 44 pages to this topic. His vision that eventually machines would be better chess players than humans has recently been proven right by the successes of Deep Blue.

On the surface, the Plankalkül possesses important characteristics of later algorithmic languages such as ALGOL [6]. For example, it contains control constructs corresponding to IF and REPEAT-UNTIL instructions; and, like other higher-level programming languages, it contains Boolean, integer, real, and complex scalar types. A more detailed comparison with other higher-level programming languages (that were invented much later) reveals important differences, in particular the visibility of the binary representation of the aforementioned data types at the programmer level and the existence of data structure types. Thus, unlike programming languages such as ALGOL or FORTRAN, which hide the binary representations of such data types from the programmer, the PK has really only one elementary data type: the single bit. All other data types are defined as structures of bits and their operations defined by the programmer. In this respect, the PK provided a lower level of abstraction than ALGOL or FORTRAN. On the other hand, the PK contains data structure types such as binary trees, arrays, or lists of tuples, which enable the representation of generalized graphs or geometric structures [5]. Structures can be dynamic, that is, generated during run time.

Lists are always dynamic: they can grow and shrink. There are list operations for generating sublists of elements satisfying a certain predicate, the determination of the number of elements in a list, the retrieval of the first or last element in a list, the search for the largest or smallest element in a list, insertion, deletion, and concatenation. These features of PK can be found again a decade later in the design of LISP and APL. Thus, PK, like LISP and APL, has a much higher level of abstraction than the typical numerically oriented programming languages that, by and large, mimic the ability of the von Neumann machine to modify one memory location at a time, as they allow the transformation of a whole data structure in one operation.

Zuse himself, as well as other authors [6], saw the Plankalkül as a forerunner of algorithmic programming languages such as Fortran and Algol. This view was based on some major characteristics the Plankalkül has in common with those languages, such as the notion of variables, including variable declaration and assignment, the notion of subroutines of the function type, and the conditioned or repetitive execution of subprograms or programs. A detailed description of the Plankalkül and its relation to other computing concepts is provided in [8].

3. Another Casualty Of War

Konrad Zuse was a visionary, and, given the originality and breadth of his contributions, it would seem appropriate to call him the father of the computer. However, in 1964 the Honorable Judge Earl Larson of the U.S. District Court in Minneapolis decided in Honeywell vs. Sperry that the inventor of the computer was John Vincent Atanasoff [9]. Atanasoff had worked at developing a computing device, later called the ABC, at Iowa State College from 1938 to 1942. The ABC used vacuum tubes and was restricted to the addition and subtraction of vectors [10]. Besides Zuse, whose "general-purpose" computer Z3 was fully operational in 1941, other losing contenders were Eckert's and Mauchley's ENIAC (Electronic Numerical Integrator and Computer), built at the Moore School of Electrical Engineering at the University of Pennsylvania from May 1943 to 1945 [11], and the Mark I built by Howard Aiken at Harvard University between 1939 and 1944 [12]. This ruling, as incredible as it sounds, is still valid today.

In 1941 Zuse applied for a patent with 51 separate claims for all important aspects of the relay-based computer Z3 [1]. The claims in particular referred to a programguided computing engine with an addressable memory, an instruction set (Rechenwerk) for floating-point arithmetic, and a program interpretation unit (Programwerk) to guide the computations. These are exactly the components of the von Neumann computer, described in the paper by Burks, Goldstine, and von Neumann. Because of the secrecy surrounding the war effort, Zuse's application did not become public until 1951. Zuse's application was contested initially by Triumph and later by IBM. It was not until 1967, 26 years after the initial patent application was submitted, that the German patent court decided to reject Zuse's patent claims without appeal: the court judged Zuse's ideas to be novel, but not of sufficient depth to warrant a patent! Thus, Zuse never realized any material gain from his epochal inventions.

Zuse did receive the IEEE Computer Pioneer Award in 1980 for his work in building the first process control computer, the S2, which featured an integrated analog-to-digital conversion under program control. However, the prize for building the first operational general-purpose digital computer went to Aiken, whose computer Mark I (which also employed relays) was not operational until three years after the Z3.

Because of the forced exodus of Germany's elite artists and scientists during the Third Reich, the German language ceased to be a major science language. Moreover, the postwar conditions in Germany hindered for years the spread of scientific publications. Zuse's monograph on the Plankalkül, for which there was no precedence, evolved in the second half of the forties. However, it was hardly known in Germany, let alone outside the country, and his work was translated into English only three decades later [5]. For the development of computer science it certainly was a loss that the Plankalkül remained practically unknown. Had Zuse's monograph become as widely known as the report of Burks, Goldstine, and von Neumann [4], developments such as the relational data base or logical programming [13] or the creation of standardized forms of knowledge representation in artificial intelligence might have evolved sooner.

Konrad Zuse himself said that "the best inventions aren't worth anything if they come too early." In saying this, he may have also been thinking of Charles Babbage (1792-1871), who had already developed the fundamental ideas of a programmable computing engine but was not able to construct a working model of the "analytical engine" (as he called it) because of the technical limitations of his time. Zuse was not happy with the term "von Neumann" computer. He was of the opinion that such a computer was more appropriately called the Babbage computer or, even better, the Zuse computer [4]. In fact, it was Zuse and not Babbage who first thought of the idea of addressable memory with binary addresses, the key feature of the von Neumann computer (Babbage realized memory by assigning each memory cell a particular position in his punch cards). However, unlike Babbage's concept, Zuse's first computer did not allow for program jumps (and thus iteration): the Z2 and Z3 read instructions from a punched tape, and they could be read only in a particular order. In contrast, the von Neumann machine stores instructions in memory and thus can randomly access them. Zuse emphasized later that he thought from the beginning of storing both data and instructions. But since he initially did not understand the ramfications of incorporating goto's, he preferred to leave them out [5,6]. We suspect that part of the reason was that he did not want to waste precious memory for storing instructions, in particular since the numerical calculations he was familiar with did not (by and large) require loops. Thus, it was not until the Z4 that Zuse's computers were "von Neumann computers" in all regards.

Acknowledgment: The author thanks Christian Bischof for his help in revising the initial manuscript.

References

[1] Zuse K. Der Computer — Mein Lebenswerk, Verlag Moderne Industrie, Munich 1970.

[2] Rojas R. Konrad Zuse's Legacy – The Architecture of the Z1 and Z3, IEEE Annals of the History of Computing, 19(2):5-16, 1997.

[3] Randell B. (ed.). *The Origins of Digital Computers*, Springer-Verlag, Berlin-Heidelberg-New York 1973

[4] Burks A. W., Goldstine H. H., von Neumann J. Preliminary Discussion of the Logical Design of an Electronic Computing Instrument, in Taub A.H.(ed.): *Collected Works of John von Neumann*, vol. 5, MacMillan, New York 1963, 34-79.

[5] Zuse K. *The Plankalkül*, GMD Report no. 175, 2nd edition, R Oldenbourg-Verlag, Munich-Vienna 1989

[6] Bauer F. L., Wössner H. The Plankalkül of Konrad Zuse, a Forerunner of Today's Programming Languages, Elektronische Rechenanlagen 1972, H.2

[7] Iverson K. E. A Programming Language, J.Wiley & Sons, New York 1962

[8] Giloi, W. Konrad Zuse's Plankalkül: The First High-Level, "non von Neumann" Programming Language, IEEE Annals of the History of Computing, 19(2):17-24, 1997.

[9] Lee, J. A. N. looking.back. In: Computer Magazine 2/1996, (looking back: monthly column in the 1996 issues of Computer to mark the Computer Society's 50th anniversary).

[10] Burks, A. W., and Burks, A. R. The First Electronic Computer - The Atanasoff Story, The University of Michigan Press, Ann Arbor, 1988.

[11] Burks, A. W., and Burks, A. R. The ENIAC: First General-Purpose Computer, Annals of the History of Computing, 3(4):310-399, 1981.

[12] Aiken, H., and Hopper, G. The Automatic Sequence Controlled Calculator, 1946, reprinted in B. Randell, The Origins of Digital Computers, Springer-Verlag, Berlin, 1982.

[13] Kowalski R. Predicate Logic as Programming Language, *Proc. IFIP Congress 1974*