The stored-program universal computer: did Zuse anticipate Turing and von Neumann? (English summary)


While the paper by Haigh focussed on von Neumann and his involvement with the ENIAC, the paper by Copeland and Sommaruga has Turing’s 1936 paper on computable numbers and his proposal for the automatic computing engine (ACE) as its centerpieces, and Zuse as an interesting side figure. The paper contains a biography of Zuse (section 2) and a discussion of Zuse’s ideas of building a machine and programming it using the schemes of a *Plankalkül* (sections 5 & 6). As with most relay-based calculator designs from the 1940s, Zuse separated the program from the machine and thus did not implement a stored-program concept, but the authors discuss some interesting passages where Zuse speculates on programs that recycle parts of themselves or even partially modify themselves.

The paper by Haigh argued that the stored-program concept as a term comes years after the events and that a historiographical finer distinction should be made between two distinct ideas present in [J. von Neumann, “First draft of a report on the EDVAC”, Moore Sch. Electr. Eng., Univ. Pennsylvania, 1945; see also IEEE Ann. Hist. Comput. 15 (1993), no. 4, 27–75; MR1241101], notably the von Neumann architecture on the one hand, and the modern code paradigm on the other hand. Copeland and Sommaruga propose another, yet finer distinction, an onion-layer model of six programming paradigms, P1 to P6 (section 4). P1 is a computer programmed by wiring (as an example they cite Colossus) and P6 the fully developed stored-program computer with the facility of modifying its own instructions. (It should be remarked that stages P3 and P4 do not correspond to any actual computer, but can only be found in theoretical form in Turing’s 1936 paper. P3 is a machine only potentially capable of modifying its instructions, viz. the universal machine that reads and executes but does not modify a description number of another machine. P4 is the same but with the use of markers.)

The authors put Turing’s ACE proposal (which they consistently date 1945 while most other researchers take 1946 as its publication date) and Zuse’s *Plankalkül* (1945) alongside the “First draft” accredited to von Neumann as the defining documents that establish the new stored programming paradigm P6. It is useful to take a step back from the priority discussions and the introduction of new historical and philosophical subtleties and distinctions found in Haigh’s and Copeland and Sommaruga’s papers. Actually, the most important conclusion that one can deduce from these articles, though the authors do not really push this point, is that the stored-program concept is the result of a confluence of ideas, has been implemented in more than one place in more than one way, and cannot be studied in conceptual isolation but needs a context.
It should finally be noted that the reader should be cautious with some of the statements in Copeland and Sommaruga’s paper. First of all, the paper at times seems to conflate the notion of the stored-program concept with that of a universal machine (as in their title), making some of their statements inaccurate. Further smaller inaccuracies are: electromechanical devices are not a factor of a hundred or a thousand faster than human computers, but only a factor of five to ten (p. 45); the English Bombe was an electromechanical device, not an electronic one as suggested, and programming by wiring made it only marginally faster (p. 78); the 2-state, 3-symbol machine mentioned on p. 88 (whose weblink is broken) is not a universal machine but only a weakly universal machine at best, since 2-state, 3-symbol Turing machines are decidable (see, e.g., the survey by T. Neary and D. Woods [in SOFSEM 2012: theory and practice of computer science, 385–405, Lecture Notes in Comput. Sci., 7147, Springer, Heidelberg, 2012; MR2958190]).

{For the collection containing this paper see MR3410164}

Maarten Bullynck

© Copyright American Mathematical Society 2018