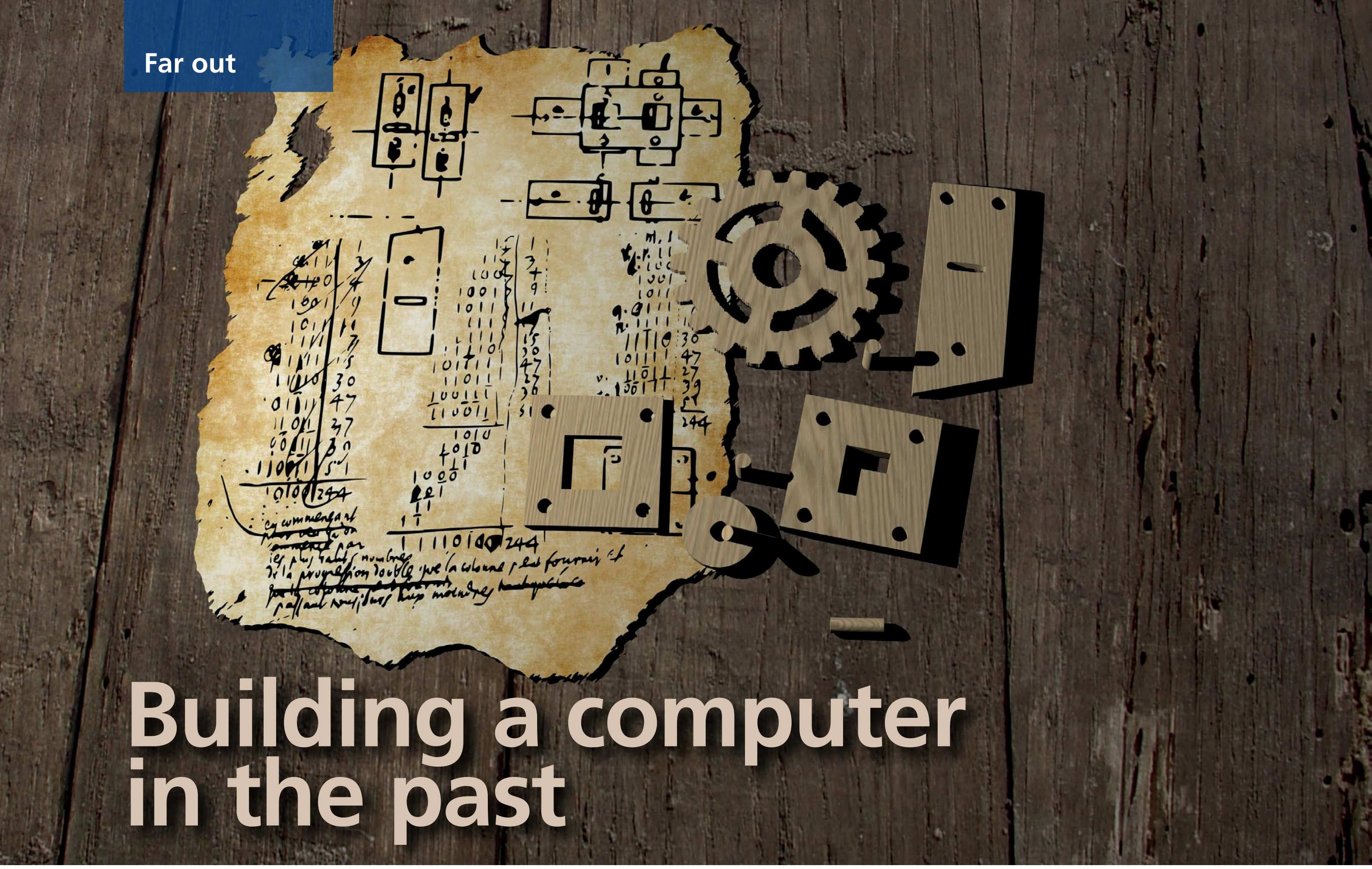


Building a computer in the past

Imagine that you are thrown back in time by hundreds of years. Somehow, you manage to make your life comfortable, but it is missing a true purpose – a computer, that is. Could you build one?

Story by Ville-Matias Heikkilä

Images by Ville-Matias Heikkilä, Oona Räisänen, Nasu Viljanmaa, Wikimedia Commons (Klaus Nahr, Bruno Barral)

Modern computers are electrical devices based on microchips. It would be too easy to assume that a computer cannot be built without first developing an enormous amount of different tools and techniques for the manufacture of different electronic components. Luckily, the outlook for a time traveller is not as gloomy. You can build a programmable computer by using more coarse techniques that have been used for several millennia already.

In the 1930s, the German engineer Konrad Zuse built a programmable calculator in his parents' living room. It is currently known as the Z1 computer. It read instructions off punched tape and followed them in order to perform basic arithmetic operations on 22-bit floating point numbers. You could save numbers in the work memory, read them from user input and print them out. In other words, it met all the criteria for a computer. From a time traveller's point of view, the Z1 was interesting because it was completely mechanical: the only electrical part was a motor that rotated the mechanism.

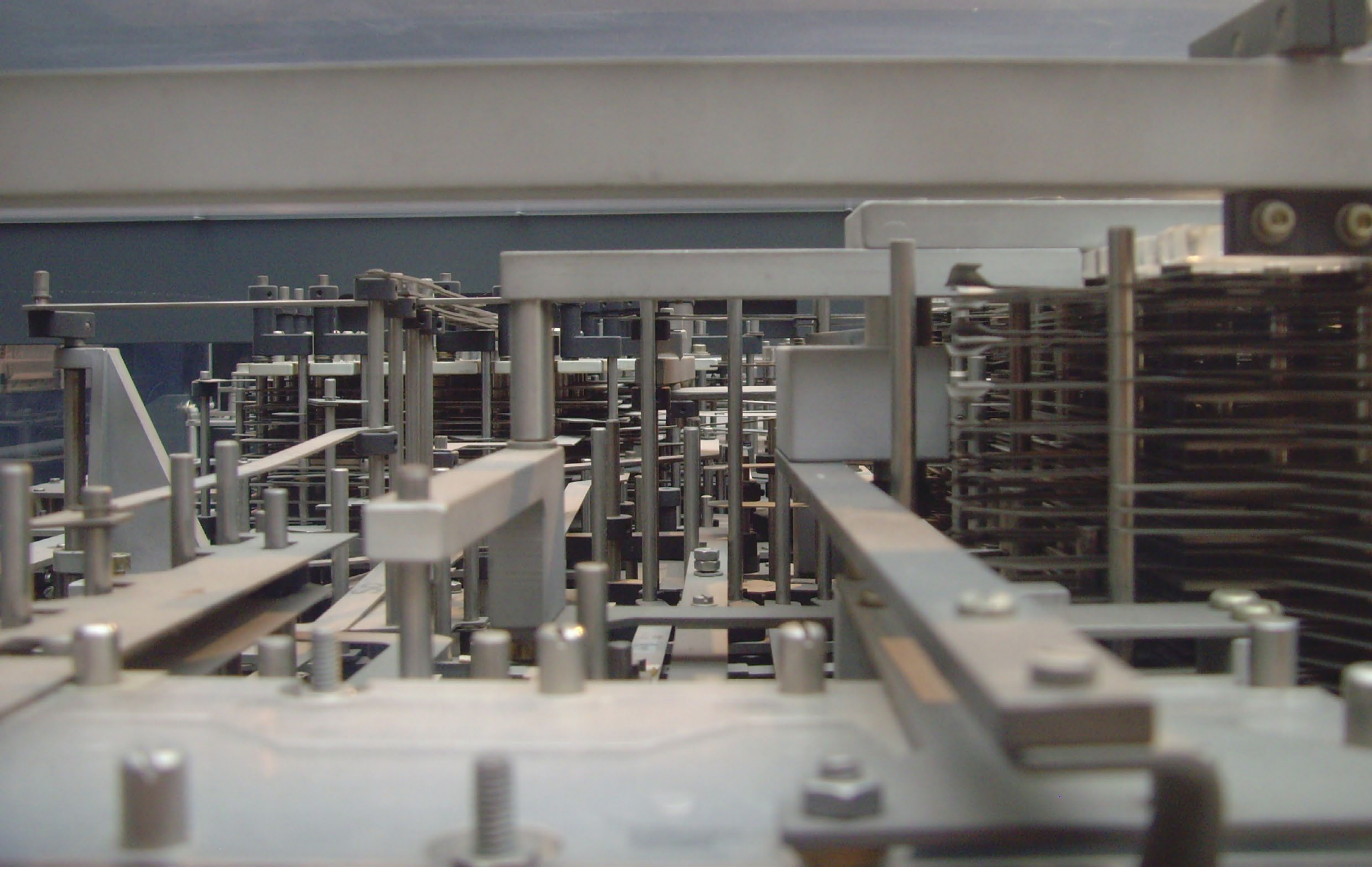
Zuse built the mechanism for the Z1 using 30,000 thin metal strips that he cut with a jigsaw. Combining strips of different sizes allowed him to build logic gates whose parts moved between two possible positions, depending on the positions of the other parts. Even the most complex binary logic is based on a few different logic gates, which means that, in principle, moving metal plates could be used to create a mechanical version of any computer.

Even the ancient Greeks were capable of constructing precision mechanics. The Antikythera mechanism that modelled the movements of the celestial bodies was based on thirty bronze gears, the largest of which had 223 teeth. The Greeks also built automatic theatres, self-opening doors, coin-operated machines, steam engines and other nearly magical devices. Hero of Alexandria, who lived during the first century, was particularly skilled at creating useless but highly innovative gadgets. It seems clear that the Greeks could have constructed the Z1 – provided that they had the blueprints. Designing the machine, however, would not have been possible without symbolic algebra and the binary system that did not arrive in Europe before the second millennium.

Gears or levers?

Binary logic built with metal strips is only one of many techniques that a time traveller can use in building their computer. Rotating gears and Leibniz wheels are also viable alternatives, since most mechanical calculators are based on them. Gottfried Leibniz developed the Leibniz wheel (stepped drum) in the 17th century for his own calculator.

The Analytical Engine, developed by the English mathematician Charles Babbage in the 1830s, would have used gears. If it had been completed, it would have been the world's first programmable computer. It could theoretically serve as a model for a time traveller's computer, as long as you did not copy its structure too exactly. The project suffered from megalomania:



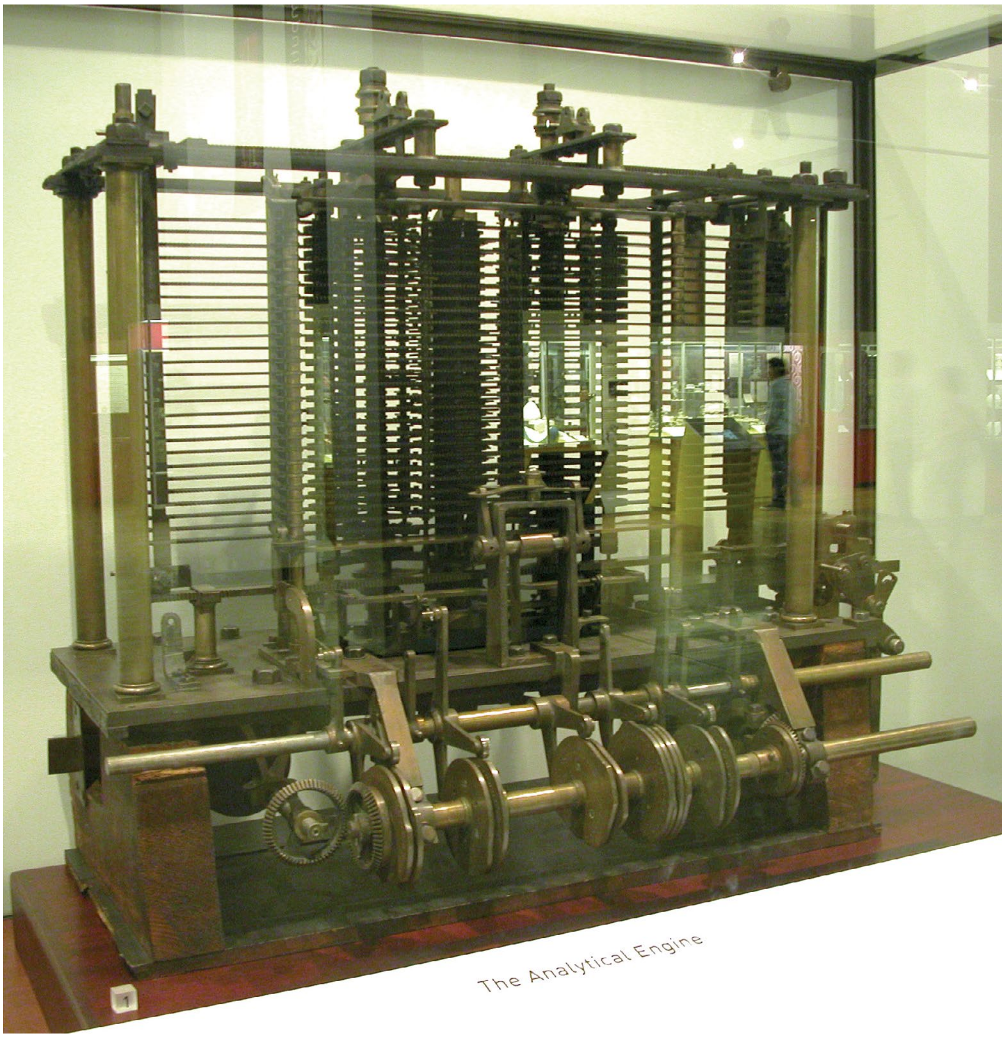
The rebuilt mechanism from the Z1. Deutsches Technikmuseum, Berlin.

the memory, for example, was supposed to contain one thousand 40-digit decimals, while Zuse made do with 64 memory locations.

Although the Analytical Engine was never completed, Babbage's humbler idea, the Difference Engine, was turned into working devices already in the 19th century. The Swede Per Geor Scheutz built a wooden prototype of his difference engine in 1843, and was later able to sell two metal versions. However, the difference engines are not programmable computers. They are only suited for producing function tables. Another predecessor of the computer from the same century was the punched card machine that was used in the US census in 1890.

Ropes, rolling marbles or pneumatics are also viable physical foundations for a computer. If the time traveller decides to invent electricity, they can build relays, electron tubes and maybe even semiconductor diodes and transistors. An optical computer, however, will most likely require a laser, so it is probably not a feasible endeavour.

Those who want to practise building a mechanical computer before travelling back in time can do so by using Lego bricks, for example. The Antikythera mechanism and Babbage's Difference



A small part of the Analytical Engine. Science Museum, London.

Engine have been built with Legos, and the British Lego hobbyist ”Random Wraith” has also used them to build logic gates. But no one has yet built an entire programmable computer.

From simple to complex

Regardless of the physical and theoretical foundation of your computer, you should always build it one abstraction layer at a time. During the first stage, you develop a sufficient selection of simple and reliable primary elements, such as logic gates. In the second stage, these primary elements are used to form more complex entities, such as memory and addition elements. By increasing the level of complexity one level at a time, we will finally arrive at a machine that can run a program, after which any higher abstraction levels can be implemented in software.

The simplest logic gate is the NOT gate that takes in one bit (0 or 1) and outputs the opposite bit (1 or 0). Figure 1 explains the operation of the NOT gate on the Zuse Z1. The input bit is determined by whether the topmost plate is in the upper or lower position. The position of the left plate indicates the output bit. The right plate is used for synchronisation, without which the gate will not operate. Mechanical parts require careful synchronisation in order to work properly, so you should not try to optimise the process by removing the sync bit.

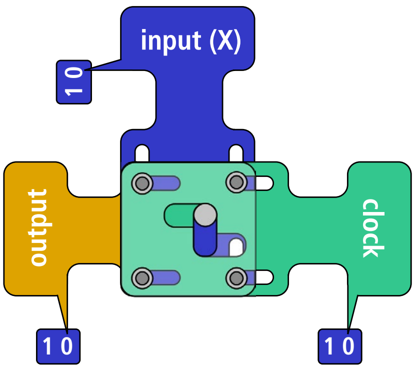


Figure 1: NOT gate on the Z1.

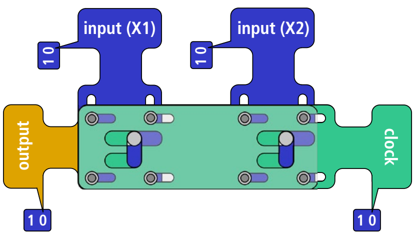


Figure 2: OR gate on the Z1.

Another gate used by Zuse was OR (Figure 2) that takes in two bits. It outputs zero if both input bits were zero, otherwise, it outputs one. In principle, binary logic can be built with only one gate type (NAND or NOR), but you can reduce the size by adding others. The XOR gate is very useful for addition, for example.

To add together two arbitrary binary numbers, we need an element known as an adder (Figure 3) that takes in three bits (A, B and C) and outputs their sum in two bits (D and E). The upper bit of the sum (D) can be routed to the input of another adder, which allows eight parallel adders to be chained together in order to form an addition circuit for two 8-bit numbers (Figure 4).

A long adder chain is not absolutely necessary, however. For example, the first Finnish computer ESKO performed additions one bit at a time, using a single adder. The smallest and slowest model of the PDP-8 minicomputer also did this. However, Zuse was not as frugal even with the Z1.

Instead of moving plates, you can also use rotating shafts and gears for the basic structure. This may even be a better option in some cases. According to Random Wraith’s observations, the preferred elementary operations for rotation-based logic are the sum, difference, halving and absolute value of the number of rotations. These analogue operations can be fairly simply used to build digital circuits – both logic gates and binary adders.

Memory and buffers

In addition to the calculation elements, a working computer requires storage space for the results. The Z1 had two registers for this purpose, the R1 and R2, and RAM memory with 64 locations. Calculations were always performed between two registers and the result was saved in either of them. There were dedicated instructions for handling memory. They copied data from the registers into the memory and back.

In the Z1’s RAM memory, each bit corresponded to a small metal strip whose position was altered. The strips were placed in a grid that was surrounded by the selector mechanism. The selector’s first three input bits selected one of the eight rows on a level and the following three bits selected one of the eight columns. This allowed reads and writes to be targeted at one strip at a time. The memory was constructed by using 22 of these bit levels that were operated simultaneously.

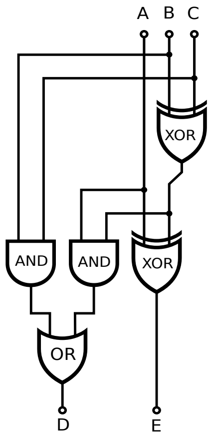


Figure 3: An adder that calculates the sum of two binary numbers.

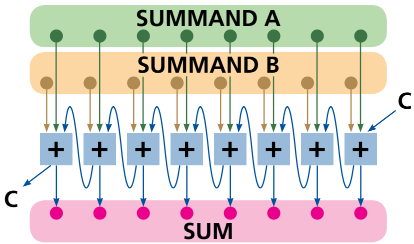


Figure 4: An adder chain that calculates the sum of two 8-bit numbers.

Modern computers run their software from RAM, but the Z1 read its instructions off of punched tape, which was a fairly popular means of storage in early computing. However, punched tape and punched cards have been used for controlling different mechanical devices since the 18th century. Barrel organs and music boxes also use a type of mechanical drum memory. It could be used in computing to save a microprogram that performs the computer’s actual instruction set.

Instruction set

The Z1’s instruction set included arithmetic instructions, memory handling instructions and number input and output commands. The problem was the lack of jump instructions. Loops had to be performed by either writing them out or by taping the beginning and end of the tape together. Therefore, you should not model the instruction set on your computer after the Z1, since there are more advanced options.

In principle, a general-purpose computer can be astonishingly simple. Many cellular automatons are Turing complete, for example. In theory, a device that winds a memory tape around endlessly and changes the state of each memory location based on the states of the previous cells could be considered a computer. In practice, and especially in mechanical form, such a machine would be extremely slow and laborious and would only provide theoretical pleasure to even the most hardened idealist.

Many hobbyists who have built computers from their elementary components have used the instruction set of the PDP-8 minicomputer due to its simplicity. This could make it a good starting point for a mechanical computer. Other ideals could be the Data General Nova, TMS1000, RCA 1802 and MOS 6502. Those who desire a higher level of elegance might develop a Forth-style stack-based instruction set.

Should I invent electricity first?

Mechanical computers have their limitations. Even if you could make one completely reliable, it will be hopelessly slow for many interesting tasks. The fastest electromechanical computers have reached approximately ten additions per second, which would also be the upper limit for fully mechanical ones. Even using electron tubes can increase the performance by a factor of several thousand.

Inventing electricity may also prove otherwise beneficial for a time traveller. Electricity allows for building a telegraph and a radio, which may give warmongering ancient rulers the upper hand against their neighbours. Once the time traveller has sold enough communications technology to be regarded with favour by the rulers, they will also be in an excellent position to start constructing a computer.

The first electrical device that a time traveller will want to build is the battery. Chances are that they are carrying a mobile phone, and its backlight alone will make it an object of immense magical power in the eyes of the people of ancient times. Charging it is recommended in order to prolong the benefits of the magic power. The electrolyte can be sauerkraut, rowan berries, citrus fruit or vinegar, for example. You also need two different metals for the anode and cathode. Recruiting a skilled blacksmith for building the conductors and connectors is recommended.

The skills required for constructing generators, transformers and relays have been around for centuries, but our understanding of physics was insufficient and did not allow for their invention before the 19th century. An electron tube can be built with traditional artisan skills as regards the glass, metal and insulation, but pulling a vacuum inside the tube may prove to be a problem. A crude form of a vacuum pump was already known in ancient times, but a time traveller who wants to build tube computers will most likely need to develop a better method for this. Those interested in semiconductors will most likely need to spend decades developing the necessary processes before the construction of the computer can begin.

How to justify my creation?

It appears that a time traveller with sufficient skill and good fortune could build a computer in the Middle Ages or even earlier. But how would the rest of the world view such a gadget?

The boundary between technology and magic was very unclear before the Enlightenment. In ancient Greece, complex machines were mostly seen in temples that competed with each other and required a steady supply of ”miracles”. In the 16th century, the Italian scientist Giambattista della Porta wrote in his book ”Magiae Naturalis” that real magic is based on natural sciences and has nothing to do with the supernatural. In other words, a time traveller should prepare for the fact that science and technology will be categorised as witchcraft by the majority of the population.

There is no point in trying to rationalise the importance of computing, either. After all, even most modern people do not understand the ideas of data processing, since computers are only collections of apps to them. The fact that the industry pioneers have struggled with finding approval for their ideas is also indicative of how difficult the concept is. After Leibniz built a working calculator, nobody resumed his work for a hundred years. Babbage's idea on the automation of brainwork was not understood, even though the automation of manual labour was already under way. Even in the 1970s, the directors of several computer companies did not believe that there would be a market for home computers.

In other words, the time traveller is not likely to encounter many people to whom they could explain the idea of a computer, even with great effort. To prevent the computer from becoming only a secret personal project, the time traveller should try to shape the culture in a more receptive direction. Becoming a philosopher might be a feasible solution. The formal systems could be injected with references to ”thinking machines” and data processing theory. If the era is very narrow-minded, however, the most radical thoughts should only be expressed at meetings with secret societies.

If the time traveller is, instead, thrown into the future, where humanity has suffered a major technological setback, this scenario has a major benefit. If people still exist, they will have at least some form of lore related to modern technology, which provides better prerequisites for understanding it. Therefore, a tribal warrior from the Neo Stone Age may well be more receptive to the idea of a computer than an ancient philosopher.

The history of computing is often told from an engineer's point of view: from mechanical parts to electron tubes to transistors and even denser microchips. However, our little thought exercise here shows that the development of culture has been at least as important. Humanity has had to undergo several changes in thinking before the idea of a computer could even come to pass. In the Middle Ages, a computer would have been completely anachronistic and incomprehensible – regardless of whether it had been made of wood or future components.

We can also reverse our scenario: if a time traveller from the far future arrived in our time, how would they view the possibilities offered by 2010s technology? Would they be able to use it to create something that is completely mind shattering, or would they prefer to advance, say, nanotechnology and quantum computing before building their magnificent invention?