20/20

# The Philosophy of Babbage's Analytical Engine


Michael D. Lore

[redacted]


March 14, 1989

Technology and Western Culture

(MECE 3301H 2667)

Lienhard

This is really good.

# The Philosophy of Babbage's Analytical Engine

Perhaps the most unfair element of the tides and churnings of history is the fact that many people are not credited for making great achievements until hundreds of years after their deaths. It must have been a great surprise when the "inventors" of the modern computer, perhaps by mishap, stumbled across the records of Charles Babbage and found that what they were "inventing" had, in concept, been invented nearly 150 years ago. The technological advancements of the past several decades have made the dream of Babbage a reality. Babbage had once remarked that he would "gladly give up the remainder of his life if he could be allowed to live three days 500 years hence, assisted by a scientific cicerone who could explain to him the discoveries made since his death" (Moseley, 1964, p. 258).

Today there are hundreds of variations on "computers," including advanced scientific calculators, portable computers, and mainframes with enormous speed. Generally speaking, however, they all work in the same fashion. One of the most important features of computers is the extremely versatile manner in which data can be operated upon. It is the organization of the computer that makes this possible. This organization is better termed the machine's <u>philosophy of operation</u>, for many varieties of mechanical or electrical hardware can produce a machine which operates with the same philosophy. (Of course, it is much easier to design a complex machine using electronics.) In choosing the word "philosophy," I am indicating that there is a level of design that is philosophical and far removed from gears, wheels, diodes, and transistors. The philosophy of the computer includes complex logic and structuring, and is undoubtedly the most important feature of the machine.

The calculating machines of Charles Babbage were designed solely with mechanical devices. According to Dubbey (1978), his work made severe demands upon the technology of his time. He designed and constructed many rather ingenious devices which were to implement the control and logic within his machines. His greatest contribution, however, was not in mechanics or the design of devices. Rather, it was his development of a philosophy by which a machine could be instructed to solve a limitless set of mathematically based problems. This same philosophy is the basis of modern computers. Also, the concept of feedback and Babbage's rationalistic outlook set the stage for part of his great contribution.

## A Non-Chronological Historical Background

Many writers have claimed that Charles Babbage was "a man born out of his time" (Moseley, 1964, p. 16). Indeed, his work in computing machines demonstrates a knowledge of such devices far exceeding that of his contemporaries. However, many of his ideas were conceptually typical of the late eighteenth century. Two of these are feedback and rationalism.

Feedback, in the general sense, is the directing of the output of some device back to its input. If the feedback is negative, meaning that it reduces the ~~total quantity of input when combined with the~~ original input, then it can be used to control the device. For a digital computer, we must consider a type of feedback which I will call *data feedback*. It cannot be considered negative or positive, and most likely does not fit a *strict* definition of feedback. However, it is a process by which a machine can examine a result which it has produced and then decide upon its next action based upon some aspect of the result. This type of feedback was included in the plans for Babbage's second calculating machine, the analytical engine. Historically, some of the most ingenious devices created owed their uniqueness to an application of feedback. Babbage called the feedback in his analytical engine "the Engine moving forward by eating its own tail" (Bernstein, 1964, p. 42).

Another theme from the eighteenth century that affected Babbage was rationalism. Rationalism is a belief that the logical application of reason will inevitably lead to the revelation of the self-evident truths of the universe. Babbage believed that the universe operated on "...great and transcendental laws...based on reason and logic" (Moseley, 1964, p. 257). By analyzing mathematics, Babbage formulated (or discovered) laws which were suitable for machine application. His first attempt resulted in the difference engine, which was designed to rapidly generate tables of sequences of numbers. Tables of logarithms, sines, and cosines were in high demand by mathematicians during Babbage's time. Dubbey (1978) explains the principle of operation of the difference engine in terms of a table of differences, shown in Table 1. The basic mathematical principle was that the successive differences in the numbers eventually becomes constant, as can be seen from the table. Because of this, a machine capable of successive additions can be set to calculate the numbers based upon the differences. This early example shows how Babbage applied his belief in rationalism to creating a real device, operating on one of the many laws of the universe, to do useful work.

The difference engine also demonstrates a simple

Table 1. A Table of Differences for Babbage's Difference Engine (Dubbey, 1978, p. 181).

| n | Sequence* | $1^{st}$ Difference | $2^{nd}$ Difference |
|---|---|---|---|
| 0 | 41 | 0 | 2 |
| 1 | 43 | 2 | 2 |
| 2 | 47 | 4 | 2 |
| 3 | 53 | 6 | 2 |
| 4 | 61 | 8 | 2 |

*The sequence is derived from $n^2 + n + 41$.

philosophy of operation. The philosophy is summed up in the table of differences. It is a strategy by which a machine could be made to find the desired answer. One can easily imagine that many different designs could be constructed to implement this particular philosophy of operation. Before we continue to examine the philosophy of operation of Babbage's more intelligent machine, the analytical engine, an introduction to the fundamental elements of a computer is in order.

## The Modern Computer

The microprocessor is the heart of the modern computer. Microprocessors are constructed on silicon wafers through many complicated processes which lay a circuit map upon the silicon. The individual microprocessors are then cut from the wafer and packaged in ceramic or plastic. All microprocessors have a similar philosophy of operation.

The type of philosophy used to design a digital circuit is called Boolean logic. Boolean logic is a mathematical way of expressing thought in terms of true or false. Operations are then defined to work on Boolean variables, which can take on only the values one (true) and zero (false). There are sixteen operations of two variables (Mano, 1984, p. 54). The simplest of these is the AND operation, which is true only if both of its arguments are true. For instance, a door might open if it is unlocked AND its handle is turned. This could be expressed as

Open = Unlocked AND Handle Turned.

By defining logic in a mathematical way, engineers can develop quite sophisticated works of machinery. When the work of George Boole, the inventor of Boolean logic, first came about, Babbage noted its importance by commenting in the margin of his copy of Boole's paper, "This is the work of a real <u>thinker</u>" (Hyman, 1982, p. 244). Boolean logic is one example of the philosophy of a computer that is so far removed from the actual machinery used to implement it.

The structure of a computer, however, better indicates what philosophy is actually required to make so versatile a machine. Figure 1 shows the general organization of a computer. The Central Processing Unit (CPU) is the center

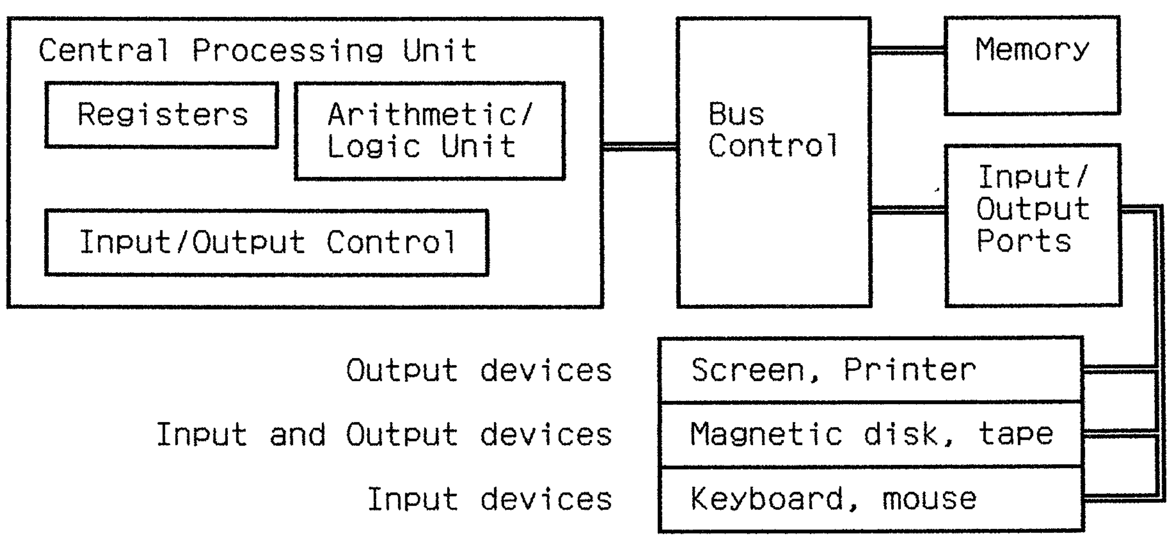


Figure 1. Computer Organization

through which all other parts are controlled. The Registers are memory locations inside the computer in which data may be physically acted upon. The Arithmetic/Logic Unit (ALU) is the Boolean logic that can actually act upon the Registers. The Input/Output Control is Boolean logic which controls the flow of data and instructions in and out of the CPU. The Bus Control is a section of logic that enforces the standards of data transfer between the CPU and the outside world. The Memory is just that, a place where data and instructions to be executed can be stored. Note that the CPU cannot in most cases act directly upon the memory. Data must be moved into the registers, acted upon, and then moved back into memory. The Input/Output Ports are memory locations through which external devices may communicate with the main portion of the computer. Although it is impossible to explain the exact operation of a computer without a few hundred more pages, it will later be shown how closely Babbage's analytical engine correlates with this structure and philosophy of operation.

The beauty of the computer is the method by which it can be controlled. By creating programs, users can make a computer perform an indefinite series of operations. We must remember, though, what those operations are. Every one of them is purely mathematical. The computer can add, subtract, compare, move data, etc. A popular misconception is that computers "think" much like a human. Fallacies like this are derived from the computer programs ~~which~~ that can write papers and play chess. Even so, all of the operations are mathematical.

A close friend of Babbage's, Ada Augusta Byron, Countess of Lovelace, foresaw this misconception:

> It is desirable to guard against the possibility of exaggerated ideas that might arise as to the powers of the Analytical Engine. In considering any new subject, there is frequently a tendency, first to _overrate_ what we find to be already interesting or remarkable; and, secondly, by a sort of natural reaction, to _undervalue_ the true state of the case, when we do discover that our notions have surpassed those that were really tenable.
>
> The Analytical Engine has no pretensions whatever to _originate_ anything. It can do whatever we _know how to order it_ to perform. It can _follow_ analysis; but it has no power of _anticipating_ any analytical relations or truths. Its province is to assist us in making _available_ what we are already acquainted with. (Bernstein, 1964, p. 45-6).

These lines are a wonderful summary of the operation of a modern computer. What makes Babbage's (and our) philosophy of operation of a computer so fascinating is the infinite versatility with which such a simple set of operations can process data. We now proceed to analyze the philosophical structure of Babbage's analytical engine.

## Babbage's Analytical Engine

Charles Babbage had already successfully designed and constructed a small version of his difference engine when he conceived the idea of a much more powerful machine, the analytical engine. Babbage borrowed Jacquard's idea of a loom programmed by punched cards to design a programmable calculating machine that "<u>weaves</u> <u>algebraic</u> <u>patterns</u> just as the Jacquard-loom weaves flowers and leaves" (Bernstein, 1964, p. 38-9). This idea led him to work out the details of the philosophy of such a machine.

Any comparison of modern computers with Babbage's machines is inherently dangerous; it is therefore important to first discuss what cannot be compared. To begin with, Babbage's machines are not binary machines with Boolean logic. Moseley (1964) makes the mistake of saying that the binary punched-card system invented by Jacquard "anticipated the zero and one, the on/off, the yes/no, the binary system, of the modern digital computer." Although it is binary as a form of input, its application to Babbage's machines led to no use of binary (or Boolean) logic. The reason for the modern use of Boolean logic is its simplification of complex logic through its simple algebraic form. Coincidentally, it also drastically simplifies circuitry.

Babbage's lack of Boolean logic in his machines causes problems for comparison. As seen from the discussion of the modern computer, the system of Boolean logic is part of the philosophy of the operation of a computer. Babbage's computers used no such logic. However, suffice it to say that if Babbage's analytical engine is comparable at all, it must have had some form of logical operation. Because Boole's work appeared very late in Babbage's career, it had little or no effect on the construction of his machines.

Thus, we may safely compare those aspects of Babbage's machine that relate to the organization and function of the machines. As will be seen, a system with Boolean logic must tackle some of the same problems as Babbage's system.

The similarities of Babbage's analytical engine and modern computers are in three major categories: architecture, design problems, and programmability. To begin with, the architecture of the analytical engine, according to Dubbey (1978), has five components. The first is the store, "in which all the variables to be operated upon, as well as all those quantities which had arisen from the result of other operations, would be placed" (Dubbey, p. 198). The store is directly equivalent to a computer's memory. The second component is the mill, in which the "grinding" of the computations was to take place. Babbage noted that the mill was the place where the "quantities about to be operated on would always be brought" (Dubbey, p. 198). In a modern computer, the mill would be called a register. Apparently, Babbage lumped what we call the arithmetic/logic unit (ALU) into the mill as well, but the separation of the location of the number and the machinery that operates on that location is obvious. The third component described by Dubbey is the control, which was the set of punched cards and mechanisms which would effectively control the sequence of operations performed by the computer. The controlling punched cards were called operation cards. The control is not quite as directly comparable to the modern computer, partially because modern computers can store their instructions in memory rather than read them only from a single source. However, the central processing unit (CPU) of the computer is effectively the "control" of the whole computer. Thus, although the terminology differs slightly, it appears that Babbage and I are talking about the same thing. The last two components of Babbage's engine are the <u>input</u> and <u>output</u>, which include the means of placing constants in the store, via <u>variable cards</u>, and the means of saving the results on paper or on new variable cards. The logical separation of the components of the analytical engine is very similar to the separation found in modern computers. Figure 2 shows the structure of Babbage's analytical engine. This organizational structure typifies the philosophy of

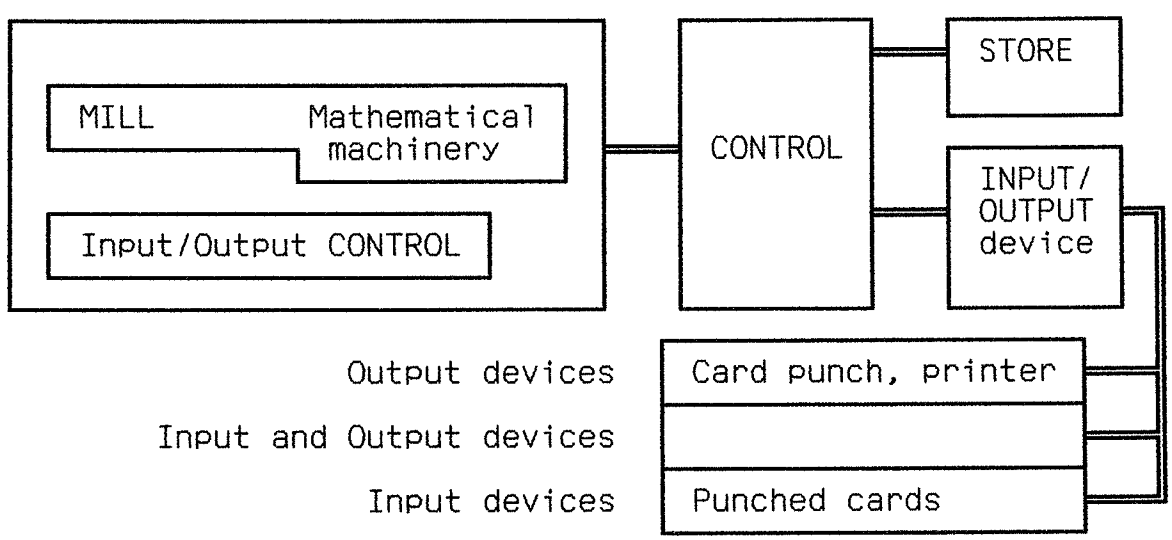


Figure 2. Analytical Engine Organization



It'd be nice if Figs. 1 & 2 were side by side.

operation of any computer or complex calculator.

Although the philosophy of operation may be architecturally sound, there were still several design problems to be faced by Babbage that are strikingly similar to the design problems faced by modern engineers. The first of these is the problem of carrying the tens in addition. Babbage's original design would add continually until all carriages were eliminated. Similarly, the adder logic used in modern computers had to run through several calculations until a stable final result was reached. Additions of this sort for critical numbers are shown in Figure 3. The problem with both methods in not accuracy, but speed. Babbage fought drastically against the time factor because his machine would hardly be worthwhile if it were no faster than a man. Both Babbage and modern designers contrived a very similar

| Addition | Babbage's | Modern (in binary) |
|---|---|---|
| 1 | 1309<br>+ 8701 | 01101111<br>+01001001 |
| 2 | = 9000<br>+ 1010 (carriage) | = 00100110<br>+10010010 (carriage) |
| 3 | = 0010<br>+10000 (carriage) | = 10110100<br>+00000100 (carriage) |
| 4 | = 10010 (result) | = 10110000<br>+00001000 (carriage)<br>= 10111000 (result) |

Figure 3. Sequences of additions required for Babbage's analytical engine and the modern adder circuit. The numbers added are not the same.

solution to the problem. Mechanisms and logic were added so that all of the carries generated could be simultaneously added in the machine. Babbage achieved this through a device using a chain (Dubbey, p. 203). Modern engineers implemented a logical "look-ahead carry generator." Other design problems included methods of handling zero and infinity, the problems of implementing and accessing a large store or memory, and the problem of managing memory (Dubbey, p. 203-5).

The most fascinating aspect of Babbage's analytical engine and our modern computer is, of course, programming. I have already mentioned the punched cards used to instruct the analytical engine. These cards, like instructions to modern computers, can be arranged in a programmed order to achieve a desired result. One of the more powerful program elements, however, is mathematical decision making. Both the analytical engine and the modern computer are capable of testing the result of some computation for its mathematical sign. If negative, the machine can decide to "jump" to a different set of instructions or punched cards (Dubbey, p. 205). This mathematical decision making facilitates looping, a powerful tool in computer programming.

The list of similarities between the analytical engine and modern computers is very long. Babbage devised a machine much like a modern computer. The basis of the machine is not gears and rods. It is an operational philosophy that can be repeated in a number of different types of technologies. The introduction of the modern computer came with the new technology that simplified the work required to construct a computing machine. Whereas Babbage was forced to think in terms of mechanical devices, modern engineers are free to think in terms of Boolean logic and functional architecture.

An engineer is now in the position to ask him- or herself what can be gained from Babbage and the analytical engine. The answer lies in the philosophy of Babbage's machine. It is a knowledge that a fundamentally intelligent machine must be thought of in terms of operational philosophy rather than only technological design. Babbage could, perhaps, have taken legitimate credit for his achievements if he could have popularized the philosophy, rather than the mystery, of the analytical engine.

I doubt he could have popularized it w/o a working model. In a way popularizing was what he put Ada Byron to doing. Even with a working engine, he was still -- as you say -- a man out of his time.

## References

Bernstein, J., 1964, The Analytical Engine: Computers--Past, Present and Future, Random House, New York.

Dubbey, J. M., 1978, The Mathematical Work of Charles Babbage, Cambridge University Press, Cambridge.

Hyman, A., 1982, Charles Babbage: Pioneer of the Computer, Princeton University Press, Princeton.

Mano, M. M., 1984, Digital Design, Prentice-Hall, New Jersey.

Moseley, M., 1964, Irascible Genius: A Life of Charles Babbage, Inventor, Hutchinson & Co., London.