# Boolean Algebra

From *Bit by Bit: An Illustrated History of Computers*
By Stan Augarten, Ticknor and Fields,1984

Boolean algebra, named after the English mathematician George Boole (1815-64), is a system of symbols and procedural rules for performing certain operations on numbers, letters, pictures, objects--whatever. (Leibniz inaugurated the search for such a system in his De Arte Combinatoria.)

While this form of algebra may seem forbiddingly abstract, it's really not much more complicated than ordinary arithmetic. For example, just as the appearance of a times sign (x) between two numbers calls for a multiplication, the appearance of a Boolean symbol between two numbers, letters, or statements, also calls for the performance of a specified operation.

There are many operations in Boolean algebra but the three most basic are called AND, OR, and NOT. They are binary in nature, able to process only two different kinds of entities, and they, along with other Boolean operations are often called gates, an apt metaphor for their functions. (Although Boole's system may be applied to any group of items, we'll confine our examples to binary numbers.) AND is a gate for 1s; if both numbers trying to slip through an AND gate are 1, AND requires the passing of a single 1 to the next gate up the road. But any other combination of digits (two 0s or a 1 and a 0) will yield a 0. OR is a less selective sieve for 1s; if either of the numbers at its gate is 1, then OR will pass on a 1. As for NOT, it acts as an inverter, transforming any 1s or 0s that come knocking on its door into their opposites (a 1 into a 0 and vice versa).

Although Boolean algebra contains other operations, AND, OR and NOT are all you--or a machine--need to add, subtract,multiply, divide and perform other logical processes, such as comparing numbers or symbols. Given the binary character of Boolean gates, it's a relatively easy matter to engineer a binary calculator's components into patterns that mimic AND, OR, and NOT. Of course, nothing could have been further from Boole's mind than the idea of incorporating his system into a machine; yet the invention of the computer owes almost as much to Boole, a self-taught mathematician who never went to college, as to anyone else.

In two epochal works, The Mathematical Analysis of Logic Being an Essay Towards a Calculus of Deductive Reasoning (1847) and An Investigation of the Laws of Thought (1854), Boole sought to identify the procedural rules of reasoning and to establish a rigorous system of logical analysis. Before the publication of these works, formal logic was a sleepy discipline with little to show for thousands of years of efforts. Its most powerful analytical tool was the syllogism, a form of deductive reasoning that proceeds from a major to a minor premise and then to a conclusion, as in "All men are mortal; all heroes are men; therefore all heroes are mortal" --not much to crow about.

One of the most important results of Boole's work was the demise of logic as a philosophical discipline and its rebirth as a vigorous branch of mathematics. Although most logicians criticized or ignored Boole's idea, they were absorbed by a growing number of mathematicians, who refined and amplified them and Boole was rewarded with a professorship at Queen's College, in Ireland. (Babbage, who knew a good idea when he saw one, wrote in the margin of his copy of The Mathematical Analysis of Logic, "This is the work of a real thinker.")

And then, in 1910, the British logicians Alfred North Whitehead and Bertrand Russell Published the first installment of their three-volume Principia Mathematica (1910-13), which transformed Boolean algebra into a formidable intellectual system known as "symbolic logic"... for the moment though, it's important to remember that the internal operations of computers are governed by Boolean algebra and that [Konrad] Zuse, in his uncanny instinct for the heart of the matter, was the first to incorporate these rules into a calculating machine.