PART 5

GOING DIGITAL

Transistors were quite a leap in the history of electronics, but another equally revolutionary development was the application of binary values to the on/off (well, really high- and really low- voltage) states, and the idea of representing questions in logic to electronic circuits. The following chapters explain how digital electronics work and how you can put them to work for you.

You will learn about integrated circuits, which are simply multiple circuits miniaturized and combined on a single chip. You will next learn how memory stores data as well as the instructions for running digital devices. You will be able to understand that writing a program is speaking to a machine in its own language.

Microcontrollers are amazing digital tools. They are small computers designed to work in embedded devices, including your own electronic projects. They have a streamlined construction and are easy to customize.

CHAPTER 14

DIGITAL THEORY

In This Chapter

- Using binary digits in electronics
- Distinguishing between analog and digital electronics
- Representing logical operations with truth tables
- Using logic gates to control circuits

The idea of linking the binary digits 1 and 0 to high-voltage and low-voltage levels, respectively, made possible revolutionary changes in how we use electronics. Great strides had already been made in electronics prior to this digital revolution, especially in the field of communications and the launch of radio and television. But once digital concepts were overlaid on electronics theory, the pace of change in technology accelerated to warp speed.

The Idea Behind Digital Concepts

In 1937, a Massachusetts Institute of Technology (MIT) graduate student named Claude Shannon wrote "A Symbolic Analysis of Relay and Switching Circuits," which has been called the one of the most important master's theses of the century. In the paper, Shannon proposed the use of Boolean algebra's two-position analysis in creating digital circuits. The world of electronics was from then on destined to change, even if it took a few decades for the idea to reach its full potential.

Titans of Electronics

Claude Shannon (1916–2001) is considered the father of digital circuits, but his impact goes much further. His work at Bell Labs and MIT with electronics, cryptography, and mathematics provided inspiration to generations of future engineers.

Shannon was quite a character, and often invented contraptions for pure enjoyment: a motorized pogo stick, a rocket-powered flying disk, and a mechanical mouse that could solve a maze. He would ride his unicycle through the halls of Bell Labs while juggling balls. He and a fellow MIT professor enjoyed taking their mathematical research on trips to Las Vegas to test drive some theories at the blackjack table and roulette wheel. Unfortunately, his final years were spent in the fog of Alzheimer's, but his lasting impact remains perfectly clear.

Analog vs. Digital

An analog signal is a continuous wave in both amplitude and in time. A digital signal is composed of individual, identifiable steps. A wave in the ocean is an analog signal; the tick-tick-tick of the second hand on a clock is digital. The following figure shows a sine wave in analog form and digital form. The digital wave approximates the analog wave by representing points along the waveform.

There are pros and cons to working with each signal type. Because an analog wave is a continuous wave, it has more information. However, because it has more information, it needs more processing and storage space.

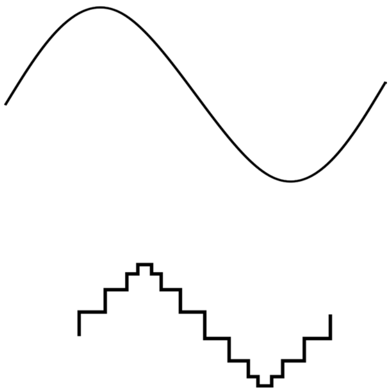


FIGURE 14.1 An analog wave (top) and its digital equivalent (bottom).

A digital signal can be much more compact than its analog counterpart, which means it requires fewer resources to transmit, receive, or store. A digital signal is also less likely to be affected by *noise.*

In electronics, **noise** is any unwanted interference with a signal. Noise sources can be natural, due to the natural interaction of electromagnetic fields, or man-made—from motors, fluorescent lights, radio or radar transmissions, wireless signals, and numerous other sources.

One of the primary benefits of digital waves for electronics is that each discrete bit of information can be expressed numerically using *Boolean algebra.* The states of "on" and "off" or "high voltage" and "low voltage" can be represented by 1 or 0, which in turn can be easily transmitted using switches, diodes, and transistors in circuits.

Boolean algebra is an approach to the study of numbers that is based on logic.

Logic is a branch of philosophy that was developed by the ancient Greeks, including Aristotle. It starts with the basic premise that an answer is either true or false. Aristotle proposed that there are laws of logic regarding a bi-valued reality in which statements are either true or false: X = X, X ≠ Y, not–X = Y, and X = not–Y. English mathematician George Boole expanded upon these classical views of logic and developed a system of algebra based on them.

Boolean numbers are not the same as real numbers. Instead, Boolean numbers represent a decision. They ask: Is it 1? If yes, 1. If no, 0. Each decision has the possibility of producing just two outcomes: 1 or 0.

Truth Tables

The most common questions asked about a relationship of numbers in Boolean logic can be expressed by the following logical operations:

- NOT: negation
- AND: conjunction
- OR: inclusion
- NOR: neither/nor
- NAND: not both

- XOR: exclusive
- XNOR: equality

To represent these concepts, we can use something called a *truth table*. One value, called an *operand*, goes across the top of the table horizontally and the second operand goes down the table vertically. For each question or logical operation, the various answers populate the box.

DEFINITION

An **operand** is a quantity that has a mathematical or logical operation performed on it.

A **truth table** is used in Boolean logic to give the results for the possible inputs and outputs.

The columns represent one side of the logical decision and the rows represent the other side of the logical decision. The result of each combination is depicted in the box where the column and row intersect.

In this table, $0 \neq 1$ is true (because 0 is NOT equal to 1), whereas $0 \neq 0$ is false. Let's now represent true with 1 and false with 0, as shown in the following figure.

This table does not represent multiplication, division, or any other real mathematical operation.

Instead, it the outputs of the question, "Is A not equal to B?" The answer can be true or false, and we are using the binary digits 0 and 1 to represent both the operands and the answers.

		A	
		0	1
B	0	F	T
	1	T	F

FIGURE 14.2 A truth table for NOT.

		A	
		0	1
B	0	0	1
	1	1	0

FIGURE 14.3 Truth table for NOT with binary substitution.

		A	
		0	1
B	0	0	0
	1	0	1

FIGURE 14.4 Truth table for AND.

		A	
		0	1
B	0	0	1
	1	1	1

FIGURE 14.5 Truth table for OR.

Truth tables can be created for each of the logical operations you just learned. Let's look at AND. The operation AND is true only if both operands are true—in other words, only when both operands are 1. Stated another way, the operand 0 represents false and the operand 1 represents true. So false AND false is false; true AND false is false; false AND true is false but true AND true is true.

The operation OR is false only if both operands are false—that is, only when both operands are 0.

The operation NOR is true only if both operands are false—only when both operands are 0.

The operation NAND is true only if at least one of its operands is false. One operand must be 0 and one must be 1.

Two common operations of Boolean logic that we use in electronics require two inputs: XOR or "exclusive-OR" and XNOR or "exclusive-NOR." These operations compare the two inputs and ask a question about them.

Here is the truth table for XOR. Looking at the two inputs, the output is true if one (and only one) of the inputs is true.

The XNOR gate is the inverse of XOR. For XNOR, looking at the two inputs, the output produces a value of true if and only if both operands are false or both operands are true.

		A	
		0	1
B	0	1	0
	1	0	0

FIGURE 14.6 Truth table for NOR.

		A	
		0	1
B	0	1	1
	1	1	0

FIGURE 14.7 Truth table for NAND.

		A = 0	A = 1
B	0	0	1
	1	1	0

FIGURE 14.8 Truth table for XOR.

		A = 0	A = 1
B	0	1	0
	1	0	1

FIGURE 14.9 Truth table for XNOR.

WATTAGE TO THE WISE

The field of logic takes some getting used to. It all makes sense—it is logic, after all—but only if you work deliberately through each statement. This is not a problem for electronic signals; they simply do what their circuit tells them to do. When working with logical statements, designers and programmers use tools such as truth tables to keep their ones and zeroes straight.

Binary Numbers

Binary numbers are distinct from Boolean numbers. Binary numbers use the same two digits as Boolean numbers, but binary numbers are real numbers that represent values beyond 0 and 1. (See Appendix C for details on working with binary numbers.) A binary system can represent numbers from the base 10 system (our familiar numbering system with digits from 0 to 9) with a string of binary zeroes and ones. Binary number representations of decimal numbers or binary-coded decimals (BCDs) can be sent as individual digits (in series) or as a group of *bits* (in parallel). The usual arrangement is in a *byte*, which is eight bits.

DEFINITION

A **bit** is a binary digit. Four bits are called a nibble. A set of eight bits is a **byte**.

Application to Electronics

So why do you need to know about binary numbers and Boolean operations? Because of the real-world application of these concepts to electronics. You've already learned that, using diodes and transistors, you can create circuits that can pass through no (or relatively low-) voltage or relatively higher-voltage signals.

If you go back to the idea of a computer as something that performs computations, you can see that creating a series of questions or logical operations could get you to a result. Of course, the number of operations required to perform even simple tasks would require lots of these operations and, therefore, lots of circuits.

Logic Gates

Each of the circuits that do these computations contains gates. In digital or logic circuits, the gates are called *logic gates*. Each gate gives one output but can have multiple inputs.

NAND gates and NOR gates are known as *universal gates* because given enough combinations, they can mimic the function of any other logic gate. A NOT gate is also called an *inverter* because it inverts one signal into the reverse.

WATTAGE TO THE WISE

When you begin designing your own circuits or doing your own programming, you need to think like a circuit or a computer. It is important to map out the processes step by step.

Designers and programmers often use flowcharts to represent the inputs and the decisions at each step. As you start out in electronics, get in the habit of planning things out. You might be going high-tech, but a pencil and paper can still be your most useful tools!

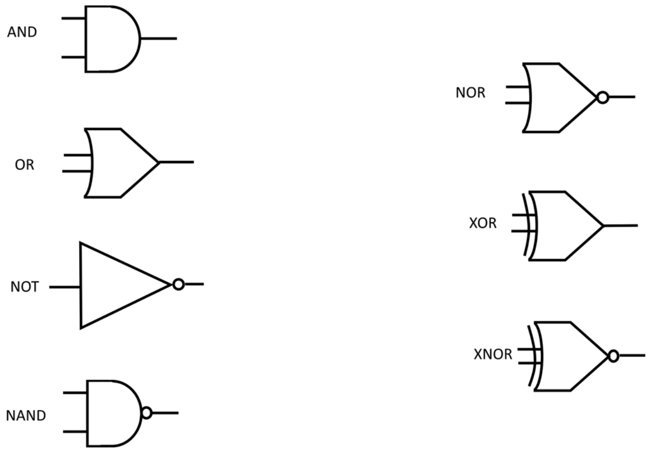


FIGURE 14.10 Symbols for various logic gates on a circuit board diagram.

Lab 14-1: AND Gate

To see how a digital logic gate works in action, let's construct an AND gate.

Materials:

1 9V battery

2 single-pole, single-throw (SPST) switches

Jumper wire

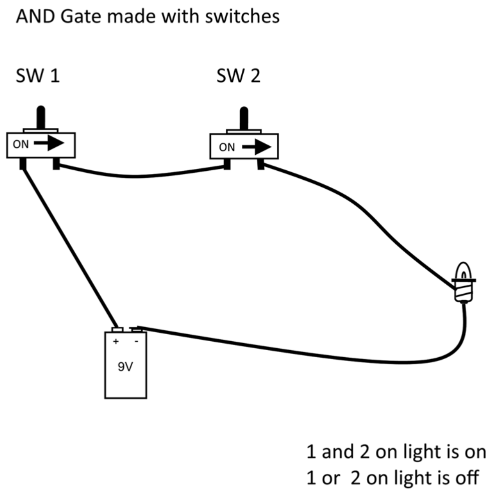
1 flashlight bulb

Instructions:

1. Connect the jumper wires as shown in the diagram to the two switches, the light bulb, and the battery.
2. Turn on SW1 to turn the light bulb ON.
3. Turn on SW2 to turn the light bulb ON.

The circuit is constructed so that if SW1 = ON AND SW2 = ON, then TRUE. The true answer is represented by high voltage so it turns the light ON.

4. Now let's generate a FALSE output by turning off SW1. This yields a FALSE (or low voltage) because with SW1 = OFF, it is not true that



LAB FIGURE 14.1. An AND gate.

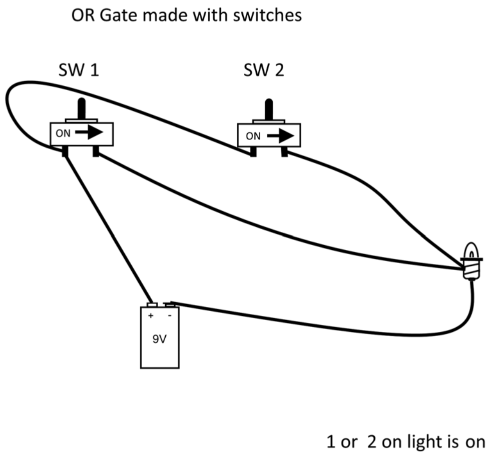
both are on. The result would be the same if both were turned off, as they both need to be ON to be TRUE.

Lab 14-2: OR Gate

Let's try another gate, the OR Gate. In this case, for the answer to be TRUE, either SW1 OR SW2 needs to be ON.

Use the same materials as in the preceding lab.

1. Connect the jumper wires as shown in the diagram to the two switches, the light bulb, and the battery.
2. Turn on SW1 to turn the lamp ON.
3. Turn off SW1 to turn the lamp OFF. The light bulb will light because if we ask the question SW1 = ON OR SW2 = OFF, the answer is TRUE.
4. Turn on SW2 to turn the lamp OFF.
5. Turn off SW2 to turn the lamp ON. Again, the light bulb will light because if we ask the question SW1 = OFF OR SW2 = ON, the answer is TRUE.



LAB FIGURE 14.2. An OR gate.

A Revolution Requiring Attention to Detail

The combination of Boolean algebra, the binary system, and the ability to create high- and low-value voltage values with electronic components is one of the most revolutionary concepts in human history. Because of this idea, and all of the steps required to invent the devices that make it all possible, we can use electrical signals to solve the most complicated calculations, to render digital images in timeframes almost too small to measure, to connect people from across the planet and into space. While it's easy to get overwhelmed with the potential of electronic decision-making, it's important that you understand that each thing you do with a computer requires many, many individual logical decisions being made—made one at a time, or simultaneously—but following a plan drawn out by the designer or programmer.

The Least You Need to Know

- Claude Shannon revolutionized electronics by combining binary concepts with voltage levels.
- An analog signal is a continuous wave; a digital signal is composed of individual, discrete steps.

- Boolean algebra is an expansion on classical logic theory composed of statements that are true in a bi-valued world. The answers to each logical question can be depicted in truth tables.
- Using Boolean logic, you can create electronic circuits that can perform logical operations.

Chapter Review Questions

1. The binary digits one and zero are linked to ______________ voltage for one and ______________ for zero.
2. True/False A sound wave can only be represented by a digital wave.
3. Two benefits of using digital signals mentioned in the text are: ______________ and ______________.

Give the name of the logical operation that express the relationship below:

4. Negation ________________
5. Inclusion ________________
6. Equality ________________
7. Not both ________________
8. Conjunction ______________
9. Equality ________________
10. Neither/ Nor ______________