Digital Electronics - Learning Unit 1

Introduction to Digital Electronics

Objective and Overview

You'll be introduced to specialized terms and concepts and review basic instrumentation and construction techniques. This lesson covers basic concepts key to working with digital electronics, with references for review, and provides a short glossary of terms. A selection of common conventions and construction methods are presented.

Student Preparation

None.

Using the Correct Terms

Digital electronics has a special language all its own. You already know that it deals with "1s and 0s", but that's a vast oversimplification of the in and outs of "going digital." It will help us communicate better if we agree on what important terms mean, so let's discuss a few.

Logic Levels

Digital electronics operates on the premise that all signals have two distinct levels. Depending on what types of devices are in use, the levels may be certain voltages or voltage ranges near the power supply level and ground. The meaning of those signal levels depends on the circuit design, so don't mix the logical meaning with the physical signal. Here are some common terms used in digital electronics:

- Logical–refers to a signal or device in terms of its meaning, such as "TRUE" or "FALSE"
- Physical–refers to a signal in terms of voltage or current or a device's physical characteristics
- HIGH–the signal level with the greater voltage
- LOW–the signal level with the lower voltage
- TRUE or 1–the signal level that results from logic conditions being met
- FALSE or 0–the signal level that results from logic conditions not being met

Be careful not to mix TRUE & HIGH or FALSE & LOW because they are not always the same! There are two types of digital electronics. Positive logic assigns ground to FALSE or 0 and a positive voltage to TRUE or 1. Negative logic does the opposite.

- Active High–a HIGH signal indicates that a logical condition is occurring
- Active Low–a LOW signal indicates that a logical condition is occurring
- Truth Table–a table showing the logical operation of a

device's outputs based on the device's inputs, such as the following table for an OR gate described in Lesson 2.

Inputs		Output
Input A	***Input B***	***X***
FALSE	FALSE	FALSE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
TRUE	TRUE	TRUE

• State–a stable combination of signals or of a set of inputs and outputs, can refer to a device, a circuit, or a set of signals

Logic Families

The two most common logic families are the 74HC and CD4000-series of devices. A logic family is a set of devices that all use the same internal technology, so their logic levels are all compatible. You will find the same logical functions, such as gates, counters, and registers, in all logic families, but their voltage and switching characteristics vary to satisfy different needs.

Both of these logic families utilize CMOS technology, which stands for "Complementary Metal Oxide Semiconductor", describing the types of transistors and the way they are connected to form logic gates. This course will rely on the CD4000 logic family of devices because they use positive logic (easier for beginners to understand) are inexpensive, draw little power, and are widely available. The 74HC family is a negative logic series. Both are widely used and have many devices with identical logic functions. We'll visit logic families in more detail in Lesson 13.

Number Systems

Digital logic may work with "1s and 0s", but it combines them into several different groupings that form different number systems. You're familiar with the decimal system, of course. That's a base-10 system in which each digit represents a power of ten. In the field of logic you'll also encounter…

- Binary–base two (each bit represents a power of two), digits are 0 and 1, numbers are denoted with a 'B' or 'b' at the end, such as 01001101B (77 in the decimal system)
- Hexadecimal or 'Hex'–base 16 (each digit represents a power of 16), digits are 0 through 9 plus A-B-C-D-E-F representing 10-15, numbers are denoted with '0x' at the beginning or 'h' at the end, such as 0x5A or 5Ah (90 in the decimal system) and require four binary bits each. A dollar sign preceding the number ($01BE) is sometimes used, as well.

- Binary-coded decimal or BCD–a four-bit number similar to hexadecimal, except that the decimal value of the number is limited to 0-9.
- Decimal–the usual number system. When used in combination with other numbering systems, decimal numbers are denoted with 'd' at the end, such as 23d.
- Octal–base eight (each digit represents a power of 8), digits are 0-7, and each requires three bits. Rarely used in modern designs.

As examples, 0x2A, 00101010B, and 42d all have the same value of 42.

Hexadecimal is useful because it is a compact way of dealing with binary numbers that have a multiple of four bits. Each hexadecimal digit can be represented by four binary bits. 0 is 0x0 in hexadecimal and 0000B in binary. 15 is 0xF in hexadecimal and 1111B in binary. Since most computers and microprocessors use 8, 16 or 32-bit words, hexadecimal notation is very common.

This course will deal with binary, decimal, and hexadecimal numbers.

As long as we're on the subject of numbers, there are some prefixes to review, as well. You are probably familiar with the metric prefixes k (kilo), M (Mega), and G (Giga) and their use to represent (loosely) some powers of two in the computer world. Because the binary numbers they're used to represent aren't equal to the exact values, they are gradually being replaced with IEC binary prefixes: The new prefixes are Ki = 2^{10} = 1024, Mi = 2^{20} = 1,048,576, and Gi = 2^{30} = 1,073,741,820. "bits" will be used to mean bits and "B" will be used to mean bytes. These are by no means common at present, so for this course we'll stick with the familiar (but slightly incorrect) k, M, and G. Be prepared to see them more frequently as the old prefixes are gradually phased out. More information on binary prefixes is available at http://physics.nist.gov/cuu/Units/binary.html.

Bit Ordering and Counting

Without some kind of agreement, it's easy to get into trouble when labeling bits and signals. In digital logic there are four different sizes or widths of numbers (see Figure 1-1).

- Bit–a single binary signal taking on the value 0 or 1, and representing a power of two in a binary number.
- Nibble–a group of four bits, containing a single hexadecimal number, such as 0x8
- Byte–a group of eight bits, containing two hexadecimal numbers, such as 0x3F
- Word–a group of bits larger than a byte, usually 16 bits, and almost always multiples of four bits

For the groups that have more than one bit, it is important to know "which way" to start reading the number. Fortunately, like for the decimal system, the least significant positions (meaning the lowest powers) are on the right as shown in Figure 1-1. The bit representing the smallest power of two (usually the zero-th power or 1) is called the

“Least Significant Bit” or LSB. On the other end of the binary number is the bit representing the highest power of two, called the “Most Significant Bit” or MSB.

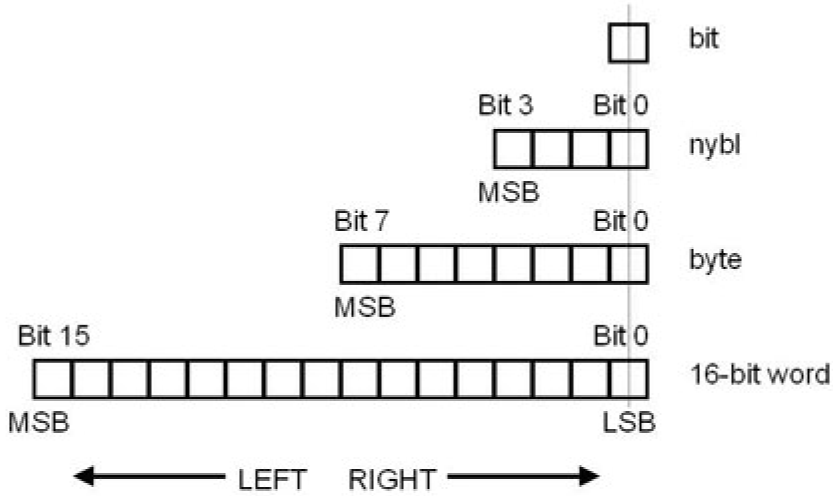

Figure 1-1 — Bit positions and names in the binary system

One of the more unexpected things about the binary system is that counting begins at zero, not one. The first of anything is labeled with a ‘0’, such as bit 0 or register 0, so a byte has bits 0 through 7. Designers also refer to “left” and “right”, meaning “towards the MSB” and “towards the LSB.” When you are just starting, it can be mighty confusing, so take your time to be sure you have it right. Even veterans sometimes make mistakes, particularly when two different groups are trying to design a complex device together.

Packaging

Logic devices are manufactured in standard sizes and organizations called packages. Package can refer to both the physical size and shape of the device or it can refer to the organization of logic elements within the physical device. The most common type of physical package is the DIP or Dual In-line Package, sporting two rows of pins along opposite rows of a rectangular body, 0.3” apart or more, with the pins spaced by 0.1”. These are also referred to as *through-hole* parts because to assemble them on a circuit board requires holes to be drilled through the board for the pins. Most products today use *surface-mount technology* (SMT) parts, whose pins lie flat against the circuit board and don’t require any holes. When ordering parts, be sure to get the right version, since there is great disparity in size. This course will use through-hole parts exclusively.

A logical package description usually refers to the number of individual logic devices grouped into a single physical device. Examples include quad gates, dual flip-flops, hex inverters, and octal drivers that include 4, 2, 6, and 8 individual logic circuits inside, respectively.

Digital Construction Techniques

Building digital circuits is somewhat easier than for analog circuits–there are fewer components and the devices tend to be in similarly sized packages. Connections are less susceptible to noise. The tradeoff is that there can be many connections, so it is easy to make a mistake and harder to find them. Due to the uniform packages, there are fewer visual clues. Here are some guidelines to help you make your first circuits.

Prototyping Boards

You'll be putting together some temporary circuits, or *prototypes*, as part of the exercises at the conclusion of each lesson using a common workbench accessory known as a prototyping board. A typical board is shown in Figure 1-2 with a DIP packaged IC plugged into the board across the center gap. The board consists of sets of sockets in rows that are connected together so that component leads can be plugged in and connected without soldering. The long rows of sockets on the outside edges of the board are also connected together and these are generally used for power supply and ground connections that are common to many components.

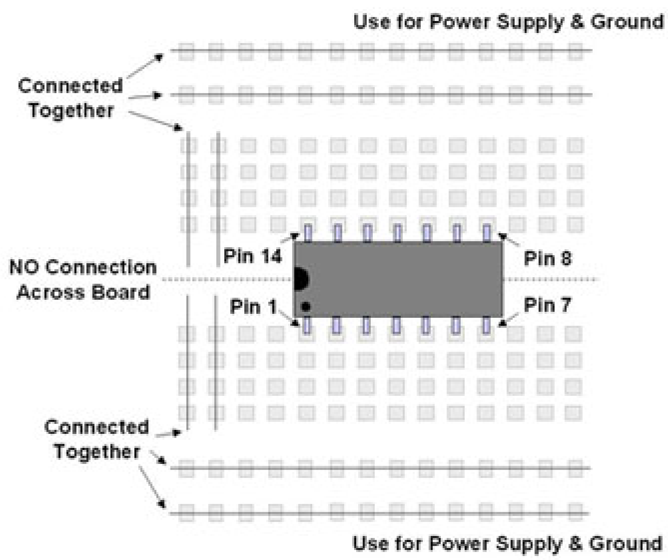

Figure 1-2 — Using a prototyping board

Try to be very systematic in assembling your wiring layout on the prototype board, laying out the components approximately as shown on the schematic diagram. Here are a few suggestions that will improve your chances of having a correct hookup on the first attempt:

- Positive and negative power supply voltages appearing at the top/bottom of the schematic should be on the top/bottom of your prototype board.
- Use wire jumpers that are about the right length to be about 1/2 ~ 3/4" above the board when they are installed. Jumpers that are too short tend to pull out of the sockets and if too long, snag on fingers and test probes and obscure the circuit.
- Remember that power supply grounds, function generator grounds, and all circuit grounds must be physically connected to be electrically identical.

- For complex circuits, use different colored wires for different functions and use masking tape or paper labels to label major signal paths or power leads.
- Finally, take your time and be careful. Make a paper copy of the circuit and use it as a guide as you build the circuit and for making notes. Use a highlighter to color each connection as it's made or component as it's installed.

Reading Pin Connections

IC pins are almost always arranged so that pin 1 is in a corner or by an identifying mark on the IC body and the sequence increases in a counter-clockwise sequence looking down on the IC or "chip" as shown in Figure 1-2. For most DIP packages, the identifying mark is a semi-circular depression in the middle of one end of the package or a round pit or dot in the corner marking pin 1. Both are shown in the figure, but only one is likely to be used on any given IC. When in doubt, the manufacturer of an IC will have a drawing on the data sheet and those can usually be found by entering "[part number] data sheet" into an Internet search engine.

Powering Digital Logic

Where analog electronics is usually somewhat flexible in its power requirements and tolerant of variations in power supply voltage, digital logic is not nearly so carefree. Whatever logic family you choose, you will need to regulate the power supply voltages to at least ±5 percent, with adequate filter capacitors to filter out sharp sags or spikes.

Logic devices depend on stable power supply voltages to provide references to the internal electronics that sense the high or low voltages and act on them as logic signals. If the power supply voltage is not well regulated or if the device's ground voltage is not kept close to 0 V, then the device can become confused and misinterpret the inputs, causing unexpected or temporary changes in signals known as *glitches*. These can be very hard to troubleshoot, so insuring that the power supply is clean is well worth the effort. A good technique is to connect a 10 ~ 100 μF electrolytic or tantalum capacitor and a 0.1 μF ceramic capacitor in parallel across the power supply connections on your prototyping board.

Review:

Logical–refers to a signal or device in terms of logical conditions
Physical–refers to a signal in terms of voltage or current or a device's physical characteristics

HIGH–the signal level with the greater voltage
LOW–the signal level with the lower voltage
TRUE or 1–the signal level that results from logic conditions being met
FALSE or 0–the signal level that results from logic conditions not being met

Active High–a HIGH signal indicates that a logical condition is occurring
Active Low–a LOW signal indicates that a logical condition is occurring

Truth Table–a table showing the logical operation of a device's outputs based on the device's inputs
State–a stable combination of signals or of a set of inputs and outputs, can refer to a device, a circuit, or a set of signals

A logic family is a set of devices that all use the same internal technology, so their logic levels are all compatible.

Binary–base two (each bit represents a power of two), digits are 0 and 1, numbers are denoted with a 'B' or 'b' at the end, such as 01001101B (77 in the decimal system).
Hexadecimal or 'Hex'–base 16 (each digit represents a power of 16), digits are 0 through 9 plus A-B-C-D-E-F representing 10-15, numbers are denoted with '0x' at the beginning or 'h' at the end, such as 0x5A or 5Ah (90 in the decimal system) and require four binary bits each. A dollar sign preceding the number ($01BE) is sometimes used, as well.
Binary-coded decimal or BCD–a four-bit number similar to hexadecimal, except that the decimal value of the number is limited to 0-9.

Bit–a single binary signal taking on the value 0 or 1, and representing a power of two in a binary number.
Nibble–a group of four bits, containing a single hexadecimal number, such as 0x8
Byte–a group of eight bits, containing two hexadecimal numbers, such as 0x3F
Word–a group of bits larger than a byte, usually in multiples of four bits

Counting in the binary system begins at zero, not one.

Now click on the Activities button and proceed with the Student Activities, which are required before moving to the Questions (click on the Questions button). Upon completion of these Questions, go to the next Learning Unit.