Laboratory Manual and Supplementary Notes:

FED 101 – Freshman Engineering Design Electrical and Computer Engineering Module

January 2005

John D. Carpinelli
Mohammed Feknous
Marek Sosnowski
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
University Heights
Newark, New Jersey, 07102-1982

Table of Contents

1   Introduction

1.1   Introduction to FED

FED 101 (Fundamentals of Engineering Design) is a required course for all NCE freshmen. It has different modules associated with different NCE departments and this manual is intended for the Electrical and Computer Engineering Module (ECEM). If you have declared that you wish to study electrical or computer engineering, you are assigned to this module. If you are still undecided, taking FED 101 - EEM may be a good way to find out what ECE is all about. If after taking this course you decide that ECE is not for you, so be it. While we hope that this will not happen, it is important to make a good decision about your professional future based on real information, not on myth.

FED 101 is very different than a typical college course focused on studying in depth a specific technical topic. FED was developed as a part on a national initiative aimed at improving engineering education carried out by Gateway Engineering Education Coalition and supported by the National Science Foundation. NJIT was among several top universities, which included Columbia, Cooper Union, Ohio State, Brooklyn Polytechnics and others, which took part in this effort.

This course covers a broad range of subjects and attempts to expose you to information, skills and experience needed for successful collage work and future professional activity. Among the prime course goals are:

- Inform you what electrical and computer engineers do and excite you about the field
- Enhance your ability to learn (learning to learn) and work effectively in peer groups
- Improve your communication skills
- Introduce you to basic electrical engineering concepts, devices and circuits as well as computer software for their simulation.

We hope that taking this course will be enriching and enjoyable experience. Good luck!

1.2   Introduction to ECE

ECE –Electrical and Computer Engineering is a broad field of theoretical and practical knowledge related to operation of devices and systems based electricity and magnetism. It would be impossible to find today any area of modern life that does not rely on extensive use of electrical power, electronic communication, or computers. Just imagine what life would be like without these technologies. That's why getting a degree in computer engineering or electrical engineering is so great. With the knowledge and experience you gain you can work in companies focusing not only on electronics or computers but also in those in communication, transportation, or aerospace industries, as well as in the areas of bioengineering and medical technology, and many others.

The ECE department at NJIT is an organization of faculty and staff devoted to providing an outstanding academic and research experience to students to prepare them to meet the needs and challenges they will face in the world of technology. Department has been recently reorganized into thrust areas to meet the challenges of future technology development, advancement of science and engineering, and industry needs. The Department has over 40

faculty members, over 1100 undergraduate and 450 graduate students with an annual research expenditure of over $6 million and growing.

The academic program in the ECE department is divided into two tracks: Electrical Engineering (EE) and Computer Engineering (CoE), with some overlap in the coursework.

Undergraduate students in the senior year of the EE program are able to select from seven specialized areas:

- **Communication Network:** The information revolution is built on an infrastructure of communications and computer networks. This infrastructure has the potential to drastically change the way we live and work. The communications networks track focuses on the analysis and design of wireless & wireline systems for information delivery. In addition to the systems elective, which emphasizes networks, a variety of courses are available in topics such as optical communications networks and wireless communications.
- **Telecommunications:** Wireless Telecommunications is rapidly becoming one of the most rapidly developing and exciting fields in today's technology. The Telecommunications track seeks to prepare individuals to contribute to the diverse fields of wireless communications. The systems elective focuses on wireless personal communications systems, while the electives cover a wide area of topics such as networks and optical communications.
- **Controls:** The mechanism of feedback pervades nature, science, and technology. The curriculum in control teaches how engineers can use the feedback mechanism to design systems for controlling a variety of dynamic processes, ranging from spacecraft, aircraft, and automobile emission systems to heating, ventilating, and air conditioning systems.
- **Computers:** The computer system elective provides an in-depth study of computer system organization and computer system design. Students study CPU design, control unit design, memory organization and I/O processing.
- **Power Systems:** The Power Systems elective includes the study of the economical generation and stable transmission of electrical energy to consumers.
- **RF/Microwave and Fiber Optics:** This area concentrates on radio frequency (rf) microwave and lightwave technologies at the component and the system levels. Applications include communication systems (rf/microwave and fiber optics), remote sensing, radar, sensors, etc.

Areas of specialization in the CoE program include:

- **Computer Networking**
- **Advanced Computer Architecture**
- **Telecommunications**

You are invited to visit various homepages of the ECE department for further information on our academic and research programs, faculty and staff and state-of-the-art facilities and resources to support high quality educational programs with life-long learning opportunities.

1.3 Design Exercise – Paper Drop Competition

In this exercise, you will play the role of the engineer. You are given a goal and must design a solution to achieve that goal.

1.3.1 Design Specification

Each team is required to design and construct a "flying" device. There are two design criteria for this device.

1. The device must stay in the air as long as possible.
2. The device must land as close as possible to a given target.

Each team must construct their device using any or all of the following materials.

- Three sheets of 8½" x 11" paper
- Adhesive tape
- One 3" x 5" index card
- Four paper clips
- A pair of scissors

1.3.2 Scoring

A design competition will be held among all teams in the class. One member of each team will launch the device from a predetermined height toward a target on the lower floor. The time will be recorded from when the device is launched until it hits the ground. Then the distance will be measured from the device to the target. Each team will perform two drop runs; the times and distances will be totaled for each team.

The scoring for this competition emphasizes flight time over accuracy. The length of time before reaching the ground comprises 70% of the overall score, and the distance from the target accounts for the other 30% of the score. The scores are scaled by the slowest and fastest times or closest and farthest distances. The formula for calculating the time portion of the score, a maximum of 70 points, is as follows.

$$\text{Time score} = \frac{(\text{Your team's time} - \text{Shortest team's time})}{(\text{Longest team's time} - \text{Shortest team's time})} \times 70$$

To illustrate how this works, consider three teams with total times of 4, 8, and 11 seconds. The formula becomes

$$\text{Time score} = \frac{(\text{Your team's time} - 4\,\text{seconds})}{(11\,\text{seconds} - 4\,\text{seconds})} \times 70$$

For the three teams, this is

$$\text{Time score} = \frac{(4\,\text{seconds} - 4\,\text{seconds})}{(11\,\text{seconds} - 4\,\text{seconds})} \times 70 = 0\,\text{points}$$

$$\text{Time score} = \frac{(8\,\text{seconds} - 4\,\text{seconds})}{(11\,\text{seconds} - 4\,\text{seconds})} \times 70 = 40\,\text{points}$$

$$\text{Time score} = \frac{(11\,\text{seconds} - 4\,\text{seconds})}{(11\,\text{seconds} - 4\,\text{seconds})} \times 70 = 70\,\text{points}$$

The longest time always earns 70 points and the shortest time receives no points. Other times earn varying numbers of points; the closer they are to the maximum time, the greater the number of points they earn.

The distance scores are calculated in a similar manner using the following formula.

$$\text{Distance score} = \frac{(\text{Longest team's distance} - \text{Your team's distance})}{(\text{Longest team's distance} - \text{Shortest team's distance})} \times 30$$

1.3.3 Acknowledgments

Thanks to Stephen Tricamo, Professor of Industrial and Manufacturing Engineering at NJIT, for allowing us to adapt this experiment from one he developed for his FED 101D class.

2 **Electricity**

This chapter introduces the physical phenomenon of electricity. First it examines the overall structure of the atom, the most fundamental basis for studying electricity. Next, it examines the basics of electricity – charge, current, and voltage. The two types of current electricity, direct current and alternating current, are introduced, along with differences between conductors and insulators.

Next, this chapter introduces the topic of resistance, and its relation to voltage and current. The color codes for resistors are presented, and circuits using resistors in series and in parallel are examined. Potentiometers, resisters whose values can be varied, and photoresistors, resistors whose values vary based on the amount of light they receive, are introduced. Laboratory Experiments 1 and 2 correspond to the material covered in this chapter.

2.1 *Electricity, Charge, and Current*

2.1.1 **Atomic Structure**

To understand current, it is necessary to first understand the basic structure of the atom. An atom is composed of three basic types of particles. The nucleus, or center of the atom, contains some number of *protons* and *neutrons*. The protons are positively charged particles, and the number of protons in an atom determines the type of atom. For example, all hydrogen atoms have exactly one proton, and all atoms with 13 protons are aluminum. Most atoms (except for most hydrogen atoms) have one or more neutrons in their nuclei as well. Neutrons have no charge, and the number of neutrons in an atom may vary. Atoms with the same number of protons but different numbers of neutrons are called *isotopes*.

Outside of the nucleus, negatively charged particles called *electrons* orbit the nucleus. Electrons are much smaller and lighter than protons. The attraction between the positively charged protons in the nucleus and the negatively charged electrons normally keeps the electrons in orbit around the nucleus. The number of electrons is the same as the number of protons, and the positive and negative charges cancel out; the atom has no net charge. A typical atom is shown schematically in Figure 2.1.

Figure 2.1: Basic atomic structure. Negative electrons orbit the positive nucleus at the center.

Sometimes an electron can escape from the orbit of its atom and be captured by another atom. The original atom now has one less electron and the atom has a net positive charge. The atom that captures the electron has one more electron than it has protons, resulting in a net negative charge. These are called *ions*, a general term that applies to all atoms with non-zero net charge, either positive or negative.

This last phenomenon, electrons moving between atoms, is the basis for electricity. Electricity is created by the flow of electrons. We'll examine this in the following subsections.

2.1.2 **Charge**

In the previous section we mentioned that electrons have a negative charge, but how much is this charge? To quantify charge, scientists have defined a *Coulomb* as the amount of charge contained by 6,250,000,000,000,000,000 (6.25×10^{18}) electrons. Although this sounds like a lot of electrons, keep in mind that one mole of hydrogen atoms, 6.022×10^{23} atoms, weighs only one gram, and the vast majority of that gram comes from the protons!

Charge is typically denoted as *Q*. Coulombs is abbreviated *C*; for example, a charge of 12 Coulombs is denoted as *Q* = 12*C*..

2.1.3 **Current**

Electric current is the flow of charged particles in a specific direction. In liquids and gases, these charged particles can be electrons or ions. In solids, such as the wire used in electrical circuits, electrons are the charged particles that cause electric current. Since the charge of a single electron is very small indeed ($q_e = 1.6 \times 10^{-19}$ C), any practical current involves flow of many electrons.

Current is typically denoted as *I* (for intensity). Current is the amount of charge flowing per unit time. This can be denoted by the equation:

$$I = \frac{Q}{t}$$

The same amount of charge flowing over a longer period of time would produce a smaller current, just as having a street where 20 cars pass through an intersection in one minute would be considered to have greater traffic than having the same 20 cars pass through the same intersection in an hour. The basic unit of current is the *Ampere*, or *Amp*, denoted as *A*. One Ampere is defined as the flow of one Coulomb of charge per second.

The following point is really important. *Although the electrons in a metal wire flow from the negative terminal to the positive terminal, the current flows from the positive terminal to the negative terminal*. The electrons carry a negative charge, and the current is defined by convention as the flow of positive charge. This is sort of like subtracting 1 and -1. 1 - (-1) is the same as 1 + (+1). The negative charge of the electrons flowing in one direction gives the same current as the positive charge flowing in the opposite direction. The convention of the current flow from positive to negative terminal was established before electrons were discovered and later people did not bother to change it.

2.1.4 **Direct Current and Alternating Current**

Two types of electric current are used in everyday life. *Direct current*, or *DC*, always flows in the same direction. This is the type of current created by batteries. The other type of current is *alternating current*, or *AC*. This is the current used to power household appliances and lights. This type of current periodically changes direction, once every 1/120 seconds in the United States (or once every 1/100 seconds in Europe).

2.1.5 **Voltage**

Voltage, also called the *electromotive force* or *potential difference*, is the force that causes current to flow. It can be helpful to visualize voltage as a difference in potential energy caused

by differences in charges. Consider a simple battery; it has positive and negative terminals. Chemicals inside the battery cause positive charges to congregate near the positive terminal and negative charges to collect near the negative terminal. If we connected a wire from one terminal to the other, electrons would flow from the negative terminal to the positive terminal, creating a current in the wire. Eventually the charges on each side of the battery will become more neutral, and the battery will die out.

Voltage is defined as energy per unit charge. The basic unit of voltage is the *Volt*, abbreviated as *V*. One volt is equal to one joule per Coulomb. The more volts a battery has, the more joules of energy it supplies per coulomb.

2.1.6 Conductors and Insulators

Some materials allow electrons to flow more freely than others. *Conductors* are materials that give up electrons easily, offering little opposition (resistance) to current flow. Copper is a very good conductor; that is why house wiring is usually made of copper.

Other materials, called *insulators*, do not yield electrons easily. They offer high resistance to current flow. They are not perfect; some electrons do flow in insulators. However, the amount is so small that, for all practical purposes, virtually no current flows. Insulators are useful for wrapping wires, causing all current to flow from one end of the wire to the other and not allowing current to escape from within the wire. This is why an extension cord that is plugged into a wall outlet can be handled safely, as long as there is no break in the insulation!

2.2 Resistors and Ohm's Law

Resistors are fundamental components in electric circuit design. As their name implies, they resist the flow of current in a circuit. The next several sections examine resistors, their color codes, and circuits that use resistors in series, in parallel, or in both configurations. Potentiometers and photoresistors, resistors whose values can be varied, are also described.

To examine the relationship between the voltage, current, and resistance in a circuit, we will start with the simple circuit shown in Figure 2.2.

Figure 2.2: A simple 1-resistor circuit

The circle on the left hand side of the figure is a power source. It has a voltage of 1.5V, the voltage level of a standard battery. The positive terminal of the battery (+) is connected via a wire, represented by straight lines, to one end of a resistor, which is denoted by the zigzag lines. The other end of the resistor is connected to the negative terminal of the battery (-) with wires, completing the circuit.

The value of the resistor is based on how well it resists the flow of electrons. A higher resistance allows fewer electrons to flow through the resistor in a given time, reducing the current. The basic unit of measure of resistance is the *Ohm*, denoted by Ω, the Greek letter Omega. One Ohm is defined as the value of the resistance that lets the current of 1 A to flow under a voltage of 1 V (one volt per ampere). The resistor in this circuit has a value of 100 Ω.

Ohm's Law defines the relationship between voltage, current, and resistance. It was developed by German physicist Georg Ohm, for whom both Ohm's Law and the unit of measure for resistance were named. It states that the voltage (*V*) in a circuit is equal to the product of the current (*I*) and the resistance (*R*), or

$$V = I \times R$$

Manipulating this equation, we can express the current or resistance as a function of the other terms in the equation as follows.

$$I = \frac{V}{R} \qquad \text{and} \qquad R = \frac{V}{I}$$

Now let's look back at the circuit in Figure 2.3. With a voltage of 1.5V and a resistance of 100 Ω, we can calculate its current as

$$I = \frac{V}{R} = \frac{1.5V}{100\Omega} = 0.015A = 15mA$$

The symbol *mA* stands for milliAmperes, or one-thousandths of an Ampere.

Worksheet 2.1 – Single-Resistor Circuit

For the circuit shown below, show the missing values for each set of given values.
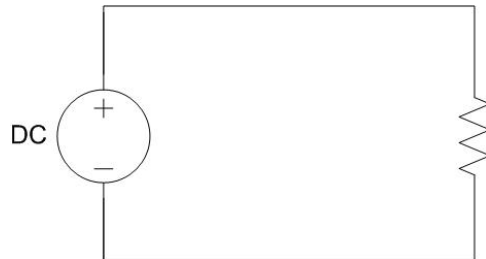
1. V = 10V   I = 0.5A   R = _____

2. V = 2V   I = _____ R = 400 Ω

3. V = _____   I = 0.1A   R = 1KΩ (=1000 Ω)

4. V = 25V   I = 0.125A   R = _____

5. V = 8V   I = _____ R = 640 Ω

6. V = _____   I = 1.2A   R = 120 Ω

9

2.3 Resistor Color Codes

If you look at a resistor, you won't see a number stamped on it indicating its value. Instead, it has several colored stripes that indicate its resistance. There may be three, four, or five stripes on a resistor. In this section we'll examine how these stripes represent the resistor's value.

Each stripe can be one of several colors. For the first two stripes (or the first three stripes for resistors with five stripes), the colors represent digits from zero to nine. The colors corresponding to each digit are shown in Table 2.1.

Table 2.1: Resistor colors and digit values for resistor magnitudes

Color	Digit
Black	0
Brown	1
Red	2
Orange	3
Yellow	4
Green	5
Blue	6
Violet	7
Grey	8
White	9

These digits give the base value of the resistor. For example, consider a resistor with four stripes with colors brown, black, yellow, and gold, respectively. The first color, red, represents the digit 1, and black corresponds to 0. Together they give a resistance value of 10.

However, the next stripe changes this value considerably. The value corresponding to yellow is 4, but this 4 is not added to the end of the 10 to create 104. Instead, it means that 10 is multiplied by 10^4, or 10,000; the actual resistance is 10×10^4, or 100,000 Ω, or 100 KΩ. Table 2.2 shows the exponent values for this stripe. Note that the values are the same as in Table 2.1, except two new colors are available. Gold, representing 10^{-1}, and silver, which corresponds to 10^{-2}, are used for very small resistances. A program that shows the values of a resistor based on its color code is available via the ECE Department laboratory web site.

Table 2.2: Resistor colors and values for resistor multipliers

Color	Multiplier
Black	10^0
Brown	10^1
Red	10^2
Orange	10^3
Yellow	10^4
Green	10^5
Blue	10^6
Violet	10^7
Grey	10^8
White	10^9
Gold	10^{-1}
Silver	10^{-2}

It is very unlikely that this resistor is exactly 100,000 Ω. Manufacturing processes aren't perfect, and the actual resistance may be greater than or less than its stated value. For the four and five stripe resistors, the last stripe indicates the *tolerance* of the resistor. The tolerance specifies the percentage that the actual resistance may vary from its marked value. Table 2.3 shows the more common tolerance values. For our 100 KΩ resistor, the gold stripe indicates a tolerance of ±5%, so our resistor may have any value from 100 KΩ – 5%, or 95 KΩ, up to a maximum of 100 KΩ + 5%, or 105 KΩ.

Table 2.3: Resistor colors and values for resistor tolerances

Color	Tolerance
Brown	±1%
Red	±2%
Gold	±5%
Silver	±10%

Resistors with tolerances of 5% or 10% have four stripes. Resistors with tolerances of 2% or less have five stripes. A resistor with only three stripes does not display a tolerance explicitly. By default, its tolerance is 20%.

Worksheet 2.2 – Resistor Color Codes

What are the resistances, and tolerances, of resistors with the following colored stripes?

1.   Green, Blue, Red

2.   White, Yellow, Green, Silver

3.   Brown, Orange, Violet, Yellow, Brown

Show the colors found on resistors with the following values and tolerances.

1.   1.2 KΩ ±20%

2.   8.8 Ω ±5%

3.   3.14 KΩ ±2%

2.4 Series Resistors

A typical circuit will have more than one resistor. Resistors in a circuit may be configured in series, in parallel, or in a combination of the two. This section examines resistors connected in series; parallel resistance is examined in the next section.

Resistors that are connected end-to-end are said to be connected in series. Figure 2.3 shows a circuit with two resistors connected in series. One hallmark of series resistance is that the same current that flows through one resistor must flow through the other resistor as well. There is only one path for the current to flow in this circuit.
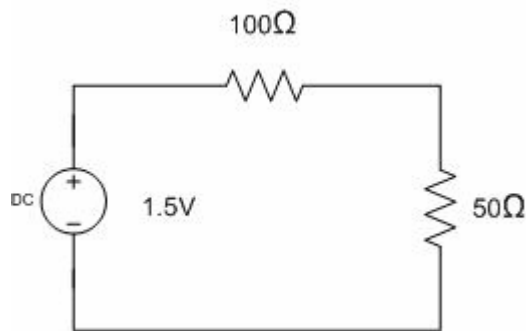
Figure 2.3: Circuit with two resistors in series

When two resistors are connected in series, their overall resistance is the sum of their individual resistances. For the circuit in Figure 2.3, the two series resistors have values of 100 Ω and 50 Ω; their overall resistance is 100 Ω + 50 Ω = 150 Ω. Using Ohm's Law, we can calculate the current in the circuit using this combined resistance.

$$I = \frac{V}{R} = \frac{1.5V}{150\Omega} = 0.01A = 10mA$$

2.5   Parallel Resistors

Resistors are not always connected in series; they can also be connected in parallel. Figure 2.4 shows a circuit with two resistors connected in parallel. Notice that both ends of the two resistors are connected together.
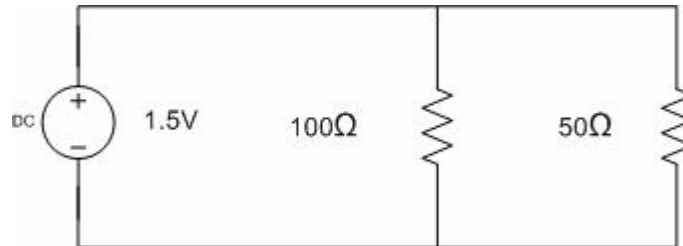
Figure 2.4: Circuit with two resistors in parallel

Although it might not seem to make sense, the overall resistance of two resistors connected in parallel is *less* than the resistance of either resistor! The basic reason this is true has to do with the current flow; adding another resistor in parallel increases the amount of current flowing in the circuit. From Ohm's Law, I = V/R; as current (I) increases and the voltage (V) remains the same, the overall resistance (R) must decrease.

Let's look at the circuit in Figure 2.4. Each resistor is connected directly to the positive and negative terminals of the battery, so each has a voltage of 1.5V. The 100 Ω resistor has a current of 1.5V/100 Ω = 15 mA, and the 50 Ω resistor has a current of 1.5V/50 Ω = 30 mA. Together the circuit has a current of 15mA + 30 mA = 45 mA. For the overall circuit, using Ohm's Law, we find R = 1.5V/45mA = 33.3 Ω.

A standard formula, called the *reciprocal formula,* is used to calculate the net resistance of two or more resistors in parallel. The reciprocal of the overall resistance is equal to the sum of the reciprocals of the individual resistors, or

$$\frac{1}{R_{OVERALL}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \cdots$$

This formula can be simplified for circuits with only two resistors. The formula becomes

$$R_{OVERALL} = \frac{R_1 R_2}{R_1 + R_2}$$

2.6   Series-Parallel Resistors

As their name implies, series-parallel circuits have resistors in series and in parallel. Figure 2.5 shows two series-parallel circuits.

(a)   (b)
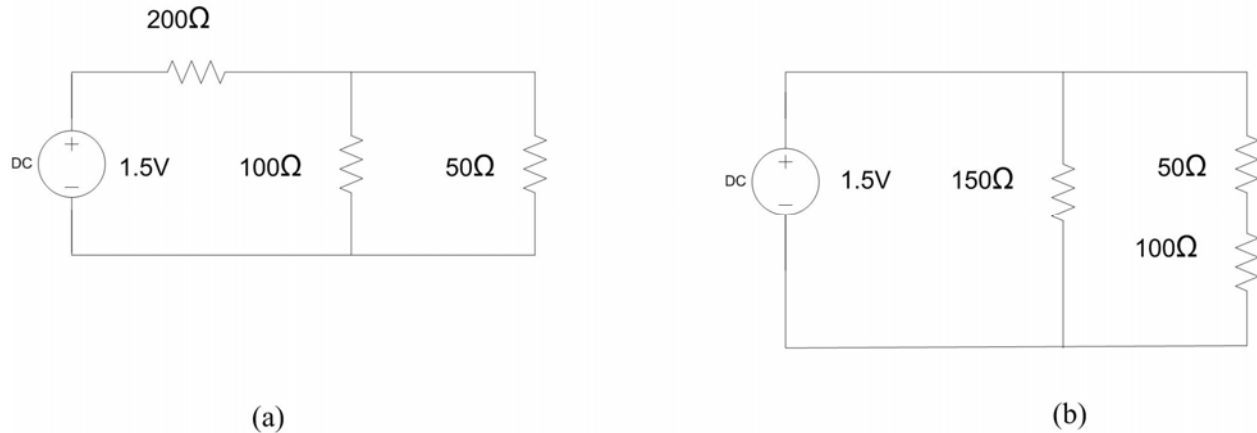
Figure 2.5: Two series-parallel circuits

We can analyze series-parallel circuits by breaking them into their individual series and parallel components. For the circuit in Figure 2.5(a), the 50 Ω and 100 Ω resistors are in parallel; their net resistance, calculated using the reciprocal formula, is 33.3 Ω. That equivalent resistance is in series with the 200 Ω resistor, producing a net resistance of 233.3 Ω in the circuit. This yields a current of 6.4mA.

For the circuit in Figure 2.5(b), we first combine the 50 Ω and 100 Ω series resistors, which results in a net resistance of 150 Ω. Combining this in parallel with the 150 Ω resistor yields a net resistance of 75 Ω, and a current of 20mA.

Worksheet 2.3 – Circuit Analysis

What is the net resistance and overall current for the following circuits?

1.

100Ω

1.5V

DC

200Ω

2.

1.5V

DC

100Ω   300Ω

3.

100Ω

1.5V

DC

150Ω   300Ω

4.

1.5V

DC

200Ω   100Ω

200Ω
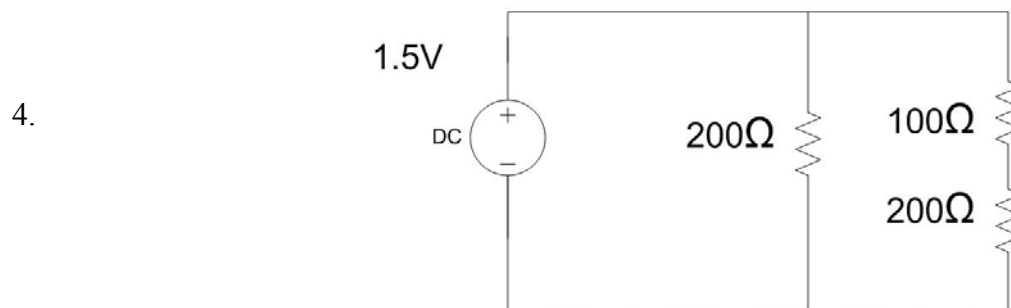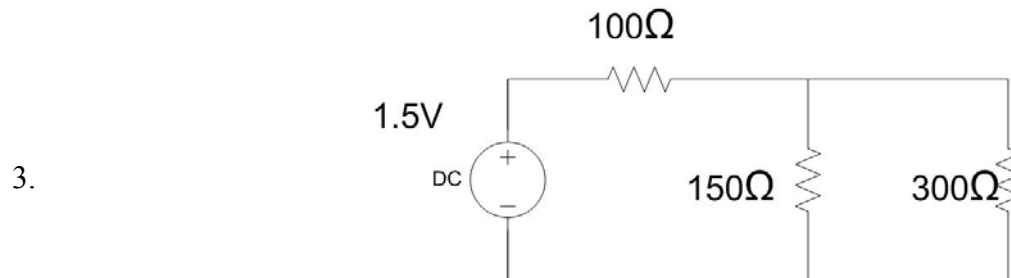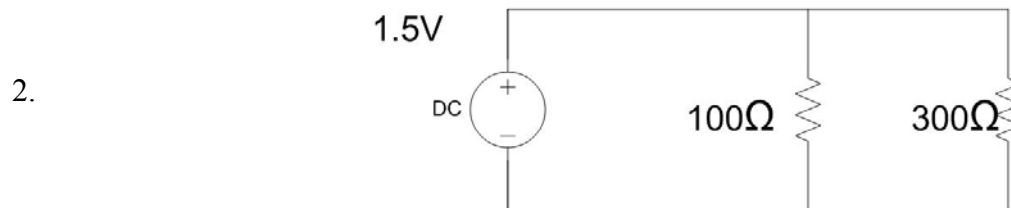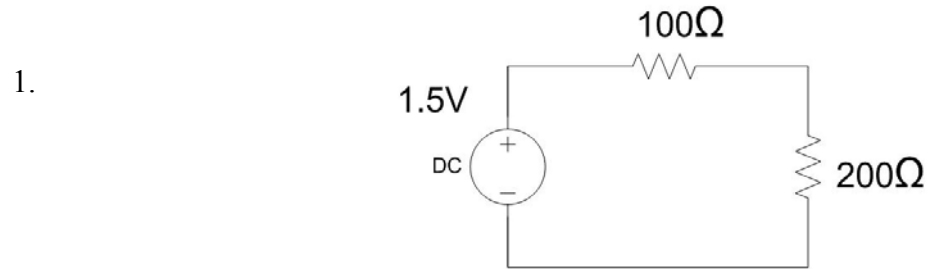
2.7   Potentiometers and Photoresistors

For some circuits, it would be preferable to allow a user to change the value of the resistance without having to re-wire the circuit. This is the role of the *potentiometer*. It is a variable resistor whose value can be changed, typically by turning a shaft or sliding a lever.

There are several applications for potentiometers. Dimmer switches used to vary the intensity of lights are typically potentiometers, or their close relatives *rheostats*. Volume control knobs on older radios and televisions are also potentiometers.

A photoresistor also changes its value, but the user does not directly change the value of the resistance. Instead, the resistance varies depending on the amount of light sensed by the photoresistor. The photoresistor has greater resistance in dim light and darkness. If the light is bright, the resistance decreases. The photoresistor can be used as a light sensor.

2.8   Kirchoff's Laws

Kirchoff's Laws are useful tools for analyzing circuits. There are two distinct laws, Kirchoff's Current Law (KCL) and Kirchoff's Voltage Law (KVL). This section examines both laws and how they can be used to analyze resistor circuits.

2.8.1 Kirchoff's Current Law

Kirchoff's Current Law can be summarized as follows.

Current entering a node = Current leaving a node

A node is a point in the circuit to which at least two elements, for example resistors, are connected. Intuitively this low makes sense if you consider that current flow is generated by electrons. The electrons flowing into a point in a circuit must come out somewhere; just like water flowing into a pipe must flow out of its other end or out of the pipe branches, if there are any.

To illustrate this point, consider the series resistor circuit shown in Figure 2.6. This is the same as the circuit of Figure 2.3, and we had previously calculated the current in this circuit to be 10 mA. For this circuit, this is the current in to and out of point A, as well as the current in to and out of point B.
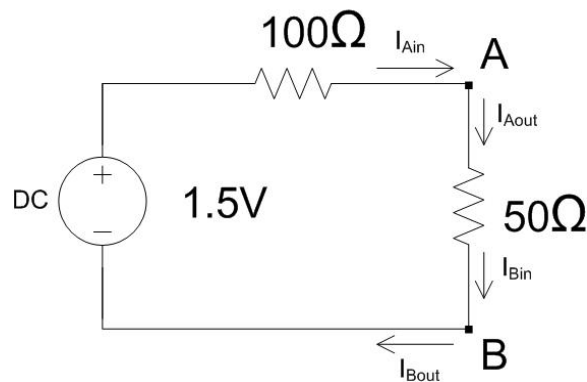
Figure 2.6: Applying Kirchoff's Current Law to a series resistor circuit

Although the KCL is valid for series resistor circuits, it isn't all that useful as an analysis tool. It is much more helpful for analyzing current flow in parallel resistors. For example, consider the circuit shown in Figure 2.7. This is the same circuit as shown in Figure 2.4, and we had calculated its overall current as 45 mA.
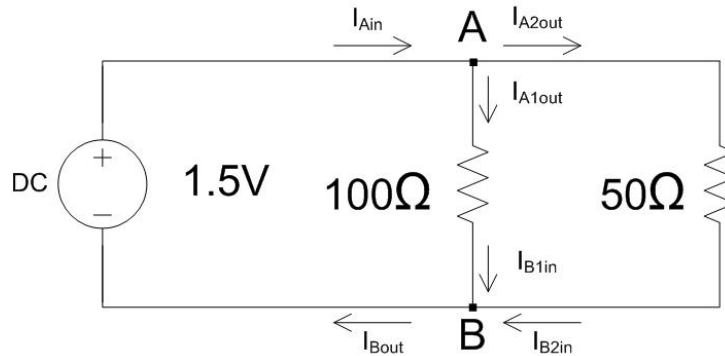
Figure 2.7: Applying Kirchoff's Current Law to a parallel resistor circuit

According to Kirchoff's Current Law, the current entering node A is equal to the current leaving the node, or

$$I_{Ain} = I_{A1out} + I_{A2out}$$

In Section 2.5 we calculated $I_{A1out} = 15$ mA and $I_{A2out} = 30$ mA, so

$$I_{Ain} = 45 \text{ mA} = I_{A1out} + I_{A2out}$$
$$= 15 \text{ mA} + 30 \text{ mA} = 45 \text{ mA}$$

Similarly for node B,

$$I_{Bin} = 45 \text{ mA} = I_{B1out} + I_{B2out}$$
$$= 15 \text{ mA} + 30 \text{ mA} = 45 \text{ mA}$$

An important question remains: How do we know how much current flows through each of the parallel resistors? In this example, why did the current split up as 15 mA and 30 mA instead of, say, 40 mA and 5 mA?

Clearly the current values are not selected randomly. In this case there is a straightforward explanation. *When two resistors are connected in parallel, they both have the same voltage drop.* For the circuit of Figure 2.8, the voltage drop across each resistor is 1.5V. Applying Ohm's Law gives us the following current values.

$$I_{A1out} = 1.5V/100\Omega = 15 \text{ mA}$$
$$I_{B1out} = 1.5V/50\Omega = 30 \text{ mA}$$

If two resistors are connected in parallel and one resistor has four times the resistance of the other, the larger resistor will have 1/4 the current flow of the other resistor.

2.8.2 Kirchoff's Voltage Law

The second of Kirchoff's Laws, Kirchoff's Voltage Law, is as follows.

The sum of all voltages in a loop is equal to zero

Before examining this law in detail, we first must define a loop. A loop is essentially a closed path within a circuit, consisting of part or all of the circuit. For example, Figure 2.8 shows a series resistor circuit and its one and only loop.
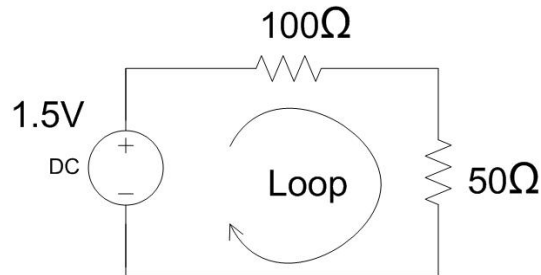
Figure 2.8: A series resistor circuit and its only loop

There are three voltages in this loop: the voltage from the 1.5V power supply and the voltages dropped across each of the two resistors. By convention, we show voltage values as positive or negative based on the direction of the current flow. Since current flows in the same direction as the loop for this circuit, the voltage across each resistor is positive. Since the loop passes from the negative to positive terminal of the power source, its value is negative. The voltage in this loop can be expressed as

$$+V_{100\Omega} + V_{50\Omega} - 1.5V = 0$$

By Ohm's Law, the voltage drop across each resistor is equal to its current multiplied by its resistance, $V = I \times R$. We had previously calculated the current flow through each resistor as 10 mA, so $V_{100\Omega} = 10 \text{ mA} \times 100\Omega = 1.0V$ and $V_{50\Omega} = 10 \text{ mA} \times 50\Omega = 0.5V$, or

$$V_{100\Omega} + V_{50\Omega} - 1.5V = 1.0V + 0.5V - 1.5V = 0$$

Circuits with resistors in parallel have more than one loop. As shown in Figure 2.9, a circuit with two parallel resistors actually has three loops. The sum of the voltages in each of the three loops must equal zero.
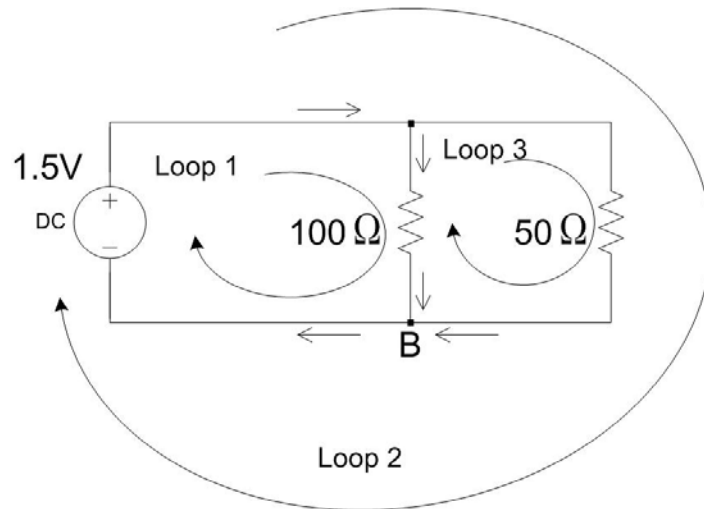
Figure 2.9: A parallel resistor circuit and its three loops

First let's look at Loop 1. It consists of the 1.5V power supply and the 100Ω resistor. We had previously calculated the current for this resistor as 15 mA, so its voltage, calculated using Ohm's Law, is 15 mA × 100Ω = 1.5V. The voltages in this loop are

$$(15 \text{ mA} \times 100\Omega) - 1.5\text{V} = 1.5\text{V} - 1.5\text{V} = 0$$

The second loop consists of the 1.5V power supply and the 50Ω resistor. Since this resistor has a current of 30 mA, the voltage equation for this loop is

$$(30 \text{ mA} \times 50\Omega) - 1.5\text{V} = 1.5\text{V} - 1.5\text{V} = 0$$

The third loop, consisting of the two resistors, might appear to fail under Kirchoff's Voltage Law since all voltage drops across resistors so far have been positive. If this is true for this loop, we would be adding two positive values and could not obtain a zero result. However, this is not the case here. Note the direction of the arrow for this loop. For Loop 1, the flow of the loop for the 100Ω resistor goes in the opposite direction of the current flow. For this loop, the voltage across this resistor is treated as a negative value, and the loop equation becomes

$$-(15 \text{ mA} \times 100\Omega) + (30 \text{ mA} \times 50\Omega) = -1.5\text{V} + 1.5\text{V} = 0$$

3   Simulation Tools

Simulation software packages have been developed to assist engineers and scientists in solving design problems using a computer. These packages rely on mathematical models of circuits (PSpice from Multisim, Electronics Workbench from Multisim to name two of the most popular ones), or on the usage of a programming language (Matlab from MathWorks, MathCad from MathSoft) to facilitate the resolution of mathematical problems in circuit analysis. The two packages that you will be exposed to in a very limited fashion will be PSpice and Matlab. The students will be shown a limited set of the features of these software packages to have an idea of their capabilities. Both packages will be utilized more extensively in the future, in the various courses that require the assistance of these software tools and the students will see how powerful these packages can be in supporting the design of engineering systems, most notably in electrical and computer engineering.

PSpice is available free of charge as a very limited version of the professional package. It can be downloaded from The ORCAD website http://www.orcad.com/downloads/form/cdrequest.asp, which is the parent company of Microsim. Matlab is available free of charge as part of the software package distributed to the students when they join NJIT. A copy of the CD may be obtained from the computer store for a nominal fee. Additional information may be obtained from www.mathworks.com.

3.1   Circuit Examples

We will initially analyze on a sheet of paper a circuit that is purely resistive (made of energy sources and resistors only). The students are advised not to be scared by the complexities introduced. The material will become familiar in subsequent courses. The purpose of these choices is to show you how powerful and useful the software packages are. Complex circuits and alternating current problems will not be included in quizzes. You will however easily master these cases and much more complex ones by next year.

Note that the circuit shown in Figure 3.1 introduces a new device called a current source. An ideal current source will deliver the same current regardless of what and how many components are connected to it, in this case a constant 1 Amp.
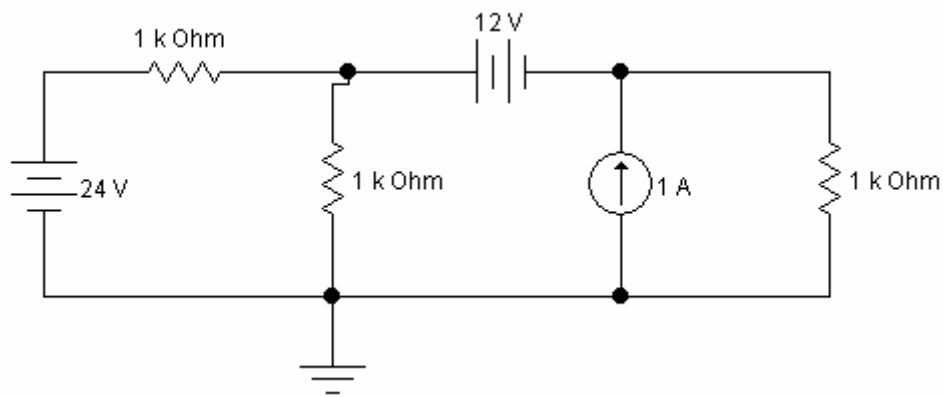
Figure 3.1: Complex Resistive Circuit

We are going to use mesh analysis (it will be covered more extensively in ECE 231) to obtain some of the currents and voltages in the circuit.

Mesh 1: $\qquad - 24 + 10^3 I_1 + 10^3 (I_1 - I_2) = 0$
Supermesh: $\qquad 10^3 (I_2 - I_1) + 12 + 10^3 I_3 = 0$
Current source: $\qquad\qquad I_3 - I_2 = 1$

Rearranging these three equations, we obtain

$$2 \times 10^3 I_1 \quad - 10^3 I_2 \qquad\qquad = 24$$
$$10^3 \ I_1 \quad - 10^3 I_2 \quad - 10^3 I_3 \quad = 12$$
$$- I_2 \quad + \quad I_3 \quad = \ 1$$

Using Cramer's rule (Cramer's rule is a simple mathematical method to easily obtain solutions from algebraic equations),

$$\text{Let} \quad \mathbf{A} = \begin{vmatrix} 2 \times 10^3 & -10^3 & 0 \\ 10^3 & -10^3 & -10^3 \\ 0 & -1 & 1 \end{vmatrix}$$

$$\mathbf{A_1} = \begin{vmatrix} 24 & -10^3 & 0 \\ 12 & -10^3 & -10^3 \\ 1 & -1 & 1 \end{vmatrix}$$

$$\mathbf{A_2} = \begin{vmatrix} 2 \times 10^3 & 24 & 0 \\ 10^3 & 12 & -10^3 \\ 0 & 1 & 1 \end{vmatrix}$$

$$\mathbf{A_3} = \begin{vmatrix} 2 \times 10^3 & -10^3 & 24 \\ 10^3 & -10^3 & 12 \\ 0 & -1 & 1 \end{vmatrix}$$

where we see that $\mathbf{A_1}$, $\mathbf{A_2}$, and $\mathbf{A_3}$ are replica of $\mathbf{A}$ except that the first column is replaced by the right side of the previously established equations (24, 12, 1) for $\mathbf{A_1}$, and so on for the second column for $\mathbf{A_2}$, and the third column for $\mathbf{A_3}$.
The solutions are obtained as:

$$I_1 = \mathbf{A_1} / \mathbf{A} \qquad\qquad I_2 = \mathbf{A_2} / \mathbf{A} \qquad\qquad I_3 = \mathbf{A_3} / \mathbf{A}$$

These operations yield:

$$I_1 = -0.321\ A \qquad I_2 = -0.667\ A \qquad I_3 = 0.333\ A$$

If one were interested in the voltage across the most-right 1 kΩ resistor, then

$$V_3 = 10^3\ I_3 = 333.3\ V$$

We see that any voltage or current can be obtained in terms of the three mesh currents.

3.2 PSpice

PSpice is a PC version of the Spice program (**S**imulated **P**rogram with **I**ntegrated **C**ircuit **E**mphasis). It will be used to simulate an electrical system, and obtain values or graphical plots of some of the responses (entities that we may be interested in such as currents or voltages through time, or frequency domain characteristics).

Much of this tutorial on PSpice will be presented through examples from which the students can extract patterns that they can apply to other cases, or be able to extend their learning capabilities to other features of these packages. The Windows operating system and many other packages before it have shown the appeal that a graphical interface has on the user. PSpice had to go that route to compete with other software packages that understood the need to develop user-friendly software packages.

3.2.1 Getting started

From **Start, All Programs, Pspice Students, Schematics**, have the PSpice program running in the **Schematics** mode.

3.2.2 Simple resistive circuits

We will first revisit the two circuits from chapter 2, shown in Figures 2.4 and 2.5.

3.2.2.1   Series Circuit
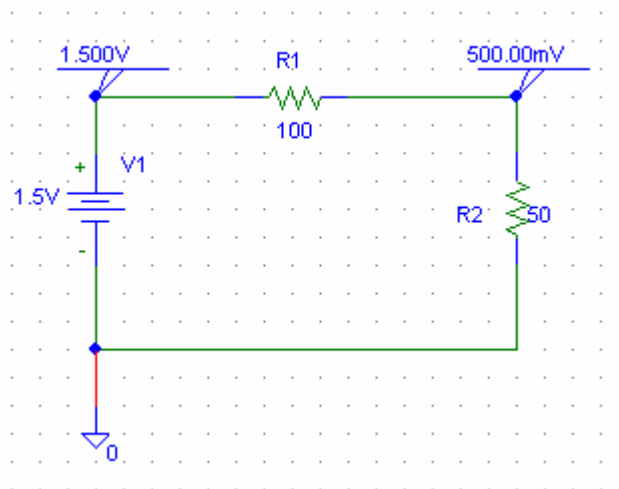
Figure 3.2: Series Resistive Circuit

To draw the circuit shown in Figure 3.2, you can either go to the menu **Draw, Get New Part** (equivalent to **Control-G**), or use the button **Get New Part** ⬚, you can either type in the name of the part that is needed, or obtain from the long list, or open the libraries which will give you a categorized listing of the components. The components are VDC for the dc voltage sources or batteries, AGND for the ground, R for resistors,. You can use the button **Draw Wire** ⬚ to connect the various components. You can also use **Control-R** to rotate the components when highlighted (use the left button of the mouse) and if needed. You can also add from the parts library Viewpoint to display the voltage at any given node referenced to the ground node. The ground node may not, as in this case, be part of the original circuit. However PSpice needs a reference node labeled as a ground node. All the voltages are referenced to it. You can see from the simulation that the voltage drop across the 50 Ω resistor is as found earlier 0.5 V or 500 mV. The voltage across the 100 Ω resistor is obtained as the difference between the two terminal-potential, $1.5 - 0.5 = 1$ V

It is also possible to simulate current measurements through the addition of a new part **IPROBE**. Since the IPROBE is equivalent to an ammeter (device that measures currents), it is imperative to break the connections so that the device can be inserted in series with whichever component whose current we are trying to evaluate. Deleting an element can be done by highlighting the element (left button of the mouse), and pressing the delete key from the keyboard.
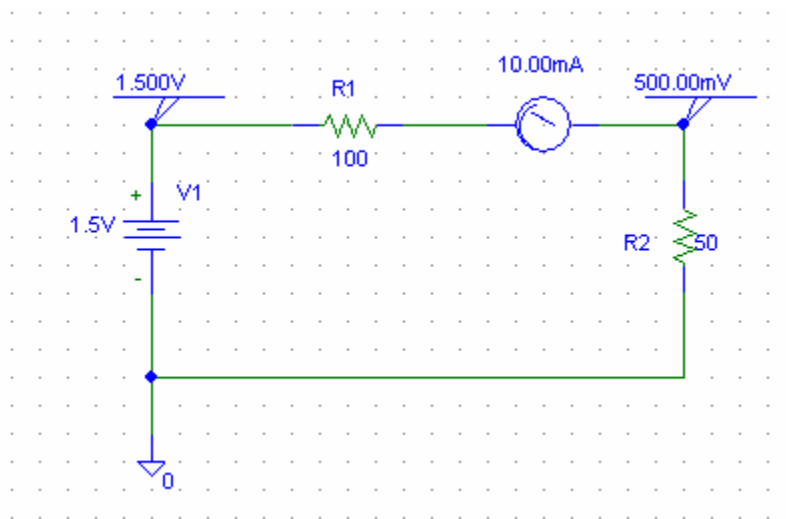
Figure 3.3: Series Resistive Circuit with Current Measurement

We see that the current is as predicted through calculations 10 mA.

3.2.2.2   Parallel Circuit

Similarly for the circuit introduced in chapter 2 (Figure 2.5), PSpice provides the desired responses.
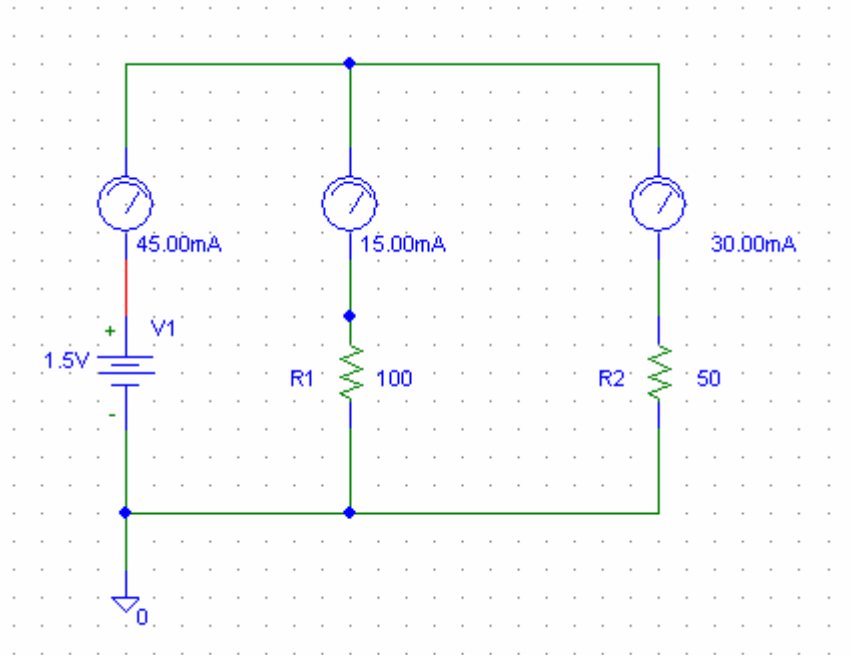
Figure 3.4: Parallel Resistive Circuit

3.2.3 **A Complex resistive circuit**
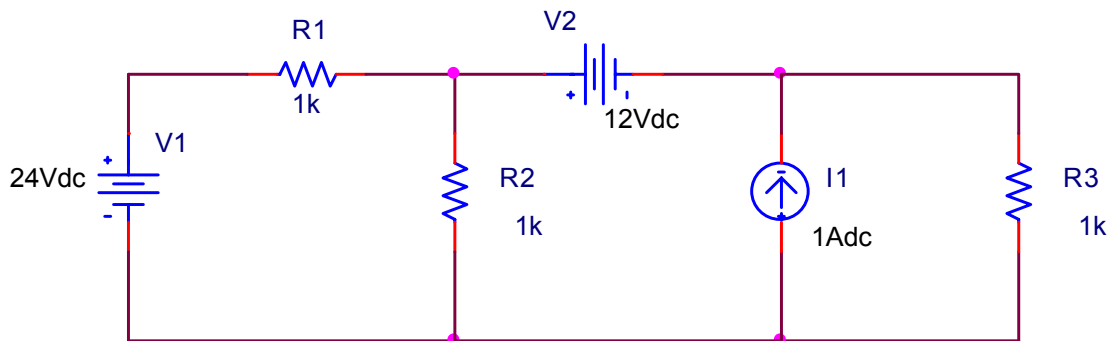
Figure 3.5: Complex Resistive Circuit

Simulation of the circuit yields the following:
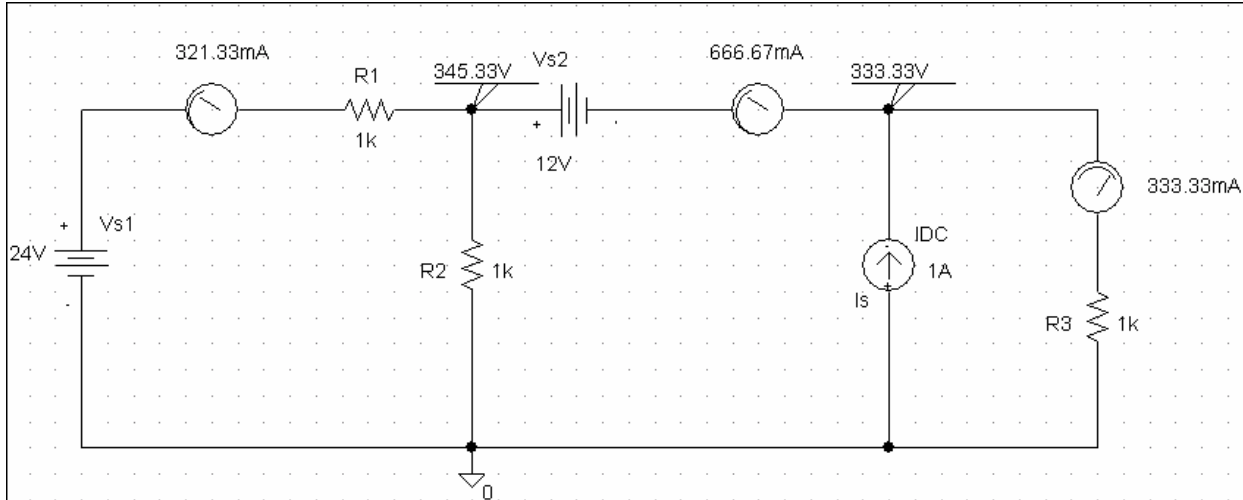
Figure 3.6: Complex Resistive Circuit with PSpice Evaluation

Below, we give the output file generated by PSpice for this circuit. It is a text description of the circuit with all possible voltage and current values of interest.

```
**** 08/22/04 23:09:30 *********** Evaluation PSpice (Nov 1999)
**************

 * C:\Documents and Settings\Mo\My Documents\FED Material\CD-FED\PSpice-
Files\Schematic1.sch


 ****      CIRCUIT DESCRIPTION


************************************************************************
* Schematics Version 9.1 - Web Update 1
* Sun Aug 22 23:09:27 2004



** Analysis setup **
.OP
.OP

* From [PSPICE NETLIST] section of pspiceev.ini:
.lib "nom.lib"

.INC "Schematic1.net"

**** INCLUDING Schematic1.net ****
* Schematics Netlist *

R_R2          0 $N_0001  1k
R_R3          0 $N_0002  1k
R_R1          $N_0003 $N_0001  1k
I_Is          0 $N_0004 DC 1A
V_Vs1          $N_0005 0 24V
```

```
V_Vs2           $N_0001 $N_0006 12V
v_V7            $N_0006 $N_0004 0
v_V10           $N_0005 $N_0003 0
v_V3            $N_0004 $N_0002 0

**** RESUMING Schematic1.cir ****
.INC "Schematic1.als"



**** INCLUDING Schematic1.als ****
* Schematics Aliases *

.ALIASES
R_R2            R2(1=0 2=$N_0001 )
R_R3            R3(1=0 2=$N_0002 )
R_R1            R1(1=$N_0003 2=$N_0001 )
I_Is            Is(+=0 -=$N_0004 )
V_Vs1           Vs1(+=$N_0005 -=0 )
V_Vs2           Vs2(+=$N_0001 -=$N_0006 )
v_V7            V7(+=$N_0006 -=$N_0004 )
v_V10           V10(+=$N_0005 -=$N_0003 )
v_V3            V3(+=$N_0004 -=$N_0002 )
.ENDALIASES

**** RESUMING Schematic1.cir ****
.probe

.END

**** 08/22/04 23:09:30 *********** Evaluation PSpice (Nov 1999)
**************

 * C:\Documents and Settings\Mo\My Documents\FED Material\CD-FED\PSpice-
Files\Schematic1.sch

 ****      SMALL SIGNAL BIAS SOLUTION      TEMPERATURE =   27.000 DEG C

*************************************************************************

 NODE   VOLTAGE      NODE   VOLTAGE      NODE   VOLTAGE      NODE   VOLTAGE


($N_0001)   345.3300                 ($N_0002)   333.3300

($N_0003)    24.0000                 ($N_0004)   333.3300

($N_0005)    24.0000                 ($N_0006)   333.3300



    VOLTAGE SOURCE CURRENTS
    NAME          CURRENT

    V_Vs1          3.213E-01
    V_Vs2         -6.667E-01
    v_V7          -6.667E-01
```

```
    v_V10          -3.213E-01
    v_V3            3.333E-01

   TOTAL POWER DISSIPATION    2.88E-01   WATTS

**** 08/22/04 23:09:30 *********** Evaluation PSpice (Nov 1999)
**************

 * C:\Documents and Settings\Mo\My Documents\FED Material\CD-FED\PSpice-
Files\Schematic1.sch

 ****     OPERATING POINT INFORMATION     TEMPERATURE =   27.000 DEG C

**********************************************************************
         JOB CONCLUDED
         TOTAL JOB TIME                  .01
```

3.3   *MATLAB*

Matlab has become a de facto standard programming language in industry and universities, where scientists and engineers are involved in systems analyses and design.

3.3.1 Getting started

Launch Matlab through the following sequence of steps: **Start > All programs > Matlab 6.5 > Matlab 6.5**. Multiple windows will be opened by Matlab. The window of interest to us at this point is the command window. In that window, the students will be able to type in the recognizable commands that Matlab will execute. The line command always starts with '>>'.

        The students are encouraged to type in 'demo' as seen in Figure 3.7. The demo program will show you some of the capabilities of Matlab and some of the code that Matlab relies upon to perform the required tasks.
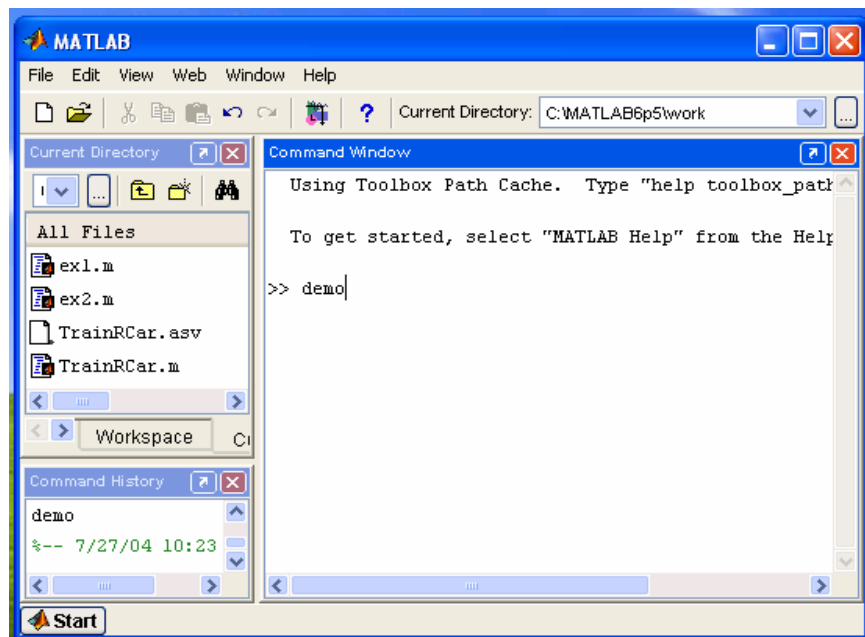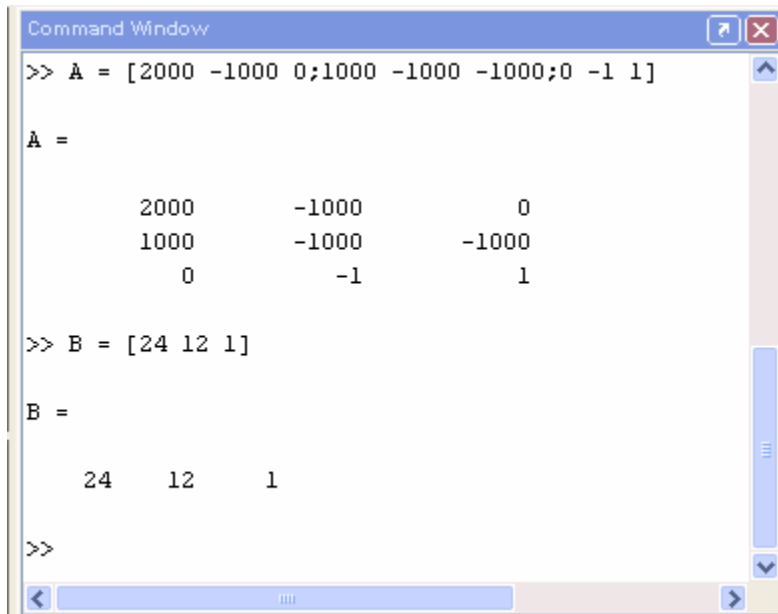
Figure 3.7: Initial Open Window at Matlab Start

3.3.2 **Back to the complex resistive circuit example**

We define two matrices **A** and **B**, where **A** is equivalent to the **A** matrix seen in 3.1 (though it was its determinant that was involved in those calculations) and B is made of the right side of those equations **B** = [24 12 1]
With Matlab, the solution takes fewer steps.
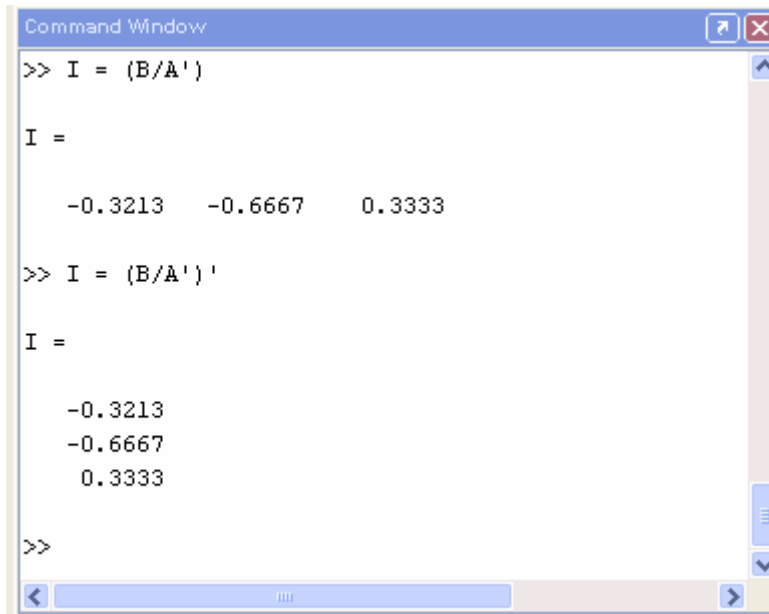
Figure 3.8: Entering into Matlab the Relevant Matrices Elements

Figure 3.9: Matlab Results for the Complex Resistive Circuit

29

We see that with a few keystrokes, Matlab gave us all the needed answers. You would appreciate the power of Matlab as your technical skills and knowledge increase. The main drawback is that it is not very user-friendly, and it may not be the most efficient language. However, most designs and analyses had been made possible with this package. Like most software packages, one always has to verify that the results obtained digitally make sense. No machine or software package is full proof. However, what would take you a day by hand, would take few minutes, and possibly seconds, with a computer.

4 **How Things Work**

4.1 *Introduction*

Among the goals of this course is gaining understanding of engineering design work, enhancing your communication skills, and practicing team work. For this purpose you are required to pick a topic related to electrical and computer engineering and make its professional presentation. A list of suggested topics is included in this chapter but you may also suggest your own (subject to approval by the instructor). The whole team of students that had been formed at the beginning of the semester will start researching the internet, the library, and other resources (see below) to prepare the elements necessary to present the information related to the description of a scientific or technological phenomenon, concept, machine, or innovation. Choose your topic wisely as you have to be excited about what you would be presenting to do it well. Your excitement will motivate others, in your team as well as in other teams in the class, to do a good presentation as well.

4.2 *List of Suggested Topics*

The following is by no means an exhaustive list. It will only help you choose when you still are undecided about your own choice. The listing will appear in no particular order.

1. Cellular technology, or in general, wireless communications
2. CD and/or DVD burners
3. How does an electric motor work?
4. How does a power station generate electricity?
5. Solar power
6. How do computers communicate through the World Wide Web?
7. How does electronics (so-called computer boards) affect the performance of a car?
8. How do we manufacture integrated circuits (ICs)?
9. How does a hard drive store data?
10. How do servomechanisms help in the design of robots?
11. How do we communicate with satellites?

4.3 *Suggested resources*

NJIT Library

Websites
   ECE website
   Encyclopedia.thefreedictionary.com
   Electronics.howstuffworks.com
   Google.com
Advertising is part of the way these and other sites will support themselves. Concentrate on the material that is of interest to you.

Other resources include the CDs 'The Way Things Work' and 'The New Way Things Work'. They contain very basic information, but could give you a starting point from where you could start to develop the required material.

4.4   Presentation

The task of an engineer is not finished unless he or she can perform a good presentation of his or her work. The outside world (from the peers and employer to clients and other people who may be interested in the results) needs to find out first hand the importance of that work. Who is better than the author to render justice to the efforts expended? Nobody can be more excited than the author in presenting it. A presentation of that work starts with a well written text. A well written text is not necessarily a piece of literature. It has to appeal to potential customers of that work (those customers may be the general public). The styles vary depending on the recipients of that information. It cannot be boring, lengthy, or too short (not enough information); and of course it should not contain spelling and grammatical errors.

The next step is to present that information in an informative and exciting way directly to the interested party. The most common way of delivering that material is through a power-point presentation. Other means could include transparencies, poster boards, video, or other visual means. Try to master as much of the material you will be presenting as you can. Otherwise, you will be reading your material (the audience does not get excited by a speaker reading his material).

A sample of a power point presentation is available in appendix A, and an electronic copy of it is also available on the website.

5 Digital Logic

One of the more recent introductions in electrical devices is the personal computer. Computers rely on voltages and currents to represent binary and Boolean values. In this chapter we look at Boolean values and Boolean logic, and basic digital logic devices that realize Boolean operations.

5.1 Introduction to Boolean Logic

Boolean logic is the basis for digital system design, from the simplest circuits to the most advanced computer systems. From a logic standpoint, a Boolean value can be true or false; on or off; yes or no; "maybe" is not allowed. You can think of this like a light switch that can be either on or off; no dimmer switches allowed. In digital logic, true is usually represented by a 1 and false is denoted as 0.

Having a Boolean value of 0 or 1 isn't very useful on its own. To perform useful work with Boolean values, and ultimately the digital circuits we design, we must be able to perform functions that combine different values to produce desired results. There are several basic functions for Boolean values, including the AND, OR, Exclusive-OR, and NOT functions.

5.1.1 The Logical AND Function

The AND function is one of the most commonly used functions in Boolean algebra. It takes two or more Boolean values as its inputs and generates a single Boolean output. This output is 1 (on) only if all of its inputs are 1; if any input is 0 (off) then the output is 0, even if all of the other inputs are 1. This is similar to what you might see if you serve on a jury. For criminal trials, the jury must return a unanimous verdict to convict the defendant. Here, every juror must vote guilty (1) to produce a verdict (output) of guilty. If any juror votes not guilty (0), there can be no guilty verdict.

As another example, consider the extreme example of an AND function with one million inputs. If 999,999 of these input values are 1, but the other input is 0, the AND function outputs a 0. The AND function can be represented using several notation. The AND of Boolean values A and B can be represented as $A \wedge B$, $A \cdot B$, or simply AB.

One way to visualize the AND function is as a circuit consisting of a battery, a light bulb, and two or more switches in series, as shown in Figure 5.1. The switches represent the input values to the Boolean function; a closed switch represents a 1 (on) and an open switch represents a 0 (off). The light bulb corresponds to the output of the AND function. The bulb is on when the function outputs a 1. Figure 5.1(a) gives an example where input $A = 1$ and input $B = 0$, producing an output of 0. Figure 5.1(b) shows the same circuit, except $A = 1$ and $B = 1$, producing an output of 1 and lighting the bulb.

33
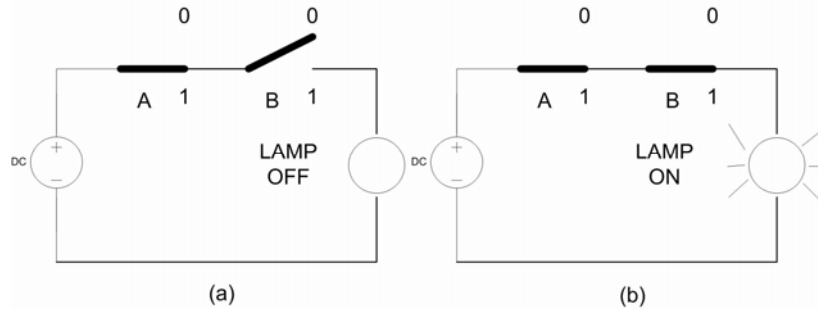
Figure 5.1: Illustration of the AND function producing results of (a) 0, and (b) 1

5.1.2 The Logical OR Function

The OR function is more tolerant of its input values. Its output is 1 if *any* of its inputs are 1. Returning to our extreme, one million input function, if 999,999 of the inputs are 0 but the remaining input is 1, the OR function outputs a 1. The OR of Boolean values A and B can be represented as $A \vee B$ or $A + B$. Be careful with the latter notation. Here, $A + B$ is not arithmetic. In standard arithmetic, $1 + 1 = 2$; however, in Boolean logic, $1 + 1 = 1$.
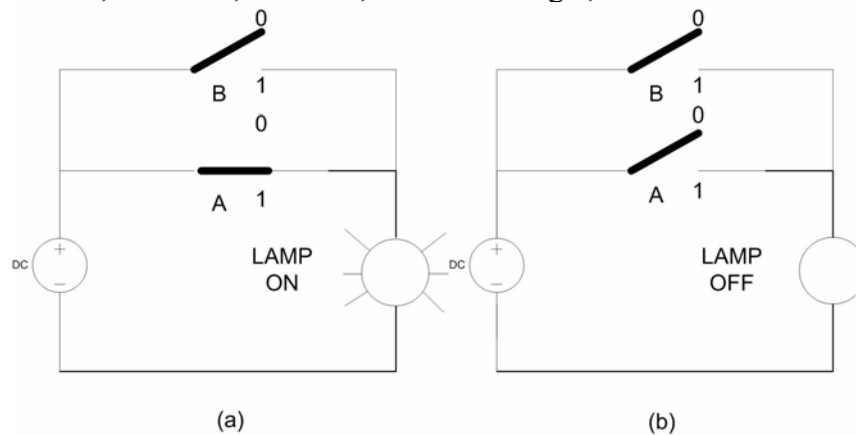
Figure 5.2: Illustration of the OR function producing results of (a) 1, and (b) 0

To visualize this function, consider the circuits shown in Figure 5.2. The switches representing A and B are connected in parallel; setting either switch to 1 completes the circuit and lights the bulb. Figure 5.2(a) shows input values of $A = 1$ and $B = 0$, which produces an output of 1. Figure 5.2(b) shows inputs of $A = 0$ and $B = 0$, generating a result of 0.

5.1.3 The Logical Exclusive-OR Function

The Exclusive-OR, XOR, function, is based on the OR function. Unlike the OR function, however, for the Exclusive-OR function the number of inputs that are 1 does matter. If two inputs are 1, their XOR is equal to 0, not 1; the logical OR of two inputs with values of 1 is 1. When more than two input values are Exclusive-ORed, the output if 1 if an odd number of inputs are 1 and 0 if the number of inputs with values of 1 is even. The Exclusive-OR of Boolean values A and B is represented as $A \oplus B$.

The Exclusive-OR function is similar to a 3-way switch you might find at the top and bottom of a stairway in a house. If one switch is up and the other is down, the light is off. If either switches are up, or both are down, the light is lit. (This may be reversed in some houses. My house has two sets of 3-way switches. The upstairs light is lit when the switches are in opposite positions; the downstairs light is only lit when both switches are in the same position.) Figure 5.3 shows an Exclusive-OR configuration in which the switches must be in opposite positions to light the bulb, or output a 1. In Figure 5.3(a), $A = 0$, $B = 1$, and the output is 1. Figure 5.3(b) shows $A = 1$, $B = 1$, and an output of 0.

(a)   (b)
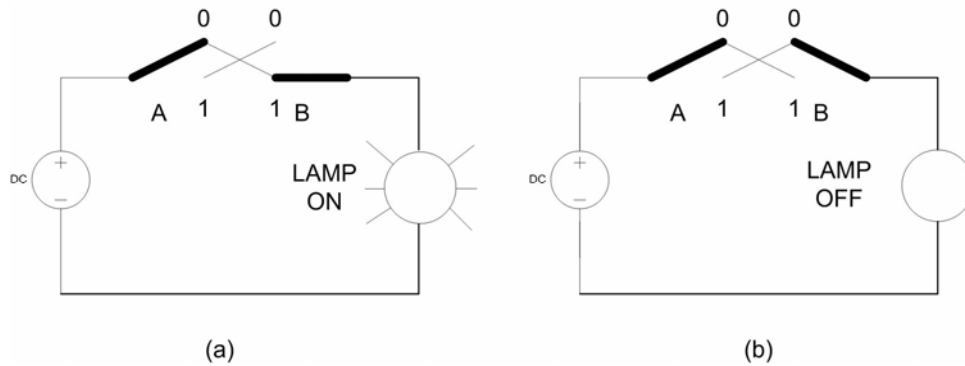
Figure 5.3: Illustration of the XOR function producing results of (a) 1, and (b) 0

5.1.4 The Logical NOT Function

Unlike the AND, OR, and XOR functions, which received two or more inputs, the NOT function operates on a single input. Its output is the opposite of its input value. An input of 0 produces an output of 1, and an input value of 1 generates a 0 output. The NOT of Boolean value A can be represented as A', $/A$, or $!A$. Figure 5.4 illustrates a NOT function for input values of 0 and 1.

(a)   (b)
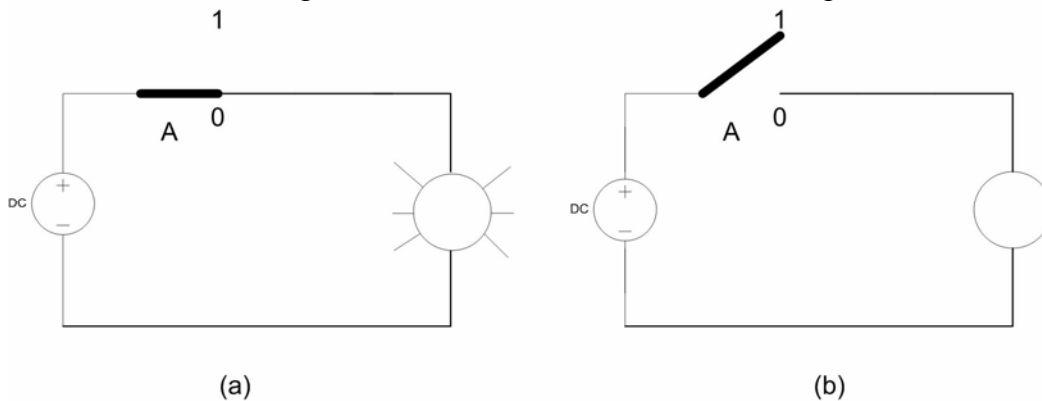
Figure 5.4: Illustration of the NOT function producing results of (a) 1, and (b) 0

Worksheet 5.1 – Boolean Logic Calculations

Show the values of each function for the given values of *A*, *B*, and *C*.

1. $A = 0, B = 0$: $A \wedge B =$ _____ $A \vee B =$ _____ $A \oplus B =$ _____

   $A' =$ _____ $B' =$ _____

2. $A = 0, B = 1$: $A \cdot B =$ _____ $A + B =$ _____ $A \oplus B =$ _____

   $/A =$ _____ $/B =$ _____

3. $A = 1, B = 0$: $AB =$ _____ $A \vee B =$ _____ $A \oplus B =$ _____

   $!A =$ _____ $!B =$ _____

4. $A = 1, B = 1$: $A \wedge B =$ _____ $A + B =$ _____ $A \oplus B =$ _____

   $A' =$ _____ $B' =$ _____

5. $A = 1, B = 0, C = 1$: $A \cdot B \cdot C =$ _____ $A \vee B \vee C =$ _____

   $A \oplus B \oplus C =$ _____

6. $A = 0, B = 1, C = 0$: $A \cdot B \cdot C =$ _____ $A \vee B \vee C =$ _____

   $A \oplus B \oplus C =$ _____

5.2 Truth Tables

Truth tables are a mechanism used to express the operations of a function for all possible input values. In the previous worksheet, we calculated the values of the functions $A \wedge B$, $A \vee B$, $A \oplus B$, A', and B' for all possible values of A and B. Truth tables show these values in a convenient format. A truth table has two groups of columns, and several rows of data. The left-hand group of columns consists of all inputs. Each row of the table shows a different combination of input values. For example, a table with two inputs, A and B, would have values of $A = 0$ and $B = 0$ in its first row, followed by A and B of 0 and 1; 1 and 0; and 1 and 1, as shown in Table 5.1.

Table 5.1: A generic truth table for a 2-input function

A	B	Function
0	0	
0	1	
1	0	
1	1	

Consider the truth table for the 2-input AND function shown in Table 5.2(a). The left hand side shows the two inputs, A and B, and all possible combinations of values for these inputs, one combination per row. The first row shows $A = 0$ and $B = 0$, followed by rows for $A = 0$ and $B = 1$, $A = 1$ and $B = 0$, and $A = 1$ and $B = 1$. The right hand side of the table shows the output of the AND function for these input values. The AND function only outputs a 1 when $A = 1$ and $B = 1$. It outputs a 0 at all other times. Tables 5.2(b), (c), and (d) show the truth tables for the OR, XOR, and NOT functions.

Table 5.2: Truth tables for the (a) AND, (b) OR, (c) XOR, and (d) NOT functions

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

(a)

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

(b)

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

(c)

A	A'
0	1
1	0

(d)

5.3 Complementary Functions – NAND, NOR, and Exclusive-NOR

In addition to the AND, OR, XOR, and NOT functions, there are three common, complementary functions. The NAND function is the opposite of the AND function; it may help to think of the NAND as the NOT-AND function. The NAND of two or more inputs is 0 if all of the inputs are 1; if any input is 0, the output of the NAND is 1. Table 5.3(a) shows the truth table for the 2-input NAND function. Note that its outputs are exactly the opposite of those for the AND function of Table 5.2(a)

Table 5.3: Truth tables for the (a) NAND, (b) NOR and (c) XNOR functions

A	B	NAND		A	B	NOR		A	B	XNOR
0	0	1		0	0	1		0	0	1
0	1	1		0	1	0		0	1	0
1	0	1		1	0	0		1	0	0
1	1	0		1	1	0		1	1	1
	(a)				(b)				(c)	

There are also complementary functions for the OR and XOR functions. The NOR function produces an output of 1 only when all of its inputs are 0. Its truth table is shown in Table 5.3(b); as with the NAND and AND functions, the outputs of the NOR function are exactly the opposite of the OR function. The Exclusive-NOR, or XNOR, function outputs a 1 when an even number of inputs are 1, and 0 when an odd number of inputs are 1. Its truth table is shown in Table 5.3(c). Again, the outputs of the XNOR function are exactly the opposite of those of the XOR function in Table 5.2(c). Note that the output of the 2-input XNOR function is 1 whenever the two inputs are the same. For this reason, the 2-input XNOR function is sometimes referred to as the *equivalence* function.

38

Worksheet 5.2 – Boolean Logic Calculations

Show the truth tables for the 3-input AND, OR, XOR, NAND, NOR, and XNOR functions.

A	B	C	$A \wedge B \wedge C$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

A	B	C	$A \vee B \vee C$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

A	B	C	$A \oplus B \oplus C$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

A	B	C	$(A \wedge B \wedge C)'$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

A	B	C	$(A \vee B \vee C)'$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

A	B	C	$(A \oplus B \oplus C)'$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

5.4   Digital Logic

Logic gates are digital components that implement the logic functions described earlier. They are usually represented by the symbols shown in Figure 5.5. For functions with more than two inputs, the gate symbols are the same, except more inputs are added to the gate. (The NOT gate is an exception; it can never have more than one input.) Each gate always has one output regardless of the number of inputs.
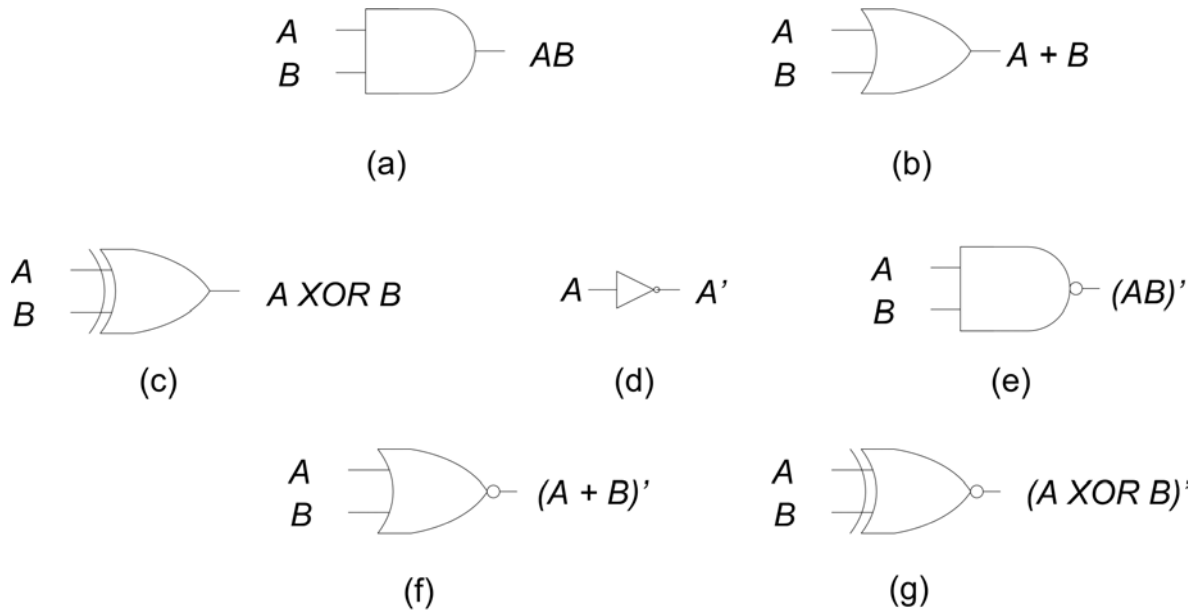
Figure 5.5: Logic symbols for the (a) AND, (b) OR, (c) XOR, (d) NOT, (e) NAND, (f) NOR, and (g) XNOR gates

Notice that the NOT, NAND, NOR, and XNOR gates have small circles at their outputs. This indicates that the output of the gate is complemented, or inverted. For example, consider the AND and NAND gates. They look the same, except the NAND gate has a small circle at its output. Remember, as shown in the truth tables, the output of the NAND gate is the complement of the output of the AND gate. Logically, you can consider the NAND gate to be an AND gate which sends its output to a NOT gate, as shown in Figure 5.6(a). The reverse is also true; an AND gate has the same logical function as a NAND gate that sends its output to a NOT gate as shown in Figure 5.6(b).
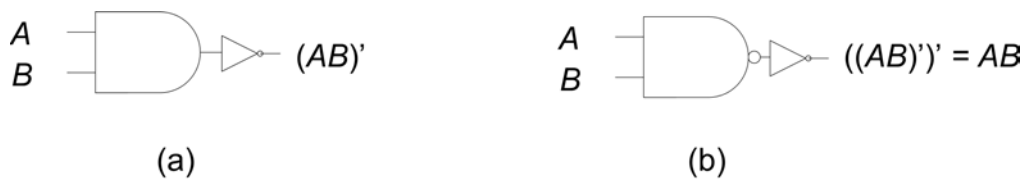
Figure 5.6: Logical implementation of (a) a NAND function using AND and NOT gates, and (b) an AND function using NAND and NOT gates

5.5   Logic Chips

To build logic circuits, designers can use Integrated Circuit chips, or ICs, or chips. Different ICs are available with the gates we've seen so far, as well as more advanced components. A chip may look like the chip shown in Figure 5.7(a). The black box in the center of the chip contains the circuitry that implements the chip's functions. The metal strips on the sides of the chip are its pins. These pins interface the logic gates in the chip to the rest of the circuit. For a NAND gate with two inputs, two of the chip's pins would connect to the inputs of the NAND gate, and the output of the NAND gate would connect to another pin. By connecting the pins of the chip, we incorporate the NAND gate into a circuit.


(a)   (b)
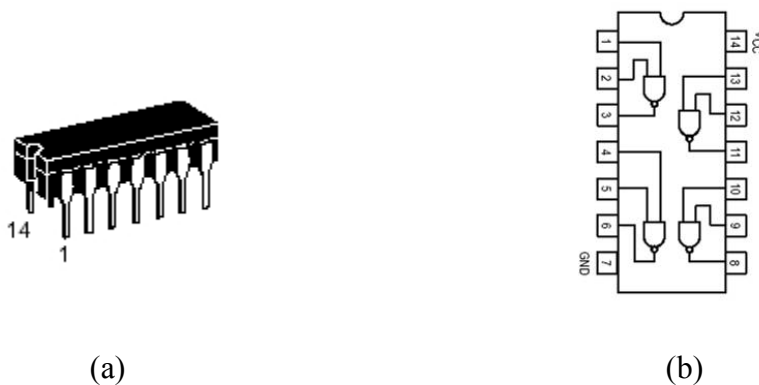
Figure 5.7: (a) A 7400 Quad 2-input NAND gate, and (b) its logic diagram

Figure 5.7(b) shows the logical organization of a 7400 TTL chip. This chip has four, 2-input NAND gates. For each NAND gate, two pins are connected to the gate's inputs and one pin connects to its output. This accounts for 12 of the chip's 14 pins. The other two pins are used for power connections. Notice that the pins are numbered counterclockwise around the chip starting in the upper left corner. Chips usually have a dot by pin 1 or a notch at the top of the chip, such as the notch shown in the figure, so you can tell how to connect it in your circuit.

The logic gates require power in order to function; we use the two power pins to supply power to the chip and its gates. Recall from the circuits covered earlier, those consisting of switches and light bulbs, that a circuit must be completed from the battery's positive terminal to its negative terminal in order for current to flow. One of the two power pins, denoted Vcc, is connected to the positive terminal. The other power pin, Gnd or ground, is connected to the negative terminal. For TTL chips, the voltage of the battery or power supply is 5 volts, denoted 5V.

5.6 *Where Boolean Logic Meets Digital Logic*

A logic chip doesn't know what binary values 0 and 1 are. To input a 1 to a gate, we supply a voltage at a given level. For TTL chips, logic 1 is a voltage at or near 5V, the voltage of the chip's power supply. A voltage near 0V is used for logic 0. The same voltage values are used for the inputs and outputs.

Other Logic Chips

Now let's look at some other chips. The 7402 chip has four NOR gates; its logic diagram is shown in Figure 5.8(a). The 7404 is a hex inverter, having six NOT gates. It is shown in Figure 5.8(b). Figure 5.8(c) shows the 7408 chip, which has four AND gates. The quad OR, XOR, and XNOR chips, the 7432, 7486, and 74266, respectively, are shown in Figures 5.8(d), (e), and (f).
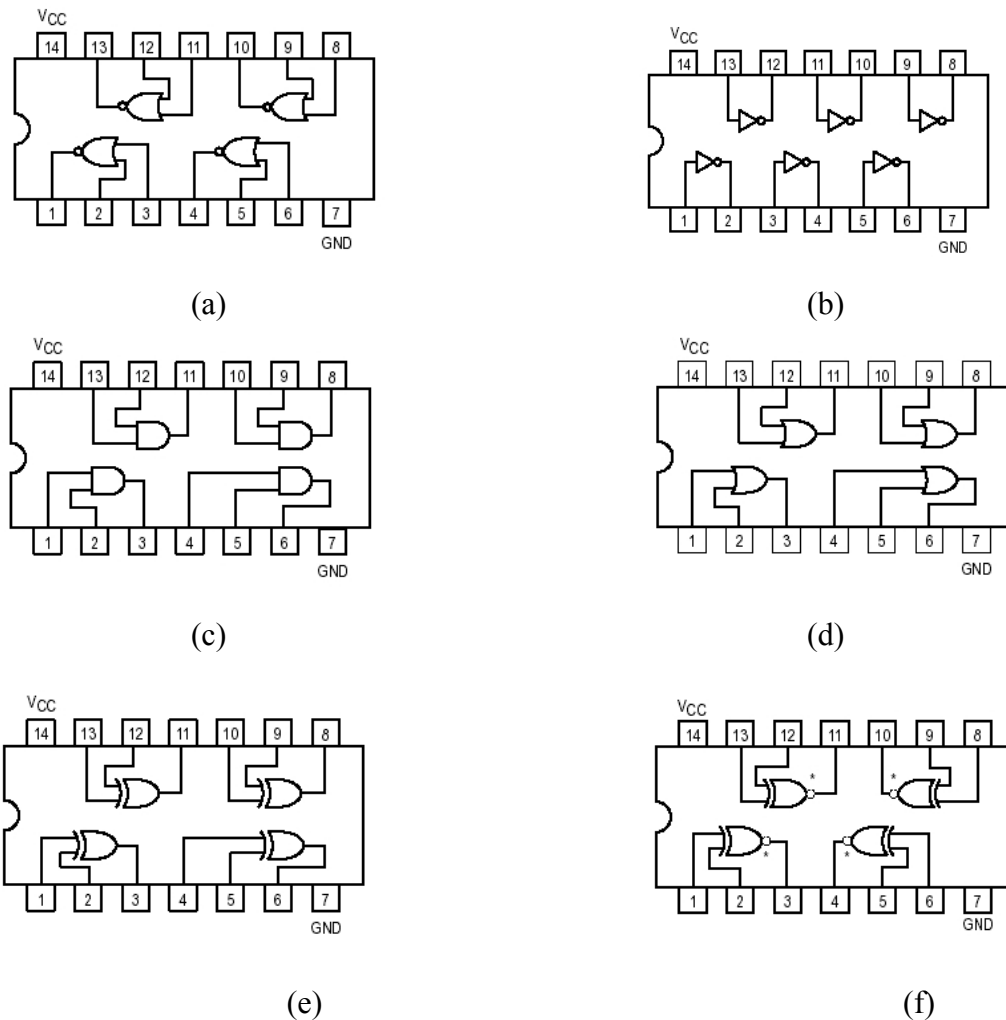
(a)

(b)

(c)

(d)

(e)

(f)

Figure 5.8: Logic diagrams for the (a) 7402, (b) 7404, (c) 7408, (d) 7432, (e) 7486, and (f) 74266 chips

5.7   Combining Logical Operations

The AND, OR, XOR, and NOT functions introduced in the previous sections are useful in their own right, but they are most useful when combined to form more complex functions. In the real world, situations that cannot be described with a single AND or OR function often can be modeled by combining logical operations.

For example, consider the following scenario. A teenager wants to buy a car but doesn't have any money for a down payment. Ignoring options that involve borrowing money from his parents, winning the lottery, or robbing a bank, the teenager has two options. One is to earn more money, possibly by working more hours or taking a second job. The other is to earn the same amount of money and spend less, maybe by giving up his or her cell phone. We could express these options by combining the logical AND and OR operations as follows.

Get $ to buy car = (*Earn more $*) OR (*Earn same $* AND *Spend less $*)

5.8   Modeling Real-world Situations

We can design a circuit to model this function. We would need three switches, one each for *Earn more $*, *Earn same $*, and *Spend less $*. We would also need one light bulb to represent the output *Get $ to buy car* and a battery to supply power to the circuit. The circuit would light the bulb if *Earn more $* is TRUE (1), or if both *Earn same $* and *Spend less $* are true. The circuit to do this is shown in Figure 5.9. Notice that this circuit combines elements of the AND and OR circuits from the previous chapter. The light can be lit if both *Earn same $* and *Spend less $* are on (TRUE). This is the AND function. The light can also be lit if *Earn more $* is true. Having either option light the bulb is an OR function.
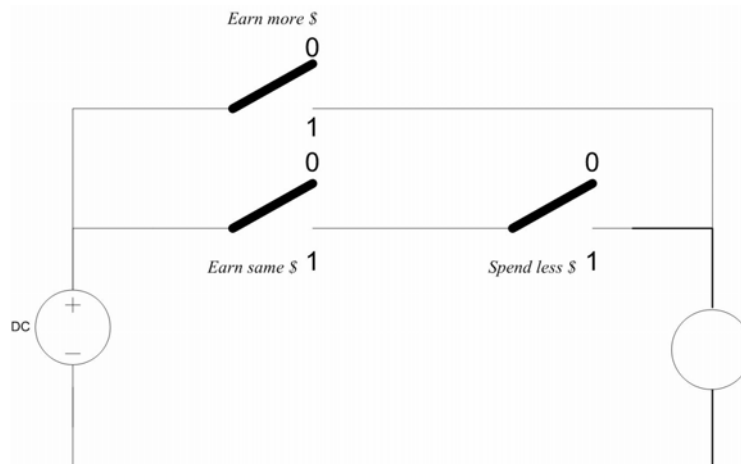

Figure 5.9: Circuit to generate the *Get $ to buy car* function

5.9   Truth Tables for Combined Functions

Just as with any logic function, we can create a truth table for our car-buying example. Here the variables are the three conditions *Earn more $*, *Earn same $*, and *Spend less $*, which we

43

represented as switches in our circuit. There is one output for this function, *Get $ to buy car*; this was represented as a light bulb in our circuit. Each input, and the output, can be either TRUE or FALSE. The truth table for this scenario is shown in Table 5.4. Table 5.4(a) shows the truth table with values of TRUE and FALSE. Table 5.4(b) shows the same table with binary values 1 and 0; 1 represents TRUE and 0 corresponds to FALSE.

Table 5.4: Truth tables for the *Get $ to buy car* function

Earn more $	Earn same $	Spend less $	Get $ to buy car	Earn more $	Earn same $	Spend less $	Get $ to buy car
FALSE	FALSE	FALSE	FALSE	0	0	0	0
FALSE	FALSE	TRUE	FALSE	0	0	1	0
FALSE	TRUE	FALSE	FALSE	0	1	0	0
FALSE	TRUE	TRUE	TRUE	0	1	1	1
TRUE	FALSE	FALSE	TRUE	1	0	0	1
TRUE	FALSE	TRUE	TRUE	1	0	1	1
TRUE	TRUE	FALSE	TRUE	1	1	0	1
TRUE	TRUE	TRUE	TRUE	1	1	1	1

(a)   (b)

5.9.1 Creating the Exclusive-OR Function

Let's look at another way of creating the exclusive-OR function. Instead of using an XOR gate, we will construct this function using AND, OR, and NOT operations. First we start with the truth table for the exclusive-OR function, shown in Table 5.5.

Table 5.5: Truth table for the XOR function

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

To create the XOR function, we will develop functions that generate each 1 output separately and then logically OR these functions together. Looking at the table, we can see that $A \oplus B = 1$ twice, when $A = 0$ and $B = 1$, and when $A = 1$ and $B = 0$. First we will create the function to generate the first 1. This function is 1 when $A = 0$ and $B = 1$, or when $A' = 1$ and $B = 1$. (Remember that A' is the NOT of A; if $A = 0$ then $A' = 1$, and if $A = 1$ then $A' = 0$.) We can implement this function as $A' \wedge B$, or $A'B$. Its truth table is shown in Table 5.6(a).

Table 5.6: Truth tables for the functions (a) $A'B$, (b) AB', and (c) $A'B + AB' = A \oplus B$

A	B	$A'B$
0	0	0
0	1	1
1	0	0
1	1	0

(a)

A	B	AB'
0	0	0
0	1	0
1	0	1
1	1	0

(b)

A	B	$A'B + AB'$ $= A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

(c)

Next we calculate the function for the second 1. This occurs when $A = 1$ and $B = 0$, or if $A = 1$ and $B' = 1$. Its function is AB', and its truth table is shown in Table 5.6(b).

Finally, we logically OR these two functions together to produce a single, combined function. Here this function is $A'B + AB'$, which is true whenever either $A'B$ or AB' is true. Its truth table is shown in Table 5.6(c). Note that this truth table is identical to the truth table for $A \oplus B$ given in Table 5.5. Therefore, $A \oplus B = A'B + AB'$.

Worksheet 5.3 – Truth Tables

Show the truth tables for the following functions.

A	B	C	A + B + C
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

A	B	C	AB	BC	AB + BC
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

A	B	C	A'BC	A'C'	A'BC + A'C'
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

A	B	C	A'C	BC	A'B	A'C + BC + A'B
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

A	B	C	ABC	ABC'	A'B	ABC + ABC' + A'B
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

5.9.2 Determining Functions from Truth Tables

Besides using functions to create truth tables, we can also work in the other direction. We can use truth tables to determine its function. Consider, for example, the truth table shown in Table 5.7. This function has three inputs, *A*, *B*, and *C*, and one output function.

Table 5.7: Truth table for an unknown function

A	*B*	*C*	*Function*
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Three different combinations of input values cause this function to be 1. The first occurs when $A = 0$, $B = 0$, and $C = 0$, or $A' = 1$, $B' = 1$, and $C' = 1$. This can be expressed as $A' \wedge B' \wedge C'$, or $A'B'C'$. The function is also 1 when $A = 0$, $B = 1$, and $C = 0$, or $A' = 1$, $B = 1$, and $C' = 1$. This is $A'BC'$. Finally, the function is 1 when $A = 1$, $B = 1$, and $C = 1$, or ABC. We can combine them to form a single function that matches this truth table, $A'B'C' + A'BC' + ABC$. Table 5.8 shows a truth table for each individual component of the function and the overall function.

Table 5.8: Truth table for the function and its components

A	*B*	*C*	*A'B'C'*	*A'BC'*	*ABC*	*A'B'C'* + *A'BC'* + *ABC*
0	0	0	1	0	0	1
0	0	1	0	0	0	0
0	1	0	0	1	0	1
0	1	1	0	0	0	0
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	0	0	0	0
1	1	1	0	0	1	1

Worksheet 5.4 – Truth Tables

Show the functions for the following truth tables. Express each entry in the truth table that generates an output of 1 as a separate term.

A	B	C	Function
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Overall function: _____

A	B	C	Function
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Overall function: _____

A	B	C	Function
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Overall function: _____

5.9.3 Simplifying Functions

Sometimes we can simplify a function by combining some of its terms. For the truth table in Table 5.8, take a look at the first two lines that produce outputs of 1. The inputs for these two lines are $A = 0$, $B = 0$, and $C = 0$, and $A = 0$, $B = 1$, and $C = 0$. Notice that both $A = 0$ and $C = 0$ in both cases. As long as A and C are both 0, the function should produce a 1, regardless of the value of B. If $B = 0$, then $A = 0$, $B = 0$, and $C = 0$, and $A'B'C'$ is 1. If $B = 1$, then $A = 0$, $B = 1$, and $C = 0$, and $A'BC'$ is 1. We can replace these two terms by the single term $A'C'$. Our function would then become $A'C' + ABC$; its truth table is shown in Table 5.9.

Table 5.9: Truth table for the simplified function

A	B	C	$A'C'$	ABC	$A'C' + ABC$
0	0	0	1	0	1
0	0	1	0	0	0
0	1	0	1	0	1
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	0	1	1

In general it is desirable to simplify functions. If you implement a function using logic gates, simplifying the function usually results in simpler circuits.

Worksheet 5.5 – Truth Tables

Show the simplified functions for the following truth tables. Express each entry in the truth table that generates an output of 1 as a separate term.

A	B	C	Function
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Original function: _____

Simplified function: _____

A	B	C	Function
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Original function: _____

Simplified function: _____

A	B	C	Function
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Original function: _____

Simplified function: _____

5.10 More Complex Digital Components

The digital components we've seen so far are fairly basic. Although they can perform some useful work on their own, they are rather limited. This chapter examines some more complex digital components that can be used to perform more complex functions. It looks at some components that change their outputs as the inputs change; these components include multiplexers, decoders, and encoders. It also presents some components that can store data for future use, including registers and counters. Finally, this chapter concludes with a laboratory experiment that uses some of the components described in this chapter.

Note that some of the material in this chapter, and Figures 5.10 – 5.13, were adapted from *Computer Systems Organization and Architecture*, by John D. Carpinelli (Addison Wesley, Boston, MA, 2001).

5.11 Binary Numbers

Many digital circuits deal with data. To understand how these circuits work, we must first understand how binary data works. In this section we will examine the binary representations of data. We will study the direct storage of data in binary, as well as the binary coded decimal, BCD, data format.

5.11.1 Binary Values

In decimal notation, each digit has a specific value. This value depends on the digit itself, as well as its position. For example, the 1 in 12 is much less than the 1 in 1,234,567.

The decimal number 1,234,567 can be thought of as $1,000,000 + 200,000 + 30,000 + 4,000 + 500 + 60 + 7$. Each digit is followed by some number of zeroes, which means that the digit is actually multiplied by 10 raised to some power. That power depends on the location of the digit. For example, the 7 is multiplied by 10^0, or 1, since it is the rightmost digit. The 6 is one position to the left, so it is multiplied by 10^1, or 10, to produce a value of 60. The five is multiplied by 10^2, and so on. We could also express 1,234,567, as follows.

$$(1\times10^6) + (2\times10^5) + (3\times10^4) + (4\times10^3) + (5\times10^2) + (6\times10^1) + (7\times10^0)$$

The same holds true for binary values. However, since they use base 2 instead of base 10, we multiply the individual digits by powers of 2 instead of powers of 10. Consider the binary value 10111. This can be expressed as $(1\times2^4) + (0\times2^3) + (1\times2^2) + (1\times2^1) + (1\times2^0)$, which is $16 + 0 + 4 + 2 + 1 = 23$ in decimal.

Binary values can get very large very fast. For convenience, we sometimes use *hexadecimal* notation to express binary values. Each group of four bits, starting at the right hand side of the number, is represented as a single hexadecimal digit. For binary values from 0000 to 1001, we use the standard decimal digits 0 to 9; 0 = 0000, 1 = 0001, 2 = 0010, 3 = 0011, 4 = 0100, 5 = 0101, 6 = 0110, 7 = 0111, 8 = 1000, and 9 = 1001. However, binary values from 1010 to 1111 do not have single-digit decimal equivalents. For that reason, we use the letters A through F to represent these values; A is 1010, B is 1011, C is 1100, D is 1101, E is 1110, and F is 1111.

For convenience, large binary numbers can be broken down into groups of bits called *bytes*. Each byte is 8 bits wide. Computers often access several bytes of data simultaneously when processing instructions.

5.11.2 **Binary Coded Decimal (BCD)**

Some applications require decimal data. Typical of these applications are those with digital displays. We could store such data in binary format, and convert the binary values to decimal to display the data, but there is a better way. *Binary Coded Decimal*, or BCD, notation stores each decimal digit as four binary bits. The binary value only uses the bit patterns 0000 to 1001 to represent 0 to9. The binary patterns corresponding to hexadecimal values A to F are not used. In this way we can store decimal data directly and operate on the data in decimal.

To illustrate the differences between binary and BCD, consider the binary value 0010 0111. As a binary value, this would be $(0\times2^7) + (0\times2^6) + (1\times2^5) + (0\times2^4) + (0\times2^3) + (1\times2^2) + (1\times2^1) + (1\times2^0)$, or 39. However, let's say this is actually a BCD value. The low four bits, 0111, would represent a single decimal digit, 7. The next four bits, 0010, also represent a decimal digit, 2 in this case. Concatenating these values, we see that this represents 27 in BCD, which is quite different from the value of 39 of the binary representation.

Worksheet 5.6: Binary and BCD values

What are the decimal equivalents of the following binary values?

1.   Binary: 01011001   Decimal: _____

2.   Binary: 10011001   Decimal: _____

3.   Binary: 00000001   Decimal: _____

What are the decimal equivalents of the following BCD values?

1.   Binary: 01011001   Decimal: _____

2.   Binary: 10011001   Decimal: _____

3.   Binary: 00000001   Decimal: _____

Show the 8-bit binary values for the following decimal values.

1.   Decimal: 43   Binary: _____

2.   Decimal: 71   Binary: _____

3.   Decimal: 53   Binary: _____

Show the 8-bit BCD values for the following decimal values.

1.   Decimal: 43   Binary: _____

2.   Decimal: 71   Binary: _____

3.   Decimal: 53   Binary: _____

5.12 Components

There are some standard functions that are frequently used in digital circuits. To simplify the design task, developers have created several chips to implement these functions. A few of these are described in this section.

5.12.1   Multiplexers

A multiplexer, or MUX, is a selector. It chooses one of its data inputs and passes it through to its output. To illustrate its operation, consider the 4 to 1 multiplexer shown in Figure 5.10. Four binary data values are passed to the inputs of the multiplexer. Two select signals, S_1 and S_0, determine which of the four inputs is passed through to the output. If $S_1 = 0$ and $S_0 = 0$, input 0 is passed through to the output. Similarly, setting S_1 and S_0 to 01, 10 or 11 would pass input 1, 2, or 3, respectively to the output. Note that the E signal is an *enable* signal. If this input is 1, the chip is enabled and performs as described. However, if the E input is 0, the chip is *disabled* and does not produce a usable output. This output, called the *high-impedance state*, or *Z*, does not represent a valid binary value.

S_1	S_0	E	Output
X	X	0	Z
0	0	1	Input 0
0	1	1	Input 1
1	0	1	Input 2
1	1	1	Input 3
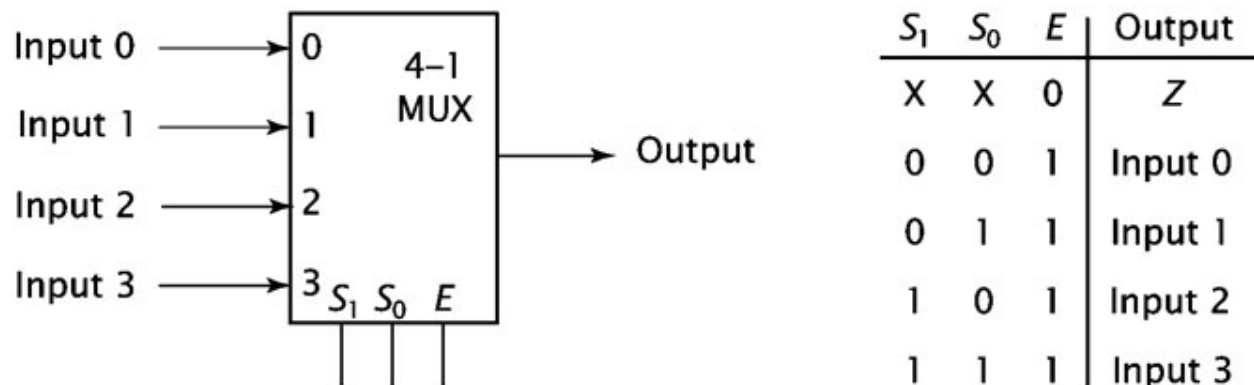
Figure 5.10: A 4-1 multiplexer

5.12.2   Decoders

A decoder, as its name implies, accepts a value and decodes it. It has n inputs and 2^n outputs, numbered from 0 to $2^n - 1$. Each output represents one possible value of the inputs. The output corresponding to the value of the n inputs is activated. For example, a decoder with 3 inputs and 8 outputs will activate output 6 when the input values are 110.

Figure 5.11 shows a 2 to 4 decoder. For inputs $S_1S_0 = 00, 01, 10$ and 11, outputs 0, 1, 2 and 3, respectively, are high when E is 1.
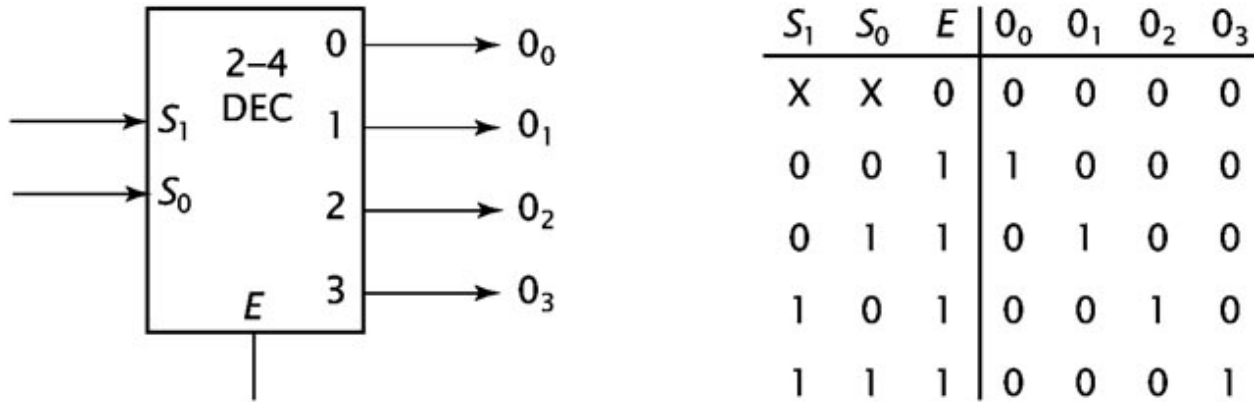
S_1	S_0	E	O_0	O_1	O_2	O_3
X	X	0	0	0	0	0
0	0	1	1	0	0	0
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	1

Figure 5.11: A 2-4 decoder

5.12.3   Encoders

The encoder is the exact opposite of the decoder. It receives 2^n inputs and outputs an *n*-bit value corresponding to the one input which has a value of 1. The 4 to 2 encoder is shown in Figure 5.12. Notice that there is a third output, *V*. This value indicates whether any of the inputs are high. As with the previous chips, the output is invalid when E is 0.

I_0	I_1	I_2	I_3	E	S_1	S_2	V
X	X	X	X	0	Z	Z	Z
0	0	0	0	1	0	0	0
1	0	0	0	1	0	0	1
0	1	0	0	1	0	1	1
0	0	1	0	1	1	0	1
0	0	0	1	1	1	1	1
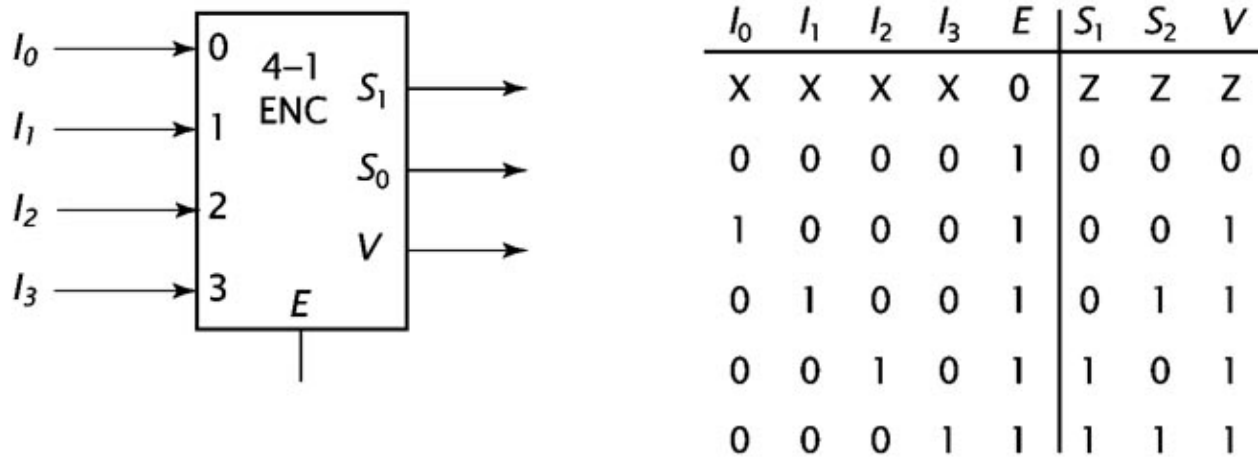
Figure 5.12: A 4-2 encoder

This encoder works if E is 1 and exactly zero or one inputs are high, but fails if more than one is high. For instance, if inputs 1 and 2 are high, the encoder might set $S_1S_0 = 01$, $S_1S_0 = 10$, or some other value which represents neither input. When more than one input can be active, a different type of encoder must be used. This is the role of the *priority encoder*.

A priority encoder works just like a regular encoder, with one exception. Whenever more than one input is active, the output is set to correspond to the highest active input. For example, if inputs 0, 1 and 3 are active, the output is set to $S_1S_0 = 11$, corresponding to input 3

5.12.4 **Registers**

The components presented so far in this section are *combinatorial*. Once their inputs are changed, their outputs may also change. When values must be stored for later use, *sequential* logic is needed. Unlike combinatorial logic, sequential components can retain their output values even when their inputs change.

The most fundamental sequential components are the *registers*. They store one or more bits of data and make them available to other components. A 1-bit register is often called a *latch* or *flip-flop*.

Most registers have a *clock* input. The clock input is usually derived from an oscillator or other circuit which alternates its output between 0 and 1. It is used to synchronize the flow of data in a digital system. When the clock input changes from 0 to 1, the data on the input of the register is loaded into it. The data is made available via output Q. Some variants also have a load signal (*LD*), which must be high as the clock changes from 0 to 1 in order for data to be loaded into the flip-flop.

5.12.5 **Counters**

A *counter* does exactly what its name implies, it counts. It stores a binary value and, when signaled to do so, arithmetically increments or decrements its value. Some counters do only one or the other, while other counters can do both. Like other registers, counters can be loaded with an externally supplied value. Some counters can also be cleared.

Figure 5.13 shows one implementation of a 4-bit up counter, along with its truth table. It uses four flip-flops to store the 4-bit value. The *CLR* control signal clears the counter. Setting *INC* = 1 increments the counter as the clock changes from 0 to 1. Signal C_{out} is set to 1 as the count increments from 1111 back to 0000.

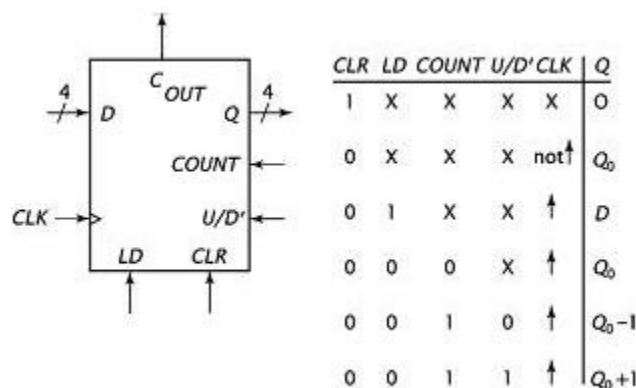| CLR	LD	COUNT	U/D'	CLK	Q
1	X	X	X	X	0
0	X	X	X	not↑	Q_0
0	1	X	X	↑	D
0	0	0	X	↑	Q_0
0	0	1	0	↑	Q_0-1
0	0	1	1	↑	Q_0+1

Figure 5.13: A 4-bit counter and its truth table

Sometimes we wish to have a counter that does not count in binary. Rather, it would count in decimal, outputting BCD values instead. There are special counters, called *decade counters*, designed just for this purpose. They count from zero to nine, and then go back to zero. Decade counters are useful in devices which require counting in decimal, such as digital clocks. We'll see how these counters can be used in the following section.

5.13 Application Note: Decimal Counter

In this section we will examine a simple decimal counter circuit. Our decimal counter will have the ability to count from 00 to 99 seconds, and its output will be displayed in BCD on LEDs.

We have already seen decade counters in the previous section. They count from 0 to 9, and then go back to zero. We could wire up an oscillator to cause this chip to count up once every second, but we'll use a pushbutton switch instead for simplicity.

This counter will work well for the one's digit of our counter, but what about the 10's digit? What can we use for that digit?

We can actually use the same type of counter for the tens digit. We just have to increment it differently. If we increment it using the pushbutton, as we did for the one's digit, our sequence would be 00, 11, 22, 33... 88, 99, which is not what we want at all. Clearly we need a different way to increment the ten's digit counter.

The key to solving this problem is to notice that the ten's digit is incremented whenever the ones digit goes from nine to zero. The decade counter has a carry output signal that is set to logic 1 when this occurs. We can then use that signal on the 1's counter to cause the counter of the 10's digit to increment. This is exactly how we want our counter to function. The circuit to do this is shown in Figure 5.14.
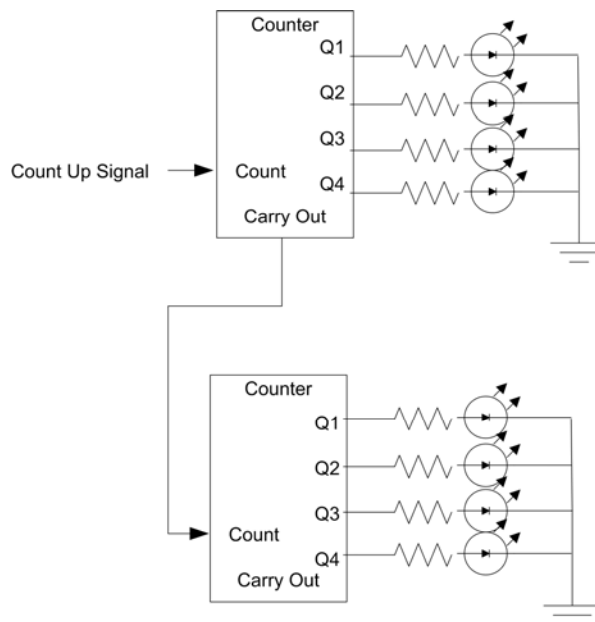
Figure 5.14: 2-digit decimal counter and display using LEDs

5.14 On Some Helpful information and Circuits
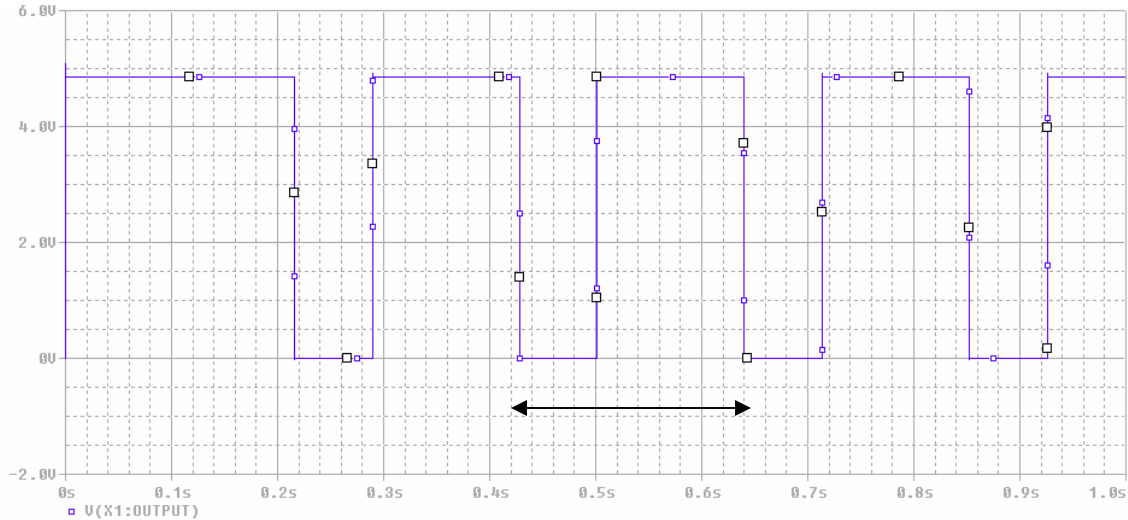
5.14.1   Constant level pulse train

Figure 5.15: Constant level pulse train

As shown in Figure 5.15, aside from the initial moment, the signal is repetitive and has the form of a pulse train. The duration of a cycle (the smallest interval of time after which the signal repeats itself) is called a period, T. The unit for a period is a second (abbreviated sec or s) or its fraction, such as a millisecond (ms). The inverse of a period is the frequency $f = 1/T$. The unit for frequency is Hertz (Hz) equal to one period per second. This if the period is equal to 1 ms, the frequency is 1 kHz.

For example, the signal that brings electrical energy to our homes is a sinusoidal signal (periodic) with frequency 60 Hz (Hertz). Its period would then be $T = 1/60$ sec. In the case of the previously mentioned signal, the period is approximately 0.21 sec.

The duration over a cycle where a signal is high (in this case 5V) is labeled T_{ON}. The duration over a cycle where a signal is low (in this case 0V) is labeled T_{OFF}. The ratio T_{ON}/T is defined as the duty cycle of the signal.

$d = T_{ON}/T$

In this case, $T_{ON} = 0.14$ s. Since $T = 0.21$s, the duty cycle d would be:

$d = 0.14/0.21 = 0.667$ or 66.7%

5.14.2   The 555 timer

The 555 timer is a key component of multivibrators. The two most popular forms of a multivibrator are the monostable multivibrator and the astable (or free-running) multivibrator. At this point we will concentrate on the astable multivibrator. This configuration will be useful in generating clock signals (constant level pulse train). That signal could act as a clock signal for some of the integrated circuits that you may need in your project, and could also help generate an audible tone through a speaker.

5.14.3   Astable Multivibrator

The 555 timer is at the heart of building these multivibrators. The circuit that generated the previous pulse train shown in Figure 5.15 is given in Figure 5.16
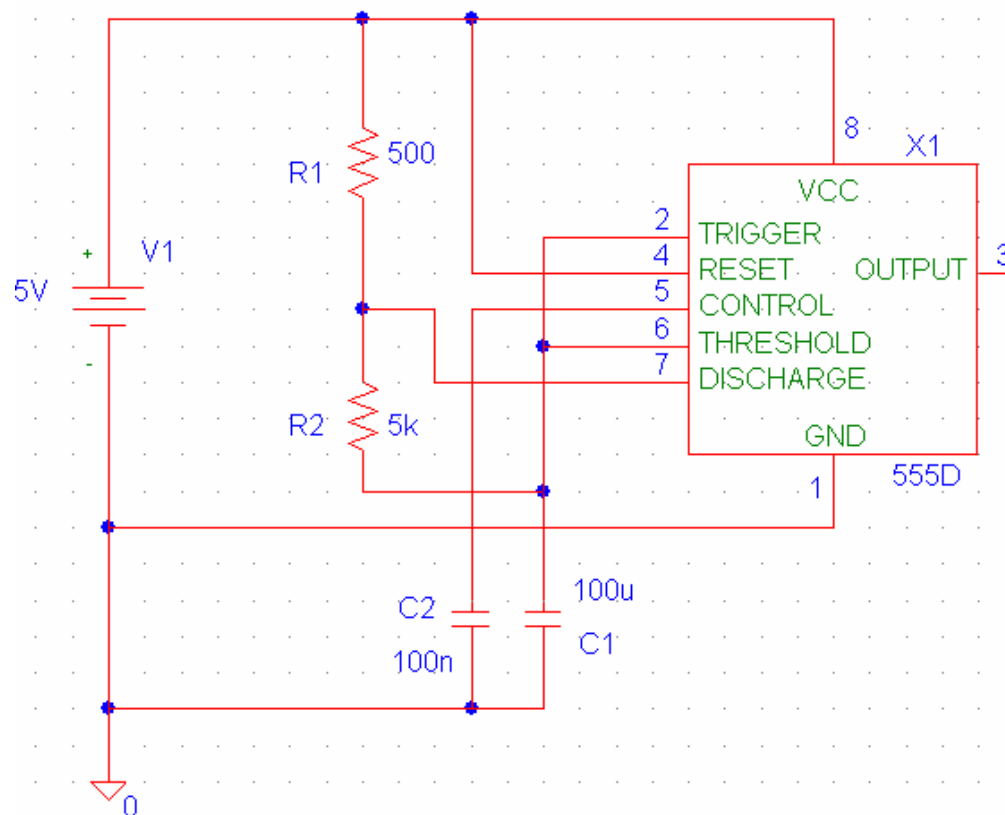
Figure 5.16: A possible astable multivibrator implementation

$T_{ON} = 0.693 \ (R_1 + R_2) \ C_1$
$T_{OFF} = 0.693 \ R_2 \ C_1$

Hence $T = T_{ON} + T_{OFF}$
    $T = 0.693 \ (R_1 + 2R_2) \ C_1$

    $d = T_{ON}/T = (R_1 + R_2)/ \ (R_1 + 2R_2)$

59

$$d = \frac{100}{1 + R_2/(R_1 + R_2)} \%$$

$T = 0.693 \ (10^3 + 2 \times 10^3) \ 100 \times 10^{-6} = .2079$ sec or approximately 0.21 sec.

In this case, since $R_1 = R_2$, the duty cycle has to be 2/3 or 66.67%.

Note that because $R_1 > 0$, the duty cycle can never be less than 50% with this configuration. However, if $R_2 \gg R_1$ ($\gg$ means much greater than), then we can approach d = 50%, as shown below where $R_2 = 10R_1$, and $R_2 = 10$ kΩ. In this case the period is

$$T = 0.693 \ (10^3 + 2 \times 10 \times 10^3) \ 100 \times 10^{-6} = 1.46 \text{ sec}$$

The display would read approximately 1.45 sec.
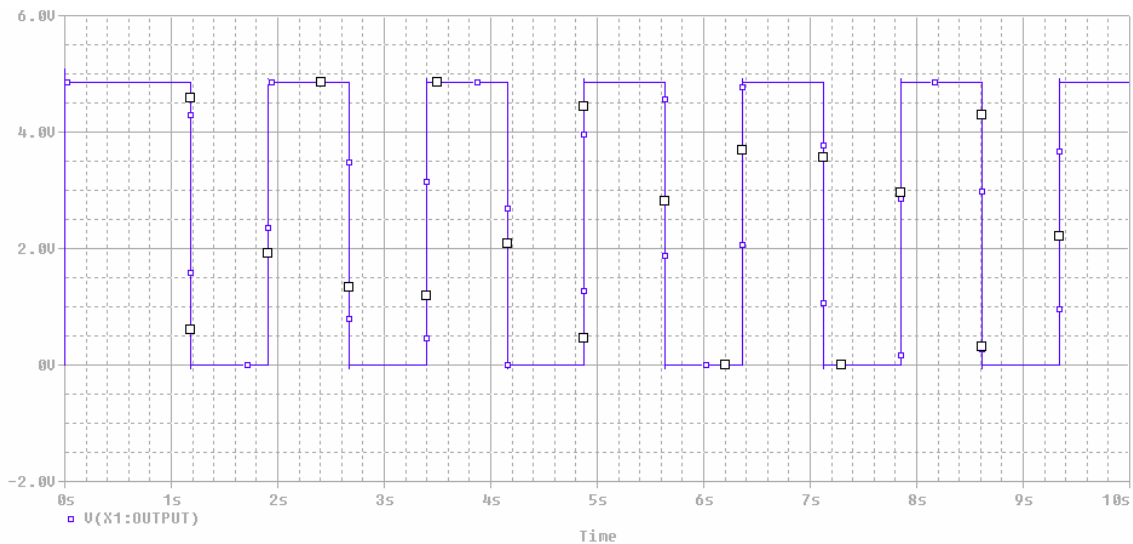
Figure 5.17: Pulse Train with $R_2 \gg R_1$ (with $R_2 = 10R_1 = 10$ kΩ)

However, other configurations will exhibit the possibility of a duty cycle covering all possible practical values (from 0 to 100%). An example of a circuit performing that task is shown in appendix B, with displays of 2 choices: 70% and 30%.

5.14.4   4-bit Binary Counter

The 4-bit binary counter that was selected for implementation is the 74LS163. We have presented a portion of the information that the manufacturer (in this case Motorola) would be making available. You will be able to understand most of the features and design parameters by the time you have passed the ECE 251 course.

At this level, this chip is capable of counting from 0 to 15 in binary format (from 0000 to 1111), and repeats itself thereafter.

The astable multivibrator that was introduced in Figure 5.16 acts as a clock for the 4-bit binary counter with $R_2 = 10R_1 = 5$ kΩ

60
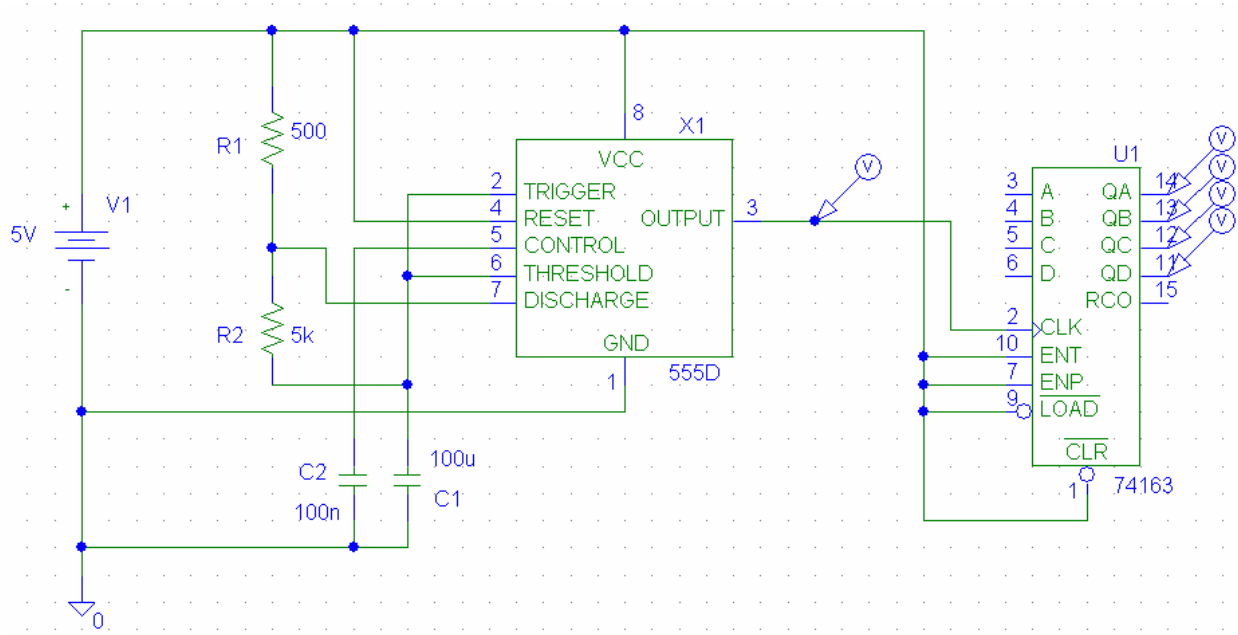
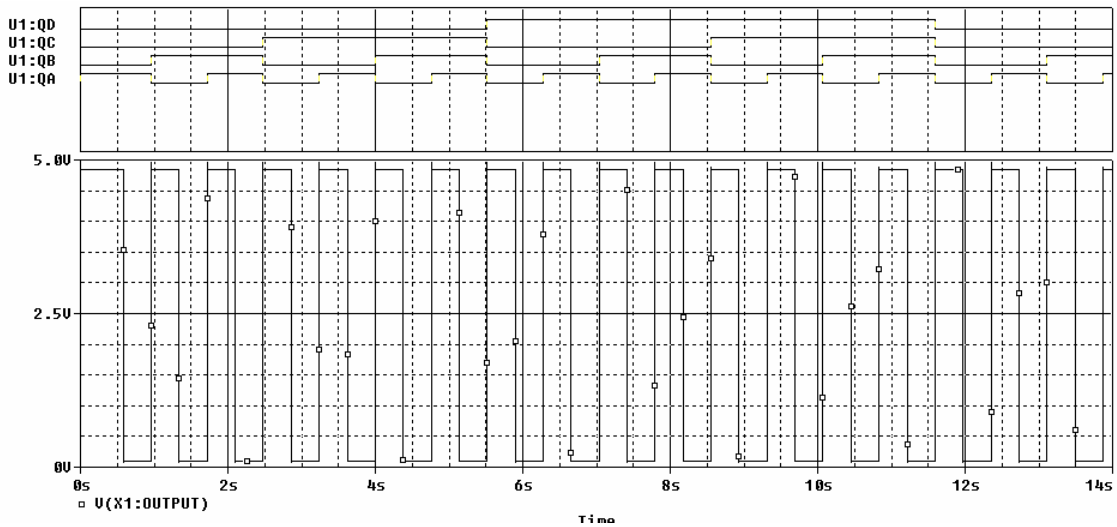Figure 5.18: 4-bit Binary Counter based on the 74LS163

Figure 5.19: Timing Diagram for the 4-bit Binary Counter based on the 74LS163

5.14.5   LED Protection

An LED is a diode that turns ON when a current flows through it (in this case, turning ON means lighting up). As all devices, and most particularly semiconductor devices, they are very sensitive to current limitations. If you exceed the maximum current that the manufacturers specify, you

will destroy those components. Hence an LED or a seven-segment display is to always be connected in series with a resistor (usually 100 to 500 Ω, within the available standard values); the higher the resistance, the dimmer the display.

For this LED configuration, since the cathode is connected to ground (0 potential), a zero potential at the anode will keep the LED off, and a higher potential, for example 5 V, will turn on the LED. The higher the potential difference, the higher the current flowing into the LED will be generated, and the brighter the display. Keep in mind that if you exceed the maximum allowable current, the LED will be turned off permanently.

When the anode is already connected to a higher potential, in this case 5 V, the way to turn on the LED is to apply a lower potential to the cathode, 0 V for example.

62

6 **Transistors and Diodes**

Transistors and diodes are wonderful devices. They are the basic elements of modern electronic circuits that made possible the rapid development of computer and information technologies we experience today. The TRANSISTOR was invented in 1947 at Bell Laboratories (AT&T), a mere 15 miles from NJIT. Initially, its importance was not fully appreciated, but years later the impact of this invention was recognized by awarding its authors (John Bardeen, Walter Brattain, and William Shockley) a Nobel Price. The full power of the transistor was unleashed with the invention of the INTEGRATED CIRCUIT (Texas Instruments, in 1958), which combines many transistors (a few initially, millions today) in one integrated device. It is this device that literally changed the way we work and live. The inventor of the integrated circuit, Jack Kilby, has been also recognized with the Nobel Price.

Transistors and simpler elements, DIODES, belongs to the class of so called SOLID STATE devices, because they are made of a solid semiconductor material, most of them crystalline silicon (earlier electronics used vacuum tubes). There are two types of semiconductors, n-type and p-type. These symbols stand for negative (n) or positive (p) charges (electrons and holes) that carry electric current in these materials. Electronic devices are made by combining p and n semiconductors We will first examine the simpler device, the diode before moving to more complex transistors.

6.1 *Diodes*

Diodes, like resistors, have two terminals for connecting to electrical circuit. Unlike resistors, however, diodes are so called NONLINEAR ELEMENTS. It means the current through them is not proportional to the voltage applied across them; they do not obey Ohm's law. Moreover, they have polarity. That is, their terminals can be assigned positive and negative signs since they are connected to p – type and n – type semiconductors, as shown schematically in Figure 6.1.
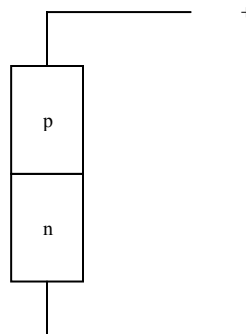

Figure 6.1: A diode as a junction of p – type and n – type semiconductors

When the positive terminal of the diode is connected to the positive terminal of a power source, a current flows with very low resistance. This is called the forward bias of a diode. If the diode is connected in reverse, practically no current flows. The diode acts like a valve, passing the current in one direction. You may recall a different device, also called a valve, found

on bicycle or car tires. The valve there passes the air only in one direction allowing you to pump a tire, while it can not escape in the reverse direction.

The curve representing the relationship between the current and the voltage across a typical diode is shown in Figure 6.2. With the values of current on the vertical axis and voltage on the horizontal scale it is called the I-V characteristic on the I-V curve. The current increases exponentially for the positive voltage values and the graph really shoots up when the voltage exceeds about 0.7 V. For the negative voltage value the graph shows zero current, but only because of the scale on the vertical axis. Expanding this scale would reveal that there is in fact some current for the reverse bias, the so-called reverse current. Only an ideal diode has zero reverse current but in real diodes this current is usually orders of magnitude smaller than the forward current.
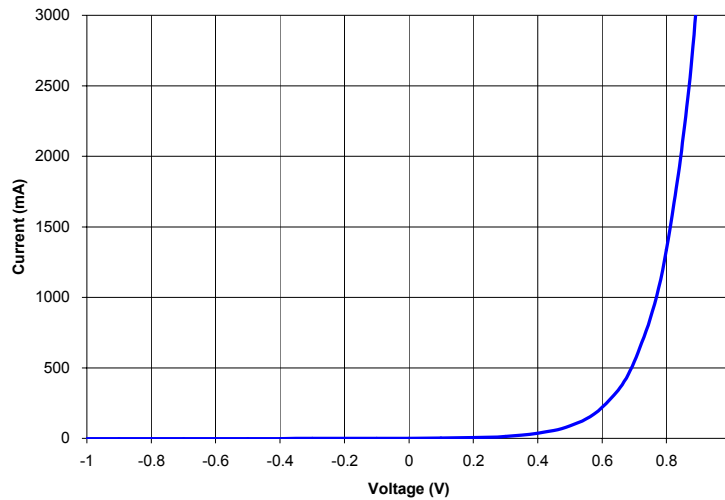
Figure 6.2: The characteristic of a typical diode (I-V curve)

The diodes are very useful in rectifiers, circuits that change AC to DC power. They are used in all electronic devices, such as desktop PCs, which require DC to operate but are supplied by the household current (AC). Battery operated devices, such as quartz watches run on batteries and do not need rectifiers but laptops still need them to recharge their batteries. Diodes can be also used in digital devices to form basic logic gates, as we will see in the laboratory experiments. A special type of diode, the light emitting diode or LED, is used in light communication and display panels. The symbols representing diodes in electric circuit schematics are shown in Figure 6.3. The forward current direction is from left to right. The arrows in the LED symbol represent emitted light.

Figure 6.3: Schematic representation of a diode (left) and an LED (right).

6.2 Transistors

There are two most common transistor types today: the Metal-Oxide-Semiconductor or MOS and the Bipolar Junction Transistor or BJT. The MOS is also designated as MOSFET because it is a field effect transistor (FET). A great majority of both types are made from silicon (Si) and a small fraction (about 2%) from gallium arsenide (GaAs). Depending on the type of the semiconductor materials used in making transistors (n-type or p-type semiconductors) there are n-type or p-type MOSFETs or n-p-n and p-n-p BJTs. The BJT dominated the market initially but now most of the transistors, particularly in integrated circuits, are of the MOS type. The BJT still holds its own, particularly in some analog and high power circuits.

The big difference between transistors and diodes is that transistors have three terminals, not two. The additional terminal controls the current flowing through the other two. This creates important functional possibilities, namely, one circuit can control (or switch on or off) the current flowing in another circuit. Another important feature of the transistor is that the controlling current can be much smaller than the current that is controlled. This leads to amplification, a very useful effect in devices such as stereo amplifiers or the control of the speed of electric motors.
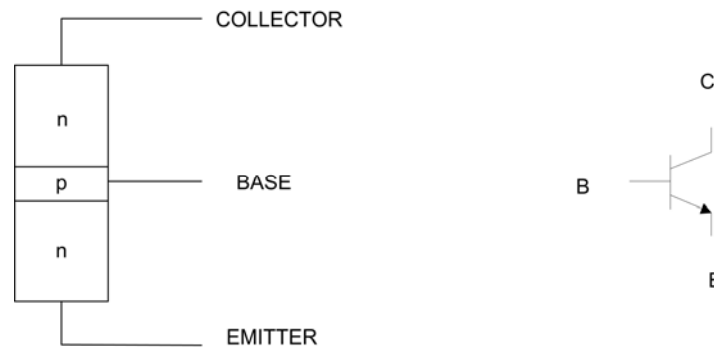
Figure 6.4: The structure of an n-p-n BJT (left) and its representation
in electrical schematics (right)

The structure and the schematics of connections to a BJT are shown in the Figure 6.4. An n-p-n type transistor consists of two n-type regions (the emitter and the collector) with a thin p – type layer (the base) between them. Note that the transistor consists of two p-n junctions connected by the common p-type base. In normal operation the collector is positive and the current flows between the collector (C) and the emitter (E) while the base (B) is the control electrode. There are also less often used p-n-p transistors with opposite types of semiconductor layers. In MOS transistors the electrodes have different designation: source instead of emitter, gate instead of base and drain instead of collector. There are also n-type and p-type MOSFETs. In the n-type, the current flows between source (S) and drain (D), while gate (G) is the control electrode.

Most transistors of any type are made today as elements of integrated circuits (ICs), which can contain many millions of circuit elements. Such a combination of many interrelated circuits can result in an extremely complex and powerful system such as the Pentium microprocessor. Single or discrete transistors, however, are still useful in many applications such as high frequency or power units.

Experiment 1 – Resistors and 1-Resistor Circuits

In this experiment you will examine resistors and build and analyze a simple resistor circuit. You will also become acquainted with the digital multimeter that you will be using throughout this course.

PARTS LIST:

- Protoboard with built-in power supply
- Digital multimeter
- Various resistors and wires

Background - Protoboards

To perform this experiment, students will need to become familiar with the protoboard and logic probe they will use. Both are described in this section.

We can construct a circuit using digital logic chips, but we need a surface to secure the chips and wires that comprise the circuit. This is the function of the protoboard. Chips and wires are plugged into the protoboard and can be rearranged easily as the design is modified. (Final circuits are wired on breadboards. Protoboards are typically used during the design of prototypes, hence its name.) A typical protoboard is shown in Figure L1.1.

Figure L1.1: A typical protoboard

The terminals at the top of the protoboard are used to connect the power supply (Vcc and Gnd) to the circuit. One pair of cables connects the power supply to these terminals, and wires connect the terminals to the protoboards.

The main protoboard surface consists of a grid of holes. The pins of the chips in a circuit are plugged into these holes; for reasons that will be explored shortly, the chips must straddle the relatively large channels, or gutters, separating columns of holes. The chips cannot be placed so that they are in the leftmost nor rightmost two columns. These special columns are reserved for Vcc and Gnd.

The protoboard has connections embedded within its structure so that certain holes are always connected to each other. As a result, wires and pins plugged in to connected holes are also connected. This greatly simplifies the task of wiring up circuits. Any hole is connected to all other holes in the same row between the same two channels; this is illustrated in Figure L1.2. When dealing with integrated circuit chips later in the course, you will see that this is the reason that chips must straddle the gutters, to avoid inadvertently connecting pins on the same chip. The power columns are also connected, but in a different way. Here, holes are connected to

66

other holes in the same column. Designers can connect wires from the two terminals connected to Vcc and Gnd to these columns, and then connect wires from these columns to the power pins of each chip. Note that the power strips often stop midway down the protoboard; watch out for this when wiring up your circuits for this and future experiments.
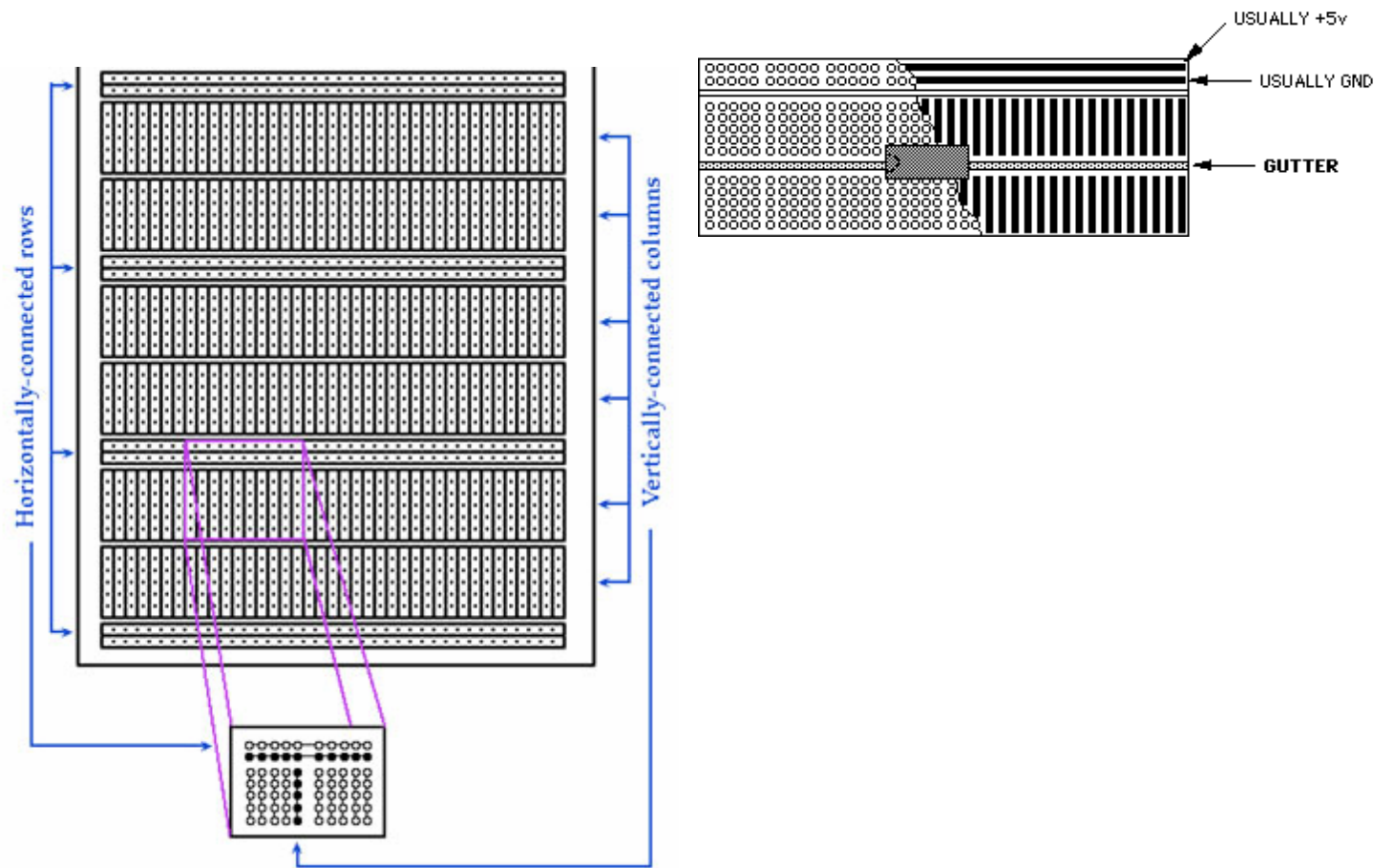

Figure L1.2: Protoboard internal connections. Connect two top horizontal rows(detail shown on the right) to the positive (red) and negative (black) power terminals.

Part I – Resistor Measurements

Using the multimeter, measure the resistance of ten different resistors. For each resistor, record its nominal value (the value indicated by its color bands) and its measured value. Is each resistor within its tolerance specification?

Part 2 – 1-Resistor Circuit

Wire up the circuit shown below. Using the multimeter, measure the voltage across the 100 Ω resistor. Also measure the current flowing through the resistor. Using Ohm's Law, verify that your measurements are valid for this circuit.

1.5V

DC

+

−

100Ω
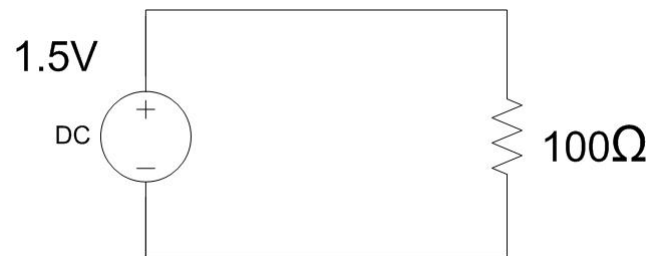
Figure L1.3: 1-Resistor circuit

Experiment 2 – Series and Parallel Resistance

In this experiment you will examine several resistor circuits. Using an ammeter, you will measure the current in the circuits to determine their overall resistances. You will also determine values for unknown resistances in several pre-wired circuits.

PARTS LIST:

- Protoboard with built-in power supply
- Digital multimeter
- Various 100Ω, 330Ω, and 1KΩ resistors and wires
- Black tape to cover "unknown" resistors

Part I – Known Circuits

In this part of the experiment, you will examine wire and test two known circuits. The schematics for these circuits are shown in Figure L2.1.
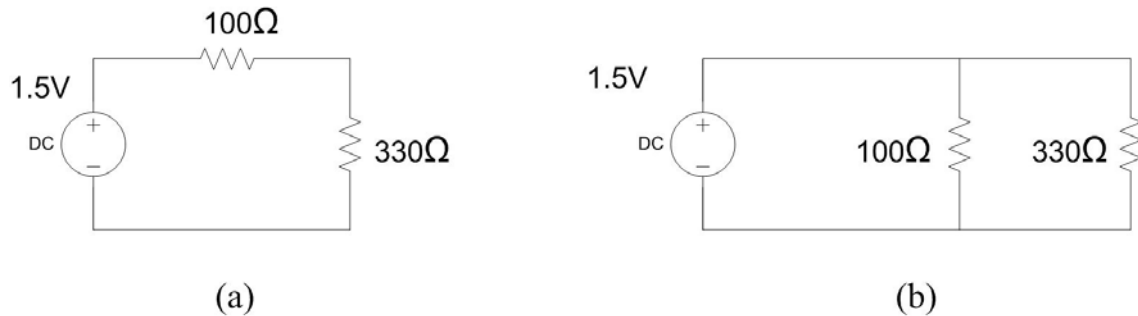
Figure L2.1: Circuits to be tested

After you have wired up and powered up each circuit, connect the leads of the ammeter to measure the current flowing through the entire circuit. Record the current reading from the ammeter on the laboratory worksheet, and then use Ohm's Law to calculate the overall resistance of each circuit. Also measure the voltage drop across each resistor and note these values in your laboratory notebook. Verify Kirchoff's Laws using your measurements.

Part II – Unknown Circuits

This part uses the same circuits as the first part of this laboratory exercise, but with different resistances. You will use one resistor in each circuit that is covered with black electrical tape so you can't see its value. Proceed as in Part I, reading the current and determining the overall resistance in the circuit. Then use the rules for combining series and parallel resistors to determine the value of the unknown resistor.

Experiment 3 – Simulation Tools

In this experiment, students will use PSpice and Matlab to simulate circuits previously analyzed in this class.

PARTS LIST:

- PC running PSpice and Matlab software

Part I – 1-Resistor Circuit

Using PSpice, "construct" the 1-resistor circuit you built and analyzed in Experiment 1. The schematic for this circuit is repeated below in Figure L3.1. Simulate the functioning of the circuit and "measure" the current at all points in the circuit and the voltage drop across the resistor. How do your simulation results compare to those you measured in Experiment 1? Repeat this procedure using Matlab.
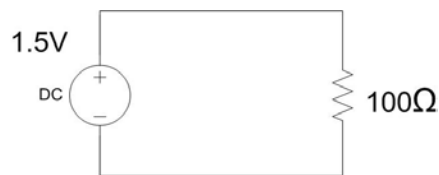
Figure L3.1: 1-resistor circuit

Part 2 – Series and Parallel Resistor Circuits

Repeat the previous procedure on the series and parallel resistor circuits shown below. These are the same circuits you built and tested in Experiment 2. Again measure the current through and voltage across each resistor. How close are the results of this simulation to the values you measured in Experiment 2? Repeat this procedure using Matlab.

(a)   (b)
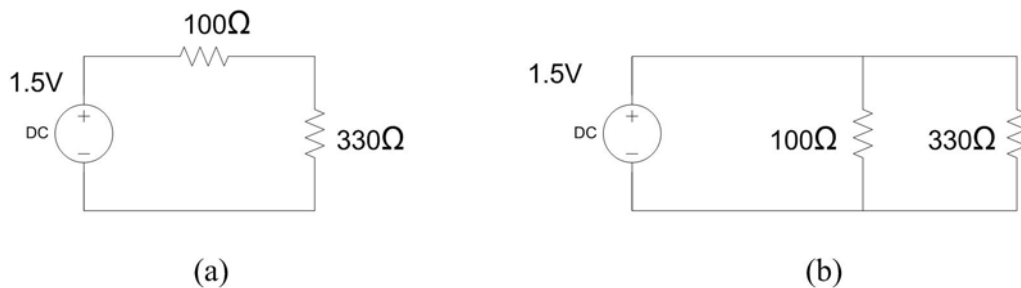
Figure L3.2: (a) Series and (b) parallel resistor circuits

70

Experiment 4 – Digital Logic Fundamentals

In this laboratory exercise, students will test several logic chips to verify their functions. Students will also test an "unknown" chip to determine the Boolean function it realizes.

PARTS LIST:

- Protoboard with built-in power supply
- Digital multimeter
- Logic probe
- 7408 (AND gates), 7432 (OR gates), and 7486 (XOR gates) chips and wires
- One additional TTL chip, either 7400, 7402, 7404, 7408, 7432, or 7486
- Black tape to cover one "unknown" chip

Background – Logic Probe

To perform this experiment, students will need to become familiar with the logic probe they will use. This is described in this section.

We will use a logic probe to test the functions of our chips. First we must connect the logic probe's Vcc and Gnd clips so the probe has power. We can clip them directly to the terminals of the protoboard. Then we touch the tip of the logic probe to the pin we wish to test. The logic probe has two LEDs that will be used in this experiment. When the probe detects a 0, its "Low" LED lights up; the "High" LED lights when a logic 1 is detected. We can use the logic probe to check the input and output values of a chip. Figure L4.1 shows a typical logic probe.

Figure L4.1: Logic probe

To test our chips, we will need to input values to the gates. To input a 1, we will connect a wire between the chip's input and Vcc. We can input a 0 by connecting the wire from the chip's input to Gnd.

Part 1 – Testing Known Chips

In the first part of this exercise, students will test several logic chips. Each group of students has a protoboard and three known chips: a 7408 (AND gates), 7432 (OR gates), and 7486 (XOR gates). Place the chips into the protoboard, ensuring that each chip straddles one of the channels. This ensures that there is no direct connection between the leads on the two sides of the chips. Connect Vcc and Gnd for each chip.

Starting with the AND gate, connect both inputs to Gnd, producing a logic 0 for each input. Use the logic probe to verify these input values, and then check the output values for the gate. Record your output values on the worksheet for this exercise.

Next, change one of the inputs to Vcc, changing it to logic 1. Again verify your input values and determine your output value using the logic probe, and record your output value on the worksheet. Repeat this for all possible combinations of the input values, and then verify that your values match those in the truth table for the AND function.

Repeat this procedure for the OR and XOR gates.

Part 2 – Testing an Unknown Chip

The protoboard for this exercise also contains a chip which has tape covering its surface. Here you will determine its function without removing the tape.

To do this, follow the same procedure you used in the first part of this exercise. Vary the values of the inputs to the chip and record its outputs on the worksheet. Then compare the truth table for your unknown chip to those given earlier in this chapter and find the table that matches the unknown chip's table. What is the function of your unknown chip?

Experiment 4 Worksheet – Digital Logic Chips

AND gate

A	B	Output
0	0	
0	1	
1	0	
1	1	

OR gate

A	B	Output
0	0	
0	1	
1	0	
1	1	

XOR gate

A	B	Output
0	0	
0	1	
1	0	
1	1	

Unknown gate

A	B	Output
0	0	
0	1	
1	0	
1	1	

Gate type: _____

Experiment 5 – Guess the Pattern Game

In this experiment, we will design a "guess the pattern" game. One player will set three inputs, *A*, *B*, and *C*, to any combination of binary values. This player does not show the pattern to the other player. The other player sets three other inputs, *D*, *E*, and *F*, to some combination of binary values. If *A,B,C = D,E,F*, that is, *A = D*, *B = E*, and *C = F*, then the game lights a single light, an LED in this case. If not, the LED is not lit.

PARTS LIST:

- Protoboard with built-in power supply
- Digital multimeter
- One 7486 chip and one 7408 chip and wires
- One LED and 330Ω resistor

To design this game, we first design a circuit to check whether or not *A* and *D* are equal. First complete the truth table in Table L5.1(a). Enter all possible combinations of values for *A* and *D*; then enter an output value of 1 if *A* and *D* are the same, or 0 if they are not. Next, determine a function for this truth table. Repeat this to develop functions that check if *B* and *E* are equal, and if *C* and *F* are identical, using Tables L5.1(b) and (c).

Table L5.1: Truth tables for the three comparisons (a) *A = D*, (b) *B = E*, and (c) *C = F*

A	D	(A = D)		B	E	(B = E)		C	F	(C = F)

        (a)   (b)   (c)

Now that you have an equation for each of our three comparisons, the next task is to design three logic circuits to generate the functions *A = D*, *B = E*, and *C = F*. You should be able to generate each of these functions using either:

- two AND gates, two NOT gates, and one OR gate, or
- one XOR gate and one NOT gate.

(You could also use one XNOR gate, but we won't do that here. The 74266 chip is a special type of chip that uses *open collector* outputs. We could use that chip, but we would have to connect additional resistors to make them work with our other chips. For now, consider it sufficient that we will use only regular, non-open collector chips to simplify our design.)

The design of the three circuits should be identical, except for their different input and output values, since each compares two different inputs to see if they are equal.

After designing and building these three circuits, test them using a logic probe. Vary the inputs to check every combination in the truth tables. Verify that each circuit correctly outputs a 1 whenever and only when it inputs are the same.
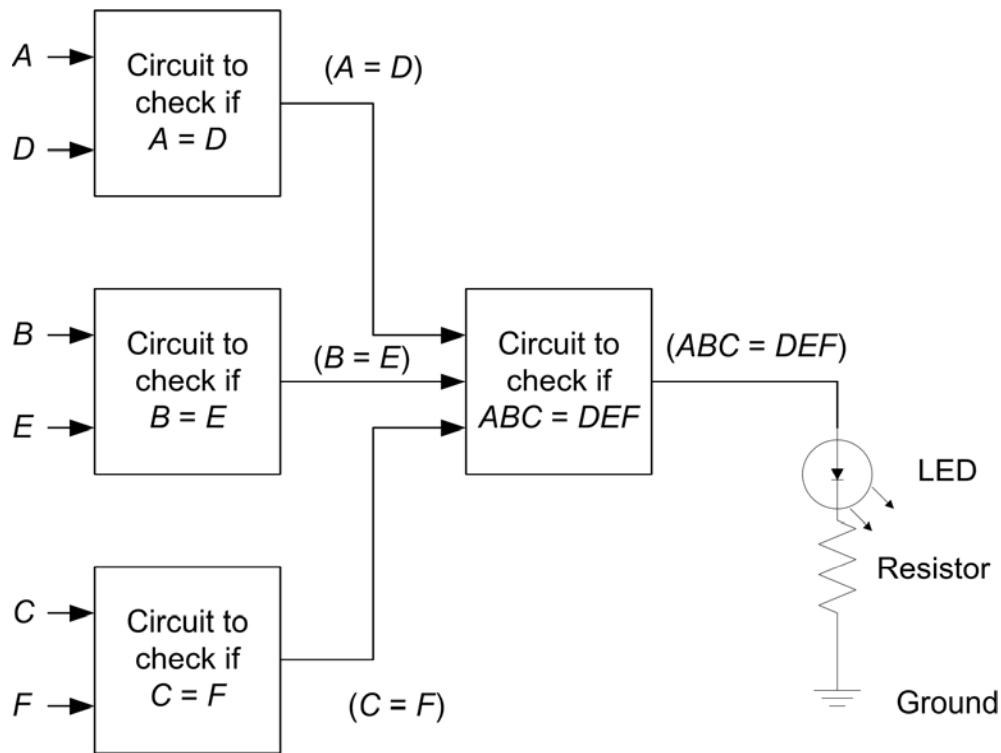

Now we must take the three circuit outputs and combine them, outputting a 1 only when $(A = D)$, $(B = E)$, and $(C = F)$ are all 1. First complete the truth table in Table L5.2. The inputs $(A = D)$, $(B = E)$, and $(C = F)$ would be generated by the circuits we just designed. The truth table should have output $(ABC = DEF)$ equal to 1 only when all three inputs are 1.

Table L5.2: Truth table for the overall function $(ABC = DEF)$

$(A = D)$	$(B = E)$	$(C = F)$	$(ABC = DEF)$

Next we design a circuit to take these three inputs and output a 1 only when all three are 1. This is a relatively straightforward design. After designing this circuit, test all possible combinations of input values with a logic probe to verify that it generates an output of 1 only when all three inputs are 1.

Now connect the sections of your circuits together as shown in Figure L5.1. The first three circuits check whether $A = D$, $B = E$, and $C = F$. The outputs of these circuits are input to the last circuit, which checks whether all three pairs of bits are the same. It outputs a 1 if the player has guessed the pre-set pattern, or 0 otherwise.

Figure L5.1: Block diagram of the "guess the pattern" game

Finally, we connect the (*ABC* = *DEF*) output to the LED, as shown in Figure L5.1. Note that the direction of the LED is important. The LED has two different length leads. The longer lead should be connected to the circuit that checks if ABC = DEF and the shorter lead should be connected to the resistor. Notice the resistor in this circuit. This is needed to keep the LED from receiving too much current and burning out.

76

Experiment 6 – Boardwalk Wheel

The Boardwalk wheel is a game of chance. It is a wheel with a pointer at its center; the outside of the wheel is divided into many different spaces. Players place their bets on which space the pointer will point to when it is done, and the pointer of the wheel is spun. It eventually slows down and comes to a stop. Whatever space the pointer points to when it stops is the winner. Here in New Jersey, such wheels are common in the amusement areas of boardwalks near the beaches, but they are also found at carnivals and amusement areas throughout the world.

In this experiment, we will create an electronic version of the Boardwalk wheel. There will be only eight spaces on our wheel. Instead of using a pointer, each space will have a single LED.

PARTS LIST:

- Protoboard with built-in power supply
- Digital multimeter
- 4-bit counter
- One chip described earlier in this chapter (to be selected by the student)
- Eight LEDs and 330Ω resistors and wires
- One 555 timer chip and components to generate a 1-second clock signal

To design this wheel, we must create a circuit that lights one LED, then the next, and so on, until the "wheel" eventually stops with one LED lit. This design can be created using only two chips: a counter and one other chip.

A counter can generate a sequence of outputs that varies from 0000 to 1111. If we only concern ourselves with the three low-order bits, it will generate the sequence 000, 001, 010, 011, 100, 101, 110, 111, and then start the sequence over again. To do this, we need to generate a clock input to the counter. This can be done using the 555 timer; see the appendix of this manual for information on how the 555 timer can be configured to generate clock pulses.

This sequence contains one value for each of the eight LEDs. Now we must use one of the components described earlier in this chapter to convert the three-bit value into eight separate signals. One signal will be active when the inputs are 000, another is active when they are 001, and so on. These eight signals will each light one LED when active. What component can perform this task?

Design and wire up your circuit. Connect it to the pre-wired LED "wheel" and verify that it works properly.

Experiment 7 – Transistors and Diodes

In this laboratory exercises, students will examine operation of diodes and use them in constructing simple logic gates. Operation of BJT and MOSFET will also be demonstrated.

PARTS LIST:

- Protoboard with built-in power supply
- Digital multimeter
- Light bulb
- Resistors and a potentiometer
- Rectifier diode, LEDs, BJT (npn) and n-type MOSFET.

Part 1 – Diodes - forward and reverse current

Set the voltage on one of the 15 V terminal to 6 V. Connect a light bulb a diode in series and connect them to the power supply terminal through an am-meter. Measure the current with the diode connected in the forward direction (the bulb lights up) and the reverse directions (the lamp is dark). To change the diode polarity, simply exchange the diode terminals on the protoboard (flip the diode position). What is the ratio of the reverse to the forward current? In an ideal diode it should be zero.

Repeat the experiment using an LED in series with a 1KΩ resistor. The resistor is necessary to limit the current, preventing the diode from burning out. What polarity is needed for emitting light, forward or reverse?

Part 2 – Logic gates with diodes

Assemble and test the two circuits shown in schematics a and b in Figure L7.1. The logic value YES (1) is represented by a voltage of 5 V and NO (0) by 0 V (ground). Make sure that the two inputs, A and B are either grounded or connected to the 5 V terminal. Determine what type logic gates are represented by these two circuits. Can you explain how they work?
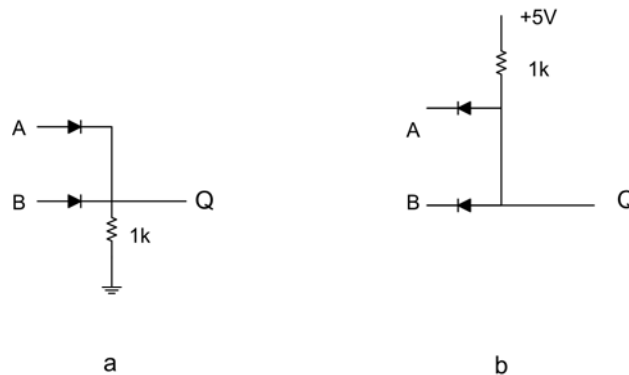
Figure L7.1 Logic gates using diodes

78

Part 3 – Operation of a BJT transistor

In this laboratory exercise we will examine both switching operation and amplification of transistors. Schematics of connections to MOS and BJT transistors are shown in Figure L7.2.
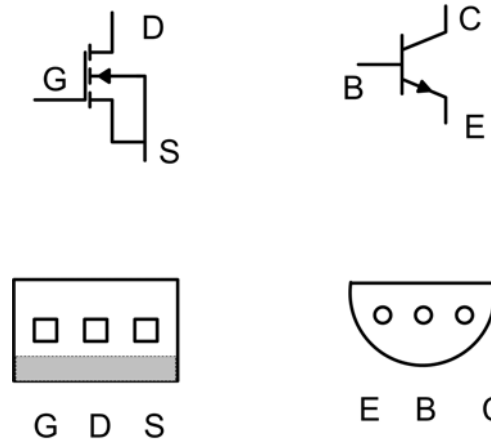
Figure L7.2: Schematics (top) and terminal connections (bottom)
of MOS (left) and BJT (right) transistors

Assemble a circuit (shown below) in which an n-p-n BJT is used to switch on and off a small incandescent lamp. Resistor R_1 should be a few k ohms. Switching a switch S should turn on or off the lamp.
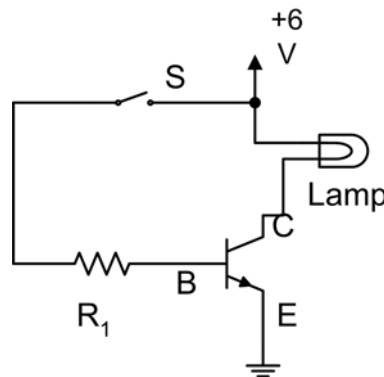
Figure L7.3: A circuit switching current to a lamp with a BJT

In the BJT, the current flows between the collector (C), which is positive, and the emitter (E) while the base (B) is the control electrode. Note that the schematic really represents two circuits (or circuit loops) connected by a transistor. One of them, the base circuit, consists of the power supply, the switch (S), the resistor R_1, and the base and the emitter of the transistor, which is connected to ground closing the loop to the power supply. The other circuit uses the same 6V power supply, connected to the lamp and through the collector (C) and emitter (E) of the BJT, closing the loop to ground. The switch turns on or off only the current to the base (in the first

79

circuit) but the current through the lamp (in the second loop) responds to the condition of the first circuit.

After successful demonstration of the circuit, measure and compare currents in the two loops (the base current and the collector current. How many times greater is the collector current then the base current? You should be able to demonstrate GAIN, a great asset of the transistor. Big currents (for example, motors of an electric train) can be adjusted by small currents from electronic control circuits.

A simple demonstration of such control can be accomplished using our lamp and a potentiometer in the circuit shown in Figure L7.4 below. The potentiometer (P) in the base circuit controls the light output from the lamp in the collector circuit. Resistor R_2 (330 ohm) protects the base circuit from excessive current when the potentiometer is turned all the way up.
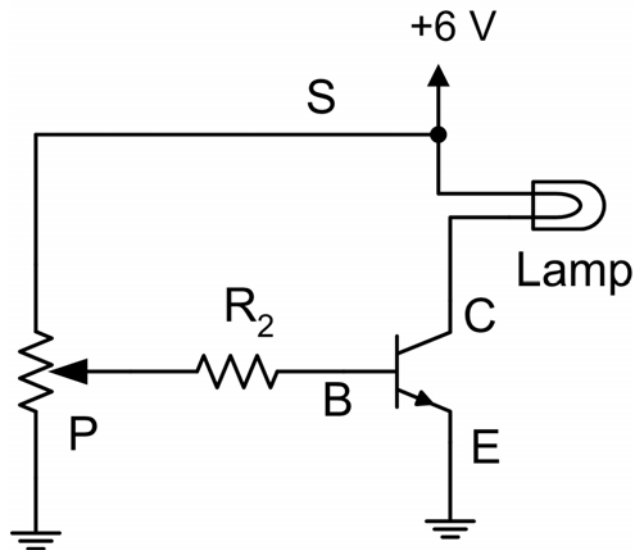
Figure L7.4: A light dimmer circuit with a potentiometer and a BJT

Part 4 – MOSFETs

In the circuit represented in Figure L7.3, replace the BJT with an n-channel enhancement mode MOSFET: the gate terminal replaces the base, the source replaces the emitter and the drain the collector, so that the drain is positive (see Figure L7.2). You do not need a switch and the resistor R_1, which greatly simplifies the circuit. Turn the lamp "on" and "off" by connecting the wire from the transistor gate either to the positive power supply terminal or to the ground.

Now a surprise: touch the end of the wire from the MOSFET gate terminal with one hand and touch your other hand to the ground and then to the positive terminal. This demonstrates a big advantage of MOSFET. Do you need much current to turn the MOSFET "on"?

Design Project: The Traffic Light Controller

The design project is an affirmation and a confirmation of your abilities as a designer and a creative engineer. Given a set of specifications that you or somebody else will set, you are to design a device that will satisfy them. You are to work as a team. The whole team will fail or succeed. Every team has a leader. A leader knows how to delegate the different tasks involved in achieving the desired results. Each member is responsible for the success or failure of the team. The end result is worth all the pain and frustration that may arise in the process.

Your project is to design a traffic light controller that responds to the following specifications. Given an intersection with EW (East-West) and NS (North-South) 2-way traffic, we want the lights to turn on in the following sequence.

Table 7.1: Traffic Light Sequence

Time	EW	NS
0 - 6	Red	Green
7	Red	Yellow
8 - 14	Green	Red
15	Yellow	Red

The sequence lasts a total of 16 seconds. The numbers are chosen to simplify some of the design and implementation steps. It is obvious that a real practical case will have a longer sequence and also allow more complex modules to be incorporated. The lights are red (stop) for the East-West direction and green (go) for the North-South direction during the first 0 to 6 seconds (a total of 7 seconds). The lights are still red (stop) for the EW direction and yellow (speed up???) for the NS direction during the 8th second (second 7). The table 7.1 describes the remaining steps.

In addition, while the lights are red in any given direction and a light beam is interrupted (equivalent to a car going through a red light), a buzzer will sound (equivalent to a camera taking a picture). Do not worry about the location of the light beam affecting cars making turns. However, there will be two light beams, each one associated with a given direction.
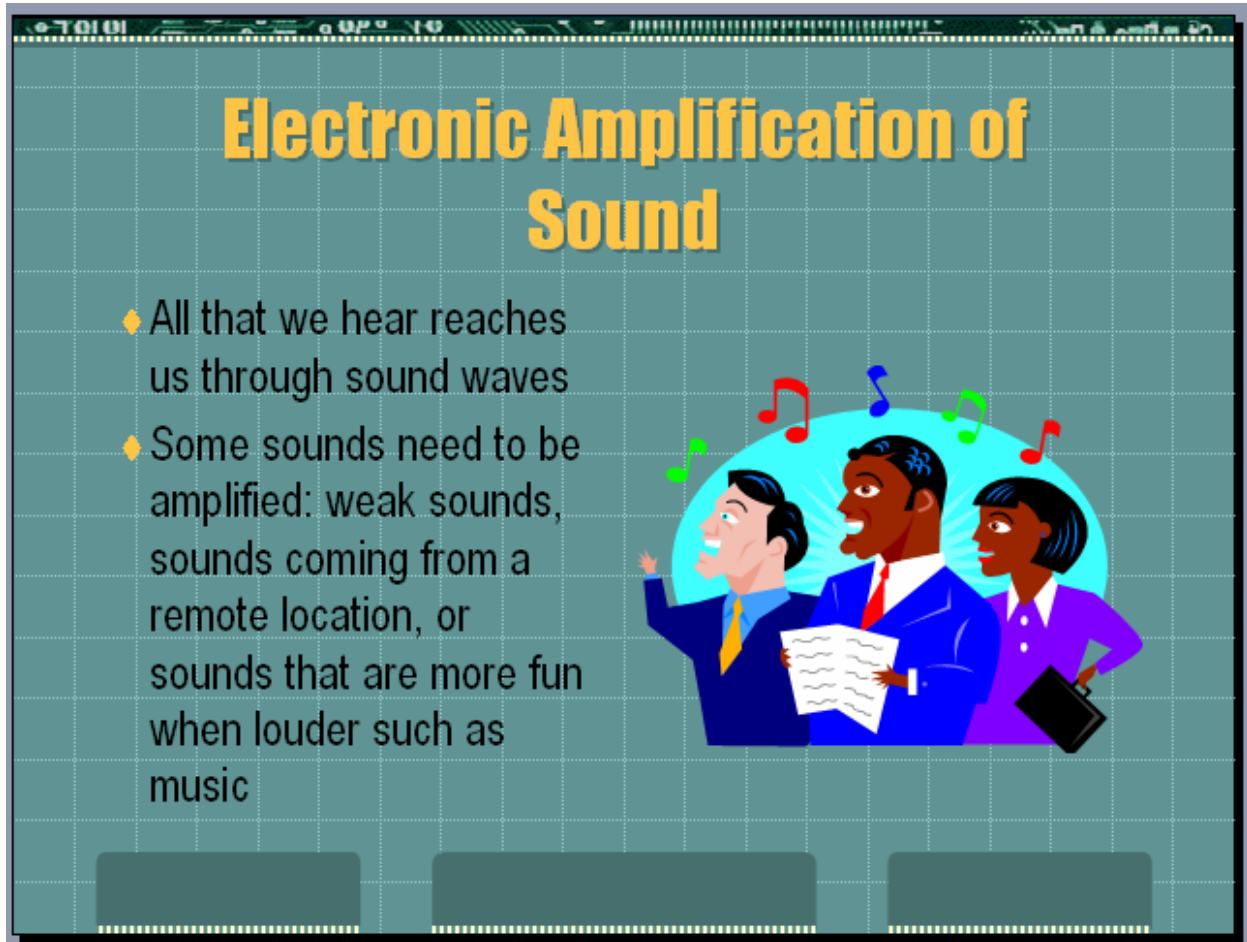
A model platform representing the intersection will be available at the laboratory, represented with LED's of appropriate colors. Your task is to design a circuit providing power from your protoboard to different LED's in an appropriate sequence.

APPENDIX A

Example of a power point presentation

The following power point presentation deals with the electronic amplification of sound. The presentation in this case relies on seven slides, where each slide is followed by a text that you may use verbally to describe the slide in more details. Do not memorize the text, but use your own words as much as you can during the presentation. The text is just a guide and a resource on what to talk about during the presentation of each slide. If you use the original version of the material prepared by somebody else, you have to reference it appropriately, with the text included within quotation marks. For presentation outside the classroom, often obtaining the author's permission is necessary to comply with the copyright laws.

Slide 1

Human beings have always relied on sounds to communicate with each other and nature. They utilize sounds to determine what their course of action would be. The sounds can be used to convey information (applause for praise, screaming for excitement or fear, talking, singing, birds chirping, car horns, etc)

Slide 2

It has become clear that the most efficient and practical way to amplify sounds is through electronic amplifiers. It is then necessary to convert sound waves to electrical signals, and electrical signals back to sound waves after power amplification.

84

Slide 3

An electronic amplifier is a device for increasing the power of a signal. The signal is in our case the electrical entity representing the sound to be amplified.
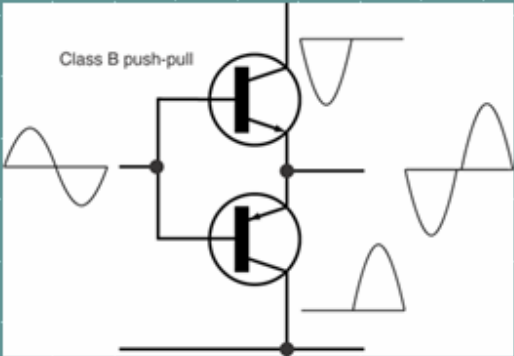
It does this by taking power from a power supply and transferring it to the output signal. The output will ideally be a replica of the input, but possessing a higher power. The process cannot be 100% efficient as some fraction of the power extracted from the power supply will be utilized by the amplifier and wasted in heat. A signal is usually measured in units of power, since a power gain is considered more meaningful than a voltage gain. The ratio is often expressed in decibels (dB). For example, if output=500mW and input=5mW then gain=500/5 = 100 which leads to 10 log 100 = 20dB, where log is a base-10 logarithm.

Amplifiers can be implemented using transistors and various other components. The transistor is a solid state semiconductor device used for amplification and switching, and has three terminals. A small current or voltage applied to one terminal controls the current through the other two, hence the term *trans*istor, a voltage- or current-controlled resistor. It is the key component in all modern electronics.

Slide 4

Each transistor is biased so that it conducts and amplifies half of the signal, minimizing the crossover distortion.

86

Slide 5

The input signal is applied to the input of this amplifier through capacitor C1, which will block any dc value but will allow the ac component to pass. Q1 and Q2 form a differential pair. This arrangement is used to conveniently allow the use of negative feedback, which is fed from the output to Q2 via R7 and R8. Feedback feeds the difference of the input and part of the output back to the input in a way that cancels out part of the input. The main effect is to reduce the overall gain of the system. However the unwanted signals introduced by the amplifier are also fed back. Since they are not part of the original input, they are added to the input in opposite phase, subtracting them from the input.

The amplified signal from Q1 is directly fed to the second stage, Q3, which provides further amplification of the signal, and the dc bias for the output stages, Q4 and Q5. R6 provides the load for Q3. So far, the entire amplifier is operating in Class A. The output pair are arranged in Class AB push-pull, also called a complementary pair. They provide the majority of the current amplification and directly drive the load, connected via dc blocking capacitor C2. The diodes D1 and D2 provide a small amount of constant voltage bias for the output pair, just biasing them into the conducting state so that crossover distortion is minimized.

Slide 6

Dynamic microphones - A dynamic microphone takes advantage of electromagnet effects. When a magnet moves past a wire (or coil of wire), the magnet induces current to flow in the wire. In a dynamic microphone, the diaphragm moves either a magnet or a coil when sound waves hit the diaphragm, and the movement creates a small current.

Dynamic microphones are the simplest and most rugged of all microphones, and do not need any batteries or external power to generate a signal; they generate their own signal with few moving parts. The two types are the moving coil which works like a miniature loudspeaker in reverse, and the ribbon which exhibits low mass similar t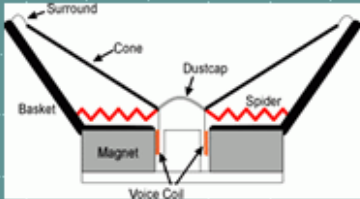o condensers but has lower output voltage. Unlike condensers which have internal amplifiers which limit the loudness of the source sound, dynamics have no problems reproducing loud sounds such as drums or starter pistols. Few dynamics exhibit the ultra flat response of condensers, but most have a response curve tailored to enhance particular sounds, voices, and instruments.

A new kind of dynamic mic uses a new kind of magnetic structure made of Neodymium. These magnets are 4 times as powerful as conventional magnets. The microphones using them can have higher output levels, better linearity and lower magnet mass allowing more compact microphone designs.

88

Slide 7

Dynamic loudspeakers
A Dynamic Cone Loudspeaker
The traditional design is in two parts, a fibrous semi-rigid cone and attached to the apex of the cone is a coil of fine wire called the voice coil or moving coil. When an electrical signal is applied to the coil, it becomes an electromagnet. The magnetic field of the voice coil interacts with the magnetic field of the permanent magnet causing the whole semi-rigid cone (diaphragm) to oscillate and reproduce sound at the frequency of the applied electrical signal. As well as the magnet, the voice coil and the cone, dynamic cone speakers also include a suspension system to provide lateral stability and make the speaker components return to a neutral point after moving.

APPENDIX B

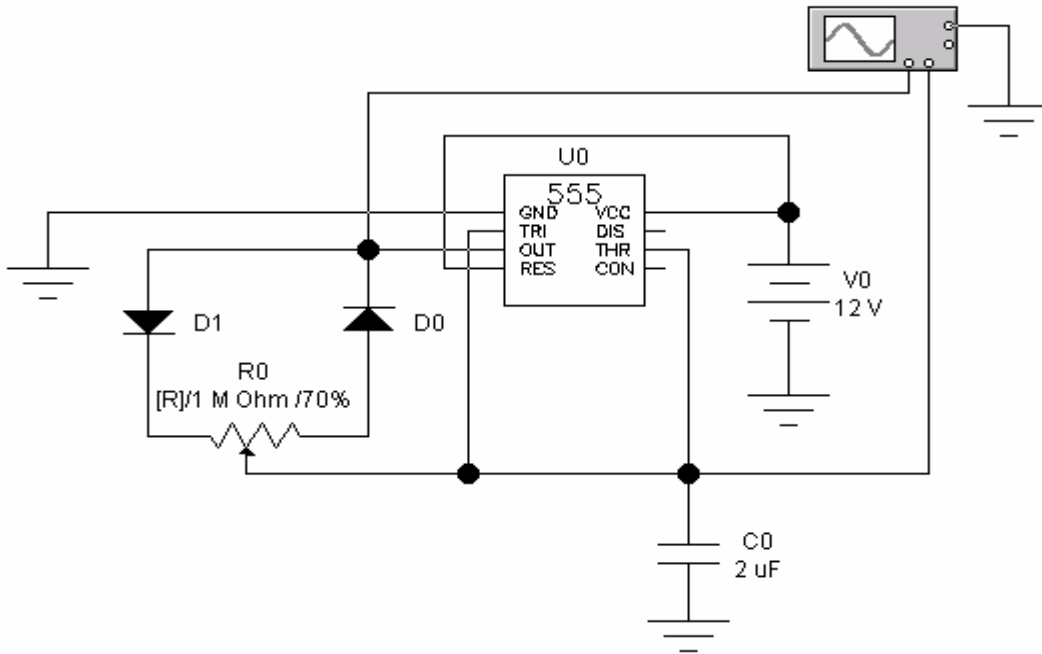**Astable Multivibrator with Varying Duty Cycle**

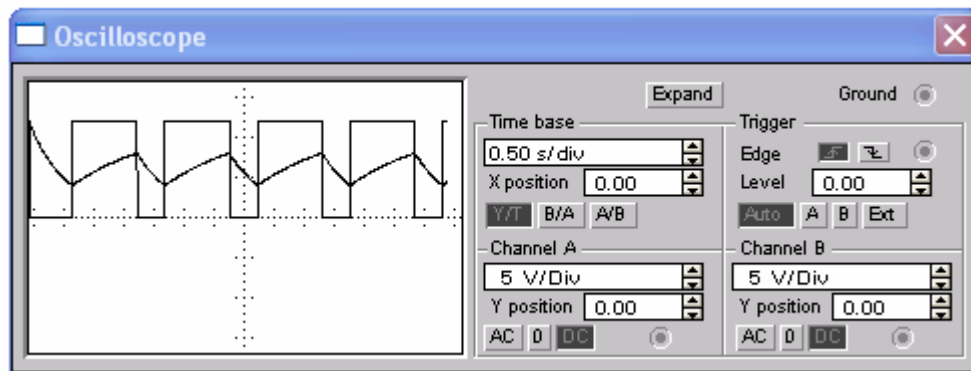Figure B.1: Astable Multivibrator with Varying Duty Cycle

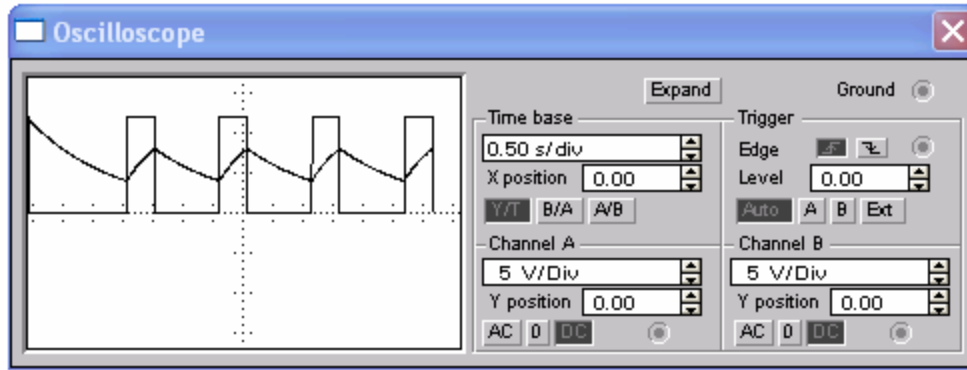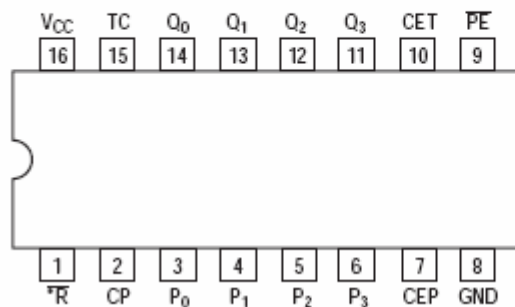Figure B.2: Constant Level Pulse Train with 70% Duty Cycle

90

Figure B.3: Constant Level Pulse Train with 30% Duty Cycle

APPENDIX C
Manufacturer's Specifications for the 74163

SN74LS161A SN74LS163A

CONNECTION DIAGRAM DIP (TOP VIEW)

V_{CC}	TC	Q_0	Q_1	Q_2	Q_3	CET	$\overline{PE}$
16	15	14	13	12	11	10	9

1	2	3	4	5	6	7	8
$^*\overline{R}$	CP	P_0	P_1	P_2	P_3	CEP	GND

NOTE:
The Flatpak version has the same pinouts (Connection Diagram) as the Dual In-Line Package.

*MR for LS161A
*SR for LS163A

PIN NAMES
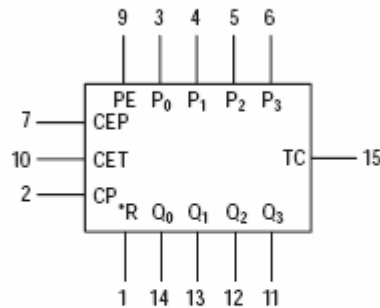
$\overline{PE}$	Parallel Enable (Active LOW) Input
$P_0 - P_3$	Parallel Inputs
CEP	Count Enable Parallel Input
CET	Count Enable Trickle Input
CP	Clock (Active HIGH Going Edge) Input
$\overline{MR}$	Master Reset (Active LOW) Input
$\overline{SR}$	Synchronous Reset (Active LOW) Input
$Q_0 - Q_3$	Parallel Outputs
TC	Terminal Count Output

NOTES:
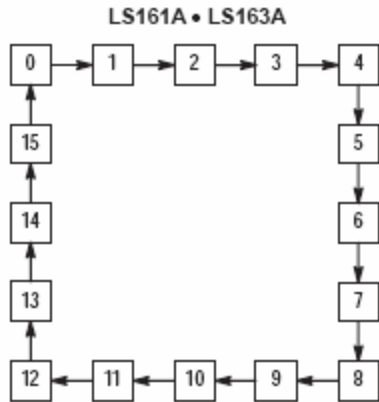a) 1 TTL Unit Load (U.L.) = 40 µA HIGH/1.6 mA LOW.

LOGIC SYMBOL

V_{CC} = PIN 16
GND = PIN 8

*MR for LS161A
*SR for LS163A

STATE DIAGRAM

LS161A • LS163A

LOGIC EQUATIONS

Count Enable = CEP • CET • PE

TC for LS161A & LS163A = CET • Q_0 • Q_1 • Q_2 • Q_3

Preset = $\overline{PE}$ • CP + (rising clock edge)

Reset = $\overline{MR}$ (LS161A)

Reset = $\overline{SR}$ • CP + (rising clock edge)
(LS163A)

MODE SELECT TABLE

*$\overline{SR}$	PE	CET	CEP	Action on the Rising Clock Edge (⌐)
L	X	X	X	RESET (Clear)
H	L	X	X	LOAD (P_n Q_n)
H	H	H	H	COUNT (Increment)
H	H	L	X	NO CHANGE (Hold)
H	H	X	L	NO CHANGE (Hold)

*For the LS163A only.

H = HIGH Voltage Level
L = LOW Voltage Level
X = Don't Care