Experiment: Debugging Circuits

I. Objective

To learn techniques for debugging circuits in a systematic manner.

II. Overview

When a complicated circuit is first built, is not uncommon for the circuit to be non-functional, due to wiring/connection errors, faulty parts, and/or incorrect equipment settings (*e.g.* wrong power-supply settings). The process of finding and correcting these problems is called **debugging**.

It is very easy to get overwhelmed by the circuit complexity and to get lost in the myriad of possible sources of error. However, it is not difficult to debug a circuit if you approach the problem systematically. Like a doctor examining a patient, an electrical engineer should find the cure for a malfunctioning circuit by observing, measuring, posing and testing various hypotheses, until the error(s) is (are) identified and corrected. We've already learned how to use two of the basic electronic measurement tools – the digital multimeter (Experiment #1) and the oscilloscope (Experiment #2); these will be our tools for debugging electrical circuits.

Firstly, we should make sure that our measurement tools are working properly, so that we can correctly interpret the measurements. We can be misled in our debugging efforts if we think that we are measuring something, when in fact we are measuring something else. This can happen, for example, if someone has set the oscilloscope to plot AC signals when you want to measure a DC signal using the oscilloscope.

FIRST RULE: Make sure the debugging tools are themselves working correctly. For example, test the voltmeter (DC voltage function on a digital multimeter) on a battery. Test the scope using the provided test signal.

In order to debug a circuit, you must understand how it is supposed to work. It is not an easy task to understand a very complex circuit. Simple circuits such as resistors in parallel, resistors in series, an RC circuit, *etc.* are easy to understand, however. Engineers typically design complicated circuits by using simple circuits as "building blocks." It is important to understand the function of each piece, or "sub-circuit," in order to understand how the circuit works as a whole. You should know how each of the various electronic components (*e.g.* resistor, capacitor, op-amp, transistor) is supposed to behave, *i.e.* the current-versus-voltage (*I-V*) characteristics of each component. Some basic questions to ask are:

- How many terminals does the component have?
- What is the voltage (or current) of a certain terminal when we apply a certain voltage (or current) at another one of its terminals?
- How does this change with varying voltages and currents (AC signals)?
- Note: The electrical properties of a component may change with various factors such as temperature, pressure, and magnetic fields.

SECOND RULE: Study the circuit diagram until you have at least a general idea of the functions of the various parts of the circuit. Know enough about the components to be able to devise tests to determine whether or not each part is functioning properly.

III. Approach to Debugging

Suppose you are given a malfunctioning circuit, and that you need to determine what is wrong with it and correct the problem(s). Even if you don't understand completely how the circuit works, there is a good chance that you can still debug it by checking for basic errors. (You probably have experienced making the common mistake of forgetting to plug the power cord into the electrical outlet!) We should always be aware that very simple errors are also very common errors.

Matching the diagram of the circuit on paper to the actual constructed circuit is part of the debugging process. Most circuits have points of fixed DC voltages (V_{CC}, ground, V+, *etc.*), so that we know what the voltages at these nodes should be — we should measure and confirm their values. If any other node has a predictable theoretical voltage, then we should confirm that, too.

Debugging Steps:

1. Check the actual wiring against the schematic. One method is to simply verify what is connected to each node.
2. Check the plug-pin circuit boards. They are notorious for bad connections.
3. Check the power bussing: Are all of the DC voltages correctly distributed?
4. If all of the fixed DC values are correct, check the operation of the actual components. (For example, measure the DC and AC voltage values at the input and output nodes of an op-amp to determine if it is working properly.)
5. If a component is suspected to be faulty, first verify that it is really hooked up correctly, then check to see if it is functional.
6. If the circuit can be divided into stages or modules, check each one of them separately. Verify that each stage or module of the circuit is functional. (Isolate the bug and eliminate it!)

To isolate an error, consider the circuit as having inputs (*e.g.* power-supply voltages, or an AC signal coming from a signal generator) and outputs. We should also be able to identify sub-circuits with inputs and outputs. We can then measure the input(s) and see if the output corresponds to what we expect, for each sub-circuit. If it does not, then something is wrong with that particular sub-circuit, and we can go down one more level to see which component(s) is (are) causing the trouble. This methodical approach to debugging a circuit may be too time-consuming in practice for very complicated circuits. As engineers, we should try to balance the knowledge we have of the properties of the circuit with this methodical approach.

Sometimes, you may pinpoint a possible source of the problem, but may not have the time to test out your hypothesis directly on the circuit. In this case, you should devise a quick and easy experiment to test your hypothesis. For example: You think a transistor is burnt out, but are not absolutely sure. You can remove it from the circuit and stick it into a curve tracer (an oscilloscope-like instrument – you can ask your TA to show one to you in the lab) to see whether it's working or not.

Engineering is the art of solving problems. If you wonder why it is an art, consider the debugging process: We want to find and correct an error, and we have algorithms and methods to help us do this. However, there is no fixed way of using these techniques – we have to figure that out, keeping in mind that we want to do this in the shortest amount of time. This requires skill and experience, but as in art, it also requires creativity.

How are circuits debugged in industry?

There is the "doing it by hand" method, but in reality, engineers have fancy programs or machines that will help you with the design of circuits. Debugging prototype circuits is always a hands-on process. Nothing beats common sense, and common sense can sometimes be a lot easier than learning to use fancy programs or machines -- and it comes in handy during technical job interviews.

Debugging advice from past EECS students:

- Always make sure you wire your circuits neatly. This will make your debugging a lot easier! (i.e. short wires for short distance connection)
- Always connect your power supply (Vcc and Gnd) to the red and blue rails of the breadboard.
- Make sure you know what your circuit is supposed to do.
- First analyze the circuit by hand to make sure you know what you're looking for (in terms of voltages and currents at various points in the circuit). Then work from the power supply down (or ground-node up), testing voltages to see if what you expect is what you're getting.
- Test your components.
- Follow the data path – from input to output – where does the error occur? (This is more of a debugging tool for digital circuits. Feedback, *e.g.* used in op-amp circuits, can complicate debugging since some of the inputs depend on the output.)

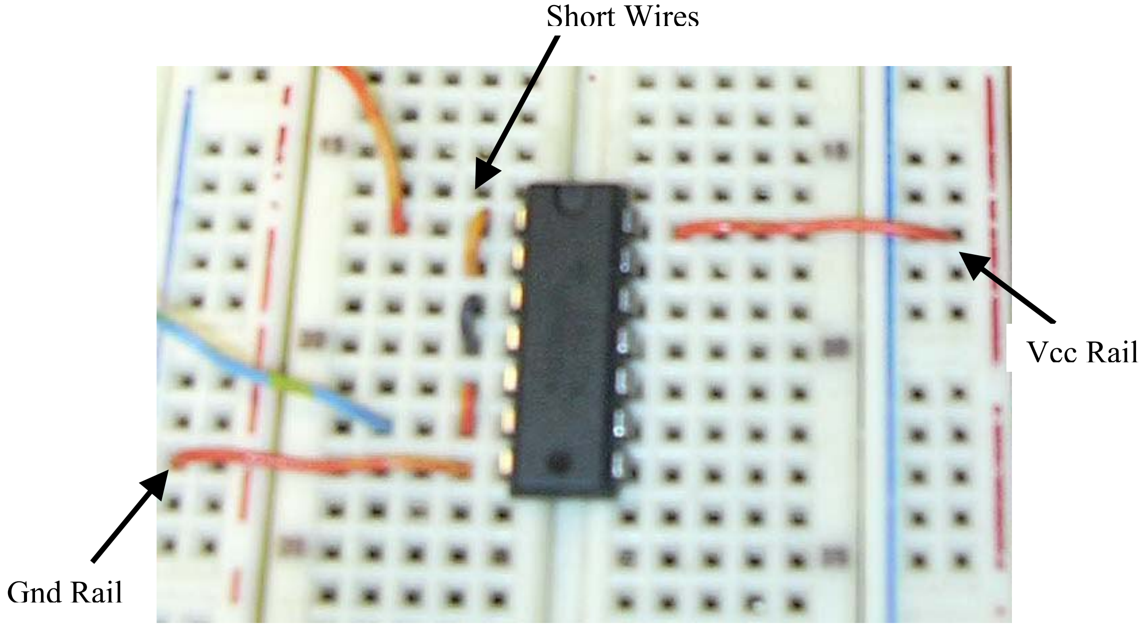
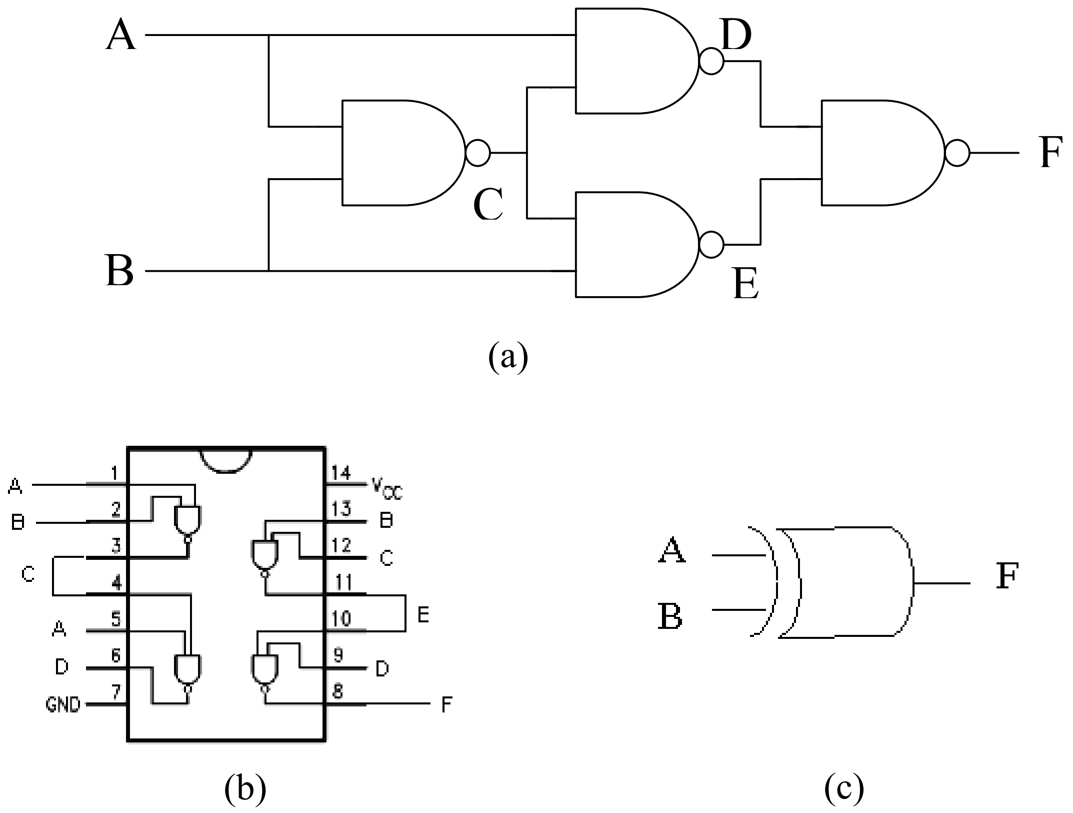


Figure 1. Example of a wired chip

IV. Hands-On

1. Building a XOR Logic Gate Using NANDs

As mentioned earlier, you should always know what your circuit is supposed to do. To illustrate this point, first construct a XOR logic gate using the MM74HC00 chip, as shown below (Fig 2a). A and B are the inputs to the logic gates, thus you need to connect them to either Vcc (1) or Gnd (0). Then, measure the voltage for nodes C to F using the digital oscilloscope probe and record the result onto the truth table in your lab report. Compare your measured result to the truth table you derived in your prelab. In this exercise, use Vcc = 5V.



(a)

(b)

(c)

Figure 2. The XOR Logic Gate (a) construction of a XOR gate using 4 NANDs (b) top view of the circuit (c) schematic of a XOR logic gate

When you are done, ask your TA to verify your working circuit.

2. Debugging Logic Circuit

By this time, your TA will kindly construct the following circuit for you (Fig 3). However, your TA will "carelessly" make some mistakes in your circuit and you need to figure out what is the bug.

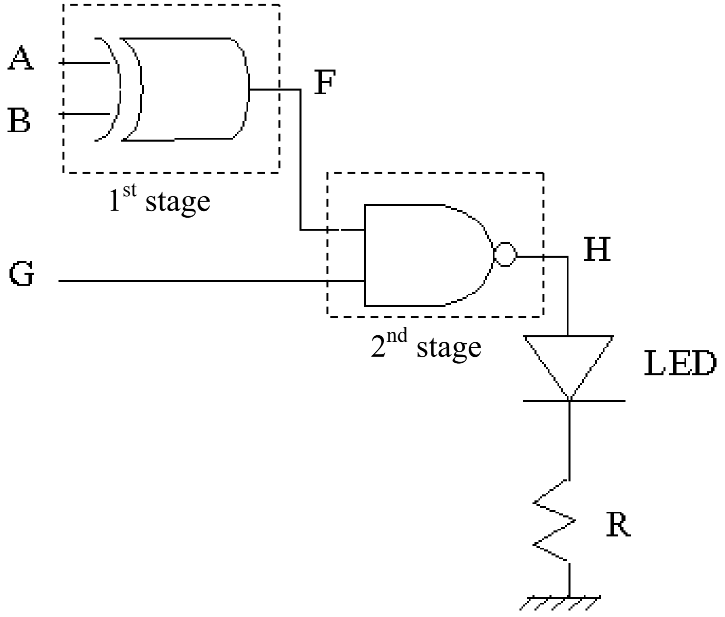


Figure 3. A Logic Circuit

INPUTS			OUTPUT	
A	B	G	F	H
0	0	0	0	1
0	0	1	0	1
0	1	0	1	1
0	1	1	1	0
1	0	0	1	1
1	0	1	1	0
1	1	0	0	1
1	1	1	0	1

Table 1. The Truth Table

By connecting the inputs (A, B, and G) to Vcc or Gnd and measure the output (H), write down your measured truth table. You should notice that your result is different form the desired truth table above (Table 1).

A useful way to debug a complex circuit is to break it into simpler and independent stages. In this case, we can break our circuit into 2 stages (see Fig 3).

First, let us concentrate on the first stage – the XOR. The inputs for this stage are A and B, and the output is F. By varying the inputs to Vcc or Gnd, measure the voltage at nodes C, D, E, and F. Record your result onto the truth table in your lab report and compare it with the one in the previous section. How are they different? Can you identify which logic gate in the XOR module is not behaving correctly? Check with your TA to verify your solution. If your TA agrees with your solution, you can correct the bug in your XOR stage.

Now, temporarily disconnect the wire that links the two modules in your circuit – so that we treat the second stage independently.

The second stage consists of a NAND gate with two inputs (F & G), and one output (H). Connect the inputs to Vcc or Gnd, and measure the voltage on the output node H. Then, record your result onto the truth table in your lab report and compare it with Table 1. Can you figure out what is wrong with the second stage? Check your solution with your TA.

Now re-connect the wire that links the two stages and record the final truth table.