EEL 3701C: Digital Logic & Computer System
Fall 2022

LAB 00: Introduction with Basic Gates in Digital Logic
(25 Points)

Instructional Objectives

At the end of this lab you should be able to:

1. Research the internet for IC datasheets and circuit diagrams to obtain the required information.
2. Understand the fundamentals of Boolean algebra as applied to digital logic.
3. Understand and use breadboards.
4. Construct combinational logic circuits and test to determine the output signal as a function of all possible combinations of several input signals.
5. Specify a logic circuit using truth tables.

Introduction:

Digital circuits make up the cornerstone of modern computational hardware. By representing binary digits (i.e. {0,1}) with voltage levels, digital circuits are able to process binary numbers electronically. Logic gates are the fundamental components within digital circuits so understanding their behavior is important. Therefore, the purpose of this experiment is to introduce you to gate behavior and logic interpretation as well as the basics of circuit wiring and troubleshooting. To do so, we will explore the function of several of the basic logic gates discussed in the lecture.

Background

Background information necessary for the completion of this lab assignment will be presented in the next few subsections.

2.1 Functionalities of Logic Gates

Digital logic *gates* are specialized electronic circuits that implement Boolean algebra expressions. Boolean algebra is the language of computer electronics and consists of logical ‘1’ (sometimes called “true”, “high”, or “on”) and logical ‘0’ (sometimes called “false”, “low”, or “off”). In the electronic circuits in this experiment, we will be using ~5V to represent a logical ‘1’ and ~0 V to represent a logical ‘0’. This is a common standard; however, other voltage representations for logical ‘1’ and ‘0’ are sometimes implemented.

From the two Boolean elements ‘1’ and ‘0’, all *binary* numbers are determined. In contrast, our common decimal numbering system has 10 elements: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, and from these, all our numbers are determined.

In Boolean algebra, three common logical operations are “AND”, “OR”, and “NOT”. Logical AND has similarities to multiplying, and a “dot” symbol such as i is used to indicate AND. The output of an AND circuit is 1 if and only if all the input signals to the AND are 1. For example, a two-input AND function gives: 0 x 0 = 0, 0 x 1 = 0, 1 x 0 = 0, and 1 x 1 = 1. Logical OR is represented by the “+” symbol. The output of an OR circuit is 1 when any of the input signals are 1. Thus, for a two-input OR function: 0 + 0 = 0, 0 + 1 =1, 1 + 0 = 1, and 1 + 1 = 1. Logical NOT is represented with a prime, “ ’ “, or an overbar “– “. NOT operation is the logical inverse of the expression, for instance, 1′ = 0, and 0′ = 1. For many types f physical logic, it is convenient to fabricate NAND and NOR circuits rather than AND and OR circuits. NAND means “NOT AND” and NOR means “NOT OR”.

2.2 The 7400 Series of Logic Gates:
Logic gates are constructed from transistors, which are analog switches. These transistors can be forced to operate in two modes, namely ”ON” or ”OFF.” In doing so, we can abstractly think of electronic signals within a digital circuit as being either HIGH or LOW (i.e. ’1’ or ’0’). A digital gate takes as input one or more digital signals and outputs a digital signal as a result of the Boolean operation. Figure *1* depicts the standard logic gate symbols and their associated Boolean operation.

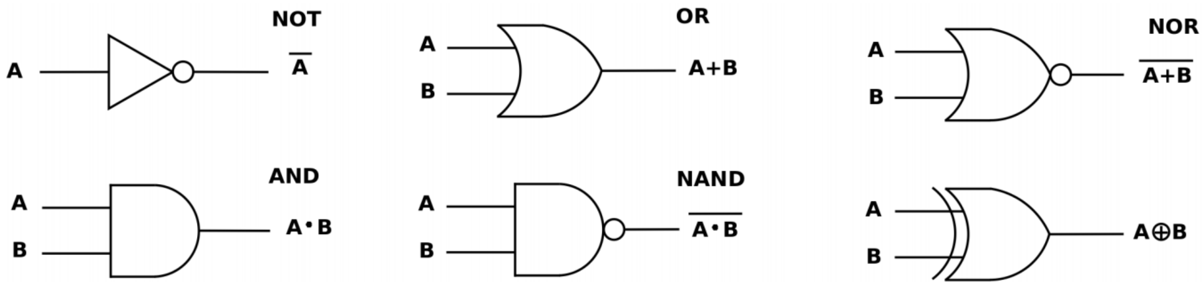


Figure 1 Basic gates in digital logic

The basic gates you will study in lectures are available in a series of Integrated Circuits (ICs) commonly referred to as the “7400” series. Within this series, there are various IC package types available; however, for breadboarding digital circuits in the laboratory, we will use the Dual-Inline Package (DIP) type as shown in Figure *2* (left). As shown, the DIP features a black plastic package with pins on both sides, slightly resembling a flat caterpillar.

Figure *2* shows the 7400 IC (left) and the DIP pinout diagrams (right). You have to supply positive voltage on the 14th pin and the 7th pin will go to the common ground. The IC contains four NAND gates. The internal connection is shown in Figure *3*. This figure clarifies the input and output pins of the IC. The other 74 series ICs maintain the same organization. You can obtain all information at www.ti.com. Put the IC number (such as SN74LS00N) in the search option in this webpage and download the datasheet to extract your required information.

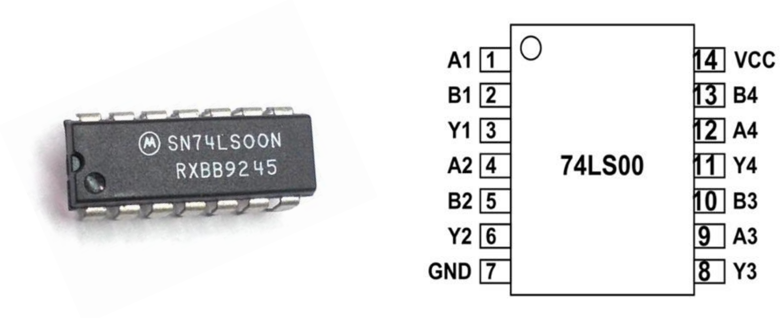


Figure 2: 74LS00 IC (left) and its pinout (right).

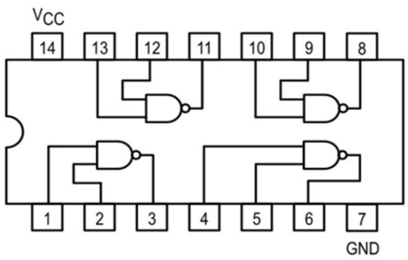


Figure 3: The internal connection of 7400

2.1 Breadboarding Techniques

A breadboard is a rectangular plastic board with a bunch of tiny holes in it. These holes let you easily insert electronic components to **prototype** (meaning to build and test an early version of) an electronic circuit, like a battery, switch, resistor, and an LED (light-emitting diode). We will implement and test most of our experiments on the breadboard. Here are a few techniques which will help you wire up your circuits:

1. Horizontal lines of points on the breadboard are electrically connected together. However, lines are not connected across the partition divisions (i.e. lines in different partitions are independent).
2. Vertical lines are NOT electrically connected, except in the case of a power or ground line (see Figure *4*).
3. Power and Ground lines shown vertically in Figure *4* are electrically connected. In the lab, you will connect the 5V supply to one of the pins in the Power line. This will connect the entire column to the 5V supply. Similarly, the Ground signal needs to be connected to one pin in the Ground line.
4. All the Integrated Circuits (ICs) in the design should be placed across one of the partition divisions as shown in Figure *4*. Do not place an IC in only one partition because this will short pins together causing the IC to burn up.
5. Wires connecting different IC pins should traverse horizontally or vertically only. Do not connect wires diagonally across the breadboard as this will result in a messy design making it difficult to debug.
6. Use smaller wires when connecting nearby points on the breadboard. This will help keep your design clean and easy to debug.
7. You may decide to always place the IC in such a way that the notch is located on the top (or bottom) as shown in Figure *4*. This may help you identify pin numbers.
8. Before placing the components on the breadboard, plan the placement of your ICs such that it minimizes wiring distance on the breadboard. ICs with high connectivity should be placed near each other. For example, if LED display inputs are connected to OR gate outputs, then try to place the OR gate IC as close as possible to the LED display.

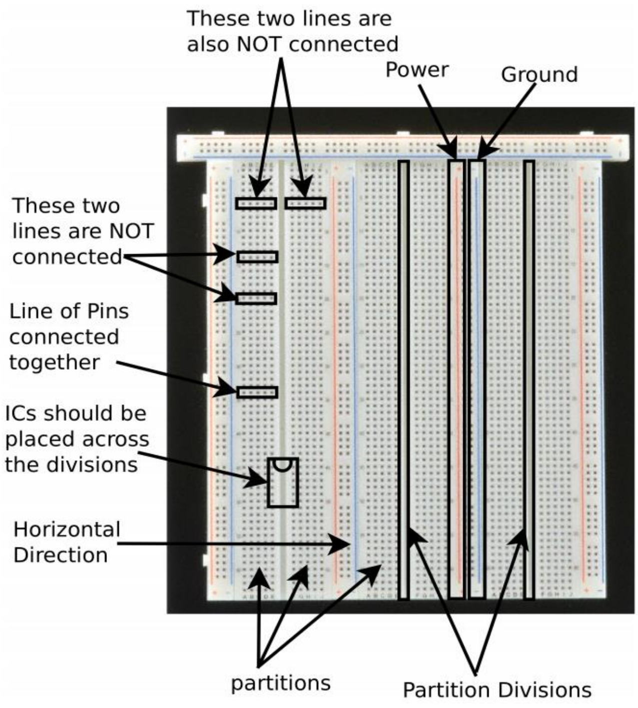


Figure 4: Overview of the breadboard

An example of connecting a power supply, resistor, and light emitting diode (LED) in a breadboard is demonstrated in Figure 5: Connecting battery, resistor, and LED in a breadboard.. Here, the battery is giving +5V power supply to turn on an LED. A resistor is connected in series with the LED. It is mainly used to limit current of LED) and prevent excessive current breakdown damaging the LED. To limit the current flow in the LED, a series resistance is inserted.

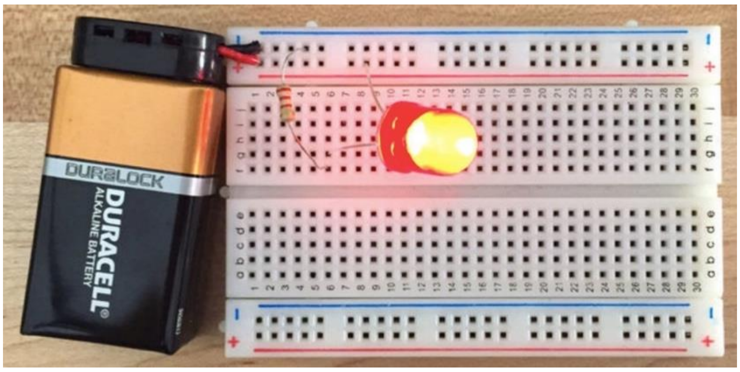
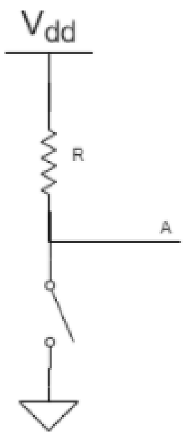


Figure 5: Connecting battery, resistor, and LED in a breadboard.

2.1.1 Pull-up Resistor Circuits

Within your lab equipment, there will be a series of Dual-inline Package (DIP) switches. These switches are meant to provide various input sequences into your circuit or FPGA board. When using these switches, you will need to use the pull-up resistor configuration, as shown below:



Recommend using the SIP resistor package for the pull-up resistor configuration. Always use this or alternative (pull-down) configuration to prevent overcurrent to logic IC and FPGA pins.

Experiment:

We will look at the behavior of logic gates in Figure *1*. Each of these gates is embedded in an integrated circuit package. Observe the datasheets of each component for the pinouts, electrical and timing characteristics of these circuits. All datasheets are available on www.alldatasheet.com and www.ti.com.

We will start by setting up the DC power supply and multimeter for our use. Be sure both are turned off. Then check to see that the multimeter is set to measure DC and be sure the red lead is connected to the red multi-meter input that is marked for voltage and the black lead is connected to the black point. To measure the voltage, connect the red lead with VCC or gate output and black lead with common ground. Set the scale to the range you need to measure (usually between 0V to 20V for digital circuits). Now, measure the DC power supply that you have and mention in your report. Once, the power supply is adjusted, you are ready to test the ICs. The experiment has two parts. In Part 1, we will be familiarized with the ICs and observe the output. In Part 2 we will solve Boolean expression.

(There is an alternative power supply. The kit you receive that has an FPGA board with some external pins. If you connect the USB cable, then the VCC and GND pin will provide your supply voltage. Though the voltage is less than 5V, but the ICs work fine with that voltage.)

Part 1

We will start by wiring an SN74HS04N (inverter) gate. Please refer to the pin configuration given in the SN74HS04N datasheet. Insert the SN74HS04N chip onto the breadboard. Be sure you are not shorting pins together. Identify the power (VCC) and ground (GND) pins for the SN74HS04N from the pin-out of the SN74HS04N in the datasheet. Connect the VCC pin to the red lead of the power supply and connect the GND pin to the black lead of the power supply. This chip (7404) contains 6 different inverter gates. Each inverter gate has an input pin and a corresponding output pin. Choose one of the gates and connect one leg of an LED with that output. The other leg of that LED must be connected with the common GND. Now apply VCC and then GND at the input and observe the output for these two cases. If the LED becomes on, then the output is High and Low otherwise. Assume A is the input to the inverter (either High or Low) and that Y is the output. At the same time measure the output voltage with the multimeter and note that voltage.

Problem 1

Fill in Table *1* in your report according to the logic behavior that you observe.

Table 1 Truth Table for an Inverter (NOT Gate)

A (High/ Low)	Y (Volts)	Y (High/Low)

Problem 2

We are going to repeat the same experiment with the gates given to you. Note that each of the gates has two inputs and one output. Fill in Table *2*, below to indicate the observed responses of these gates. Here, we will label High with “1” and Low with “0”. Also, identify the type of chips from your obtained output.

Table 2: Truth table for two input Gates

Input		Output (Y)				
A	B	7400	7402	7408	7432	7486
0	0					
0	1					
1	0					
1	1					
Identify Gate Type						**For Example: Ex-OR**

Problem 3

Again, we will repeat the same experiment for three input gates and obtain different output for 7410 and 7411.

Part 2:

In this part you will learn how to map different Boolean expression using the ICs. A simple Boolean expression might be $X = A + \bar{B}\ x\ C$

The Boolean expression tells that we need an AND gate and an OR gate to solve this problem. But we can implement this using a NAND and NOR gate. NAND gate is the logic equivalent to NOT and an AND gate. The outputs of NAND and NOR gate you have observed in the part 1. Now construct your circuit according to *Figure 6* Circuit diagram for the given Boolean expression. We will not use all gates in the chip.

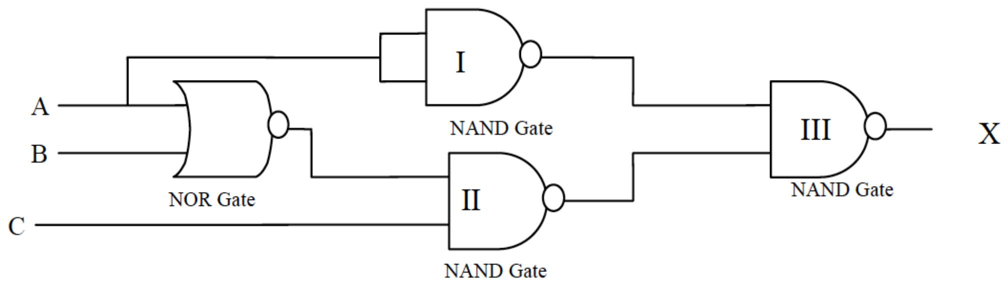


Figure 6 Circuit diagram for the given Boolean expression

Problem 4

Now measure the output voltage at X for all different combinations in A, B, and C. You need to connect the LED at the terminal X to demonstrate your output. Generate the truth table for the given combinational logic based on your design for eight different input patterns.

Pre-Lab Homework (15)

- Take a time to read the UF Laboratory Safety Manual. http://webfiles.ehs.ufl.edu/labsafe.pdf
- Go through the assembly guide ***MXDB Assembly Guide.pdf*** . Familiarize yourself with the procedure.
- Read a tutorial on how to solder from one of the multiple online materials (for instance https://www.makerspaces.com/how-to-solder/)
- Summarize your readings in 1 -2 paragraphs. **(5 pts)**
- Solve the problem 1,2,3, and 4. **(10 pts)**

In-Lab Implementation (10)

- You will receive a toolbox ***(3701 Parts List.pdf)*** Verify that all parts are available.
- With the help of your PI, build and test a simple AND-gate circuit using two switches, a LED and a 74HC08 chip. Verify that your results match the expected results for this logic combination, and re-check your circuit if there are any discrepancies **(10 pts)**

Deliverables:

- Summarize of your work in a report. Template for lab report is provide on Canvas