

Logic Design for Physical and Virtual Laboratories

Tri Nguyen, Bachelor Student, Lars Koester, Bachelor Student, and Astelyn Saint-Juste, Bachelor Student
Florida Atlantic University, USA, tringuyen2014@fau.edu, lkoester2015@fau.edu, asaintjuste2017@fau.edu
Mentor: Luis Felipe Zapata-Rivera, lzapatariver2014@fau.edu

Abstract– This research project looks at the usefulness of logic commands, 0’s and 1’s, in real world applications. The goal of this project is to help students see the usefulness of logic design as more than just blinking lights of a breadboard. The problems that were encountered during the research was finding a way to taking output₁ from switches taken for the breadboard, finding a way to check whether the answer taking from the breadboard from the user was incorrect or correct based on the output₂ from the circuit, as well program the software to check the users output₁ and comparing to the answer stored on the program. A prototype has been designed to show the implementation of the usefulness of logic design.

Keywords—Hybrid, CSV File, Analyzer, Laboratory, Binary values, Logic Design

I. INTRODUCTION

Laboratories are a crucial part of any engineering course taught in higher education. This is especially useful for introductory course, which require students to learn the fundamentals of engineering. By using breadboards to design and implement circuits that were taught in class, the students get a hands-on experience of understanding the logic behind electrical circuits. However, lighting up LED’s is not stimulating to the student. For that reason, we wanted to create a program that is centered on the laboratories to make them more interactive to the students.

This type of implementation where students interact with the physical lab and a virtual lab is called a hybrid laboratory. Hybrid laboratory “can be composed of combination of physical laboratories ... or a combination of on-line laboratories ..., or a combination of a physical and an online laboratory.” [1], as shown in Figure 1.

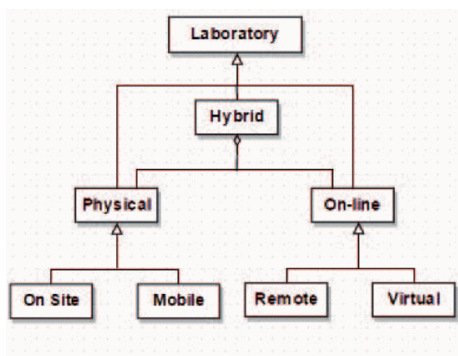


Figure 1. Proposed UML Model of Laboratory Taxonomy. [1]

This paper focuses on the concept of hybrid laboratories that can be used in an academic environment to enhance student comprehension of physical and virtual laboratories of logic design understanding.

II. UNDERSTANDING SALEAE RESEARCH PROJECT (PROTOTYPE 2)

This Project was designed using two keys components the Saleae Analyzer and the Application program interface (API). The Saleae Analyzer is able to read in electrical signals and send this information to the computer for the API to process. The Saleae Analyzer was connected to a computer via USB and 8 channels that receive electrical signals. For this prototype, 6 channels were used to gather information from the breadboard. 4 channels were used to gather the outputs from the dip switches that the user would input by flipping the switch up and down. 2 of the 6 channels were used to gather the output from a circuit that was designed on the breadboard. The API will receive the data from the Saleae Analyzer and process the data into a file and export the data to the program.

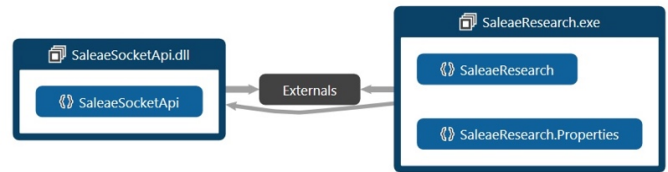


Figure 2. UML Overview of Relationship between API and Research Program.

The Program itself is designed to combine hands-on experience with a virtual laboratory. By not only lighting up LED’s on a Breadboard, the students can see how their input on the breadboard makes a decision in the virtual laboratory on the computer screen. This shows the students that the logical circuits they are building in class, can decide whether a person can get into the party. Using this analogy, we started to implement that idea. We produced a Boolean expression and based on the input on the breadboard, the expression is either right or wrong.

The first step was to find characters that we could use commercially and modify it. We found a website that let us use their images commercially and modify them. We created what a Boolean expression that fits the description of the image and named them accordingly. On the Breadboard, there are four switches that allow the student to input statements. The Boolean expression we created is $AC\bar{D} + AB\bar{C}D$. Each letter represents criteria for the person to be allowed in. For example, A means that the person is wearing a hat, and B means that the person is wearing glasses. The student also needs to build the circuit on the breadboard and the program will compare those two results and give appropriate feedback based on the input and output of circuit. The UML describes the process of the user input and how the whole process works.

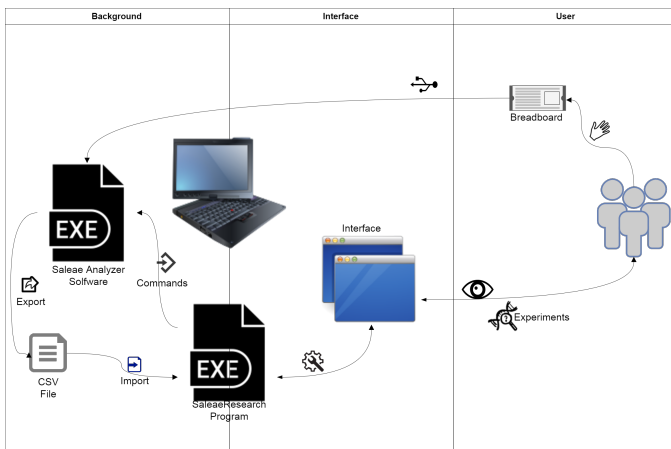


Figure 3. Diagram Showing How The Program Works.

In order for this Program to work, the Saleae software must be running and the Saleae Analyzer must be connected to the computer. Once it's loaded up, it will display the experiment along with an equation and also one of the pictures that associate with that specific experiment; more importantly, the program will also get a connection that connect to the Saleae software by using the given API. By connecting to the Saleae Software, the Program can setup how the analyzer read the data from the breadboard and can also send command to the Saleae Software to perform task as needed. Then, the students can implement the experiment on the breadboard and also connect the needed channels of the Saleae Analyzer to the breadboard. For this specific experiment, we will be using six channels; the first four channels being connected to the switch (the inputs of the circuit), and the next two channels being connected to the output of the circuit, supposedly that the channels are connected in the correct order. The figure below shows an example of how the experiment would be.

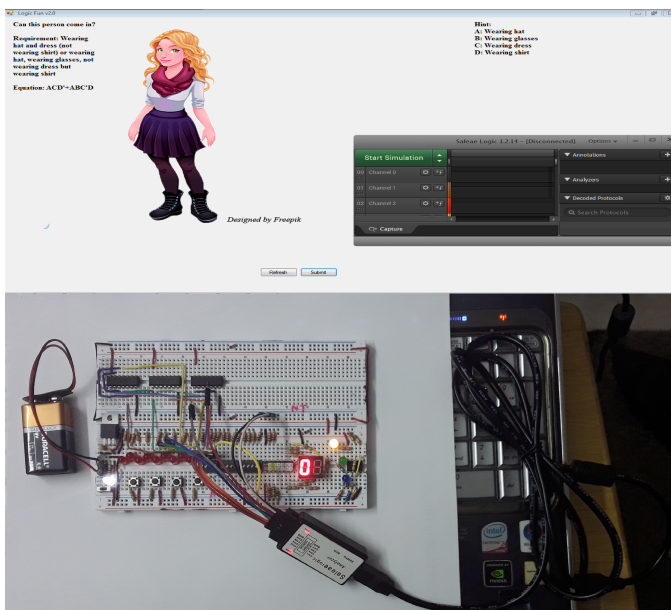


Figure 4. Picture of Real-Life Experiment.

After that, the students will have to set the switch according to the picture shown in the experiment, and then click the "Submit" button. This is the most important part of this program. First, the program will split the equation into separate terms and then convert each of those terms into binaries of 1 for high values, 0 for negated values, and -1 for don't care values, which is a combination of variables whose logical value is not specified [2]; it will also convert the name of the chosen picture, which is the term for that specific picture, into binaries of 1 and 0. Second, the program will give the command to Saleae Software via the mentioned connection to start capturing the data from the breadboard.

Once the capturing process is finished, another command is sent to the Saleae Software telling it to export the read data out as a comma-separated values (CSV) file. CSV files are plain text; they can contain numbers and letters only, and structure the data contained within them in a tabular, or table, form [3]. In this case, the file only contains one line of six zeros and ones separated by comma. Then, the Program will read the CSV file for needed data. Skipping the first value since it's the time of capture, the next four values are the values of the input channels, and the last two values are the values of the output channels. After having all needed data, the program will start comparing the data to generate the result message; below is the flowchart for this process.

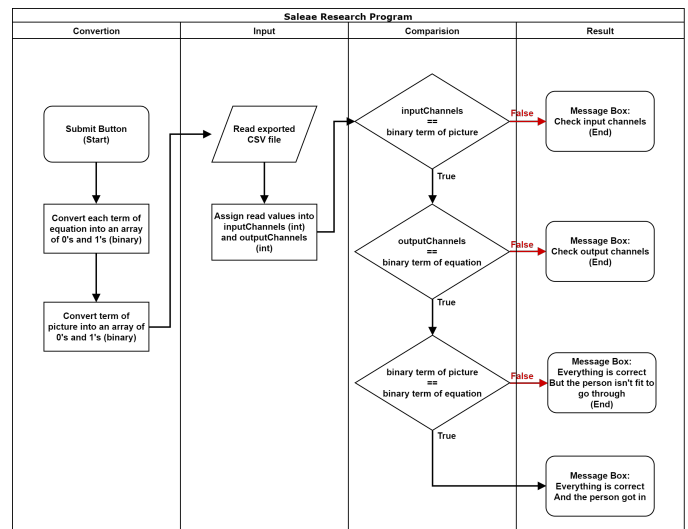


Figure 5. Flowchart of the Process when the Submit Button Being Clicked

First, the program will compare the values from the input channels to the values of the term given to the displayed picture, which was converted into binaries; if those sets of values don't match each other, then a message prompting the students to check the input channels since the input doesn't match the value that it supposed to be. Otherwise, the program can move onto the next step.

Second, each value from the output channels will be checked against the values from each term of the equation that was converted into binary; if the value from the output doesn't

match the associate value from the equation, then the program will show a message prompting the students to check the output channels and tell the students the which output channel(s) they need to check. Otherwise, the program can move onto the final step.

Finally, the Program will need to check the values of the term given to the displayed picture against the values from each term of the equation; except for the don't care variable, if the two sets of values don't match each other, a message will be shown to the students to inform them that they did everything correctly, however, the person from the picture isn't supposed to be able to go through. Otherwise, the students will be informed that the person in the picture got in.

III. CONCLUSION

We are at the final step of the second prototype for this project; however, this project is still at the very beginning stage. There are so many way to keep developing this project and we have not achieved our ultimate goal for making this program. The next stage that we are aiming at is to connect this project with the remote laboratories, which laboratories use real physical laboratories that are capable of being accessed through a network [1], that is being developed by Luis Felipe Zapata Rivera and Leonardo Ribeiro; with remote laboratories, the instruments can be accessed, monitored and controlled at a distance [1]. Below is an example of the Remote Lab.

[3] T. Fisher, "Life Wire," 16 05 2017. [Online]. Available: <https://www.lifewire.com/csv-file-2622708>. [Accessed 13 07 2017].

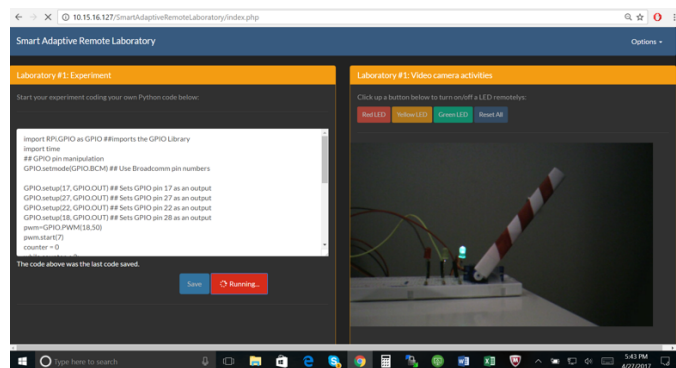


Figure 6. Picture of Remote Laboratory.

For that, we can replace boring message with the actual action of the remote lab. By doing this, we are certainly one step closer to our ultimate goal for making the laboratories more realistic and interesting.

REFERENCES

[1] M. M. Larrondo-Petrie and L. F. Z. Rivera, "IEEEExplore," 24-26 Febuary 2016. [Online]. Available: <http://ieeexplore.ieee.org/xpls/icp.jsp?arnumber=7444517>. [Accessed 7 July 2017].

[2] M. M. Mano, Digital Design, London: Prentice Hall, Inc., 1991.