Lab 7: Introduction to LabVIEW

EG 1003, Section G

Ayman Ahmed

­­­

Lab Partners: None

Experiment Date: 04/09/2019

Due Date: 04/16/2019

**Abstract**

The objective of this lab is two design two systems using the LabVIEW software. The two systems are a lighting system and a heating/cooling system. The systems will then be prototyped using a NI Elvis II board. In LabVIEW, logic and graphical programming will be used to make the systems function. In the end, the results will be that LabVIEW can be control heating and cooling for one system, and multiple LED lights for the other system.

**Introduction**

LabVIEW is a developmental environment that allows for prototyping on a computer using interactive virtual interfaces that simulate actual physical equipment, or can pair with physical equipment for easy prototyping of a system. LabVIEW components are known as VIs, which stand for virtual instruments. Each VI has two parts, the front panel and the back panel.

The front panel is the interface where someone can input data or control switches, while the back panel is where the graphical programming is done and contains the logic behind the instrument. The front panel uses controls and indicators, which both deal with information. The control allows the user to input information, while the indicator displays the output of the VI based on the inputs.

The back panel of the VI contains the source code of the instrument in a graphical format, with the inputs from the front panel showing up as terminals. Terminals show the values of the inputs which can be interacted with by the code. The code consists of two parts - structures and functions. Structures are used to create conditions, while functions give a new calculated value when one is provided based on the function. Some examples of structures are for loops, while loops, and case structures. Case structures act as multiple if/else statements, while for loops iterate over the given length. While loops iterate until the original statement is false. Every piece in the back panel, including terminals, functions, and structures, are all called nodes.

Sometimes LabVIEW controls physical pieces of a system through a prototyping board called an NI-ELVIS II+, which is essentially a large breadboard with many extra functions built in to allow easy prototyping. Some of these features include LEDs attached to the board, detachable boards, significant amounts of different power inputs, and a large breadboard size allowing for many complex circuits and hardware pieces to interact with it.

**Procedures**

The materials required for this lab were a lab PC, the LabVIEW 2017 software, an NI-ELVIS II+ prototyping board, two 100k Ohm resistors, and wires to allow LabVIEW to control our system through the prototyping board. Other materials used involved the LEDs on the NI-ELVIS board, a heating coil, and a cooling fan, which were used as the functional physical parts of the system that we controlled using LabVIEW.

After the materials were all gathered, a VI was opened to design the lighting system. The front panel required five switches and four LEDs, one for each physical light. The back panel required another case structure for one of the switches to act as a master switch, being able to turn off all the lights upon activation.

To control the LEDs on the NI-ELVIS board, the NI-ELVIS digital writer was imported from the functions tab of LabVIEW into the back panel of the VI. The DO Lines are a Boolean array that connect directly to the lights, while the Lines to Write determine which DIO rows on the physical board function. To determine the lines to write, a constant was created with the value 0-7 and attached to Lines to Write. Afterwards, a Build Array structure was created, which was attached to four false constants and the four LEDs.

To make the physical LEDs function, the 24 rows labelled DIO on the board had to be wired to the LED rows on the board. Numerically, the rows 0-23 had to be wired to rows 35-42. Finally, the lighting system was tested.

**Data/Observations**

The result of the original test VI using the thermometer as a control was very effective, showing that the logic behind the heating/cooling system worked, and was an efficient way to test the code before the heat box was actually attached to the board. The LabVIEW software was very efficient to use, and was well organized through the front panel / back panel method. Adding in different structures and functions was easy, with all of them being organized in neat sections that are easily available with one right click.

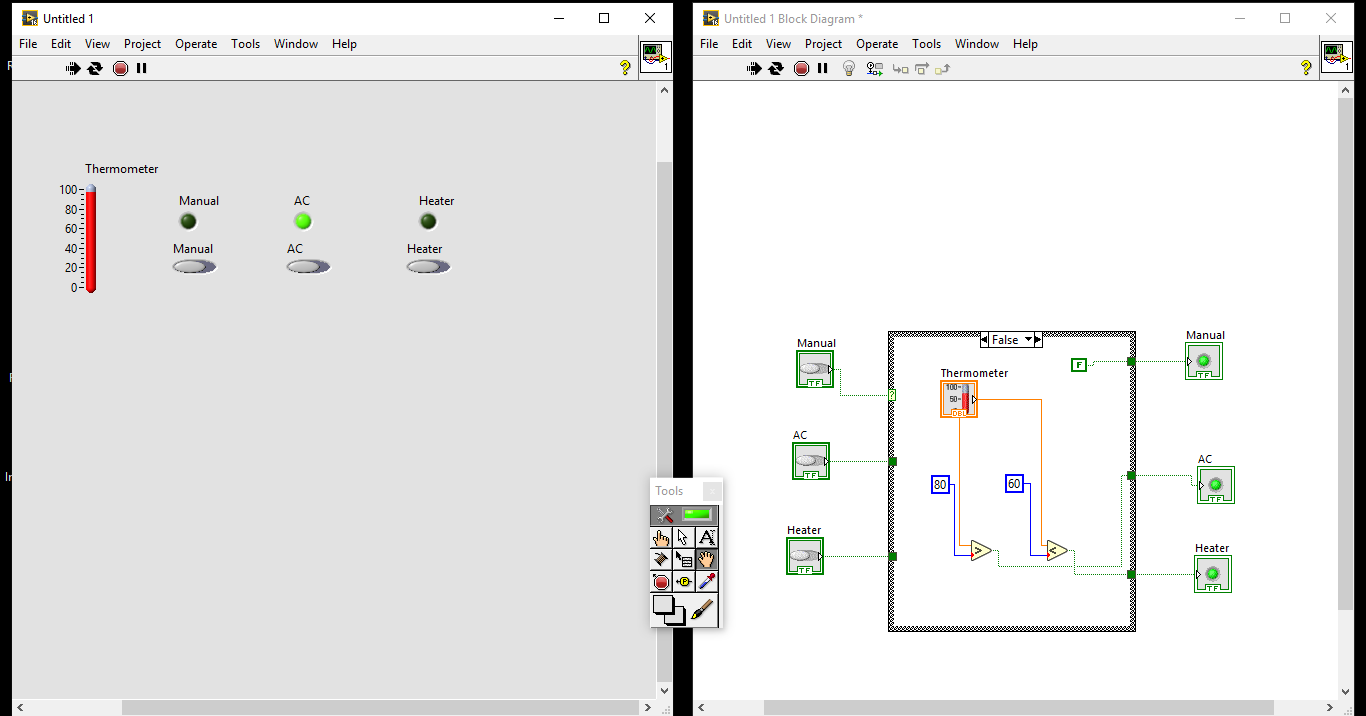


Figure 1 – Temperature Control Prototype VI

The heating/cooling system using the NI-ELVIS II+ board was very performant, and worked according to specifications. When run continuously, the physical heating/cooling parts activated quickly once the control switch on the VI was activated. LabVIEW also has well defined error messages, with good documentation for issues in the case structure in the situation that anything was wired improperly.

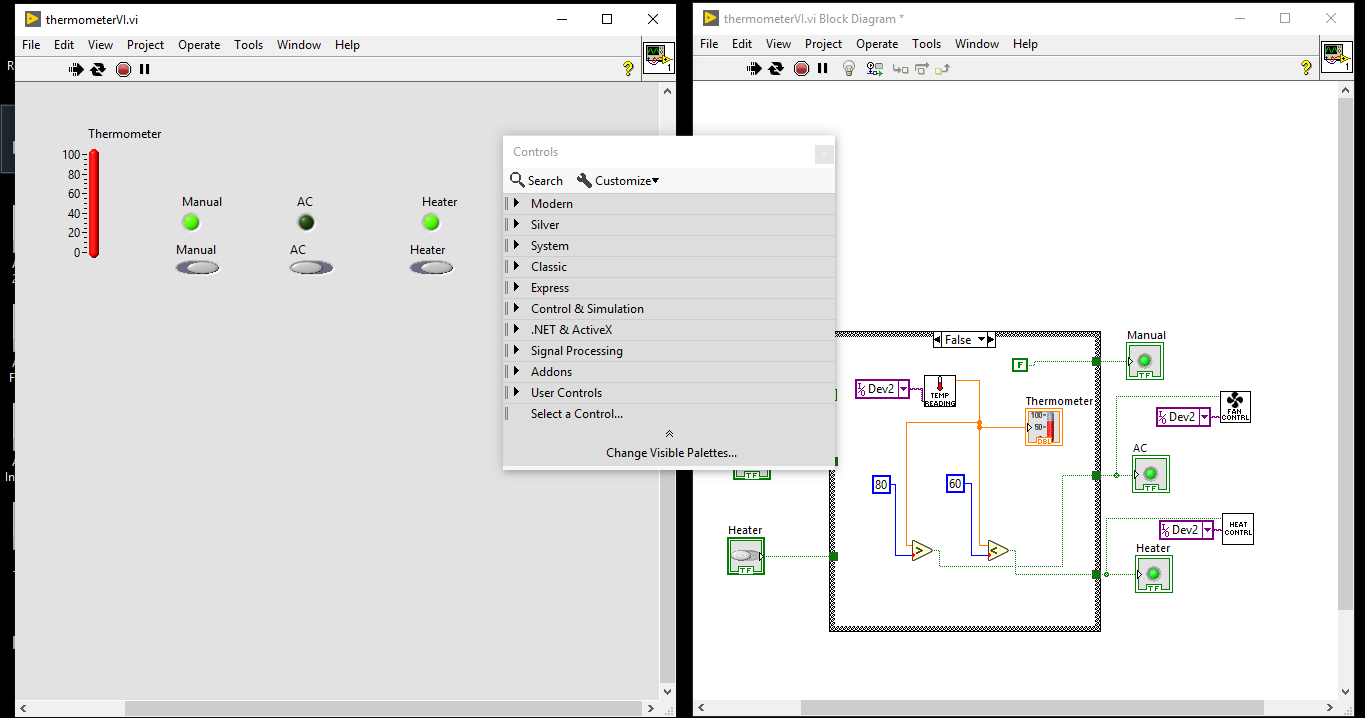


Figure 2 – Heating / Cooling System VI

The lighting system was by far the easiest one to assemble, since the LEDs were already physically embedded into the NI-ELVIS board. This was another situation where error messages in the back panel were extremely useful, with the error message for the Build Array function showing that a different build array piece is required for each part added to it.

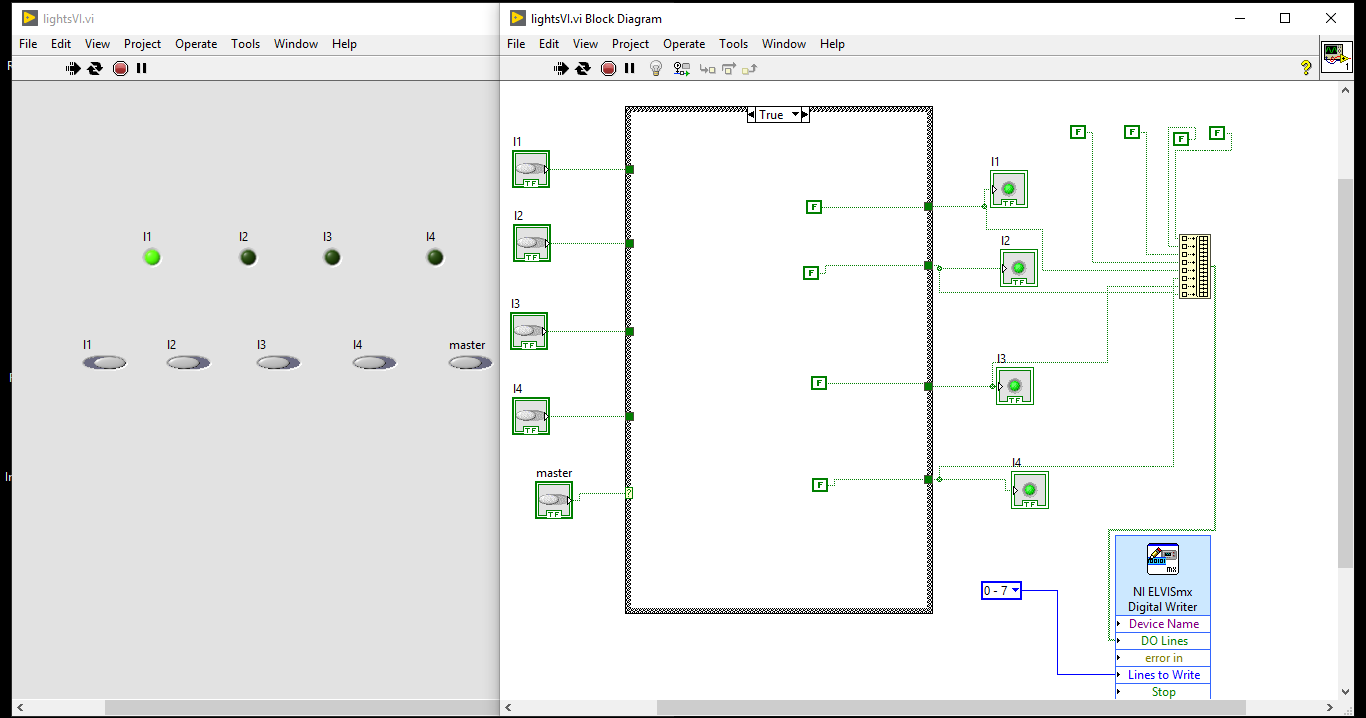


Figure 3 – Lighting System

**Discussion/Conclusions**

Although the objectives of the lab were met, there were improvements that stood to be made in the procedure. Much like how the original temperature system allowed code to be tested before attaching the heat box, a pre-built VI to test a proper circuit would be helpful in attaching the heat box to the NI-ELVIS in order to use it in LabVIEW. Despite that, the VIs themselves were very minimal and cleanly designed, with little room for shortening code. All in all, the objectives of the lab were well met, with both a proper cooling/heating system and a lighting system functioning by the end of it, with circuits and VIs created for both of them.

**Works Cited**

“Introduction to LabVIEW.” Introduction to LabVIEW *- EG1003 Lab Manual*, https://manual.eg.poly.edu/index.php/Introduction\_to\_LabVIEW.

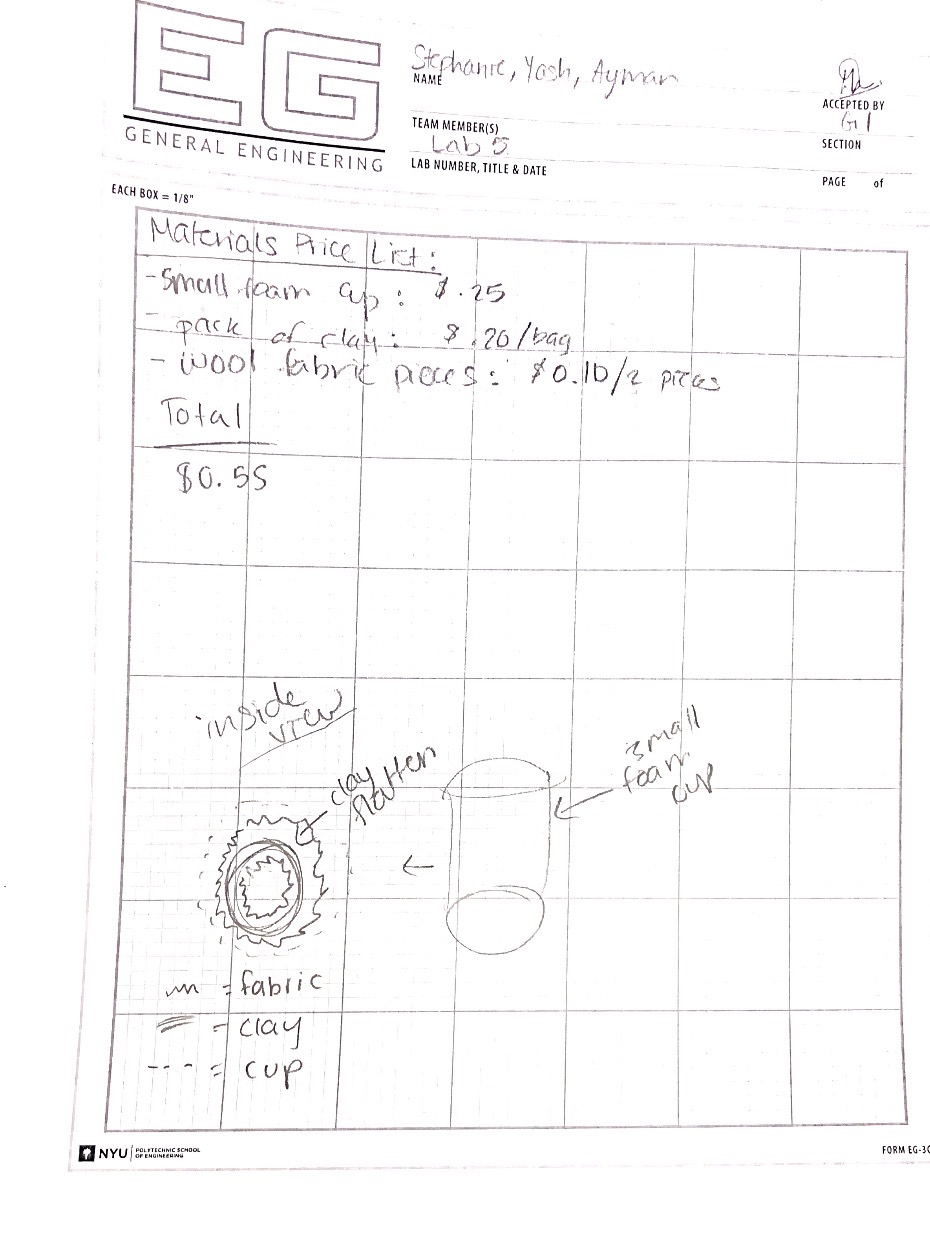


Figure 3 - Lab Notes