Christmas LED Final Project

12/13/2022

College of Engineering: Department of Biological Systems Engineering

BSEN 460: Instrumentation and Control

Caleb Ginting

Carter Hancock

Joshua Arejola

Edwin Thomsen

Summary

Our team set out to design and implement hardware and software that could play Christmas light patterns. The group used supplies found in Chase Hall, as well as LabVIEW software to program Christmas lights. After developing the hardware portion, two separate programs were made using LabVIEW. One program manually coded lights to shine during verses and used the volume of music to shine lights. Another program recorded keyboard strokes made by the user and played back LED flashing patterns exactly as recorded. Our group concluded based on ease of use, customizability, and performance, that recording and playing LED lights was a much more viable option than using sound amplitude or manually coding verses. In the future, our group would like to design, test, and demonstrate how LabVIEW could be used to design a control system that could be manufactured with real controllers.

Background Theory

Our project was focused on creating an LED system similar to LED strips available on the market today. With that in mind, our group decided to extend Lab 6 by creating complex patterns. Our vision was to have programs that could easily be implemented onto a remote control which could make our LED strip have two functions, 1) react to different button inputs to change colors, turn on specific lights, and create patterns manually and 2) react to amplitudes from a sound.

Our project consists of two important elements: the physical component and the coding. Both were essential elements, but the coding aspect did a significant amount of the heavy lifting. There was not much in terms of theory related to the physical component. We decided to use 220 Ω resistors because it was the same resistor from lab 6. This breadboard setup was connected to a MyDAQ microcontroller which changed our digital input into an analog signal turning on the light.

Most of our project's work is the team's knowledge of LabVIEW. Labview is a graphical programming experience to design and test instrumentation controls and sensors. The language is easy for beginners but can take many years to master. Unlike most programming languages that run line by line, LabVIEW uses a framework of icons, structures, and wires to work. Simply put, using LabVIEW is like putting together a Lego building; the user pulls whatever they need from a component box, drags them onto the computer screen, and connects them to wherever a component is needed. LabVIEW divides programs into two parts. The front panel allows users to control the program and see what is occurring with the code. The block diagram allows users to design what a program does using icons, structures, and wires.

Objectives

1. Design a hardware-software system to display Christmas lights on a breadboard
2. Compare and recommend designs

Procedures

17 male-to-male wires, 8 270 Ohm resistors, 8 LED lights, a MyDAQ, a breadboard, and a computer were needed to build the physical structure.

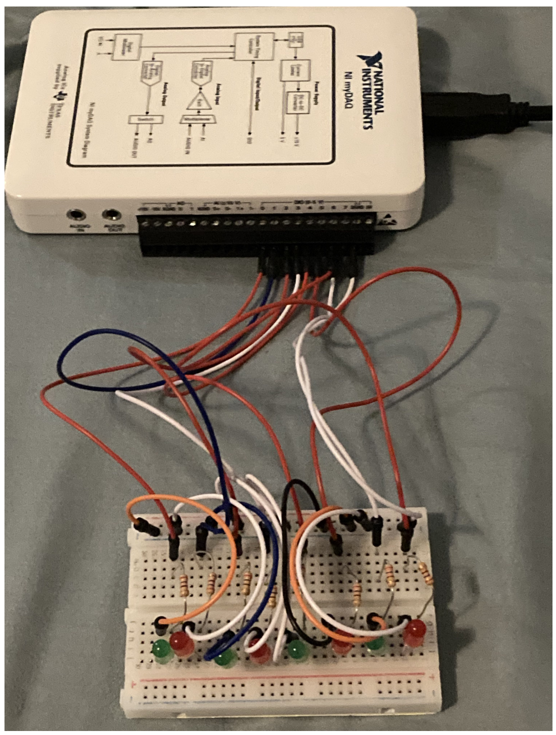

Figure 1. Christmas Lights Apparatus. The MyDAQ (top) is connected to the computer via the cable on the right.

The wires are connected to the DIO 0-7 and DGND, with the leftmost LED circuit corresponding to the leftmost DIO and so forth. There were 8 separate circuits total, each the same. A digital ground channel is connected to a resistor, which connects to the positive LED end. From the negative LED end, a wire connects to the ground rail. The ground rail is connected to DGND.

The LabVIEW software needed two codes. For the sake of conciseness, this report will discuss these VIs in a low-level setting. This is because these VI's contain so many elements that it would be confusing to explain on paper on a high level. Also, the VIs contain so many parts that they cannot be put into a six-page paper. A OneDrive link will be given along with a table so each VI can be looked at at a higher level or for copying.

The first VI, named "460 Proj alpha" plays Jingle bells. The first part of the lights display includes coding that requires a coder to specifically tell the computer which lights to turn on and when to turn them on. The second part of the VI plays the song, and at the same time, the first part of the VI is running. The second part of the VI will also begin shining lights based on the song exceeding a certain volume. If the song was loud, all the lights would be on. If it was quiet, none of the lights would be on.

The second VI, named "recorder" and "player" record and play user keystrokes. The recorder begins a timer and calculates the time in between keystrokes on the keyboard. The VI also turns on LEDs if any of the number keys are pressed. This is so the user can see that their keystrokes are registering. The in-between times are recorded on an excel sheet, and that excel sheet is put into "player". Player VI allows the user to custom design a pattern of LED lights, and those LED lights will move left or right based on the timing recorded in the recorder VI.

Table 1. Information regarding VIs in a OneDrive file. To see the code structure, open each program in the latest version of Labview, click on window, then click on "show block diagram".

https://uofnelincoln-my.sharepoint.com/:f:/g/personal/cgintnig2_unl_edu/ErL5nHWk3W9Jun0uUttuK8YB_o9JQwPCUGA0X0uwBiV3Zw

File	Function
460 Proj Alpha	Plays first verse with manually entered timings, then plays lights based off amplitude
recorder	Recorders keystroke timing
player	Plays keystroke timing while rotating a user

	selected pattern of 3D lights
keyboard	Necessary to initialize keyboard

<u>Results and Discussion</u>

This section combines results and discussion for the sake of comparison. Two software designs and one hardware design were created. The hardware design is exactly as shown in Figure 1. The software, programmed in LabVIEW, created two VIs. The first VI played a Christmas song with lights programmed to light up for verses and audio amplitude. The second VI (actually two VIs) records user keystroke timing and shines the LEDs in the exact pattern played.

After 3-4 days of coding, each VI worked as intended. However, there are some notable differences between each VI relating to practical application. “460 proj alpha.vi” plays Jingle Bells and shines lights going along with the song very well. The manual programming of lights was not exactly in sync with the song, but it roughly corresponded with the lyrics. The second mode of lights which displayed lights based on audio amplitude was less effective. The background of the song picked up a lot of noise, causing the lights to flicker on and off in an incoherent pattern; the parts of the song with a lot of instruments had a more incoherent pattern than when it was just the main singer.

The combination of the recorder and player VI accurately recorded the keystrokes of the user and played them back in a user-given light pattern. However, there were still some problems. The first is that LabVIEW can only register keystrokes so quickly, so some melodies were impossible to play on the keyboard. Another issue is that the keyboard pattern offers great customizability. This is because each pattern that can be played can be recorded on the spot. The recorder-player VI only took the length of time the user wanted to spend recording, but the first VI took over 2-days to record the first verse of a song.

Below is our analysis table of the practical criteria of each of the programs used to create the Christmas patterns.

Table 2. Practical Criteria of each VI

VI System	Ease of use	Customizability	Performance
460 proj alpha	Easy; plays with no user input	Hard: to program the verses of a single song, it may take days or even weeks	Moderate: there are rough discrepancies between when the song and when the lights glow up
recorder-player	Easy-Moderate: if the user can think of a rhythm to play, it can be recorded	Easy: songs can be made in the same duration as the song	High: The VIs replicate the user's pattern with high fidelity

Overall, it would seem that the recorder-player VIs are the better option for programming lights. This is because it offers higher customizability and fidelity to whatever the user wants to create.

If a device were to be manufactured in real life, a simple remote control-LED system would adequately replicate the recorder-player VI. The remote would have buttons to begin and stop recordings and to save the recordings to a slot. That slot could then be dedicated to a button. When pressed that button would then begin playing the pattern that the user made by clicking on the remote.

Conclusions

This project successfully created two programs in LabVIEW and one hardware design to display Christmas lights. Based on the ease of use, customizability, and performance of each software, our group recommends that the recorder player be used if it were to be marketed to the public. To continue on with the recorder-player, one would need to design a control system that allows someone to easily set up, program, and view their own LED patterns. This could be done with a controller-LED system, but more will need to be researched to find the best way to make such a control system.