QiLin Xue, xueqilin, 1007000610, Partner: Andrew Li

Multimodal Lighting System Prototype for Greenhouse Plants

I. INTRODUCTION

There is a need to create a multimodal lighting system in greenhouses to account for both worker comfort and optimal plant growth conditions.[1] We have designed a widget prototype to model how such a system may look like. Our design showcases how a single button can be used to control the light settings, and can be easily scaled up and be modified.

To build our prototype, we used a Raspberry Pi Pico as our microcontroller, and 3D printed the case using PLA filament.

II. CONCEPTUAL DESIGN PROCESS

As this is a prototype, we operated with the design philosophy of keeping everything as simple as possible. Our product should be a proof of concept that such a multimodal lighting system can work and is easy to operate. Because of this, we avoided the temptation of designing a proof of concept for other features that may be unnecessary.

A minimalist simple design will also be cheap to manufacture, which can allow for different designs to be prototyped quickly. We used this overarching design principle to guide the design of the structural, electrical, and lighting control systems.

A. Structural Component

There were two main considerations when designing the structural component. First, the casing must fit the breadboard and electrical components. For example, openings must be created for the LEDs and the button. These are considered in our first conceptual design shown in figure 1.

However, in line with our overarching design principle, we decided to cut back on unnecessary components, such as the overlapping cover. Instead, we opted for a simple lid. Instead of creating a complex geometry for the holes, we instead created holes for each. These changes were reflected in our second sketch in figure 1, and is very close to our final design.

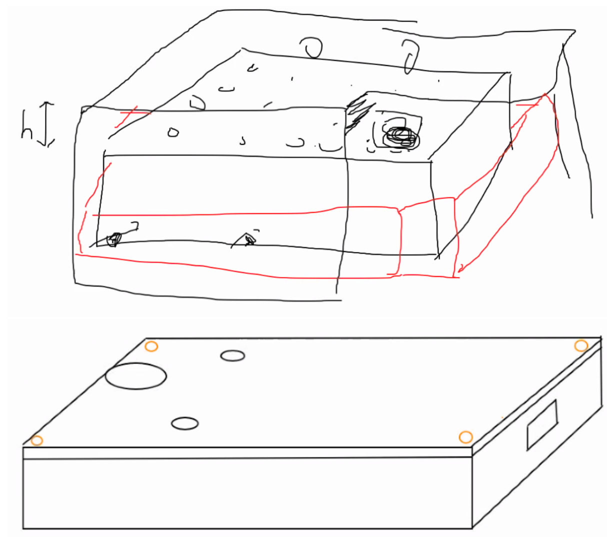

FIG. 1: Early conceptual sketches of the structural element.

B. Lighting Control

The most straightforward method of controlling the light is to use a single button to loop between the three possible states, as shown in table I. However, we found that this doesn't make sense on a practical level. When a worker leaves the greenhouse, they will need to press the button twice. In other words, to enable the plant only lighting, one needs to turn off all the lights first. Even if the button loops in the other direction, the worker needs to turn off the plant lights before turning on the lights every time they enter the greenhouse.

Mode	Red LED	Green LED
1	OFF	OFF
2	OFF	ON
3	ON	ON

TABLE I: Behaviour table for the LED modes. Each button press should advance to the next mode. After mode 3, it loops back to mode 1.

The idea of turning off lights in order to turn on lights seems very hostile and unintuitive. Furthermore, the original paper that inspired this design challenge suggests that the supplemental LED lights need to be on for 16 hours a day, so they only need to be turned off once a day. Combined with the fact that the LEDs will be on for the majority of the day, having to turn them off temporarily when a user enters/exits the greenhouse doesn't make any sense.

Instead, we propose that a single button press should alternate between two states: worker & plants, and plants only. To turn it off, the button should be held down for two seconds. This design aligns with common design standards most people are used to. For example, we tap a single button to toggle between turning on and sleep mode for our electronic devices. If we wish to shut down the power, (i.e. for an iPhone or most laptops), we need to hold down the button.

The behavior table for our improved design is shown in table II and its corresponding finite state machine diagram is shown in figure 3. However, there are a few subtleties. In order to design for familiarity out of empathy for our test users, the corresponding change by holding down the button is immediately reflected after two seconds, not when the button is released. This way, the user does not need to estimate how long two seconds will take to pass before the button is released.

Previous State	Action	Next State
Worker or Plant	Hold	Power Off
Power Off	Press or Hold	Worker Mode
Worker Mode	Press	Plant Mode
Plant Mode	Press	Worker Mode

TABLE II: Behaviour table for the LED modes using a more user centered system. Each action is determined when the button is released. The default mode when the system is turned on is worker mode because that likely corresponds to the worker arriving to the first shift of the day.

III. IMPLEMENTATION

A. CAD Model

Our CAD model can be seen in figure 2, where the top lid was made translucent to better visualize where the breadboard fits. The measurements, while using as much care as possible to ensure is accurate and precise, were taken to be conservative when possible. This, along with expanding the space for the breadboard by an extra 1%, ensures that the breadboard will fit in the case, even if the measurements were slightly off or the 3D printers introduced some error. We made two key changes from the conceptual design shown in figure 1. First, the bridge between the USB slot and the lid was removed.

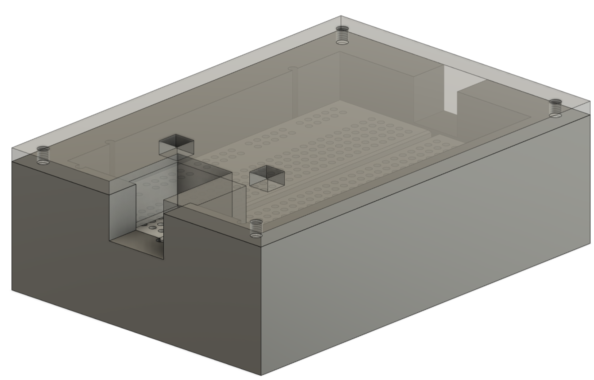

FIG. 2: CAD model of the structural component.

This was because in order to print the bridge with good results, specific changes need to be made to the printer. This extra work will by annoying to the MyFab staff, who are dealing with hundreds of orders, and will delay the prints of other teams. Therefore, we decided to keep our design as simple as possible to reduce the strain on the printing system.

Second, the slot for the button was extended to the edge, and extra vertical space was given. By having the button be in an easier to reach spot and having the slot shaped like a human finger makes the usage of the button less awkward and more intuitive.

B. Software

Careful work was done to ensure the software was not only functional, but easy to change by other people who may or may not have a programming background. The mapping of each state to the LED states and the mapping of how each action should change the state are represented using two Python dictionaries. Having these rules being in a human friendly format makes it easy for people without a technical background to adapt the software to their needs.

The rest of the code was written very similar to how a finite state machine operates, with both a control path and a data path, and follows the updated behaviour state diagram shown in figure 3. By following standard practices, it is easier for future designers to understand the code and make changes to it.

C. Electrical Circuit

The initial design for the electrical circuit was shown in figure 4. However, changes due to practical reasons had to be made when taking the structural component into consideration. Namely, to prevent collisions between individual wires and the casing, both LEDs had to be connected to the same side of the Raspberry Pi, as shown in figure 5. To choose the resistor, we used the data sheet and the lab manual[1][2] and applied Ohm's Law. The calculations can be found in a separate document.

IV. EVALUATION

The finished product is shown in figure 6, and a full demonstration of the capabilities of the system is shown in the linked outlook folder containing video demonstrations. We are able to verify that the objectives were met. Namely, the casing covers the majority of the breadboard while leaving the LEDs and buttons visible. We have also demonstrated from the physical prototype that rotating the breadboard will not shift the elements, and the casing can be constructed without the use of adhesives.

A. Qualitative Considerations

There were several qualitative issues that we found while interacting with the physical prototype ourselves, which while minor, should be addressed moving forward. First, the case appears to be symmetrical, with an opening for the button on one of the long ends and an opening for the USB on the other end. However, it is not, and the case will not function if the lid is flipped. We found this out while trying to construct the prototype. It was very easy to confuse directions. While this is only a minor inconvenience (similar to putting in the USB in the wrong orientation), we should make future designs actually symmetrical so orientation doesn't matter, or at least make it obvious to the user which orientation the lid should be in.

Secondly, the case was designed and tested by our team, which is consisted of Asian males with fingers that may be slightly skinnier than average. Thus, the calculations for our

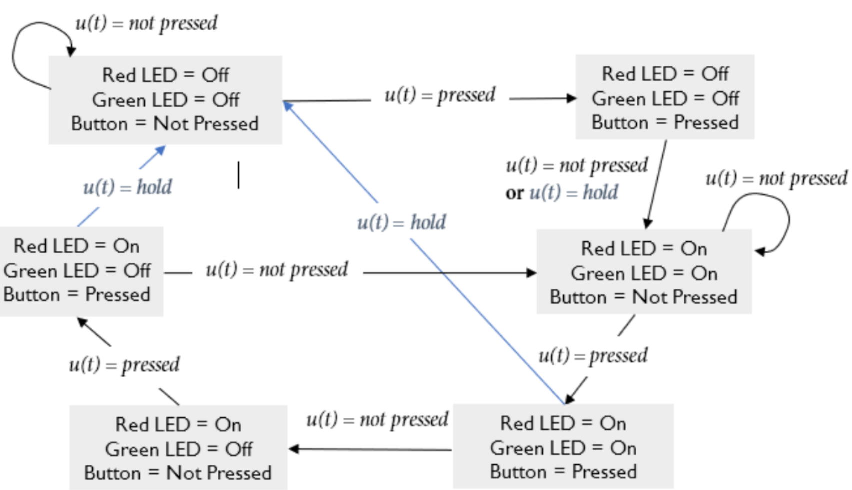

FIG. 3: Behaviour state diagram for our most updated design. The complexity at the technical level is needed in order to ensure a more intuitive user-centered design.

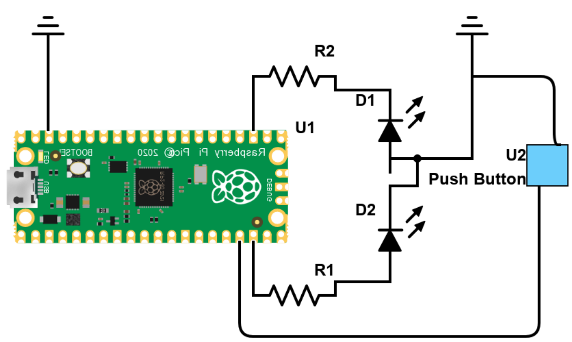

FIG. 4: The electrical circuit diagram for our initial design.

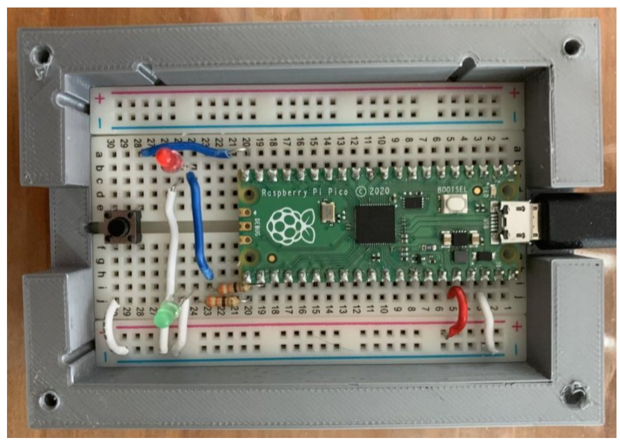

FIG. 5: The electrical component of the lighting control system of our updated design. White wires connect to the ground, red wires connect to the power supply, and blue wires connect circuit elements to each other.

CAD model reflected our finger sizes. Even though we found the slot for the button to be the perfect size for us, that may not be true for people who are on the opposite side of the spectrum. We did not consciously consider how someone with larger fingers would use our product, and how it may make them uncomfortable. In future designs, we must practice more empathy and ensure the hole is big enough or even better, find an alternative method of pressing down the button (such as with a plunger).

Finally, we found that we were a bit too conservative with how high the case should be. We did not want the lid to interfere with the wires and the Raspberry Pi, but we made the case a bit too tall. While the LEDs still stick out of the case, it is definitely possible to lower the case in order to make the LEDs stick out more in order to be more visually appealing, and illuminate a greater region.

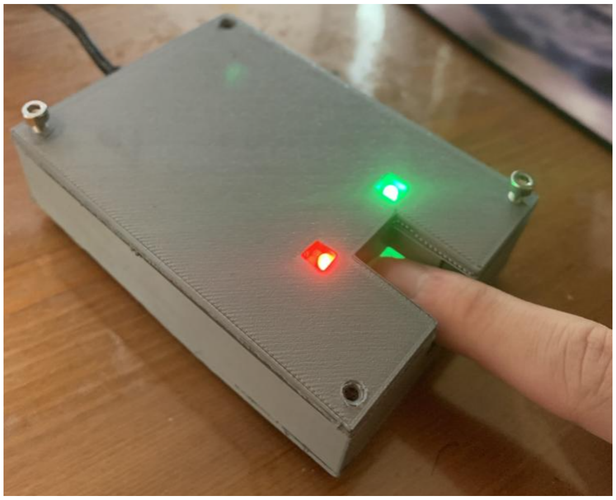

FIG. 6: The final 3D printed prototype in the worker mode. Note that only two screws were installed since we only had access to two screws. The effects of material constraints and our own poor manufacturing devices (not the MyFab printers) are discussed in section IV C.

B. Verification of Control System

To verify that the controls work as intended, we manually tested that every single action to every possible state provided the correct response. Since the logic of our program was more complicated than a simple loop, we had to be extra careful there are no logic gaps. This was done through both testing through as many edge cases as we can think of, and writing several unit tests for automated testing.

First, since our button had to respond to time-sensitive data, we tried several extreme cases. For example, we tried rapidly clicking the button several times in fast succession and tried holding down the button for a long period of time. When we held down the button continuously, the LED would first turn off (as intended), then turn back on and then off. This was because the state had been changed to `OFF` and doing a `HOLD` action would turn the lights on. To fix this, we had to ensure that when the system turns off, further actions can only be registered after the button has been released.

Second, we were able to create an automated and comprehensive unit testing system that checks all the desired behavior as well as extreme cases. This is done by instead of passing through signals via the circuit, we manually passed through actions by giving it `button down`, `button up`, and `delay` commands. This unit testing was written not just to verify the logic of the program, but out of empathy for future designers who may wish to modify the program. These future designers will be able to quickly verify that their changes do not break the overarching logic immediately, instead of through manual testing.

C. Technical Issues

The structural component was submitted to 3D printing at MyFab, but the prototype we present in this report was printed from one of our team member's own 3D printer. Our 3D printer was not calibrated properly, so the walls were rough and a bit curved. Secondly, the case was designed in CAD to be put together using the recommended M3x6 screw. However, we did not have any in stock and had to use M3x10 screws. For these reasons, our case was not very polished, and the screws stick out because they were too long.

However, we are confident that given access to the proper materials, the finished product will be better presented. This is because even with the inferior manufacturing process and the inaccess to the right screws, the case still meets on the requirements set forth.

D. Individual Contributions

This widget was created through a collaboration of my partner, Andrew Li, and I. We worked on the conceptual design process together, and we took turns doing the CAD, going through several prototypes. Because I had access to a 3D printer at home, I was able to construct and test the physical prototype. We both worked on programming, though I wrote the latest version and the unit tests. Several of the diagrams, including figure 4 and 3 were made by Andrew, and he made the circuit measurements in order to pick the resistor and verify the circuit was working.

V. REFLECTION

While I had programmed before, worked with CAD, as well as electronics, I had never worked with everything integrated with each other. With so many components working together, I learned that at every step, I need to look at the bigger picture and try to visualize how different pieces fit together. This made me develop better CAD skills as I could no longer use my previous lazy practices, especially working with a partner. I learned that with careful planning of how the CAD model is created, we can easily make changes without having to redo a lot of our past work.

One thing that was completely new to me was working with microcontrollers. It was very fascinating to see how I could have complete control over the electrical components using a programming language I am very familiar with. In fact, it's the first time I wrote a program that interacted with the real world, instead of solving some hypothetical digital problem. This was made further interesting by the fact that I took ECE253 last semester, which had an overview of how microcontrollers work on a low level. Being able to relate the theory we learned last semester with practice was very helpful.

Other than being able to relate topics to what we learned last year, I learned a lot through both the internet and my partner. For example, I consulted the internet to better learn about how to create screw holes. I found that fortunately, similar to programming references and resources, there was a plethora of resources for CAD modelling in general and specific to all the different software out there. My partner was also very skilled with electronics and gave good advice of how to arrange the circuit. We learned from each other when we worked on the CAD together. Similar to pair programming, having another set of eyes can help you look at the problem differently and suggest a solution you haven't thought of.

What interests me the most is the potential that I could do with microcontrollers. I opted to challenge myself further in this lab and create a more advanced lighting control system. However, with countless other input and output devices out there, I want to create something using microcontrollers that is bigger in scale and has a better practical use. Hopefully, I will be able to accomplish this goal in the big design project for Praxis this year.

Not only did this design process relate to course content I learned in ECE253, but it reminded me a lot of the design project I did in Praxis II, where we made and 3D printed a grafting tool for roses that people with Parkinson's can use. I didn't know a lot of CAD when I did that project and operated under a "if it works, it works" principle. As a result, the finished product was very simplistic and awkward to hold. I believe that if I redid that project, I could create a design that is more sophisticated (by fitting to the rose better, perhaps using a rose 3D model), easy to modify, and more human-centered.

Next time, I would aim to be more ambitious with the design such as incorporating a fastening method that doesn't require screws or adhesives. One idea is the use of a snap-fit joint, which use the slight flexibility of the printed plastic to help connect two parts together. This design will be much harder than inserting four screws, and I suspect it must be carefully designed in order to work. However, given that I have time, I will be more than willing to take on this challenge.

[1] “Widget lab 1 assignment.” Praxis FaCT, Jan 2022.

[2] T. Cooper, “The led datasheet.” Ada Fruit, Feb 2022.