**Studio 1A Challenge Scenario**

## **Introduction:**

The purpose of this report is to present a simple light fixture that operates in three modes to promote proper growth of crops in greenhouses at northern latitudes. The goal of the design is to allow for optimal lighting conditions for plant growth in a greenhouse, while allowing a switch to a more worker-friendly setting when workers are around, and an option to turn the system completely off.

## **Stakeholders:**

The primary stakeholders are:

Myself: This studio challenge activity will allow me to practice making conscious system design decisions, and obtain experience implementing electronic circuits that respond to human input

Company: The (hypothetical) company I work for wants a design for this product that is functional, efficient and minimizes cost. They also want prototype to be scalable and intuitive

Plant and Workers: The product is being designed for these groups. The success of our prototype will depend on the extent to which they are content with our design.

## **Requirements:**

Objective: Implement a system where input is button press and output is LED color assignments; series of button presses result in series of state/mode changes depicted below

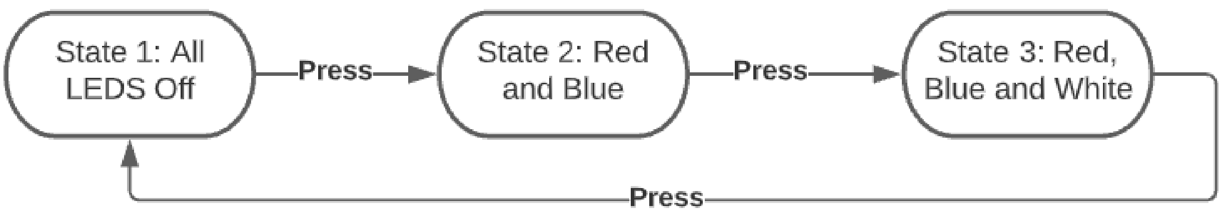


*Figure 1: State diagram of system. Box represents state, arrows are conditions for state change*

Constraints:

Equipment Constraints: We are limited to 2 buttons, 6 resistors, 3 red LEDS, 1 breadboard, 1 Pico, 3 jumper cables and 10 wires.

Design Constraints: The input response must follow Figure 2; LEDs must toggle at the positive edge of button input (i.e: immediate when pressed). To minimize printing time and save time/cost for the company, the entire design should fit halfway (94mm) across the breadboard.

Criteria:

- Design should take the smallest space on the breadboard, while still being easy to use.
- Wires should be as short as possible without risking failure, to minimize costs
- Product should be intuitive to use in terms of functionality and placement of lights

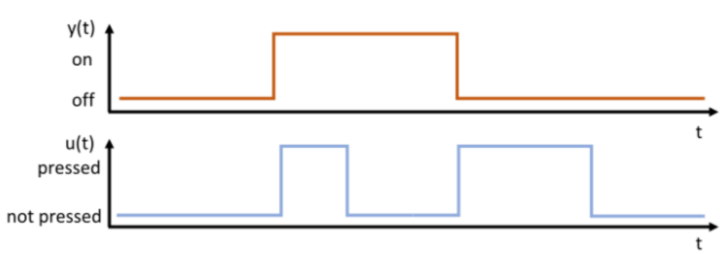


*Figure 2: Timeline diagram of button input response*

**Design Implementation:**

Picking current value: The current must be enough to turn on the LED but not so high as to exceed the maximum recommended current drawn by Pico pins. Current to turn on a LED is subjective, but 5-10 mA is easily visible[1]. Current limit on a single GPIO pin also has no specific value, but anything below 20mA should be ok. At default drive strength of 4mA, voltage at GPIO pin when current is 5-10 mA is ~3V, which is higher than minimum voltage to be considered “on”. So we can select any current i where 5mA <= i <= 10mA. Lets select **i=5mA**

Calculating resistor value: We will calculate under ideal conditions. Vs = 3.3V (votlage at GPIO pin). Vf = 1.85V (from LED datasheet). i=5mA (justification above). Finally:

$R = (V_s - V_f) / i = (3.3-1.85)/ 5*10^{-3} = 290$ ohms.

The resistors given to us are R1 = 1000 +- 1% (x2), R2 = 220 +- 1% (x2) and R3 = 330 ohms (x2). So, we pick the closest ones to 290 ohms: **2 R3 resistors** and **1 R2 resistor**. This gives a current of (3.3-1.85)/330=4.4mA and (3.3-1.85)/220 = 6.6mA, both of which are easily visible.

Button Implementation: We have two design choices here: single button to switch between modes, or 3 buttons (one for each mode). Since we only have 2 buttons, this is easy: we pick a **one-button approach.** This is also easier to use- workers will just have to press one button till they get the configuration they want, rather than remembering which button corresponds to each mode. It also restricts costs when we consider scalability, and helps simplify design while limiting space on the breadboard. If there were more lights, this approach may have been troublesome(keep pressing till succeed), but with a 3-state system, we can get to any configuration with a maximum of 2 button presses.

LED arrangement: For aesthetic appeal, we will align these all in one row. The 3 LEDs we have are all the same color; this is fine as this is just a prototype. Another detail is spacing the LEDs as close as possible to each other and centralizing in the greenhouse to obtain a uniform-color, uniform-intensity light source throughout the greenhouse.

Software Implementation: We value modular code. So, we have separated the configurations into 3 functions; each one specifies led values for a different state. In order to have the system accept input for all time, we embed main code in a while loop. In this loop, we wait for a button press and change state every time we detect a downwards button press(requires a “holding” variable, see code.py file attached)

**Final Product and System Verification:**

Final product is shown in Figure 3. To verify correct implementation, we use a multimeter to measure current through each LED. The current through the LEDs for the 2 different types of resistors are 5.16mA and 6.8mA, which are close to theoretical values (error due to tolerance, external noise, multimeter was fluctuating a lot before settling, etc). So, this design serves the original goal, and can be scaled to greenhouses at northern latitudes.

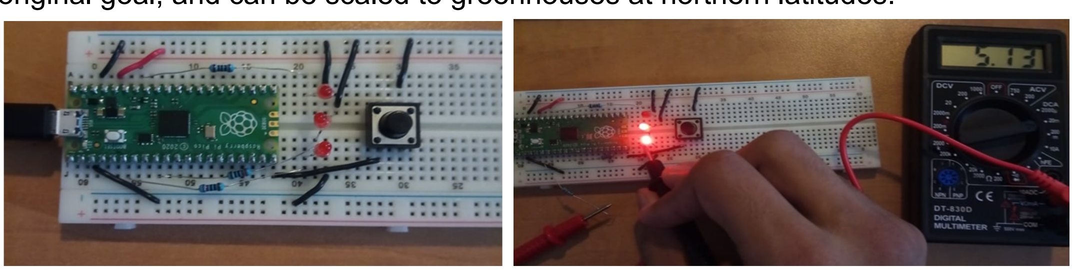

*Figure 3: a) Final System Prototype and b) System Verification*