

ESP32 Lighting Controller

CS39440 – Major Project

Kieran Todd [ktt1] - 170040473

Richard Shipman [rcs]

G400 – Computer Science

1.0 Release – 07/02/2021

Project Description

The aim of the ESP32 lighting project is to create a piece of software that easily gives the user a way of configuring and displaying multiple different LED and motor sequences when a condition is met. The lighting controller will have an easy-to-use web based front end for the user to assign lighting profiles to the different LEDs and the conditions they will start lighting up under, when and how motors will spin and to create and edit lighting profiles that the LEDs display.

There are 3 different LEDs that the ESP32 will have to display a brightness to, which are addressable LEDs, PWM LEDs and standard LEDs. The standard LEDs are pretty basic in their functionality as they have 1 colour and can only be set as HIGH (on) or LOW (off). The Addressable LEDs are RGB LEDs which means along with saying if they are on or off, they will have a value and will send that value to the LED and display the colour associated with that value. Lastly we have PWM LEDs, PWM stands for Pulse Width Modulation which means that the LED is turned off and on multiple times every second and depending on the percentage of the second the LED is on will depend on how bright it is. While addressable and standard LEDs come as a singular LED, PWM LEDs normally come as a string of LEDs.

When setting the LEDs profiles and changing the motors configuration you will have the option of setting the LED/motor(s) reactant. The reactant determines when the LED/motor(s) will start playing either the LEDs profile or what is set for the motor to do. The different reactants are time, weather and sensors. The sensor reactant is the simplest reactant, when a sensor is tripped (when there is movement in its range) it will trigger the specified output set by the user for the LED/motor. When the reactant is set to the weather the LED/motor will start its output while the weather in the region matches the weather specified by the user (e.g. spin motor 3 while weather equals raining). Lastly, when the reactant is set to be timed, the LED/motor will perform the specified output while the current time is equal or between two different time slots specified by the user.

Possible Extras

As the scenes people models completely differ, there are a lot of different possible reactants and outputs people might want for their different scenes. Something that could be implemented is a short music clip that could be played when the LED profile is played. Also the reactants possible depends on the hardware connected and the APIs that have been added to the software, for instance if a button is connected to the LED profile could be played when a button is pressed, or if a twitter API is implemented you could have a motor spin whenever you get a new follower.

Proposed Tasks

Firstly I need to thoroughly look through the process of making and hosting a website that is visually appealing on the ESP32, to do this I will have to brush up on my CSS and do some

spike work and pseudocode planning as to how I will run the lighting controller and the website in parallel.[1]

Also, as the LEDs and motors will have to react and preform their desired actions when it is a certain time, or a weather condition is met I will have to look up an API or an SDK for the weather [2] and the Clock functionality [3].

Next I will have to do some experimenting with PWM LEDs[4] and addressable LEDs[5] as I am still unsure how to actually output the correct colour/light intensity to the LEDs.

Also, I wanted to do some research into the customers this project is designed for, so I did some research into a model maker who came to my supervisor with this project idea. [6]

Lastly I wanted to do some research into the current lighting devices used for lighting in models [7]. From what I found I saw that at the moment the devices being used are very limited in terms of customisability. One of the more popular devices in the model community is one by Malcolm's miniature's, which only has 2 lighting profiles, one for fireplace and one for a light.

Project Deliverables

Simple and easy to use website that looks good – As the intended audience (Model Creators) may not have in depth knowledge on websites and coding, it has to be able to be used by everyone, with a good look and feel.

LEDs and motors respond to changes the user makes on the website

Final system – A system that meets all of the points discussed earlier in the document which will be programmed in the Arduino IDE. I will be able to demonstrate the system fully meeting all of the points discussed and with working examples.

Documentation – The documentation will cover all aspects of setting up and operating the system, including the initial setup of the system, creating new LED profiles, how to assign them to LEDs and the conditions you can set the LEDs and Motors to activate.

Testing documentation – Documents containing the outcome of the System, how it meets my initial expectations of the system and if it falls short, where and why it does

Final Report – The final report is a full report containing the process I would have went through making the system, the changes I would have made along the way and then a full description on what the system does and how it operates. The report will also contain a full comparison from the final report to what I expected to make, including the testing documentation and how I would develop the system further if I had the time following this project.

Bibliography

[1] When it comes to hosting the webserver from the ESP32 I have done some research into how to host the website and how it can be accessed, most of the information I have obtained has come from <https://randomnerdtutorials.com/esp32-web-server-arduino-ide/>

I have yet to do any research into CSS.

[2] When it comes to the weather API I have had a look around and one of the better more recognised weather APIs would be Open Weather : <https://openweathermap.org/>

This will return hourly weather temperatures and weather forecasts for the users area.

[3] I have done some research into some Clock APIs that will return the time one of them being: <http://worldtimeapi.org/>

[4] <https://randomnerdtutorials.com/esp32-pwm-arduino-ide/>

[5] <https://www.hackster.io/donowak/internet-controlled-led-strip-using-esp32-arduino-2ca8a9>

[6] <http://www.nemo.me.uk/index.html>

[7] <http://www.malcolmsminiatures.co.uk/Lighting+controls>