

Bachelor Project



**Czech
Technical
University
in Prague**

F3

**Faculty of Electrical Engineering
Department of Measurements**

Intelligent LED illuminator for growing plants sensitive to light conditions

Lukáš Kornel Gajdár

**Supervisor: Ing. Radek Sedláček, Ph.D.
Field of study: Cybernetics and Robotics
May 2022**

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Gajdár** Jméno: **Lukáš Kornel** Osobní číslo: **492371**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávací katedra/ústav: **Katedra měření**
Studijní program: **Kybernetika a robotika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Inteligentní LED osvětlovač pro pěstování rostlin citlivých na světelné podmínky

Název bakalářské práce anglicky:

Intelligent LED illuminator for growing plants sensitive to light conditions

Pokyny pro vypracování:

Na základně předchozí rešerše odborné literatury navrhnete a realizujete funkční vzorek inteligentního LED osvětlovače na podporu růstu i květu rostlin citlivých na světelné podmínky jakými jsou např. orchideje. Zvolte plně škálovatelné řešení, co se týče volby požadovaného výkonu použitých LED, jejich barevného spektra i řídicího obvodu. Pro konstrukci celého zařízení zvolte vhodný mikrořadič, který umožní nejen řízení použitých LED, ale též i bezdrátovou komunikaci s ovládacím softwarem pomocí Wi-Fi či Bluetooth technologie. Vytvořte software, který umožní celé zařízení ovládat a uživatel bude moci v rámci týdenního plánovače nastavit dobu osvětlení, zvolit barevné spektrum i intenzitu osvětlení.

Seznam doporučené literatury:

- [1] MATOUŠ, M. – HUTLA, P.: Světlo a rostlina. Světlo, 2002, č. 4.
- [2] HLADKÝ, Luděk: Osvětlení z pohledu rostlin. Světlo, 2010, č. 4.
- [3] POSPÍŠIL, Tomáš. Osvětlování rostlin pomocí technologie LED [online]. Brno, 2014 [cit. 2022-02-09]. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_veřejne.php?file_id=86118. Bakalářská práce. VUT v Brně - FEKT. Vedoucí práce Ing. Michal Krbal.
- [4] RITTERSHAUSEN, Sara. Zdravé orchideje: aby dobře rostly a krásně kvetly. Přeložil Markéta SCHUBERTOVÁ. Praha: Euromedia Group, 2019. Esence. ISBN 978-80-7617-342-2.
- [5] ESP32 Module Overview [online]. [cit. 2022-02-09]. Dostupné z: <https://www.espressif.com/en/products/socs/esp32>

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Radek Sedláček, Ph.D. katedra měření FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **11.02.2022**

Termín odevzdání bakalářské práce: _____

Platnost zadání bakalářské práce:

do konce letního semestru 2022/2023

Ing. Radek Sedláček, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Acknowledgements

I would like to thank our Lord and saviour Jesus Christ for guiding me on this difficult journey we call life.

Declaration

I declare that this work is all my own work and I have cited all sources I have used in the bibliography.

Prague, May 16, 2022

Prehlasujem, že som predloženú prácu vypracoval samostatne, a že som uviedol všetku použitú literatúru.

V Prahe, 16. května 2022

Abstract

Growing number of people around the globe are becoming interested in growing plants either for food or enjoyment. Living in small apartments limits the types of plants one can grow because of the less than ideal lightning conditions. Limited space issues also makes people want to maximise the available growing area and yield. The goal of this thesis is to create an LED illuminator that would allow people to grow plants sensitive to light conditions. Firstly I will examine what spectrum and intensity of light do generally plants need. Then I will propose what specifications and intelligent functionalities the illuminator should have and the principle behind them. Lastly I will build a prototype based on these recommendations.

Keywords: LED, lux, PPFD, WIFI, BLE

Supervisor: Ing. Radek Sedláček, Ph.D. Assistant Professor at the Department of Measurement, CTU in Prague

Abstrakt

Rastúce počty ľudí po celom svete sa začínajú zaujímať o pestovanie rastlín. Buďto pestovanie pre potravu alebo potešenie. Malé byty limitujú aké typy rastlín môže človek pestovať, najmä kôli nevyhovujúcim svetelným podmienkam. Limitovaný priestor taktiež núti ľudí hľadať spôsoby, ako maximalizovať úrodu a vhodný priestor na pestovanie. Cieľom tejto bakalárskej práce je vytvorenie inteligentného LED osvetlovača, ktorý umožní pestovanie rastlín citlivých na svetelné podmienky. V prvom rade zhrniem dostupnú literatúru a predložím, aké spektrum a intenzitu osvetlenia rastliny potrebujú. Potom navrhnem špecifikácie a inteligentné funkcie, ktoré by daný osvetlovač mal spĺňať, a popíšem ich funkcie. Na záver zostrojím prototyp daného osvetlovača na základe spomenutých špecifikácií.

Klíčová slova: LED, lux, PPFD, WIFI, BLE

Překlad názvu: Inteligentní LED osvětlovač pro pěstování rostlin citlivých na světelné podmínky

Contents

1 Introduction	1	3.2.1 Hardware diagram	23
2 Theoretical background	3	3.2.2 PCB design	23
2.1 Ideal light spectrum for plants . . .	3	3.2.3 Case	25
2.1.1 Measuring illumination	5	3.3 Software	26
2.1.2 Picking light spectrum for my application	6	3.3.1 Scheme for software solution .	26
2.1.3 How do LEDs work?	7	3.3.2 Web server	27
2.1.4 Creating white light using LEDs	9	3.3.3 Bluetooth	30
2.1.5 Blue and Red LEDs	10	3.4 Light brightness map	34
3 Implementation	11	3.5 Price comparisons	35
3.1 Choosing components	11	4 Conclusions	39
3.1.1 Selecting the right LEDs	11	A Contents of included zip file	41
3.1.2 LED driver	14	B Bibliography	43
3.1.3 Micro controller unit (MCU)	17		
3.1.4 LUX meter	19		
3.1.5 LED cooling	20		
3.2 Hardware	23		

Figures

2.1 Comparison of wavelengths necessary for photosynthesis and for overall growth of plants [20]	4
2.2 Comparison of PAR and Lumen [21]	6
2.3 Wavelengths of different white color temperature LEDs from CREE [29]	7
2.4 Relative spectral emission of blue and red LEDs from OSRAM [11][10]	7
2.5 Illustration of electron passing from conduction band to valence band [3]	8
3.1 Relative luminous flux fro white LED[29]	13
3.2 One LED driver, there are 5 of these on the PCB	15
3.3 AL8843 efficiency[1]	15
3.4 LM2576 schematics for fixed voltage of 5 V	18
3.5 Spectral responsivity of the two photodiodes of TSL2561 [28]	19
3.6 Thermal image of the heat sink, when LEDs were at 100% power.	22
3.7 Thermal image of the heat sink, the 6 hot spots are placed where they are expected, above the white LEDs	22
3.8 Hardware schematics	23
3.9 PCB top (right) and bottom (left) side	24
3.10 LED PCB naked and with Blue and Red SMD LEDs soldered onto it	25
3.11 3D model of the illuminator realisation	25
3.12 Placement of the LEDs on the heat sink, sides of the reflector are sprayed with chromium spray for better reflectivity	26
3.13 Diagram	27
3.14 Web interface on desktop[24]	28
3.15 BLE terminal example, showing basic functions	34
3.16 Lux maps, showing how uniformly the light is distributed, grid scale is 10 cm, distance between the illuminator and the lux sensor at the center of this mas was 25 cm	35
3.17 Final illuminator and comparable comercial solutions	37

Tables

3.1 Comparison of PWM and Analog dimming of LEDs	16
3.2 Price of the LEDs	36
3.3 Price of the LEDs, grow area is in cm	36



Chapter 1

Introduction

People living in the cities were long unable to grow most light-sensitive plants because their apartments do not get enough sunlight. Advancements in LED technology might change this. LEDs and the complimentary electronics needed to drive them are becoming cheaper every year. This leads to more widespread adoption of this technology for all sorts of lighting applications. One of these applications is horticulture. To maximize yield, some indoor greenhouses are utilizing some sort of artificial lighting to supplement the sun's light or replace it altogether. The events of the last couple of years have shown the world how vital food production and supply are. Using greenhouses with artificial lighting is one of the proposed solutions. This thesis will focus on all of these issues on an individual scale. I will propose a modular LED illuminator that will be able to provide plants with the light needed for their healthy growth. This solution will be easy to use, scalable, modifiable, intelligent, and inexpensive. The goal is to create an LED illuminator that can be used for plants throughout their lifespan, from germination, through flowering, until possible harvest. The solution will also work for plants that are for decoration only. Different plants have distinct needs for the light spectrum. Therefore the illuminator needs to have the option to change its spectrum to be useful to the most amount of plants. Adjusting the brightness of the light is also essential, as some plants are more shade-loving while others thrive in the full sun. The best way to adjust LED brightness is to use a constant current LED driver. Users have to be able to change the current to satisfy the needs of the plant that is currently being grown. To do so, analog dimming or PWM dimming can be used. For the reasons discussed below, I opted for the latter. The microcontroller unit will create a PWM based on the user's inputs. To give commands to the MCU, ESP32 in this case, there are two options. One is to use the onboard antenna to connect to WIFI, and the ESP32 will act as a server. The connected clients will be able to set

ON/OFF timers and the required brightness of the LEDs. Another one is to use the antenna for Bluetooth. The ESP32 will again act as a server, and the connected client will have the ability to change the brightness of the LEDs. Bluetooth mode will serve as a backup if WIFI is not available.

Chapter 2

Theoretical background

2.1 Ideal light spectrum for plants

The most important thing for any growing plant is a source of light. For their growth, plants need some wavelengths of the light spectrum more than others. Apart from the light spectrum, the intensity of the light source also affects plant growth. The wavelengths necessary for plants change according to the phase they are currently in. The blue and red parts of the light spectrum are the most important ones for photosynthesis. In figure 2.1, we can see that wavelengths around 480 nm and 660 nm are best absorbed and used by the plant for photosynthesis. In the past, lights with only blue and red types of light were used. Today grow lights using just these two types of light are fading away and are replaced by lights with a "full" spectrum. Parts of the light spectrum used by plants are as follows:

- UV Light 10 nm - 400 nm. While UV light most likely does not contribute to photosynthesis, it has been shown to improve plant health by eliminating bacteria and fungi and improving taste. [5]
- Visible light, PAR (Photosynthetically active radiation), 400 nm - 700 nm. Currently, this range is accepted to be the most important for photosynthesis. The photons in this range have enough energy to cause photosynthesis, but unlike photons at shorter wavelengths, they do not have enough energy to cause damage to cells.
- Blue Light 400 nm - 500 nm. It contributes to photosynthesis and is used

to control the flowering stages of the plants. It also regulates chlorophyll in the plants and affects their thickness by making them denser. [14].

- Green Light 500 nm - 565 nm. This type of light contributes to photosynthesis slightly. It used to be mostly ignored part of the light spectrum. However, it has been shown to add biomass production to some plants. It increases stem and leaf length. It also induces shade avoidance phenotypes. Since it penetrates the plant cells better than the red light, they contribute more to the photosynthesis in the leaves on the bottom of the plant. [25]
- Red Light 600 nm - 700 nm. Contributes to photosynthesis. This part of the spectrum affects the plant's morphology, causing the plant to grow longer leaves and stems, effectively stretching the plant. It is the most efficient light to produce using LEDs, producing the most light per watt.
- Far-Red Light 700 nm - 750 nm. Contributes to photosynthesis slightly. Far-Red light controls the flowering stages of plants such as strawberries. It also increases the size of leaves. With a larger area, the leaves can absorb more light leading to increased growth of the entire plant. [23]

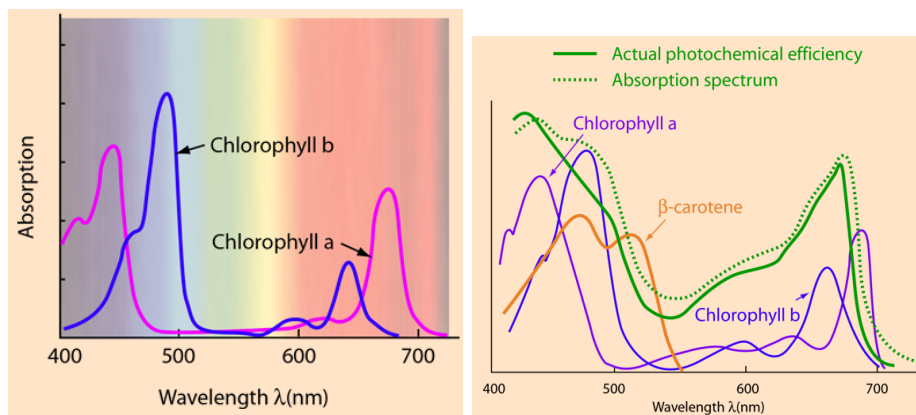


Figure 2.1: Comparison of wavelengths necessary for photosynthesis and for overall growth of plants [20]

There are multiple standards used to compare LED grow lights:

- Lumen [lm] is a unit of luminous flux, an amount of total visible light emitted by a source per time period. It is adjusted to reflect the human eye's sensitivity, which peaks around 550 nm [3].
- Lux [lx] is derived unit from illuminance. It is the amount of luminous flux per area. One lux is equal to one lumen per square metre.
- PAR (Photosynthetically Active Radiation) - is a spectral region of light radiation from 400 to 700 nm. Its comparison against lumen is in figure

2.2. It also correlates to the spectral range of light visible to humans, but while the human eye is more sensitive to the center of PAR (550 nm)[3], plants are more sensitive to blue (450 nm) and red (660 nm) light. The irradiance of PAR is measured in W/m^2 . On the one hand, if the intensity of PAR is too low, the plants will grow slowly and become unhealthy. On the other hand, if the light intensity is too high, the plant will go over the light compensation point and suffer burns. The light compensation point is when the rate of photosynthesis caused by the intensity of the light is precisely the same as the rate of cellular respiration. When this occurs, the plant consumes the same amount of O_2 and CO_2 as it releases. Since the amount of CO_2 in the air directly affects the compensation point, it is sometimes necessary to increase the amount of CO_2 to improve photosynthesis and reduce photorespiration (A process when plants consume some of the energy produced during photosynthesis).

- PPF (Photosynthetic Photon Flux ($\mu mol/s$)) - the total amount of PAR produced by a light source per second. This number can help choose the number/strengths of LEDs needed for growing areas. PPF is not entirely ideal because not all plants need the same light spectrum for their growth, and PAR takes each part of the light spectrum as equal in importance. Calculating the ideal light spectrum for each plant is not practical outside of huge greenhouses, and also, once this spectrum is fine-tuned, it can not be easily adjusted for other plants. Since I want this illuminator to be modular, it will be possible to adjust this value by full spectrum LEDs and specific LEDs to enhance the light spectrum according to the plant that is currently being illuminated.
- PPFD (Photosynthetic Photon Flux Density ($\mu mol/sm^2$)) - how much light falls onto a given surface. Once the grow area is known, we can easily calculate this value and adjust the LED output to match the desired PPFD.
- DLI (daily light integral ($mol/day \cdot m^2$)) - is the sum of photosynthetically active photons that are irradiated on a given area during some time period, usually 12 to 18 hours. Not all plants need the same wavelengths and therefore produce different results when only DLI is used, but some generalizations can be made [22]. Each plant has its own saturation point.

■ 2.1.1 Measuring illumination

Illuminance can be measured with a lux meter. Most lux meters utilize a silicon sensor with optical filters that imitate human eye sensitivity [26].

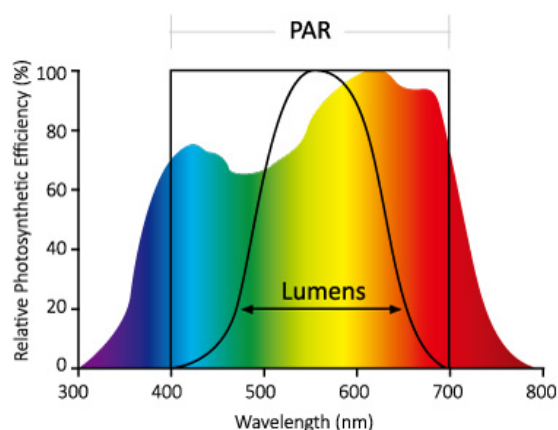


Figure 2.2: Comparison of PAR and Lumen [21]

The Lux readings are reasonably accurate for incandescent lights even with cheaper light sources because they have similar light spectrums. For LEDs, it is generally necessary to know the spectral output of the LED for the lux measurement to be accurate. The lux sensor should be at a right angle to the light source. In this project, a lux sensor is used to reduce the power consumption of the LEDs if the plant is placed, for example, on a window still or near a window and gets at least some direct sunlight during the day.

■ 2.1.2 Picking light spectrum for my application

For this project, it is best to use white light, 3000K, as the base light and then supplement it with deep blue (445 nm) and hyper red light (660 nm). The 3000K white light spectrum of chosen LEDs can be seen in figure 2.3. The spectrums of the blue and red LEDs are in figure 2.4,. By doing this, we can adjust the light spectrum later according to the plant being grown while not wasting any power. Based on the comparison of figures 2.1 and 2.3, I chose a 3000K white LED, whose spectrum is in figure 2.3. This decision was also based on price and the fact that this base spectrum will be amended. Using a full-spectrum white LED will also be more comfortable for humans taking care of the plant and living next to the illuminator. It will allow for easier plant inspection and enable us to see early signs that the plant has some problems, e.g., leaf discoloration, compared to using only blue and red LEDs. Considering humans, the decision was made not to use UV LEDs; the potential harm to the human eye is not worth the limited gain provided by using UV light to amend the light spectrum. It might be worth considering using UV LEDs in a large horticulture operation, where workers can use adequate eye protection. Lastly, additional green LEDs to improve the spectrum will not be used since the 3000K white LEDs already include

more than enough light in the 500 nm to 565 nm wavelengths.

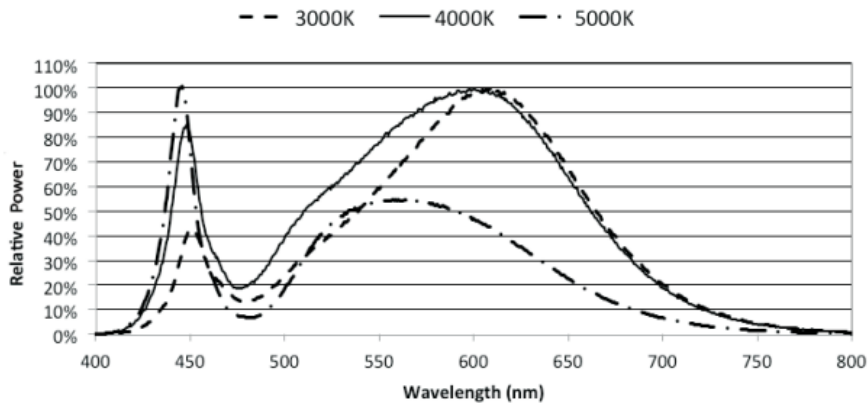


Figure 2.3: Wavelengths of different white color temperature LEDs from CREE [29]

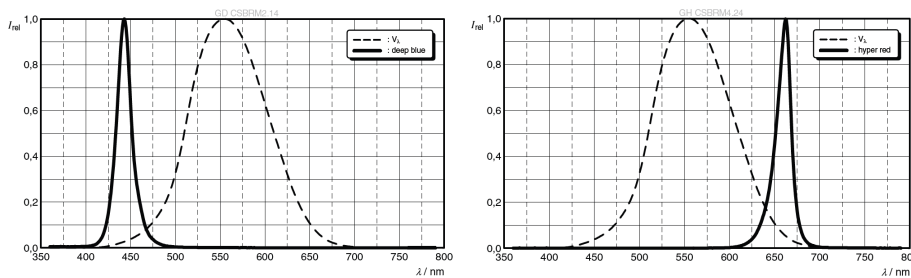


Figure 2.4: Relative spectral emission of blue and red LEDs from OSRAM [11][10]

2.1.3 How do LEDs work?

Light-emitting diodes are diodes made out of semiconducting materials. Semiconductors used in LEDs are crystals whose properties are similar to insulators, but their energy bandwidth is much narrower [3]. The band structure of semiconductors is divided into three parts based on their energy levels—figure 2.5.

- Valence band- highest band in the semiconductor structure filled with electrons
- Conducting band- lowest band in the semiconductor structure that has no electrons
- Band gap- dividing valence and conducting bands

The relative energy levels of these three bands can be seen in figure 2.5. Photon's energy, and thus its wavelength and color, is directly influenced by the band gap size. The band gap divides the valance and conduction bands and depends on the materials used to create these two bands. The valance band is a p-type semiconductor, and the conduction band is an n-type semiconductor. The energy of photons declines with their wavelength. This can be seen from:

$$E = hf = \frac{hc}{\lambda}$$

Where:

- E is energy (J)
- h is Plank's constant $\approx 6.626 \times 10^{-34} (m^2kg s^{-1})$
- f is frequency of the light (Hz)
- c is the speed of light (m/s)
- λ is the photons wavelengths (m)

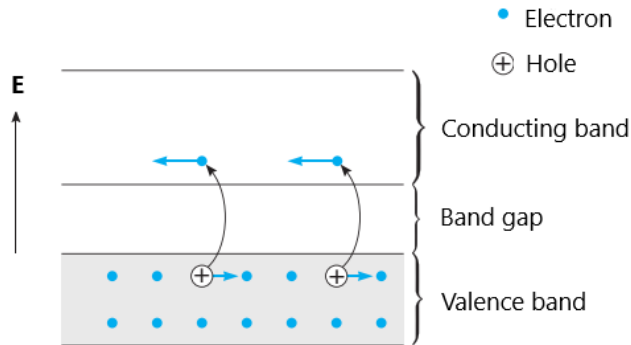


Figure 2.5: Illustration of electron passing from conduction band to valence band [3]

When the temperature is close to absolute zero, no electrons are present in the conduction band. In a typical ambient temperature of 25°C , electrons have a high chance of being in the conducting band due to the Fermi-Dirac distribution [3]. Due to this fact, most semiconductors can conduct electrical energy at typical ambient temperature and can conduct electricity better at higher temperatures, unlike conductors. Semiconductors used for LED manufacturing include: Ge, Si, GaAs, GaP, etc. [12] [3]

The p - n junction in direct band gap materials emit electromagnetic radiation when an electric current passes through it. The p stands for the

positive side containing holes, and n stands for the negative containing excess electrons. The electrons cross from the n -region (created by doping an intrinsic semiconductor with an electron donor element, such as phosphorus or arsenic, this is done because intrinsic semiconductors are essentially insulators) [9] to the p -region, where it combines with existing holes [13]. These holes are created by doping an intrinsic semiconductor with an electron acceptor element, such as boron, aluminum, or gallium [3]. The consequence is called electroluminescence; electroluminescence is a radiative combination of electrons and holes they fill in the valance band [6]. Since energy levels of the holes in the valance band are lower than the electrons' energy levels in the conduction band, figure 2.5, energy is being released. Energy is being released in 2 forms heat and light, while the latter is more favorable for LEDs. Light is being emitted from the band gap and can be seen by the naked eye if the wavelength of the released photon is in the visible spectrum (≈ 380 nm-750 nm).

The process of electroluminescence is more efficient than using, for example, incandescence, the process of emitting light and other forms of electromagnetic radiation from a hot body due to its large temperature. The typical efficiency of LEDs is between 25%-40%, [12] meaning they convert at least 60% of energy into heat, the efficiency of incandescent light bulbs is just around 5%, and halogen bulbs have an efficiency of around 10% [2]. People often think that LEDs are 80-90% efficient, but this number comes from manufacturers comparing LEDs to incandescent light bulbs.

■ 2.1.4 Creating white light using LEDs

There are two ways of manufacturing white LEDs. The first one is to combine three individual LEDs, red, green, and blue, and combine them into white light. This method has the disadvantage of poor color rendering since only narrow light wavelengths are emitted. The more favorable method, especially for high-power LEDs, is to use blue GaN chips and yellow phosphorus dispersed in an organic resin; this creates white light through wavelength conversion [30]. The disadvantage of this approach is that resin has a high thermal resistance and thus does not allow for the heat generated by LEDs to dissipate. Since heat causes LEDs to fail prematurely and shortens their lifespan, this property of resin is not ideal. The phosphorus concentration directly influences how much of the blue light emitted from the chip is converted into green and red light. Most white LEDs have their spectrum calibrated to suit the human eye, which more or less corresponds with the light spectrum needed for photosynthesis.

■ 2.1.5 Blue and Red LEDs

From figure 2.1 and previous research into what wavelengths of blue and red light are most beneficial to the plant's values, we can establish optimal wavelengths. Peak wavelengths should be in the 440-480 nm range for the blue light and 650 nm - 670 nm for the red light [15],[4]. Since the proposed light in this project should be able to adjust the intensity of these LEDs to suit multiple plant growing scenarios, the final value is not particularly important. It is impossible to choose the perfect wavelength for every plant at once. Therefore value will be based on other criteria, such as forward voltage, forward current, cost, and availability.

Chapter 3

Implementation

3.1 Choosing components

3.1.1 Selecting the right LEDs

An important parameter while choosing LEDs for grow lights is the area on which the plants will grow. There are some solutions on the market for big and medium growers, but the solutions for small growers are not smart, not adjustable, and quite expensive. Therefore I will focus on an area under 1 m^2 . This solution is planned to be easily scalable and modular. Any part of the system, except for the driver, can be exchanged to improve efficiency or to utilize the components the user already has on hand. This includes the power supply, LEDs, and the smart controller controlling the illuminator. For common plants like strawberries or tomatoes, DLI of at least $20 \frac{mol}{day \cdot m^2}$ is needed. If the light will be on for 18 hours a day, a PPF of approximately $308.64 \frac{\mu mol}{s \cdot m^2}$ is needed.

$$DLI\left(\frac{mol}{day \cdot m^2}\right) = PPF\left(\frac{\mu mol}{s \cdot m^2}\right) \cdot TIME(s)$$

$$PPF = 20 / (60 \cdot 60 \cdot 18) \cdot 10^6 = 308.64 \frac{\mu mol}{s \cdot m^2}$$

Plants without fruits need lower DLI; therefore, this value is sufficient for almost all plants one could want to grow. Some heat-loving plants can tolerate DLI of over $40 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$, but more shade-loving plants have the most significant amount of biomass at $\approx 19 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$. According to the American Institute for horticultural science, most greenhouse experiments are limited to $25 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$ [8].

I decided to aim for $20 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$ in standard operation mode, meaning using a typical forward current on the LEDs so as not to overheat them to prolong their lifespan, with the possibility to increase the current and, therefore, the LED brightness. This will increase the number of photons hitting the plant so that it can receive the recommended DLI. To grow, for example, lettuce DLI of $16\text{-}17 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$ is recommended [8]. Using a typical or slightly elevated forward current to drive the LEDs will allow lowering the current to match the necessary DLI plants, such as lettuce, need.

Most LEDs are made for lightning application with the human eye in mind. Therefore the data sheets include brightness values in lumen. To convert these values without knowing the exact spectrum of the LED, we can use the conversion factor. The final value will not be exact but is close enough for this application. The conversion factor K for 3000K white LED, with CRI (color rendering index) equal to 80, is approximately 14.5 [27].

$$PPFD[\frac{\text{mol}}{\text{day} \cdot \text{m}^2}] = K * LUX (klx)$$

$$LUX = \frac{308.64}{14.5} \cdot 10^3 = 21241 lx$$

The following equations have been used to convert the lux value to lumen so that a particular white LED can be chosen:

$$LUMEN = LUX \cdot A$$

$$LUMEN = 21241 \cdot 0.25 = 5310 lm$$

For an area of 50 cm by 50 cm, the necessary light output of LEDs is 5310 lumen. That is too much for one LED, so multiple will be used to achieve this number. As a white base, we will use the CXA1304-0000-000C00B230H. I decided on this LED mainly because of its price to light output ratio. The maximum efficacy is 117.5 lm/W. This LED has 423 lm while drawing 0.4 A. I will use 6 of these LEDs driven at 0.6 A to produce 600 lm each, figure

3.1. This LED is rated up to 1A. If the cooling allows, the LED can produce two times as much light; therefore, it can be used on a larger area or for more light hungry plants. Using 0.6 A for each LED will provide 3600 lm or approximately $216 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$.

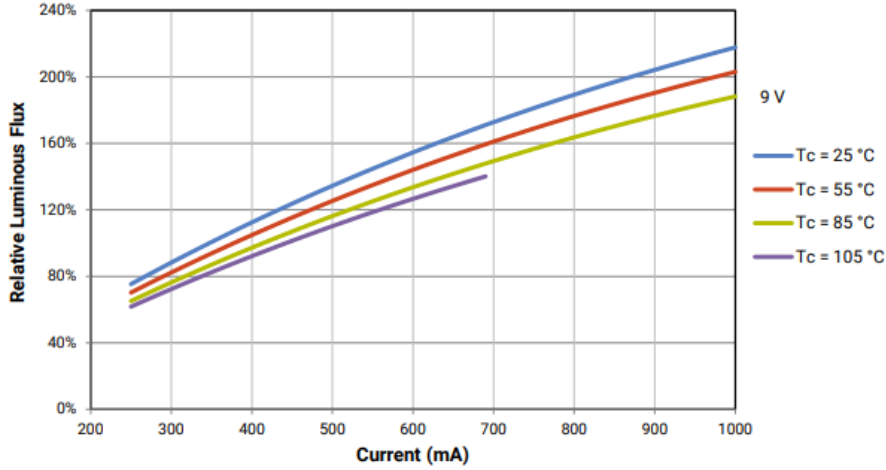


Figure 3.1: Relative luminous flux fro white LED[29]

To improve the white LED spectrum with blue and red LEDs, I chose OSRAM square horticulture lights made specifically for growing plants. Fortunately, the former calculations are not necessary since the data sheets include PPF values, and therefore the final values will be much more exact. They have wavelengths of 445 nm and 660 nm, respectively. Their spectrum, see figure 2.4, adds the most beneficial wavelengths for plants. From their data sheets we directly get a PPF value of $5.69 \mu\text{mol/s}$ at 0.7A for the red LED [11]. It can be driven up to 1.4A, almost to double its light output. The blue LED had PPF: $5.53 \mu\text{mol/s}$ and can be driven with a current up to 2 A [10]. Doing so would increase its output by 2.5-times. While increasing current to the LEDs, they lose efficiency, overheat and have a shorter lifespan; therefore, it is not desirable to do so. Two of each type of this single-color LED will be used. The total light output will be:

$$PPFD = \frac{PPF}{Area} = (5.53 \cdot 2 + 5.69 \cdot 2) / 0.25 = 89.76 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$$

Therefore the total output of all 10 LEDs driven at safe currents will be $216 + 89.76 = 305.76 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$, which is enough to support the healthy growth of most plants. It is just shy of the $308 \frac{\text{mol}}{\text{day} \cdot \text{m}^2}$ calculated earlier, but this can be easily solved by increasing the LED current slightly or running the light for a couple more minutes. This final PPF value is based on the assumption that 100% of the light will be reflected onto the surface, and there will be no

light losses to the outside environment. To accomplish this reflector will be used.

3.1.2 LED driver

To drive LEDs at a constant current, it is best to drive them in series. No 2 LEDs, even from the same manufacturer, have the same volt/ampere characteristics. If driven in parallel, the strings would be shining slightly differently. This could result in one LED failing during prolonged periods, sending its current to the remaining strings, possibly damaging them. The white LEDs have a 9 V forward voltage and will be connected in series. I have decided to use 2 LEDs in one string, which means that driver capable of driving the LEDs at max 1 A and over 22 V (2· 9 V, plus approximately 20% safety margin) will be needed. With these constraints, I opted for AL8843. AL8843 is a DC-DC step-down converter designed for driving single or multiple LEDs connected in series efficiently from a voltage source higher than the LED voltage. It is rated up to 40V and 3A, perfect for this application. Since the solution is planned to be modular and up-gradable, this driver will allow the user to use any LEDs that do not exceed the maximum voltage and current thresholds. I will use 5 of these drivers for the whole design with their complimentary parts mounted on a PCB

Setting the maximum current for LEDs

Using sense resistor it is possible to set the maximum current for driven LEDs. Using formula:

$$R_{set} = \frac{0.1}{I}$$

To maximize the lifespan of the LEDs, they should be driven at 0.4A, for which a resistor value of 0.250 Ohm should be chosen. For this application, I opted for a resistor value of 0,110 Ohm. This will allow the user to drive the LEDs at nearly the maximum rated current (0.91 A) and use them to their full potential. Driving them on higher current also reduces their effectiveness. This drop in effectiveness is more noticeable from around 0.7 A, see picture 4. This resistor only sets the maximal current, but the current can be adjusted via CTRL pin on the driver with either PWM or analog signal.

AL8843 schematics will be similar to the one from the data sheet. My schematics is in figure 3.2.

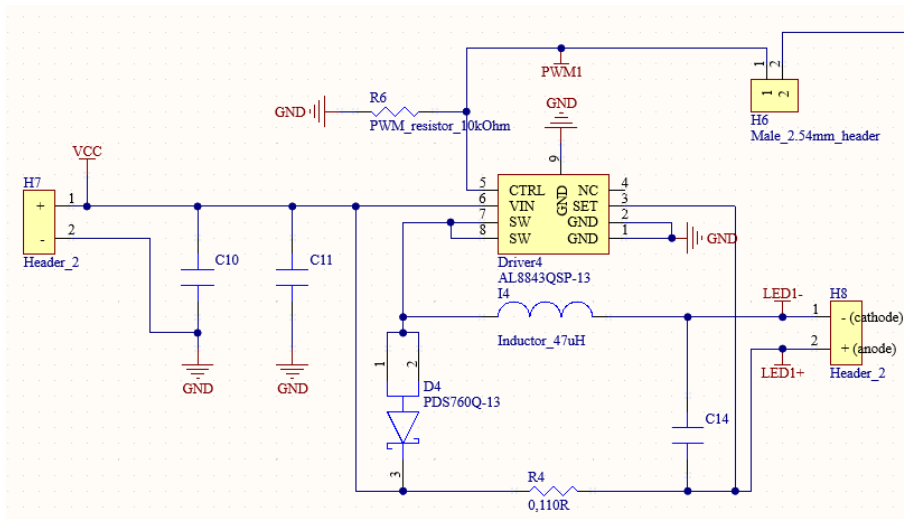


Figure 3.2: One LED driver, there are 5 of these on the PCB

Five of these drivers will be used to drive the LEDs in this configuration: 3x2 White LEDs, 1x2 Hyper Red LEDs, and 1x2 Deep Blue LEDs.

Driver efficiency

Figure 3.3 is from the data sheet, it shows that for while driving the white LEDs at ≈ 18 V from 22-24 V the efficiency of the AL8843 will be $\approx 97\%$. For driving the colored LEDs the efficiency will be lower $\approx 90\%$. This is due to the fact that the colored LED are driven at only ≈ 6 V from 22-24 V and the voltage drop is too big.

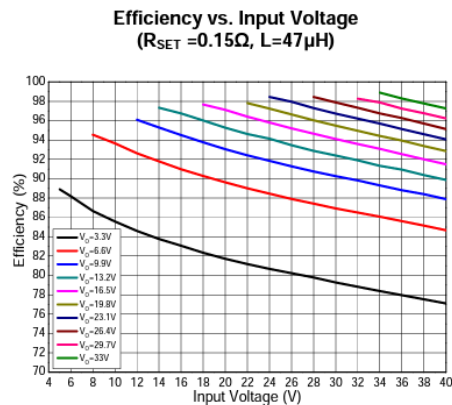


Figure 3.3: AL8843 efficiency[1]

■ Dimming

To save space on the board and simplify dimming, the drivers will be dimmed using 4 PWM signals. One signal will dim two drivers that drive four white LEDs in total. I do not deem it necessary for each white LED to be at a different brightness setting since this would mean that one part of the area that we are illuminating would receive more photons than different area, forcing the plant to grow in the direction of the higher brightness LED. There are 2 possibilities to dim the LEDs using AL8843. Analog and PWM signal, their comparison can be found in the following table 3.1.

PWM dimming	Analog Dimming
Brightness adjusted by modulating the peak current in the driver	Brightness adjusted by changing the DC current going to the LED
No color shift	Possible color shift as LED current changes
Possible current inrush	No inrush current to device
Frequency limitations	No frequency problems
Almost perfectly linear change in brightness	Worse brightness linearity
Lower optical to electrical efficiency	Higher optical to electrical efficiency (lumens per watt)

Table 3.1: Comparison of PWM and Analog dimming of LEDs

■ Complimentary components for LED driver

A low equivalent series resistance (ESR) capacitor should be used for input decoupling. According to the datasheet, I chose 10 μF , which will improve performance at lower input voltages. To maximize stability while the temperature changes, capacitors with X7R, X5R, or better dielectric are recommended. In parallel with the 10 μF capacitor there, I also placed a 100 nF capacitor to smooth out the input voltage even further, even though it is not mandatory. To minimize output current tolerance 47 μH inductors were chosen. This value is sufficient for 24 V supply voltage which I will use.

Low capacitance Schottky diode with forward current 7 A and forward surge current 274 A was chosen so that there is sufficient reserve and the diode will not be damaged. The reverse leakage current is 20 mA. Moreover, the typical forward voltage is 840 mV. This will ensure that the power dissipation on the diode will be as low as possible.

■ 3.1.3 Micro controller unit (MCU)

The MCU is required to be able to dim the LEDs with both PWM and analog signals. Another requirement for the MCU is that it will be possible to control it with either WIFI or Bluetooth. Moreover, it should have enough GPIO pins for users to add new functionalities. Examples of possible expansions are a LUX sensor, with which the LED brightness can be automatically adjusted according to the surrounding light intensity, moisture sensor, PWM controlled air fan... Using two separate MCUs for WIFI and Bluetooth, for example, EPS8266 and BTM-112, is not the optimal solution for multiple reasons. Firstly, using two modules will take up a lot of space, and communication between them adds complexity to the project. Secondly, the price difference between these two and ESP32. The former two modules cost from 350 Kc upwards, the ESP32, on the other hand, starts at 200 Kc, and it comes with the benefit of one package solution, with WIFI and Bluetooth low energy (BLE).

There are countless ESP32 modules to choose from. I chose ESP32 devkitc-V4 with WROOM-32D chip based on the price and availability. ESP32 has 32 GPIO pins. They are labeled GPIO 0 - 39. GPIO pins: 20, 24, 28-31, 37, 38 do not exist. From the 32 GPIO pins, another 5 are input only. Some pins also have the ability to use I2C, SPI, UART, and more. I will use 1 pin for VCC and GND each, 4 pins for PWM and another 4 for lux sensor. This means that I will still have enough pins for dimming the LEDs and adding extra functionality in the future. Another advantage of ESP32 is its lower power consumption than its predecessors, especially while using BLE.

■ Powering ESP32

There are three possibilities to power the MCU. Use a micro USB cable, power it from 3.3 V, or use 5 V. Using a micro USB cable would mean another cable and power-brick connecting it to mains electricity. It would also mean that the ESP32 would have to be disconnected from power and connected to the PC to upload new firmware. Powering from 3.3 V means bypassing the voltage regulator that is on board the dev-kit, and therefore the module has no protection against over-voltage. Using the 5 V option has the upside that the chip has another layer of protection on board and can be powered using 5-12 V. Using voltage higher than 6-7V would mean losing power in the form of heat on the regulator. I decided to use a switching regulator to reduce wiring complexity to deliver 5 V to the ESP32. This regulator will be mounted on the PCB. Using a step-down switching regulator is more power-efficient than a linear regulator. I will be stepping the voltage down from 24 V to 5 V, which would have resulted in much heat dissipated on

the PCB from the linear regulator. The regulator I chose is LM2576. It is capable of driving a 3.0 A load with fixed or adjustable output voltages. I opted for the fixed version since I only need to set it once to 5V. Schematics 3.4:

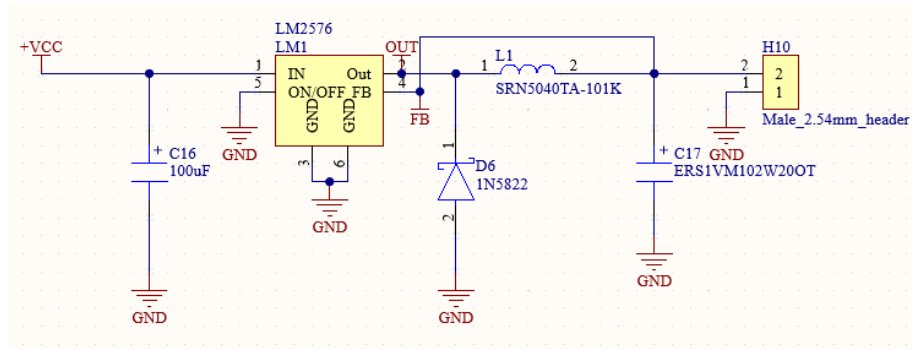


Figure 3.4: LM2576 schematics for fixed voltage of 5 V

The step-down switching regulator requires only four external components, which were chosen with the information from the datasheet. Input voltage can be from 7 V to 40 V. That is in line with the driver's maximum input voltage of 40 V. It also provides the freedom to use different input voltages to maximize the efficiency of the LEDs and the drivers. The efficiency of the regulator, in this case, will be around 88%. This value was obtained from the datasheet. From this, we can calculate the heat dissipation on the regulator:

$$P_{dissipated} = (V_{in} \cdot I_Q) + d \cdot I_{load} \cdot V_{sat}$$

$$P_{dissipated} = (24 \cdot 5 \cdot 10^{-3}) + \frac{5}{24} \cdot 0.5 \cdot 1.5 = 0.28 \text{ W}$$

Where:

- $P_{dissipated}$ is power dissipated on the regulator
- V_{in} is input voltage, 24 V in this case
- V_O is output voltage, 5 V in this case
- I_Q is quiescent current, typical value is 5 mA, value obtained from the data sheet
- d is duty cycle, and for buck converter is calculated: $d = \frac{V_O}{V_{in}}$
- I_{load} is the load current, maximum is 0.5 A, but typically values of no more than 0.1-0.2 A should be achieved
- V_{sat} is saturation voltage, conservative typical value of 1.5 V was chosen according to the data sheet [16]

A comparison to a linear regulator.

$$P_{dissipated} = (V_{in} - V_{out}) \cdot I_{out} = (24 - 5) \cdot 0.5 = 9.5 \text{ W}$$

Both values are worst case scenario for my final design, ESP32 will usually only draw 0.1-0.2 A and using lower V_{in} for linear regulator would improve its efficiency and lower heat losses.

3.1.4 LUX meter

The LEDs are capable of consuming up to 61.11 Watts. With the current prices of electricity and their possible rise in the future, it would be attractive to lower the power consumption while delivering the same amount of photons onto the plant. This illuminator is tailored to grow spaces equal to or smaller than 0.25 m^2 . Therefore if one places it near a window, the sun could provide some light needed for photosynthesis. To do so, the illuminator must know the current illuminance, which can be measured using a lux meter. I decided to use TSL2561 for its price, size, compatibility with the ESP32, and active low power (typically 0.75 mW [28]). The TSL2561 will communicate with the ESP32 using the I^2C digital interface.

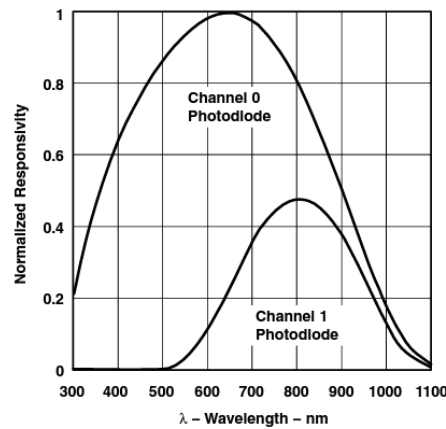


Figure 3.5: Spectral responsivity of the two photodiodes of TSL2561 [28]

TSL2561 uses two silicon diodes to calculate the brightness of ambient light as perceived by the human eye. Typical silicon detectors respond heavily to the infrared eye. This lux sensor was made with human vision in mind. Since humans cannot see the infrared eye, this device has two photodiodes. One (Channel 0) is sensitive to wavelengths from 300 nm to 1100 nm, while the other one (Channel 1) is sensitive primarily to infrared light (500 nm-1100 nm) figure 3.5 [28]. TSL2561 uses two integrating analogs to digital converters (ADC) to convert the photodiode currents to digital output. The digital output from channel 1 is then used to compensate for the effect of

wavelengths of light in the infrared spectrum registered by the photodiode on channel 0. The digital values from the ADCs are then used in a formula that approximates the illuminance in lux [28].

Using this approach is not ideal since plants have a slightly different response to light in the PAR region than the human eye does, but it gives us an idea of the current lighting conditions that the plants is currently in. Therefore the value coming from this sensor will be used to lower the current coming to the white LEDs to decrease their light output. The assumption made here is that the light coming from the sun has the same base spectrum as the white LEDs. This is not the case, and the light coming from the sun has a better spectrum for the plants than the white LEDs used, so the plants should benefit from the sun's light even more than from the extra 3000 K white LED light. It will also lower the power consumption, allowing the LEDs to run cooler and prolonging their life span. One disadvantage of this approach is that throughout the year, and even during the day, the sun lights spectrum changes slightly because the light's angle hits the earth. These small changes will be ignored while using this lux meter. However, if one wishes to conduct experiments with plants in a more controlled environment, he can disconnect the lux meter and place the plant away from any outside light sources.

■ 3.1.5 LED cooling

The most important thing for LED longevity is sufficient cooling. Cooling can be split into two categories. Active and passive. Using a fan as a form of active cooling has the advantage of dissipating more heat in a smaller footprint and having a cheaper upfront cost than a passive cooling solution. Heat sinks, on the other hand, do not consume electricity. Therefore over their lifetime, they will be cheaper compared to a fan. They also are entirely silent and require zero maintenance. For these reasons, I chose to use a passive heat sink.

■ Heat sink size

To find out the necessary thermal resistance of the heat sink, formula below can be applied:

$$R_{c-a} = \frac{T_c - T_a}{0.75 \cdot V_f \cdot I_f \cdot N}$$

$$R_{hs} = R_{c-a} - R_{tim}$$

In this case R_{hs} , the thermal resistance of the heat sink will have to be smaller or equal to:

$$R_{c-a} = \frac{398.15 \text{ K} - 298.15 \text{ K}}{0.75 \cdot (10 \text{ V} \cdot 0.9 \text{ A} \cdot 6 + 2.1 \text{ V} \cdot 0.9 \text{ A} \cdot 2 + 2.9 \text{ V} \cdot 0.9 \cdot 2)} = 2.12 \text{ K/W}$$

$$R_{tim} = \frac{\Delta x}{A \cdot k} = \frac{0.0005}{(0.015^2 \cdot 6 + 0.0007 \cdot 0.04 \cdot 2) \cdot 6} = 0.059 \text{ K/W}$$

$$R_{hs} = 2.12 \text{ K/W} - 0.059 \text{ K/W} = 2.06 \text{ K/W}$$

This formula is derived from thermal resistance circuit, figure 3.6. Thermal resistance circuit is an electrical analogy to conduction of heat transfer. Where:

- T_c is case temperature obtained from data sheet
- T_a is expected ambient temperature
- $P_{dissipated}$ is the power dissipated from the LEDs
- 0.75 is constant that says how much power was turned into heat, 0.75 is a conservative estimate recommended by the manufacturer
- V_f is forward voltage of the LED
- I_f is forward current of the LED
- N is the number of LEDs
- R_{c-a} case to ambient temperature thermal resistance
- R_{hs} thermal resistance of the heat sink
- R_{tim} is thermal interface material resistance, calculated from data sheet
- A area of thermal interface between LEDs and heat sink
- k is thermal conductivity of thermal interface material (manufacturers usually give this value instead of R_{tim} in this case it is $6 \frac{W}{m \cdot K}$)
- Δx is thickness of the material (measured in parallel to the heat flow)

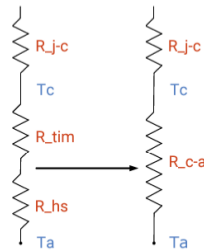


Figure 3.6: Thermal image of the heat sink, when LEDs were at 100% power.

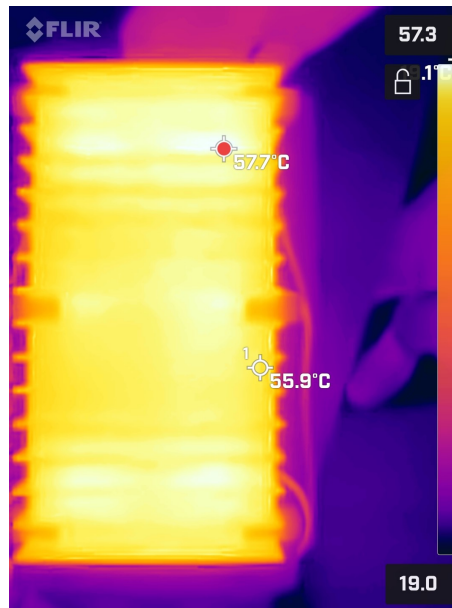


Figure 3.7: Thermal image of the heat sink, the 6 hot spots are placed where they are expected, above the white LEDs

The thickness of the chosen thermal interface material is 0.5 mm, but its thickness will be smaller since it will be pressed between the LEDs and the heat sink. Therefore we can assume its thermal resistance will be smaller. This means that the final thermal resistance of the heat sink should be equal to or smaller than 2.06 K/W. I chose an aluminum heat sink with a standard thermal resistance value equal to 1.8 K/W. This value is enough for the LEDs to run cool during regular operation (0.6 A for white LEDs). Therefore their lifespan should not decrease by excessive heat. The bottom plane dimensions of the heat sink are 152x80 mm, which is enough to mount the chosen LEDs to it. I tested the illuminator at 100% power for a couple of minutes, and the temperature of the heat sink from the top settled down at around 60°C; the result can be seen as a thermal image in figure 3.7.

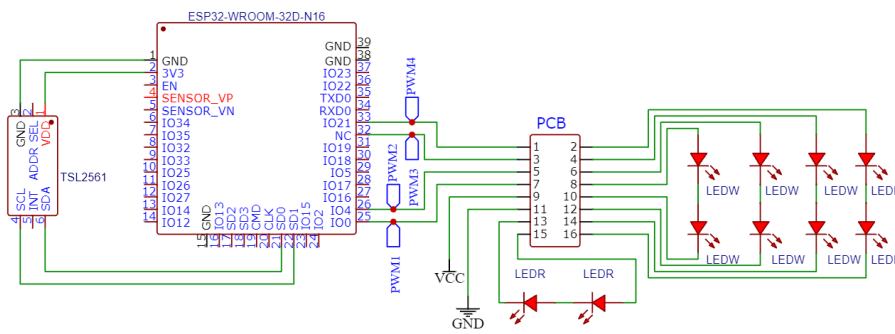


Figure 3.8: Hardware schematics

3.2 Hardware

3.2.1 Hardware diagram

The hardware for this illuminator consists of 4 blocks. The device's brains are provided by the ESP32, which will process the data from the lux sensor and set the desired PWM value. The second block is before mentioned lux sensor, powered by 3.3 V from the EPS32. The third block is the PB, which sets the suitable current to the LEDs and steps down the voltage to the ESP32. The last block is the LEDs. The Block diagram of this scheme can be seen in figure 3.8. These blocks will be described in more detail in the following sections.

3.2.2 PCB design

The driver AL8843 and its complimentary components must be mounted on a PCB. The size of the components does not allow us to use a perf board; therefore, a custom-made solution is necessary. I decided to use Altium Designer to design the PCB. Firstly, the schematics were made, in figure 3.9, using the previously selected components. Then after the PCB arrived, I soldered the components onto the PCB using a PCB reflow oven, hot air soldering, and classical soldering iron. The results can be seen in the same figure bottom picture. The schematics for Blue and Red LED drivers are the same as for white ones figure 3.2. The LM2576 schematics can be seen in figure 3.4.

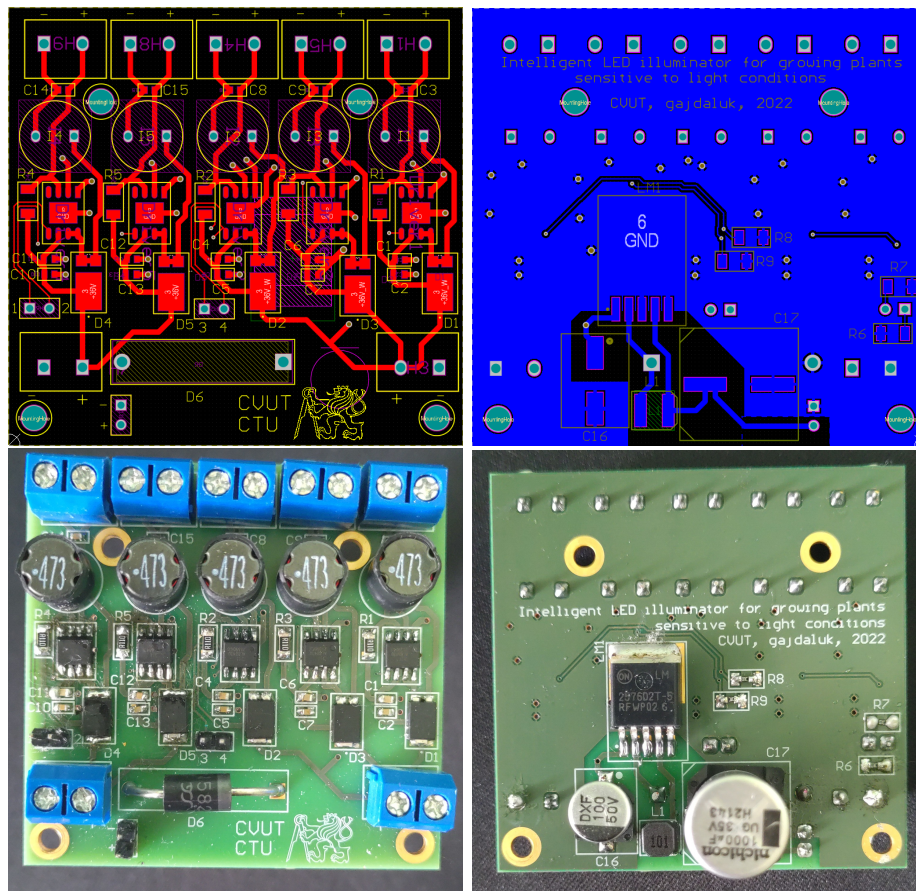


Figure 3.9: PCB top (right) and bottom (left) side

Colored LEDs PCB

The colored LEDs are SMD - surface mount devices, and their size is just 2.7 mm x 2.7 mm. Therefore it is necessary to mount them on a PCB. Their size does not allow using a prefabricated PCB, like perfboard. Since I will only use four of these LEDs, the price for fabricating the PCB is cost-prohibitive. Therefore I decided to use a custom-made PCB using a process called PCB etching. The PCB document was made in Altium, and then it was printed by a laser printer on special glossy paper. Then it was transferred with heat onto a plain copper board. The thickness of the copper is a standard 35 *um*. Then using $FeCl_3$, the unwanted copper was dissolved from the board. Afterward, the remaining ink was cleaned away with acetone. The result is a PCB that can be used for 2 LEDs; therefore, I made 2 of these. The result can be seen in figure 3.10. I decided to connect the thermal pads of the LEDs and leave as much copper on the PCB as possible to help with the heat dissipation.



Figure 3.10: LED PCB naked and with Blue and Red SMD LEDs soldered onto it

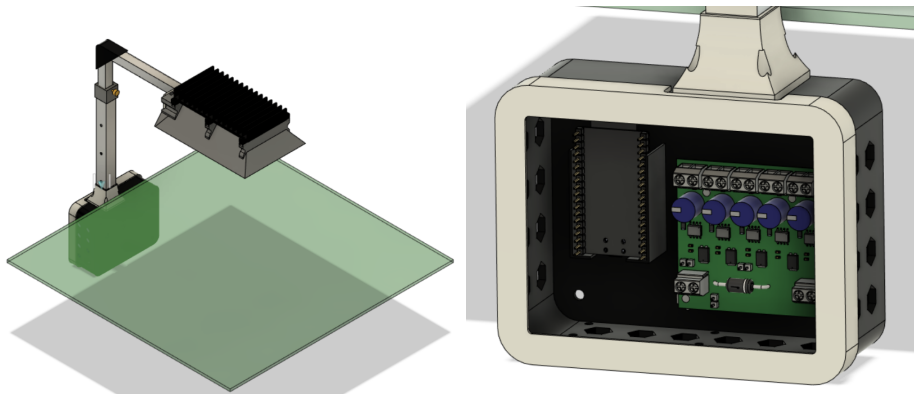


Figure 3.11: 3D model of the illuminator realisation

■ 3.2.3 Case

To house the electronics and place the LED illuminator above the plant, I decided to use a combination of 3D printing and aluminum extrusions. Modeling was done in Autodesk Fusion 360. Height will be adjustable, and I chose values from 20cm to 35cm. The height from which LEDs will be shining on the plant is also impacted by where the bottom part is placed. The reflector is mounted to the heat sink, and chromium finish coats the inside part to reflect as much light as possible. The benefit of this design is that changing the reflector angle allows users to illuminate an area of their choosing, whether it be a square, rectangle, or circle. The last one is the least optimal for this particular heat sink. This will allow growers to illuminate their preferred pot size and change it later by printing only the adjusted reflector.

The electronic housing, figure 3.11 on the right, has holes for standard M3 and M4 bolts so that they can be mounted horizontally and/or vertically. Since it will be around plants and, therefore, water, the holes for airflow are not placed on the top but only from the sides and bottom of the printed part.

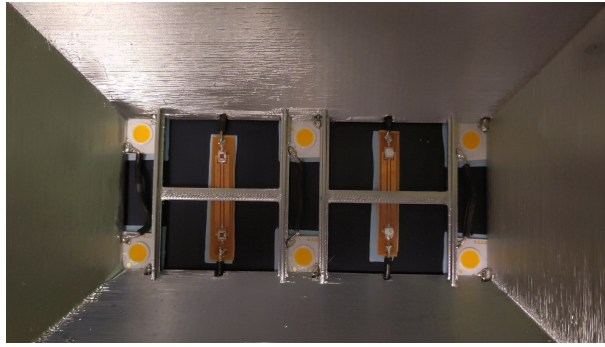


Figure 3.12: Placement of the LEDs on the heat sink, sides of the reflector are sprayed with chromium spray for better reflectivity

Plexiglass shielding the electronics will be in front of the electronic while allowing everyone to see inside. Using aluminum extrusion also allows for easy extension of the illuminator if needed. Figure 3.12 shows how the LEDs are placed on the heat sink. They are held in place by 3D printed case that serves as a reflector and is held onto the heatsink with screws.

■ 3.3 Software

■ 3.3.1 Scheme for software solution

The software controlling the ESP32 can be divided into two modes. The default mode is WIFI mode. If the device can connect to WIFI, it will connect to it and will host a web server. It will provide clients with an up-to-date state of the LED brightness and ON/OFF times for the illuminator. Clients will have the option to change any of these values to suit their plant's needs. The process is illustrated in figure 3.13 and will be discussed in more detail in the following section. If the device fails to connect to WIFI, it will go into Bluetooth Low Energy mode named BLE mode in figure 3.13. When in BLE mode, the ESP32 waits for a device to connect to it. Once the device is connected, the ESP32 starts broadcasting current PWM values. The connected device, the client, can send a message that the ESP32 will process and adjust the PWM value, or it will turn all of the LEDs off or on to 100%. A more thorough description will follow in the following chapter.

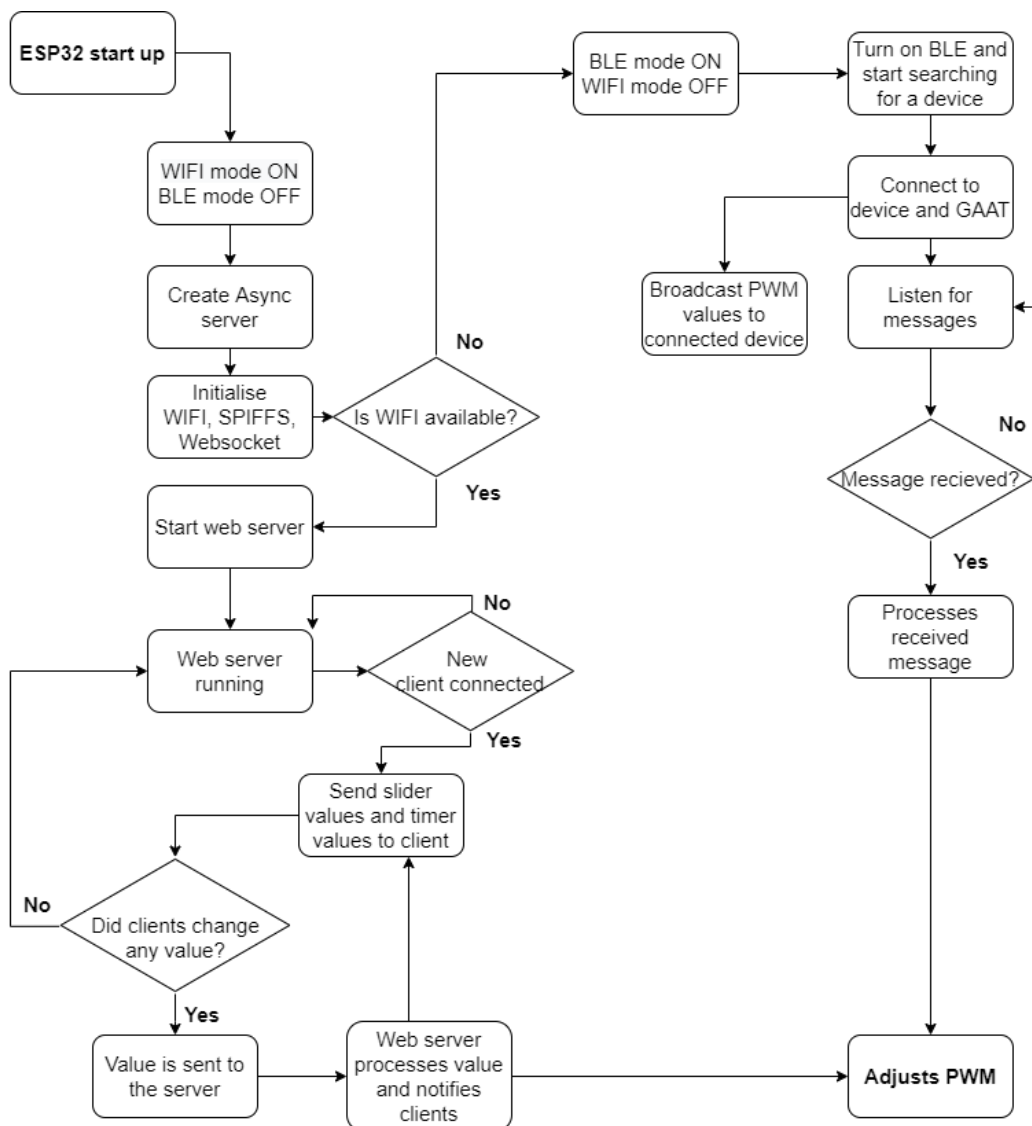


Figure 3.13: Diagram

3.3.2 Web server

To make this truly intelligent LED illuminator for growing plants, there needs to be a way to adjust brightness and light spectrum. To do this, the ESP32 will act as a web server and use Arduino programming language based on C++, JavaScript, HTML, and CSS. Controlling the ESP32 is done through the web browser. To connect to the device, it is necessary to know its IP address. To find out what it is, it is necessary to either have the Arduino serial port opened or it can be found in router settings. If the WIFI's SSID and password were added to the Arduino file, the ESP32 connects to the WIFI. When using the device on home network for example, it is convenient

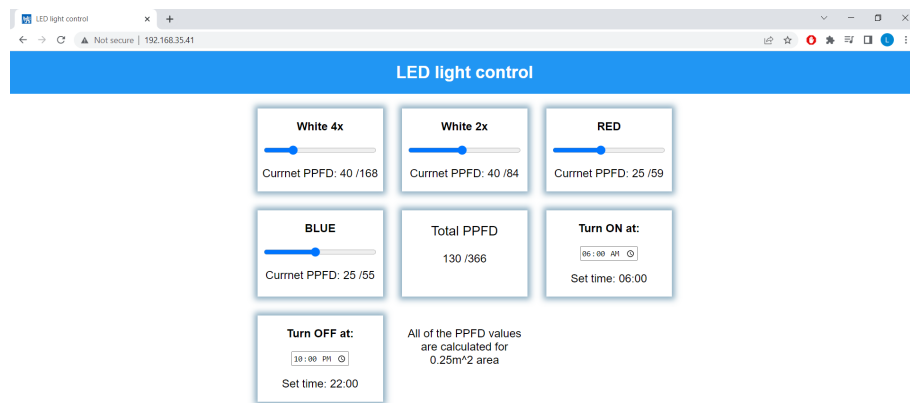


Figure 3.14: Web interface on desktop[24]

to set the ESP32 to have a static IP address, so that after the ESP32 connects to the WIFI, clients can access its website from multiple web browsers. Figure 3.14 shows GUI viewed on desktop web browser.

I decided to utilize WebSockets. WebSocket is a communication protocol located at layer 7 of the OSI model (Open Systems Interconnection model). It provides full-duplex communication between ESP32 and the user. This communication facilitates simultaneous and real-time communication between the web server, in this case, ESP32, and the web browser, in this case, the end-user. The communication is done through a TCP port, usually port number 443. However, since I will use the unsecured connection, the communication will be done over port 80. When the user moves one of the sliders, the ESP32 receives the current slider position and sends it to all connected clients. This means that all clients are always up to date. The same is done with on/off times set by the user. This means that if one wants to, he can control the illuminator from multiple devices and will see up-to-date information on all of the connected devices. To do so following libraries are necessary: `Arduino.h`, `WiFi.h`, `AsyncTCP.h`, `ESPAsyncWebServer.h`, `SPiffs.h`, `Arduino_JSON.h`. Initialization of the asynchronous web server can be seen on listing 3.1.

Listing 3.1: Basic code needed to run the initialise and start the asynchronous web server [7]

```
//Create AsyncWebServer object on port number 80
AsyncWebServer server(80);
//Create a WebSocket object
AsyncWebSocket ws("/ws");
//Initialise SPIFFS and WIFI
initSPIFFS(); initWiFi();
//Initialise AsyncWebSocket
ws.onEvent(onEvent); server.addHandler(&ws);
```

```

//Web server handles requests when it recieves request on the URL
server.on("/", HTTP_GET, [] (AsyncWebServerRequest *request){
    request->send(SPIFFS, "/index.html", "text/html");});
//Finally start the Async server
server.begin();

```

When WIFI is being initialized, SPIFFS memory needs to be initialized because it will contain HTML, javascript, CSS and .jpg files necessary to run the web server. Once the web server starts and is ready to receive data from clients and notify them of current values, it needs to be able to process incoming data. Partial code showing how the received data is processed is on listing 3.2. The code is simplified to fit on one page, but the fundamental principle is there. After the function checks, if a new message was received, it sorts which slider value was changed, calculates the total PPF, and saves this information into flash memory using the preferences library. Afterward, it broadcasts the new value to all connected clients. The function is then called in function onEvent(), which is initialized in listing 3.1.

Listing 3.2: Processing incoming data and save variable states to flash memory
[7] [24]

```

void ProcessWebSocketMessage(void *arg, uint8_t *data, size_t len) {
    //If the value was saved on flash memory if it was already set
    sliderValue1 = preferences.getString("sliderValue1", "");
    //recieved data packet is in a single frame
    AwsFrameInfo *info = (AwsFrameInfo*)arg;
    if (info->final && info->index == 0 && info->len == len) {
        if (info->opcode == WS_TEXT){
            data[len] = 0;
            message = (char*)data;
            //adjust the proper slider values
            if (message.indexOf("1s") >= 0) {
                //get the actual slider value
                sliderValue1 = message.substring(2);
                //calculate total PPF (sliderValue7), map dutycycle
                sliderValue7 = String(sliderValue1 + ... + sliderValue4);
                dutyCycle1 = map(sliderValue1.toInt()/1.68, 0, 100, 0, 255);
                //save all of these values to flash memory
                preferences.putString("sliderValue1", sliderValue1);
                preferences.putString("sliderValue7", sliderValue7);
                preferences.putInt("dutyCycle1", dutyCycle1);
                //broadcast current slider values to all clients
                ws.textAll(getSliderValues());
            }
        }
    }
}

```

}

ESP32 has an onboard real-time clock (RTC), but it is inaccurate and can shift the time throughout the day. It also needs to be initialized while connected to WIFI, and if it loses power, it forgets what time it is. This could be solved by writing the time to SPIFFS memory, but it has only a limited amount of writes, between 100 000 and 1 000 000. Therefore Every hour that the ESP32 is connected to the internet, it will synchronize time using the network time protocol. While the ESP32 has no access to WIFI, it will be possible to turn it on and off only via Bluetooth. The ESP32 will synchronize the network time protocol (NTP) every hour while connected to the internet via WIFI to get the real-time and stay up to date.

Since normally the the sliders are set from 0 to 100% and send values from 0 to 255, in this use case it is necessary to adjust the value the ESP32 receives by dividing it with the appropriate constant. Constant is calculated by dividing the maximal value of the slider (maximal PPFd that is driven by the PWM signal), by 100. I decided to use this approach instead of displaying percentages to show the LED brightness to make it easier for the user to set the wanted PPFd value. This approach on the other hand has the disadvantage that if the distance between the plant and the illuminator is changed, this value will no the accurate any more, but if the light is used for the previously calculated $0.25 m^2$ at height of 25 cm, no further calculations are needed to figure out the PPFd.

Users can set on/off time for the device to suit his/her needs. The ESP32 is equipped with flash memory, and I will utilize it to permanently save data on it in case of a power loss. To save data on the flash memory, one can use the preferences library. Every time user sets on/off time or moves one of the sliders, the new value of the timer/slider will be saved onto the flash memory. The flash memory is also used to store files necessary to host the web server, like HTML, JS, CSS files, and pictures used by the site. These files are modified from [24]. To use flash memory SPIFFS, a file system is used. It is intended for SPI NOR flash devices on embedded targets [23]. It creates a flat structure and utilizes only around 75% of the partition space, 4 MB in my case, but this will not be a problem. All of the files are just under 50 kB.

3.3.3 Bluetooth

Since ESP32 has only one antenna for Bluetooth (BT) and WIFI, it is not desirable to use them simultaneously. In this application, it would also not make much sense to simultaneously control the LED light via WIFI and BT. Therefore the BT will be set up as sort of a backup. If the ESP32 cannot

connect to WIFI, it will switch to BT mode.

ESP32 can use classic Bluetooth and Bluetooth 4.0, also known as Bluetooth Low Energy (BLE). Compared to older versions of Bluetooth, BLE has the advantage of drawing less power. It was developed for applications where it is unnecessary to send data continuously to save power compared to previously used solutions [18]. This allows BLE to be used in battery-powered devices that need to consume as little power as possible. In my case, the power consumption of the ESP32 will be pretty much negligible compared to the power consumption of the LEDs.

Nonetheless, I want to minimize the power consumption where possible. Therefore I will utilize BLE. To do so, I included necessary BLE libraries in the Arduino file. Afterward, if the BLE_mode is active, the following code will execute and initialize BLE 3.3. The Universally Unique Identifier (UUID) is used to uniquely identify information about what kind of information the type of device is transmitting. I will use three unique UUIDs, one to create the BLE service, meaning the ESP32 will act as a server while the connected device will be a client. The second one will create a channel for transmitting data (TX), and the third UUID will be used to create a channel to receive data (RX). In the initialization code, two classes are used to handle BLE communication. The first one, MyServerCallbacks(), handles device connection and disconnection. Once a device disconnects, the device has to start BLE advertising again, using the same function as the last line in listing 3.3.

Listing 3.3: BLE initialisation [18]

```
//initialize the device name and create the BLE server
BLEDevice::init("LED_illuminator");
BLEServer *pServer = BLEDevice::createServer();
//MyServerCallbacks() checks if a device is connected or not
pServer->setCallbacks(new MyServerCallbacks());
BLEService *pService = pServer->createService(SERVICE_UUID);

//create TX and RX services to send and receive data
pCharacteristic = pService->createCharacteristic(
    CHARACTERISTIC_UUID_TX,
    BLECharacteristic::PROPERTY_NOTIFY
);
pCharacteristic->addDescriptor(new BLE2902());

BLECharacteristic *pCharacteristic=pService->createCharacteristic(
    CHARACTERISTIC_UUID_RX,
    BLECharacteristic::PROPERTY_WRITE
);
//MyCallbacks() handles received messages
pCharacteristic->setCallbacks(new MyCallbacks());
//start BLE service and advertising
pService->start();
```

```
pServer->getAdvertising()->start ();
```

The second class, named MyCallBack(), handles received data from the client. The way the control of the PWM signal with BLE works here is that the client will send a message containing which PWM channel should be changed and a character indicating if the brightness should be lowered or increased. I decided to use 5% increments from 0% to 100%. This allows me to use two buttons for each PWM channel. I will also add a possibility to turn all LEDs on to 100% or off to 0%. When the off message is received, the ESP32 will turn off all LEDs. The following excerpt on listing 3.4.

Listing 3.4: Part of the MyCallbacks class which handles incoming messages to the ESP32 this section show how the PWM on channel 1 can be increased by 5%

```
class MyCallbacks: public BLECharacteristicCallbacks {
    void onWrite(BLECharacteristic *pCharacteristic) {
        std::string BLE_message = pCharacteristic->getValue();
        if(BLE_message.length() > 0){
            Serial.println(F("BLE_msg_rcieved"));
            //first number is the PWM channel, second number is eather 1 or 0
            if (BLE_message.find("11") != -1){
                Serial.println(F("Increase_channel_1_PWM"));
                if(dutyCycle1 < 252){dutyCycle1 = dutyCycle1 + 12;
                    preferences.putInt("dutyCycle1", dutyCycle1);
                } //add 5%
            }
        }
    }
}
```

After power off, the illuminator will return to the previous state as soon as the user turns it back on. This also means if in the WIFI mode the values are changed, after the power off, they will be the last saved in that mode. Code 3.4, is less complex than the WIFI mode because the only thing that needs to be written into the flash memory is the duty cycle and no conversion calculations are necessary. The 5% increments are 12 and not 12.75 because the duty cycle value is stored in integer, not float. This could be changed, but float memory takes up twice as much space, and in this case, it is not worth spending it. Also, when setting the duty cycle to the LED in Arduino, the function `ledcWrite(channel, dutyCycle);` is used, and it accepts integers for the duty cycle. Therefore it will just round it up anyway. The duty cycle is a global variable, and even though the current global variables take up only 15% of the dynamic memory, which has 320 kB. When the program is running, it needs the rest of the RAM for local variables. The BLE libraries

are so extensive that the default partition scheme of ESP32 did not have enough space for all of the program files. It needs to be changed, and since I will not utilize Over The Air (OTA) updates, the NO OTA partition scheme will be used, which gives 2 MB of space to user files, of which I will utilize around 70% and 2 MB to spiffs memory of which only around 40% will be utilized.

One way to connect to the ESP32 BLE server is to use a Serial Bluetooth application on an Android device, but there are multiple choices for any smartphone device. After scanning for BLE devices, the option to connect to LED_illuminator should show up. After connecting to it, the app will display the message Connected. It will display received messages from the ESP32 BLE server displaying the current PWM values every couple of seconds (around 5). This interval is this long to reduce power usage. Figure 3.15 show an example of communication between the client, smartphone in this case, and the server, ESP32. The advantage of this particular app is that the user can set buttons to send predefined values so that it is not necessary to type them out. The messages sent to the ESP32 consist of 2 characters, the first one (1-4) indicates which PWM channel should be modified, and the second (0 or 1) indicates whether the PWM value should be increased or decreased by 12.

■ Adjusting the LED brightness based on the reading from LUX sensor

After the duty cycle is set, whether with BLE or through web server, it needs to be adjusted based on the current lighting conditions. The reading from the lux sensor provides the current illuminance levels in lux and therefore it is necessary to use this value to adjust the PWM signal going to the white LEDs to decrease their brightness. To do so the following formula was used:

$$dutyCycle_new = \frac{\frac{Slider_PPFD \cdot dutyCycle}{255} - PPFD \cdot \frac{n}{6}}{\frac{slider_PPFD \cdot dutyCycle}{255}}$$

$$PPFD = \frac{LUX \cdot 15}{1000}$$

Where:

- Slider_PPFD is the maximal PPFD value of the slider
- dutyCycle is the old duty cycle that would have been used if not for the lux sensor

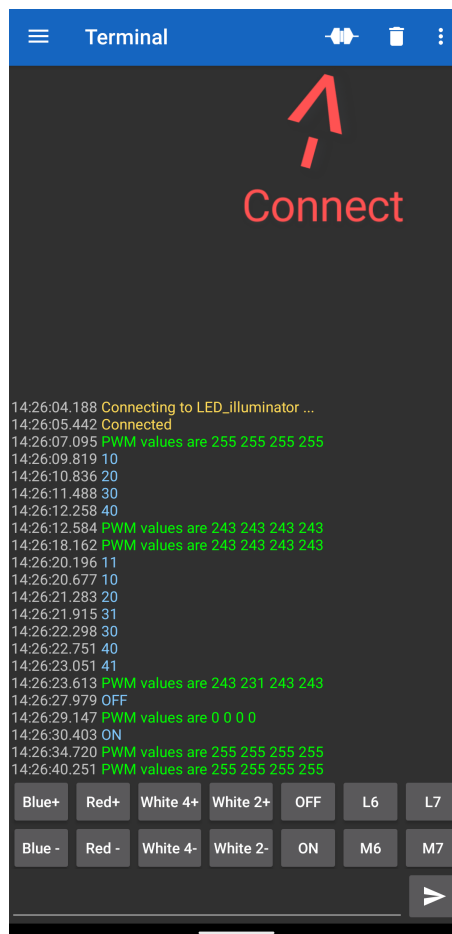


Figure 3.15: BLE terminal example, showing basis functions

- PPF is calculated from the LUX reading based on calculations made earlier
- LUX is reading from the lux sensor
- n is the number of LEDs controlled by the PWM (2 or 4)

3.4 Light brightness map

Brightness maps can be used to see how well is the light distributed on the surface. I measured the illuminance values with the before mentioned lux sensor TSL2561. The illuminator was placed 25 cm above the lux sensor. The values themselves are not as important as the how evenly is the light distributed. The results for using duty cycle of 51%, which results in current

PWM was set to 51%

2300	3300	4000	3450	2000	1000
3500	6300	9200	8400	4800	1400
4750	11500	15000	14900	7750	1900
4900	12200	17000	16000	9400	2500
4000	8400	10500	9900	6650	1900
2300	4100	5700	5200	3450	1000

Figure 3.16: Lux maps, showing how uniformly the light is distributed, grid scale is 10 cm, distance between the illuminator and the lux sensor at the center of this mas was 25 cm

on the LEDs of 0.56 A. At maximum the current can be set to 0.88 A, the calculated value is 0.91 A, but imperfections in the electronics used cause this value to be slightly lower. Also imperfections in the sense sensor can cause deviations in the current. However the difference is only approximately 3%. The resulting lux map can be seen in figure 3.16. The values are shifted towards left. This is because as seen in figure 3.12 the red LEDs are more on one side and blue LEDs are more on the other. As discussed before, this lux sensor is more sensitive to the red side of the spectrum, and even though it is trying to compensate for it, the compensation is not perfect. In this application the TSL2561 is not meant to be under the illuminator so the light from it will not affect it. It will only receive sun light. The figure 3.16 is mainly included to show that the biggest light intensity will be in the centre of illuminated area. This is beneficial for larger plants, like before mentioned tomato, but is not ideal for growing more smaller plants, like microgreens.

3.5 Price comparisons

To find out if this illuminator is a financially viable solution and can be compared to other commercial solutions its price needs to be calculated. Total price for LEDs is in table 3.2

The final price for working system is around 2501.04 Czk if shipping and manual labor is not counted. To compare this to other solutions, I looked at most popular devices (Alza, Amazon) for comparable growing area and compared them in following categories: Grow area, maximal power, smart

	Part	Quantity	Price in Kč
LEDs	CXA1304	9	29,84
	Deep Blue	2	41,57
	Hyper Red	2	59,16
Driver and its components	AL8843	5	29,116
	Schottky diode	5	19,41
	Capacitor 10 uF	5	1.65
	Capacitor 100 nF	10	0.83
	Coil 47 uH	5	24
	Sense resistor	5	0.66
	Resistor 10 kΩ	5	10.75
LM2596 and its components	LM2596	1	55
	Shotky diode	1	3.9
	Capacitor 100 uF	1	15.2
	Capacitor 1000 uF	1	37,5
	Coil 100 uH	1	14.03
ESP32	ESP32 module	1	199
PCB	PCB	1	350
Heat sink	Heatsink	1	250
Power supply unit	PSU	1	700
Total price			2501.04

Table 3.2: Price of the LEDs

capabilities, price. 3.3

Name	Grow area	Power	Intelligence	Price
Learn Mini4	50x50	63 W	Yes	2501.04
Microgreens by leaf [19]	30x30	10 W	Limited	3990
TS 600 [17]	60x60	100 W	No	2500

Table 3.3: Price of the LEDs, grow area is in cm

From the table 3.3 we can see that Learn Mini4 has some smart capabilities, but they are not meant to be played with and come pre-configured for the manufacturer's seed and plants. It is also quite more expensive than my solution. This sort of illuminator is made for herbs and therefore their power is quite low.

The MARSHYDRO TS 600 LED lamp on the other hand is quite more power-full but it has no smart capabilities, and can not be dimmed at all. It is priced same as my solution.

To sum up the price comparison, my illuminator is in between the most solutions on the market today. It is made for more advanced growers that want to adjust the spectrum and the brightness for their plants. It is also competitively priced considering that it is one of and economics of scale would make it significantly cheaper.

Figure 3.17 shows my finished illuminator, and the two LED lamps that is



Figure 3.17: Final illuminator and comparable commercial solutions

was compared to in table 3.3.



Chapter 4

Conclusions

In conclusion, the task of designing and realizing an LED illuminator for growing plants sensitive to light conditions was successfully completed. The illuminator is able to change its light spectrum to fit most plants needs. It is also power-full enough to provide enough energy for photosynthesis even for plants that require more light to grow fruitfully. The solution is scalable, and if one wishes he can add more drivers to drive more LEDs or use more LEDs per string, up to 2 times more per string in the case of the used white LEDs, and up to 6 times more in the case of the blue/red LEDs. Even without making modifications to the PCB it should be able to use LEDs that need twice as much power as the LEDs currently used. One thing that could be changed is to use the same kind of LEDs, i used a combination of SMD and LEDs that require a wire to be soldered to them. For the easier manufacturing and designing it would be beneficial to use a pre-made PCB and using only SMD LEDs. Another modification could be to use white LEDs made for horticulture to improve the light spectrum and efficiency, but buying them at low quantities is not favourable. Lastly I would target a smaller growing area with the possibility to drive 2 or more LED lamps from the same PCB, to illuminate multiple pots on a windows still.

The intelgent part of this illuminator was done using ESP32 dev board. This enabled me to utilise WIFI as well as BLE to control the illuminator. With websockets the ESP32 serves as a asynchronous web server and can send ass well as receive data. This allowed me to control the LED brightness to be exactly as the plants need it. The web server clearly states the total PPF_D that the plants are receiving in the web browser, so that no complicated calculations are necessary if the plants spectral needs are known. Utilising BLE for lower power consumption, it is also possible to control the LED brightness. This manual mode of operation can be use full for testing purposes of if WIFI is unavailable.



Appendix A

Contents of included zip file

1. Altium designer files for manufacturing the PCB
2. Arduino files needed for ESP32 (file types included: ino, html, css, js)
3. Files for 3D printing the case

Appendix B

Bibliography

- [1] *40V 3A STEP-DOWN LED DRIVER*. AL8843. DS40367 Rev. 1 - 2. DIODES incorporated. 2017. URL: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1119993/DIODES/AL8843.html>.
- [2] *All about energy efficiency and efficacy of modern LED lighting*. URL: <https://lamphq.com/led-energy-efficiency/>.
- [3] Michal Bednařík. *Studijní texty k predmetu Fyzika 2*. 2020. URL: <https://fel.cvut.cz/cz/aktuality/2021/bednarik-F2>.
- [4] N.G. Bukhov, I.S. Drozdova, and V.V. Bondar. “Light response curves of photosynthesis in leaves of sun-type and shade-type plants grown in blue or red light”. In: *Journal of Photochemistry and Photobiology B: Biology* 30.1 (1995), pp. 39–41. ISSN: 1011-1344. DOI: [https://doi.org/10.1016/1011-1344\(95\)07124-K](https://doi.org/10.1016/1011-1344(95)07124-K). URL: <https://www.sciencedirect.com/science/article/pii/101113449507124K>.
- [5] Anastasios I. Darras et al. “Low doses of UV-C irradiation affects growth, fruit yield and photosynthetic activity of tomato plants”. In: *Scientia Horticulturae* 267 (2020), p. 109357. ISSN: 0304-4238. DOI: <https://doi.org/10.1016/j.scienta.2020.109357>. URL: <https://www.sciencedirect.com/science/article/pii/S0304423820301850>.
- [6] JAROSLAV DOLEČEK. *Moderní učebnice elektroniky - 3.díl: Optoelektronika a optoelektronické prvky*. 2005. ISBN: 80-7300184-5.
- [7] *ESPAsyncWebServer*. Mar. 2022. URL: <https://github.com/me-no-dev/ESPAsyncWebServer/#async-event-source-plugin>.
- [8] James E. Faust and Joanne Logan. “Daily Light Integral: A Research Review and High-resolution Maps of the United States”. In: *HortScience horts* 53.9 (2018), pp. 1250–1257. DOI: [10.21273/HORTSCI13144-9](https://doi.org/10.21273/HORTSCI13144-9)

18. URL: <https://journals.ashs.org/hortsci/view/journals/hortsci/53/9/article-p1250.xml>.
- [9] Bruno Gayral. “LEDs for lighting: Basic physics and prospects for energy savings”. In: *Comptes Rendus Physique* 18.7 (2017). Demain l’énergie, pp. 453–461. ISSN: 1631-0705. DOI: <https://doi.org/10.1016/j.crhy.2017.09.001>. URL: <https://www.sciencedirect.com/science/article/pii/S163107051730052X>.
- [10] *GD CSBRM2.14*. OSLO Square Deep Blue. OSRAM. 2021. URL: <https://eu.mouser.com/ProductDetail/ams-OSRAM/GD-CSBRM214-ARAK-24-1-700-R33?qs=sGAEpiMZZMvONwlthflBi8MStFIEebi9y06HaqoE0b4%5C%3D>.
- [11] *GH CSBRM4.24*. OSLO Square Hyper red. OSRAM. 2021. URL: https://www.osram.com/ecat/OSLO%5C%C2%5CAE%5C%20Square%5C%20GH%5C%20CSBRM4.24/com/en/class_pim_web_catalog_103489/prd_pim_device_18145105/.
- [12] Ahlem Ben Halima et al. “Energy efficiency of a LED lighting system using a Peltier module thermal converter”. In: *Case Studies in Thermal Engineering* 34 (2022), p. 101989. ISSN: 2214-157X. DOI: <https://doi.org/10.1016/j.csite.2022.101989>. URL: <https://www.sciencedirect.com/science/article/pii/S2214157X22002350>.
- [13] Jiří Hospodka. *Elektronické obvody interaktivní studijní materiál*. 2020. URL: <https://moodle.fel.cvut.cz/mod/page/view.php?id=204733>.
- [14] Rebecca E. Johnson, Yun Kong, and Youbin Zheng. “Elongation growth mediated by blue light varies with light intensities and plant species: A comparison with red light in arugula and mustard seedlings”. In: *Environmental and Experimental Botany* 169 (2020), p. 103898. ISSN: 0098-8472. DOI: <https://doi.org/10.1016/j.envexpbot.2019.103898>. URL: <https://www.sciencedirect.com/science/article/pii/S0098847219311499>.
- [15] Yan Li et al. “Mixed red and blue light promotes tomato seedlings growth by influencing leaf anatomy, photosynthesis, CO₂ assimilation and endogenous hormones”. In: *Scientia Horticulturae* 290 (2021), p. 110500. ISSN: 0304-4238. DOI: <https://doi.org/10.1016/j.scienta.2021.110500>. URL: <https://www.sciencedirect.com/science/article/pii/S0304423821006075>.
- [16] *LM2576/LM2576HVSeriesSIMPLESWITCHER@3AStep-DownVoltageRegulator*. LM2576. SNVS107C– JUNE1999– REVISEDAPRIL2013. Texas instruments. 2013. URL: <https://www.alldatasheet.com/datasheet-pdf/pdf/543767/TI1/LM2576.html>.
- [17] *MARSHYDRO TS 600 LED Grow Light*. May 2022. URL: <https://www.amazon.co.uk/dp/B07X34HZLT?linkCode=gs4&tag=rs107-21&th=1>.

- [18] *METEOSTANICE S VYUŽITÍM BLUETOOTH LOW ENERGY*. CVUT, 2017. URL: <https://dspace.cvut.cz/handle/10467/69611>.
- [19] *microgreens by leaf*. May 2022. URL: <https://www.alza.cz/microgreens-by-leaf-learn-mini-d6249392.htm>.
- [20] R Nave. *Light Absorption for Photosynthesis*. May 2022. URL: <http://hyperphysics.phy-astr.gsu.edu/hbase/Biology/ligabs.html>.
- [21] Robert Pavlis. *LED Grow Lights – Getting the Right Color Spectrum*. May 2022. URL: <https://www.gardenmyths.com/led-grow-lights-color-spectrum/>.
- [22] Hendrik Poorter et al. “A meta-analysis of plant responses to light intensity for 70 traits ranging from molecules to whole plant performance”. In: *New Phytologist* 223.3 (). DOI: <https://doi.org/10.1111/nph.15754>. URL: <https://nph.onlinelibrary.wiley.com/doi/abs/10.1111/nph.15754>.
- [23] Meyer Prisca et al. “Blue and far-red light control flowering time of woodland strawberry (*Fragaria vesca*) distinctively via CONSTANS (CO) and FLOWERING LOCUS T1 (FT1) in the background of sunlight mimicking radiation”. In: *Environmental and Experimental Botany* 198 (2022), p. 104866. ISSN: 0098-8472. DOI: <https://doi.org/10.1016/j.envexpbot.2022.104866>. URL: <https://www.sciencedirect.com/science/article/pii/S0098847222000880>.
- [24] Sara Santor. *ESP32 Web Server (WebSocket) with Multiple Sliders: Control LEDs Brightness (PWM)*. May 2022. URL: <https://randomnerdtutorials.com/esp32-web-server-websocket-sliders/>.
- [25] Liesbeth Schenkels et al. “Green light induces shade avoidance to alter plant morphology and increases biomass production in *Ocimum basilicum* L.” In: *Scientia Horticulturae* 261 (2020), p. 109002. ISSN: 0304-4238. DOI: <https://doi.org/10.1016/j.scienta.2019.109002>. URL: <https://www.sciencedirect.com/science/article/pii/S030442381930888X>.
- [26] Ethan Schwartz. *How to calculate light intensity*. Mar. 2019. URL: <https://sciencing.com/calculate-light-intensity-7240676.html>.
- [27] Anton Sharakshane. “An easy estimate of the PFDD for a plant illuminated with white LEDs”. In: *bioRxiv* (2018). DOI: <https://doi.org/10.1101/289280>. URL: <https://www.biorxiv.org/content/biorxiv/early/2018/03/30/289280.full.pdf>.
- [28] *TSL2561 Light-to-Digital Converter*. Apr. 2018. URL: <https://ams.com/en/TSL2561>.
- [29] *XLamp® CXA1304 LED*. CXA1304. CREE. 2021. URL: <https://cree-led.com/media/documents/ds-CXA1304.pdf>.

- [30] Tao Zhou et al. “Highly thermal stable and color tunable composite fluorescent ceramics for high-power white LEDs”. In: *Ceramics International* (2022). ISSN: 0272-8842. DOI: <https://doi.org/10.1016/j.ceramint.2022.03.145>. URL: <https://www.sciencedirect.com/science/article/pii/S0272884222009361>.