Solar Powered Sanitation Device

By:
John Cerce
Ryan Kent

Professor James O'Rourke, Advisor
March 1, 2019

Abstract

This project identifies how UV-C LED technology can be used for medical instrument surface sanitation within a portable application. The necessary light intensity, power requirements and general portability of such a system were evaluated to design a testable device. With the test results gathered, a prototype device was created to show the feasibility of portable solar powered sanitation with UV-C LED technology.

1.0 Introduction

In the healthcare field, there is a necessity to sanitize medical instruments before and after use. In areas with modern conveniences like electricity and water utilities, advanced sanitization methods such as the use of autoclaves are commonplace in healthcare institutions. However, not everywhere in the world has the luxury of having access to electricity and water. In many developing countries, healthcare professionals find themselves not being able to properly sanitize equipment or obtain sanitized equipment. This issue causes many patients run the risk of further sickness or injury even though they are getting treatment.

In order to solve this issue, our team has designed a solar powered sanitization device. This device was designed assuming there was no sufficient source of water or an available powergrid. With these restriction the device implements UV-C technology in order to sanitize various medical instruments and waste. The UV-C LEDs are powered by on-board battery packs that are recharged by a solar panel with a tracking system. Solar powered sanitation with the use of UV-C LEDs offers a portable, lightweight, and low powered option for remote areas and underdeveloped regions to obtain needed medical equipment.

2.0 Background

2.1 Current Sanitation Devices

Many different types of sanitation devices are used, all used to kill dangerous pathogens such as bacteria, viruses, and spores. Autoclaves both using dry heat and steam are used to sanitize medical equipment and waste. All autoclave systems work by heating the equipment to a certain point for an extended period of time. But, the heat and amount of time needed to sanitize equipment can be reduced by using UV-C LEDs. These LEDs operate at a certain wavelength allowing them to kill and disable harmful pathogens. Although this system will operate by line of sight contact, with a high enough UV output, the time needed to sanitize can be significantly reduced. UV-C LEDs provide a fast, low powered sanitizing option for portable applications.

2.1.1 Autoclaves

For a long time most medical sanitation has been completed by autoclaves. There are different types of autoclaves, dry heat as well as steam. Both have different variables including temp, pressure, time which are manipulated depending on the type of sanitation, what's being sanitizes and how much is being sanitized (CDC).

Steam sterilization is a nontoxic and inexpensive option but can also lead to corrosion and combustion depending on the sanitized materials (CDC). They work by exposing tools and other materials to steam at very high temperatures, about 250 to 270 degrees fahrenheit, for an extended period of time (CDC). The time needed to sanitize depends on the pressure held within the system (CDC). A vacuum and the resulting pressure, allow the autoclave to hold hotter temperatures for a longer amount of time because hot steam is kept within the system (CDC). Disinfecting with steam is done in one of two ways, gravity displacement or high-speed pre-vacuum (CDC).

A gravity displacement autoclave is a system used often in laboratories, pharmaceuticals, medical waste and nonporous articles (CDC). This works by inserting hot steam into the top of the chamber, since steam is lighter than air, the air is pushed out the bottom of the chamber (CDC). With almost all of the air removed, the hot steam is able to fully sanitize the medical equipment (CDC). However, most gravity displacement systems cannot fully remove the air which extends the time needed to fully sanitize equipment (CDC). It will often take about 45 minutes to sanitize about 10 pounds of tools and waste (CDC).

The high-speed pre-vacuum system is similar but uses a pump to create a vacuum within the chamber (CDC). This helps speed up sanitization and increase penetration power of the steam (CDC). Sanitation time decreases within this system because once the air is removed, more hot steam is able to come in contact with the equipement (CDC). This is why this type of system is often used to sanitize porous materials, the vacuum allows the steam to penetrate into the material’s pores.

Another autoclave option uses dry heat to burn and oxidize bacteria (Finkiel). These systems do not corrode metals as steam will, but they are not used as much because of how inefficient the system is (Finkiel). Dry heat is much less effective than heat transferred through steam and for this reason it takes much longer to sanitize equipment (Finkiel). With a autoclave heated to at least 320 degrees fahrenheit, it can take up to 2 hours to fully sanitize the equipment (Finkiel). This measures just when the equipment is sanitizing, to heat up the autoclave and cool down after use, the full cycle can take up to 10 to 11 hours to complete (Finkiel).

2.1.2 UV-C Sanitation

Light is proven to affect microorganisms and depending on the wavelength of the light it will have different effects on the organisms (Hessling). Specifically ultraviolet light has been tested to kill or disable microorganisms occupying many different types of surfaces and liquids. Ultraviolet light has a wavelength within a range of 10 to 400 nanometers and is divided into three different types of UV light, UV-A, UV-B and UV-C (Clordisys). Only UV-C is used for germicidal applications and operates at with a wavelength between 100 and 280 nm (Clordisys). When operating at a specific wavelength of 265 nm provides the peak for UV-C LED sanitation (Clordisys).

This specific ultraviolet frequency has commonly been used to sanitize water, air, as well as high touch surfaces (Connolly). The beginning of UV-C technology applied low pressure germicidal lamps in different ways to get rid of microbial problems. The medical field has many applications for UV-C sanitation such as large stand up lamps to disinfect high touch surfaces in hospital rooms (Clordisys). Smaller air duct lamps are also used in doctor’s offices to sanitize the air going into and out of a room with high traffic of sick patients (Connolly). Extensive studies by Harris and Bilenko explore the use of UV-C lights to sanitize water (Hessling, Harris, Bilenko). These studies analyze how effective the use of UV-C is when sanitizing water as well as determining system factors such as flow rate of the water (Hessling).

For all of these circumstances, UV-C is able to successfully sanitize because of how the UV-C wavelength of 265 nm affects the DNA and RNA of microorganisms. UV-C is a form of radiation that affects the structure of DNA and RNA in microorganisms causing them to become incapable of functioning properly and replicating (Clordisys). This means that bacteria would not be able to replicate to an infectious level, therefore it is no longer a threat to spread sickness (Clordisys). Due to its radiation properties, UV-C lights are effective against viruses, bacteria, and spores (Clordisys). The dosage of UV-C light needed to sanitize for many different viruses, bacteria, and spores is shown in Appendix A (Clordisys). Although the light waves themselves do not penetrate organic material very well, the UV-C light is also harmful. The National Toxicology Program deems UV-C radiation as “reasonably anticipated to be a human carcinogen” (Clordisys). Although dangerous, there have been a lot of progress for this technology making sanitation processes more efficient than ever before.

UV-C sanitation was first implemented using low pressure lamps which often used mercury to produce the wavelength needed (Hessling). These systems were therefore highly toxic with limited lifetimes (Hessling). This toxicity and inefficiency is pushing UV-C light toward LED applications. These LEDs no longer need toxic materials to operate and they require no special handling and disposal making them a green alternative to the original lamps (Clordisys). LEDs are overall much more versatile and electronically predictable.

UV-C LEDs allow for much more circuit planning in order to manipulate the speed of sanitation as well as the lifetime of the LEDs themselves. The UV-C LEDs are constant current devices which means that the current is proportional to the output (Connolly). But, with an increase in current also comes increased in degradation to the LED itself and as the LED accumulates more operating hours at higher currents, the max output power decreases as shown in the figure below (Connolly).

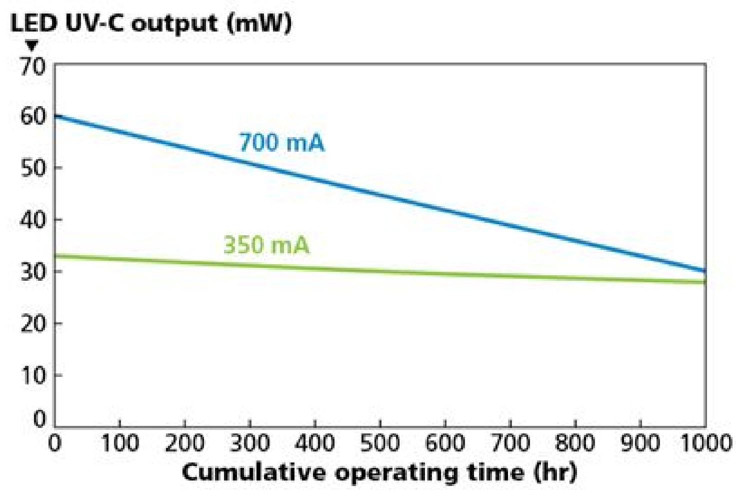


Figure 1: LED operating time vs Output

With this information, the lifetime of the LED as well as the desired power output can be manipulated to fit the required application. Figure 1 shows just 1000 hours of the LED's lifetime but industry standards require a minimum of 6000 hours to claim lifetime abilities like many manufacturers claim (Connolly). 6000 hours is a very long lifespan for UV LEDs. Through studies of water sanitation, the LEDs are only operated for about 50 hours per year (Connolly). This is because to obtain 6 log disinfection for medical purposes and high pathogen reduction, the UV-C LED only has to be operated for 30 seconds, if the output power is high enough (Connolly).

This tradeoff between the lifetime of the UV-C LED and the output power can be tested and changed depending on the target application. LED emission from the UV-C LED can be measured and recorded using a UV spectrometer then compared to the input current (Hessling). Doing this will allow further design of the power system to maximize the LED output as well as the lifetime of the LED and the battery power for our portable disinfection application.

2.2 Solar Technology

This section details solar power and it's benefits and drawbacks. It provides information on the factors that cause power loss such as setup and non-ideal energy conversion. Additionally, this section will discuss the theory behind the design of solar tracking systems and their benefits and drawbacks.

2.2.1 Solar Power Potential

The sun has been used as an energy source throughout human civilization. Although modern silicon photovoltaics, which take power directly from light radiation, have only been around since the mid 20th century, solar thermal technology or the act of focusing the sun's light to produce steam, heat or power has been around since the times of Archimedes (Breeze). As a near limitless resource, the sun's radiation is one of the most widely available resources on the planet and can be accessed anywhere on the planet, given that the sun is in view. The total solar flux that reaches the Earth's surface is somewhere between 7000 and 8000 times the global primary energy consumption. If just 0.1% of this energy was translated into usable energy at an efficiency of 10%, four times the global generating capacity of 5000GW would be generated (Breeze). This shows the incredible potential of solar powered applications around the world.

2.2.2 Photovoltaics

Photovoltaics are electrical components that when subjected to solar radiation generate a current. Also known as solar cells, photovoltaics are made of various semiconductor materials such as silicon, cadmium telluride, gallium arsenide and copper indium gallium diselenide (Breeze). Different materials used in solar cells require different amount of light energy to produce a current. As light increases in frequency, it increases in energy which makes some wavelengths of light not strong enough to be absorbed by certain materials (Breeze). In figure 2, there are some example bandwidth energies for different materials.

TABLE 13.2 Bandwidth of Some Common Solar Cell Semiconductors

Semiconductor	Bandwidth (eV)
Silicon	1.11
Cadmium telluride	1.44
Gallium arsenide	1.43
Copper indium gallium diselenide	0.9–1.7

Figure 2: Bandwidth of various semiconductors (Breeze)

As well as differing bandwidths, different semiconductor materials also produce different voltages. Using a semiconductor material with a small bandwidth allows more wavelengths to be absorbed, however the voltage across the solar cell is related to the bandwidth, also resulting in a lower voltage (Breeze). A Silicon cell for example has a voltage of 0.6V (Breeze). For most power requirements, this would not suffice. However, solar cells are connected in parallel and series in order to achieve necessary voltages. This is how solar panels are created which are used for most power generation applications.

2.2.3 Passive Tracking

In the solar tracking field, there are three main designs which are the passive tracker, the active tracker and the open loop tracker. The passive tracker is a design that makes use of liquid with a low boiling point which is placed along the east-west axis (Sidek). As the sun rises, one side heats up more than the other, heating up the liquid to a gas and either increasing pressure on one side of a movement piston or moving to the other side of the tracker, moving the panel with pure weight. Passive trackers are known to be unreliable as they are dependent on the weather around them, not just the heat from the sun. Cloudy days can often cause problems for these types of trackers and can make them inaccurate.

2.2.4 Active Tracking

Active trackers also run into problems on cloudy days. Active trackers are similar to passive trackers in the way that they follow direct sunlight but do it through photosensors. Many systems have the photosensors on all four sides of a square panel and use a control systems to make sure all photosensors are getting the same amount of light. The control system will take the information it gets from the sensors and use it to activate one or many actuators to point the panel where it needs to be. This can cause power cost issues on cloudy days as direct sunlight can be variable and cause the control system to move the panel around to balance the system, wasting energy that could be collected.

2.2.5 Open loop tracking

Open loop trackers take a different path with solar tracking. Once calibrated, open loop trackers make use of a database that the controller itself either creates or is downloaded to the controller. This database contains the direction of the sun and it’s path related to the panel’s location and points the panel along the downloaded path.

Although this design does not actually sense where the sun is, it does not run into the same issues caused by non-ideal weather that the active and passive methods do.

The position of the sun can in relation to a panel can be described with the use of two angles, the altitude and the azimuth. The Altitude is the angle between the horizon and the sun itself while the Azimuth is the angle from true north to where the altitude was taken from. This can be seen in figure 3 (Sidek).

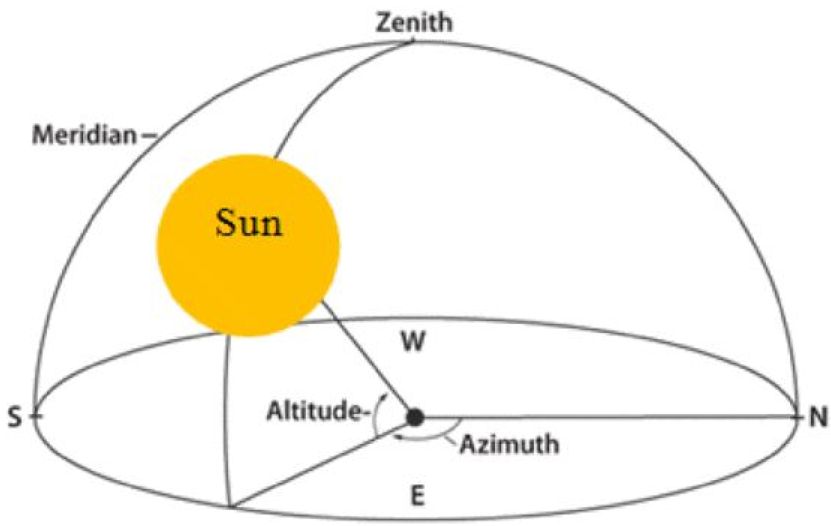


Figure 3: Depiction of the Altitude and Azimuth Angles

With the use of GPS coordinates, the time of day and the date, the sun's location can be determined with the use of the following equations ("The Sun as a Source of Energy"). First, the declination angle must be calculated. This angle takes an input n which is also known as the day number where n is equal to 1 on the first of January.

$$Declination\ Angle\ \delta\ = 23.45(\frac{\pi}{180})sin[2\pi(\frac{284+n}{36.25})]$$
(Eq. 1)

The next quantity needed is the hour angle (ω) which can be calculated using the result from the declination angle (δ) and the latitude (φ) of the solar panel. This can be calculated by the following(“The Sun as a Source of Energy”).

$$cos(\omega) = -tan(\varphi)tan(\delta)$$
(Eq. 2)

$$Hour\ Angle\ \omega = cos^{-1}(-tan(\varphi)tan(\delta))$$
(Eq. 3)

With these quantities, the Azimuth (Az) and Altitude (α) angles can then finally be computed as follows (“The Sun as a Source of Energy”).

$$sin(\alpha) = sin(\delta)sin(\varphi) + cos(\delta)cos(\omega)cos(\varphi)$$
(Eq. 4)

$$Altitude\ Angle\ \alpha = sin^{-1}(sin(\delta)sin(\varphi) + cos(\delta)cos(\omega)cos(\varphi))$$
(Eq. 5)

$$sin(Az) = -sin(\omega)cos(\delta)$$
(Eq. 6)

$$Azimuth\ Angle\ Az = sin^{-1}(-sin(\omega)cos(\delta))$$
(Eq. 7)

With the use of both the Azimuth and Altitude angles either computed or stored, a viable control system could point a solar panel in the direction of the sun.

2.2.6 Rotating Axis Tracking

In the world of solar tracking, there are a few types solar tracker axis types. They are mainly categorized into three main groups as fixed-axis, single-axis and dual-axis trackers. Both types of tracking options use a mechanical operation such as a stepper motor or passive option in order to point a panel as close to perpendicular to direct sunlight.

Fixed-axis trackers are in reality not trackers at all. A fixed-axis system is a solar panel without mechanical assistance and has a solar panel that does not move throughout the entire day. Some fixed axis panel designs are tilted to the north or south at installation to absorb more energy but often are also parallel with the ground.

Single-axis trackers are used to follow the path of the sun along one axis from east to west ("Solar Flexrack"). The most common design is the use of a single stepper motor placed at either the end of a horizontal rotating rod or at the base of the panel providing a possible change from 0 to 180 degrees in panel rotation. This type of tracker has been found in studies to improve the power gained from a solar panel in comparison to a fixed-axis design. A single-axis setup also only takes one stepper motor to operate which is less costly in means of energy used.

Dual-axis trackers also follow the sun from east to west but have a second degree of rotation that allows for tracking along the north-south axis ("Solar Flexrack"). This allows the panel to be positioned in direct sunlight in more locations and scenarios than that of a single or fixed axis system. Because of this, the dual-axis system performs better when it comes to energy absorbed ("Solar Flexrack"). However, this design also often takes the use of two motors which can be power costly on a solar system that does not have guaranteed sunlight every day.

2.3 Control Systems

This section details control systems, specifically using the Arduino family of products in order to implement sensing and control capabilities. Our sanitizer will require a control system for a solar tracking system as well as numerous sensors and timers for the sanitization operation.

2.3.1 Arduino

Arduino is an open-source electronics platform that makes use of a number of different easy to use microcontroller boards. Arduino was originally developed as a way to prototype new electronics and software design at the Ivrea Interaction Design institute (Arduino). Since then, Arduino has been grown by it's open source users with code sharing and design sharing over the internet. Arduino boards are also offered at relatively low prices on their website, with many of their boards selling for around twenty to thirty US dollars (Arduino).

One of the reasons the Arduino family of products has been so successful has been it's programming language. As well as the previously mentioned vast library of user created code at the disposal of anyone with an internet connection, Arduino's language is relatively simple to use. Using a structure nearly identical to C++, Arduino users with basic coding knowledge will find the language easy to pick up. The language also has variables and functions that are pre programmed to map directly to jumper pins and I/O ports (Arduino). This makes interfacing software with hardware very easy and not code intensive.

This easy interfacing capability becomes another benefit with the Arduino system when the open source ideology comes back into play. Being able to easily interface with I/O allows user created boards and other arduino compatible boards to be connected together. Making the Arduino ideal for design projects with a number of different hardware pieces. Because of this, a number of companies who already produce sensor boards or motor controllers have adopted the Arduino interface and it is easy to find I/O for nearly any application. This makes the Arduino a viable choice for new design projects.

3.0 Methodology

Throughout designing and testing of this system, there are many issue that were faced and solved through evaluation and correction based on the problems which arose. The final engineering prototype became a tube case with rows of UV-C LEDs spaced out by 120 degrees around the sanitation chamber. Analysis of the spacing and height of LEDs as well as how the system will be powered is analyzed and explained throughout this section.

3.1 LED Design

The LEDs which are being used for this UV-C sanitizer are 3 mm by 3 mm UV-C LEDs with a 105 degree viewing angle.. To start implementation and testing we needed to design a housing in which we can power the LED as well as dissipating the heat which is created. This includes mounting the LEDs and designing a heat sink which will mount properly to the LED.

3.1.1 LED Hardware

The UV-C LEDs have three contacts which had to be accounted for, the positive, negative, and thermal pad which is in the middle. This thermal pad is where a heatsink can be soldered or attached with thermal paste. Due to time restraints, we were unable to design PCB boards for the LEDs to be mounted on. Instead we had to use premade PDB boards and manipulate them in a way which the LEDs could be soldered on two different rails which can be used as the positive and negative power rails. From there we looked to design a heat sink which was small enough and had good connection with LED.

The first heatsink which was used was a small strip of copper tape that went between the positive and negative rails and acted as a third contact that we could manipulate the size of. The UV-C LED was then soldered to all three contacts and tested. When powered on in this configuration, the LED got much too hot to touch and handle within 10 seconds. Due to this we realized that we had to implement a larger heatsink.

To create a larger heatsink we used copper strips because copper is good to conduct and absorb a lot of the heat transmitted from the LEDs. For the two components, the copper strip and the PCB board, to fit together they had to be shaped such that the copper strip fit through the PCB board. Using a dremel with different attachments, the copper strip was shaped so one end was the correct size to fit the thermal contact of the LED, and the PCB was shaped such that the copper strip fit through in the correct location for the LED. Once shaped and testing the fit of all the components, we had to solder together the PDB board, heatsink and LED. However, when soldering the heat sink there was never a good enough connection to the LED for the heatsink to stay connected.

This is believed to be because of two factors, either the heat sink was not smooth enough or that when soldering the heat sink was not at a high enough temperature to connect to the solder correctly. In the case that we did solder the heatsink to the LED, it was a weak bond and each time the LED was turned on and heat was created, the heatsink broke from the LED. This problem means that we had to secure the heatsink in a different manor. The other option which we decided upon was a connection using thermal paste between the heatsink and LED and anchoring the heatsink to the PCB using wire. This was done by drilling a hole in the heatsink and feeding a wire from the PCB board and through the heatsink and soldering the wire to each surface. This secured the the heatsink firmly to the LED and allowed for further LED testing.

With the new heatsink, the LED was tested to determine the thermal reliability of the new heatsink system. The testing looked to analyze whether we had the proper sized heat sink and if the temperature got too high. We did this by turning the LED on for different lengths of time and determining the temperature along the length of the heatsink throughout the time period. This test showed that the new heat sink dissipate the heat along its entire length and stayed within a reasonable heat range for the LED. With the addition of a small MOSFET heat sink at the end of the copper strip we have plenty of mass to absorb the heat from the LED and surface area to radiate the heat out of the system.

3.2 LED Placement: Resulting Intensity and Time

The placement of LEDs within the sanitation chamber depends on both how far the LED is from the object as well as the spacing between the LEDs. As the LED moves away from the target, the intensity of the resulting UV-C light decreases. Also, depending on the space between LEDs, both linear and angular, the intensity of the

UV-C light may change due to the overlapping of LED viewing ranges. These two factors and the resulting UV-C intensity determines the length of time which the desired object must be exposed to UV-C lights.

All tests and evaluations of the LEDs were completed using UV-C LEDs which do not have the correct specifications. This means that the LEDs may not have the correct power output. For this reason, all LED power measurements were normalized, recorded and used to calculate the correct power intensities based on the company's power output specifications for the LEDs.

3.2.1 LED Height

The intensity of the UV-C power is inversely proportional to the distance between the LED and the target object, as the LED moves away from the object the intensity decreases. This characteristic is shown in the figure below from a range of 1 cm from the object to 10cm.

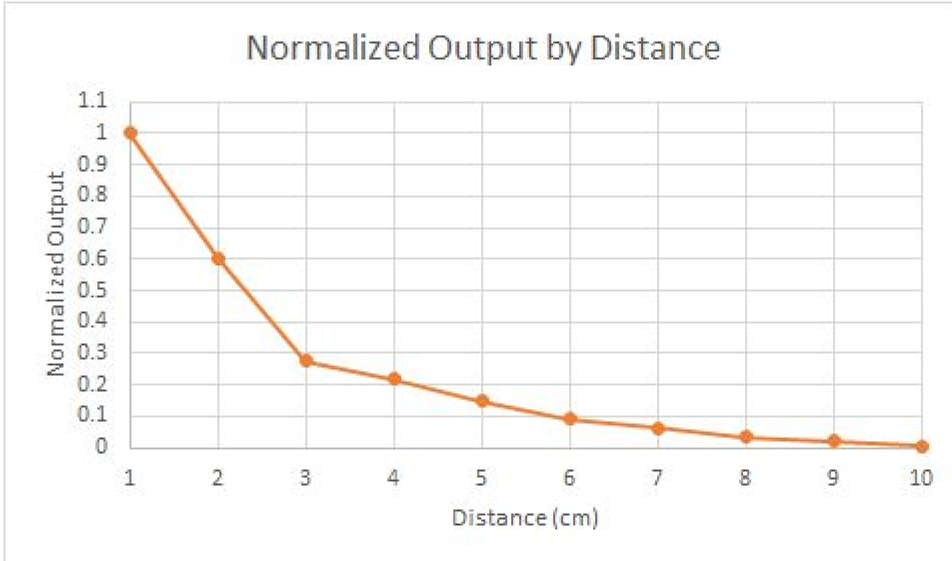


Figure 4: Normalized Output by Distance

By this graph, it is shown that the LED has an exponentially decaying curve when moving away from the target object. When we combine this LED characteristic with the expected power ratings of the UV-C LEDs we get the graph shown below.

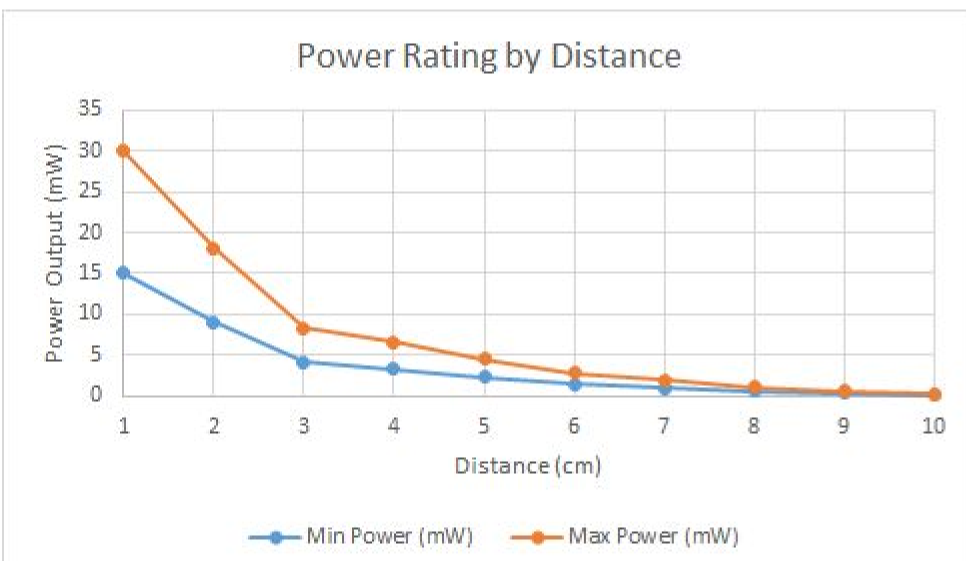


Figure 5: Range of Power by LED Height

This figure shows the range from minimum to maximum power output at each distance from the target object. The minimum intensity shown in blue, starts at the rated minimum power output of 15mW and decreases as the intensity decreases with distance. Maximum power output is also shown in orange starting at 30mW and continuing to decrease from there with movement away from the object.

3.2.2 Linear Distance Between LEDs

Next, we evaluated the effect of the spacing between LEDs on the intensity of the LED output power. This analysis was necessary because depending on how much the viewing angles of adjacent LEDs overlap the output intensity may change. The output intensity based on viewing and and height of the LED is shown in the below figure.

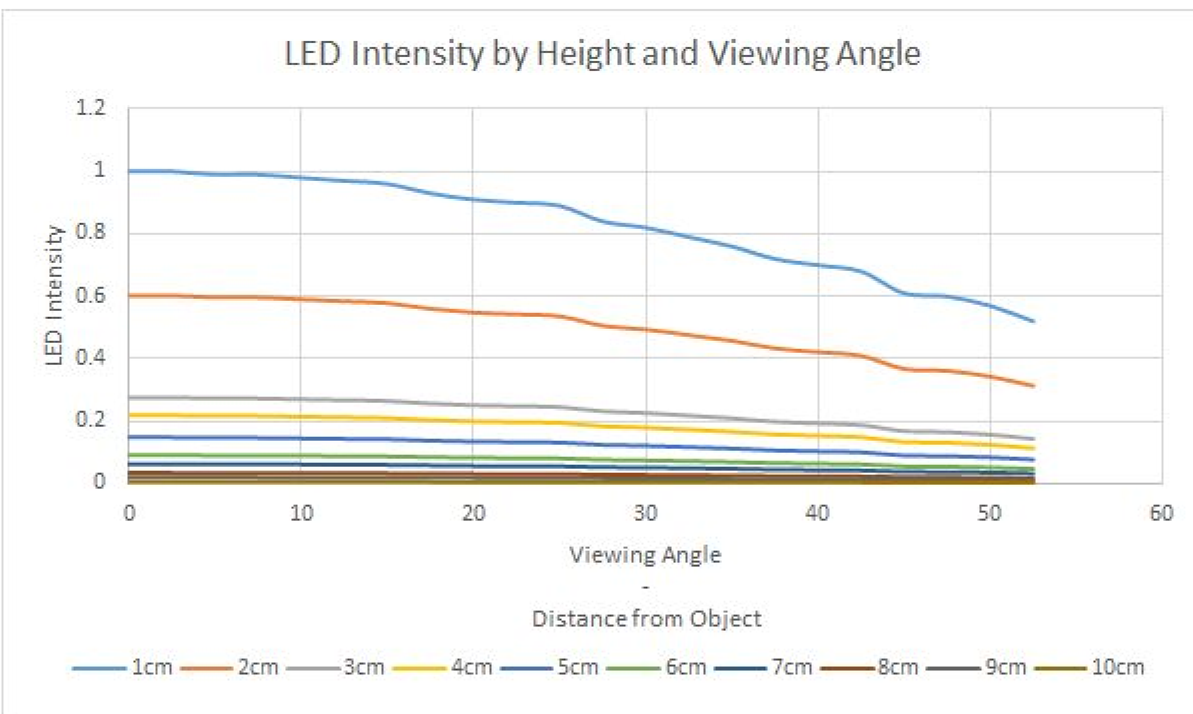


Figure 6: LED Intensity by Height and Viewing Angle

With the change in intensity based on viewing angle we can evaluate the distance between each LED based on the desired LED output intensities, 20%, 50%, and 80%. The results of this analysis is shown in the below.

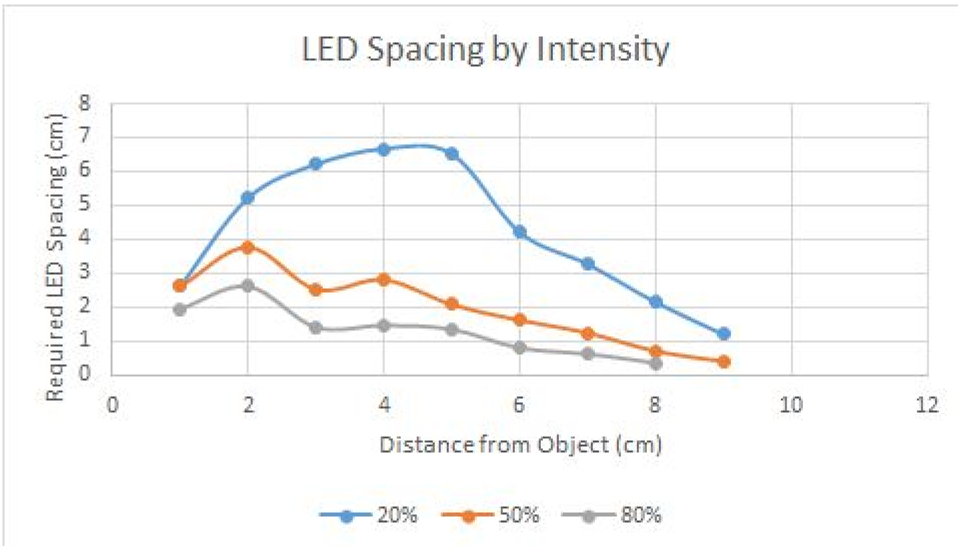


Figure 7: Intensity due to LED Spacing

This analysis above compares the required distance between the LEDs based on the LED distance away from the object and the desired LED intensity. The analysis was completed by overlapping the intensities at different LED viewing angles and calculating the distance between the viewing angles. As shown above, as the desired intensity increases, the distance between the LEDs decrease, shown by the comparison between the blue, orange, and grey curves. Using this relationship we will be able to design our system by choosing the desired intensity and the distance between LEDs within our system.

3.2.3 Time Required Based on Size of the Sanitation Region

With the analysis on the height and spacing of LEDs we can then determine the time required to sanitize objects based on the size and design of our system. If we assume that we want a 5cm of space in our sanitation range we first need to figure out which viewing angle at each LED height that supplies that amount of room. This calculation is shown in the figure below which compares the distance between the viewing angles depending on the height of the LED. The green values are the ones that provide 5cm or more for our sanitation region.

	0	2.5	5	7.5	10	13	15	18	20	23	25	28	30	33	35	38	40	43	45	48	50	53
1	0	0.1	0.2	0.3	0.4	0.4	0.5	0.6	0.7	0.8	0.9	1	1.2	1.3	1.4	1.5	1.7	1.8	2	2.2	2.4	2.6
2	0	0.2	0.3	0.5	0.7	0.9	1.1	1.3	1.5	1.7	1.9	2.1	2.3	2.5	2.8	3.1	3.4	3.7	4	4.4	4.8	5.2
3	0	0.3	0.5	0.8	1.1	1.3	1.6	1.9	2.2	2.5	2.8	3.1	3.5	3.8	4.2	4.6	5	5.5	6	6.5	7.2	7.8
4	0	0.3	0.7	1.1	1.4	1.8	2.1	2.5	2.9	3.3	3.7	4.2	4.6	5.1	5.6	6.1	6.7	7.3	8	8.7	9.5	10
5	0	0.4	0.9	1.3	1.8	2.2	2.7	3.2	3.6	4.1	4.7	5.2	5.8	6.4	7	7.7	8.4	9.2	10	11	12	13
6	0	0.5	1	1.6	2.1	2.7	3.2	3.8	4.4	5	5.6	6.2	6.9	7.6	8.4	9.2	10	11	12	13	14	16
7	0	0.6	1.2	1.8	2.5	3.1	3.8	4.4	5.1	5.8	6.5	7.3	8.1	8.9	9.8	11	12	13	14	15	17	18
8	0	0.7	1.4	2.1	2.8	3.5	4.3	5	5.8	6.6	7.5	8.3	9.2	10	11	12	13	15	16	17	19	21
9	0	0.8	1.6	2.4	3.2	4	4.8	5.7	6.6	7.5	8.4	9.4	10	11	13	14	15	16	18	20	21	23
10	0	0.9	1.7	2.6	3.5	4.4	5.4	6.3	7.3	8.3	9.3	10	12	13	14	15	17	18	20	22	24	26

Figure 8: Desired Viewing Angle by Viewing Degree and LED Height

With the desired sanitation range calculated we can then match the desired values to the LED intensity at the same viewing angles and LED height. This is shown by transferring the desired region to the comparison between the viewing angle and the LED intensity shown below. As shown in figure 9, the viewing angles which will give the greatest LED height and highest LED intensity is at the left end of each green row. We can see that there are almost no intensities above 20%. This means that on the outer edge of each row, there will be an LED intensity of 20% or less. By this, we are going to design our system around the 20% intensity curve in Figure 7 to determine the height and spacing of LEDs.

	0	2.5	5	7.5	10	12.5	15	17.5	20	22.5	25	27.5	30	32.5	35	38	40	43	45	48	50	53
1cm	1	1	0.99	0.99	0.98	0.97	0.96	0.93	0.91	0.9	0.89	0.84	0.82	0.79	0.8	0.7	0.7	0.7	0.6	0.6	0.6	0.5
2cm	0.6	0.6	0.6	0.6	0.59	0.58	0.58	0.56	0.55	0.54	0.54	0.51	0.49	0.48	0.5	0.4	0.4	0.4	0.4	0.4	0.3	0.3
3cm	0.28	0.28	0.27	0.27	0.27	0.27	0.27	0.26	0.25	0.25	0.25	0.23	0.23	0.22	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.1
4cm	0.22	0.22	0.22	0.22	0.21	0.21	0.21	0.2	0.2	0.2	0.2	0.18	0.18	0.17	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1
5cm	0.15	0.15	0.15	0.15	0.15	0.14	0.14	0.14	0.14	0.13	0.13	0.12	0.12	0.12	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
6cm	0.09	0.09	0.09	0.09	0.09	0.09	0.09	0.09	0.08	0.08	0.08	0.08	0.08	0.07	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0
7cm	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.05	0.05	0.05	0	0	0	0	0	0	0	0
8cm	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0	0	0	0	0	0	0	0
9cm	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0	0	0	0	0	0	0	0
10cm	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0	0	0	0	0	0	0	0

Figure 9: LED Intensity by Viewing Angle and LED Height

Based on the required length of out container we have three LEDs in a row at about 5.5 cm apart. Using the 20% intensity curve in figure 7, in blue, we can then determine how high the LED has to be away from the target object to provide optimal sanitization. At 5.5 cm apart, the LEDs have to be about 5 cm away from the object to have 20% intensity between the LEDs. Since this calculation in figure 7, does not account for the outside edge of the row we have to calculate the sanitation time based on the lesser value between figure 7 or figure 9 based on the height of the LED in each chart. For a 5cm height, we are evaluating between 12% and 20% intensity. We then have to evaluate the time required to sanitize based on the smallest intensity within the sanitation region so the entire sanitation region is completely covered. This analysis and calculation is completed by first taking the max required UV output of 330 mW, as shown in Appendix A, which will kill all bacteria listed. Also, knowing the maximum and minimum power output of the UV-C LED, 30mW and 15mW respectively, shown in Appendix B we can then calculate the time required to sanitize. We do this by taking the lowest intensity percentage within the sanitation region and multiple it with the specified

LED output. Also with the knowledge of the relationship 1W = 1J/s we can then calculate how long it takes by the equation, time = 330 [mJ/cm] / (LED intensity) [mJ/s]. This gives the seconds needed and using this equation to determine the time at each height from the object we obtain the results shown below.

Figure 10: Time to Sanitize by LED Height

Since our design called for a 5cm LED height, the length of sanitation ranges from about 88 to 177 seconds. This range is a one which can be managed by the LED and heatsink system. If the time period becomes a lot longer than this the LED has the possibility of becoming overheated and breaking. Also, if we want to decrease the time which it takes to sanitize, we have to decrease the LED height and therefore by Figure 7, increase the distance between the LEDs.

3.2.4 Angular LED Placement

Similar to how the linear distance between LEDs effects the overlapping intensity, the angular spacing of the UVC-LEDs around a cylindrical sanitation chamber also effects the UV intensity at angular points on the target object between LEDs. This analysis was completed by creating a test chamber along with the use of a UV sensor. The test chamber was configured with LEDs and the UV sensor hanging in the center of the chamber. The UV LEDs were set to a specific distance from the center of the chamber. With this distance, we also know the intensity of the LED and can compare this intensity to the normalized power distribution angularly around the object being sanitized. The UV-C sensor used was hung from a rotating lid in the center of the chamber and directly in front of an LED. The lid was labeled for to show a rotational displacement of 60 degrees in each direction so the sensor may be turned to determine the UV intensity at angular spots around the device being sanitized.

The results from this test are shown in the figure below. Intensities angularly around the device are symmetrical, centered at zero degrees.

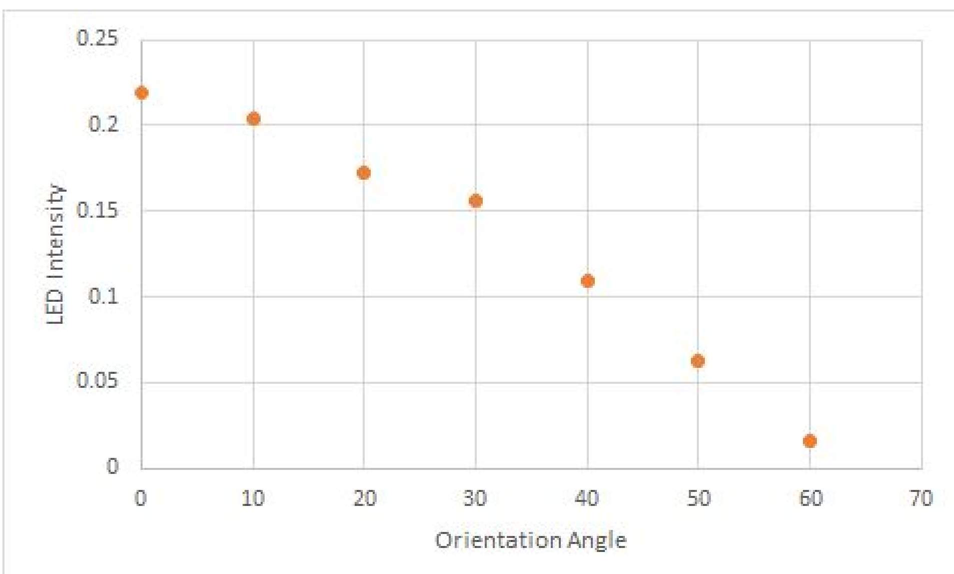


Figure 11: LED Intensity by Angle Orientation

As shown, the UV intensity diminishes as the sensor turns away from the LED. With this information we can tell the intensity at the target device in between the LEDs due to the overlapping of UV radiation. For example, if the spacing of the LEDs is 120 degrees around the sanitation chamber, there is little to no overlap in the intensity of the LEDs. Meaning, at 120 degrees between LEDs the lowest intensity becomes just under about 2.5 percent of the LED output. With two LEDs emitting radiation at that point, the resulting LED intensity becomes 5 percent of the LEDs output. This means, depending on the angular spacing of the LEDs, the time required to fully sanitize the device will change accordingly.

3.2.5 Time Requirement Due to LED Angular Orientation

With 120 degrees between each LED around a cylindrical chamber, the weakest area of UV radiation on the target object is at and angle of 60 degrees from the LED itself, directly between the two LEDs. The two LEDs therefore contribute to the radiation at that point and due to figure 11, the intensity at that point is just 5 percent of the LED’s full intensity. Since this is spot that will get the least amount of radiation we can calculate the time to sanitize using this point, to ensure all parts of the device are sanitized.

The resulting amount of time needed to sanitize due to this reduced intensity is shown in the figure below.

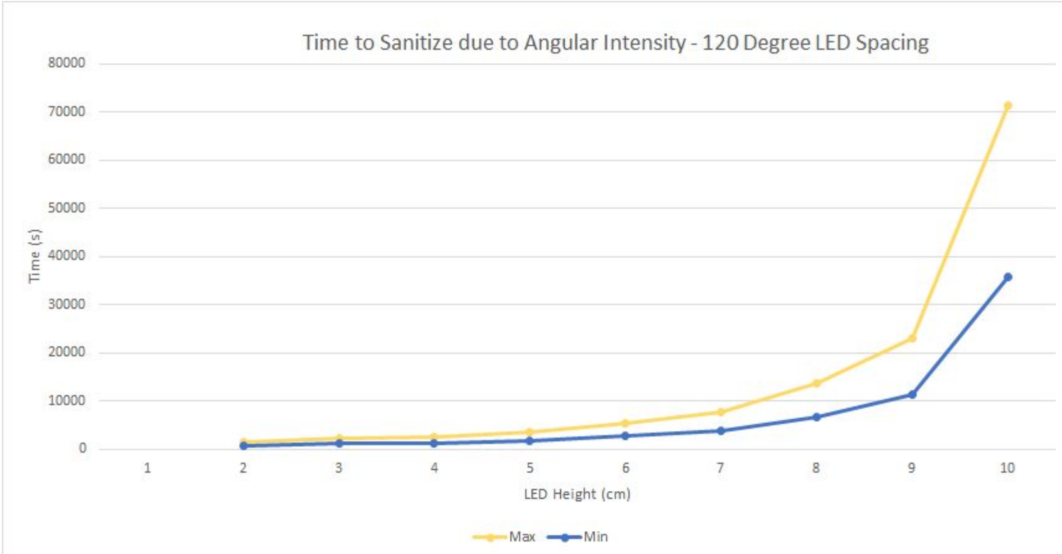


Figure 12: Time to Sanitize due to Angular Intensity - 120 LED Spacing

As shown in the above figure, the required time to sanitize greatly increases. At 5 cm the range of time needed to sanitize becomes from about 1765s to 3530s. Compared to the past time analysis due to the linear spacing of the LEDs this significantly increased the time to sanitize from a max of about 3 minutes to about an hour. This makes the power storage requirements much harder to obtain and will inevitably decrease the amount of times the system can be used due to increased energy consumption.

This can be fixed by bringing the LEDs closer together both linearly and angularly. For example, if instead of three rows of LEDs and there were 4 rows, the LEDs would be spaced 90 degrees apart. Because of this the angular intensity of the LEDs will overlap increasing the LED intensity at the weakest point and therefore decreasing the time to sanitize. The point on the target object which is directly between the LEDs becomes 45 degrees from the LED itself. Therefore, the weakest point of sanitation from 1 LED is 7.5 percent, and due to the 2 LEDs overlapping that intensity at the weakest point becomes 15 percent of the LEDs total intensity. The new time required to sanitize is shown in the figure below.

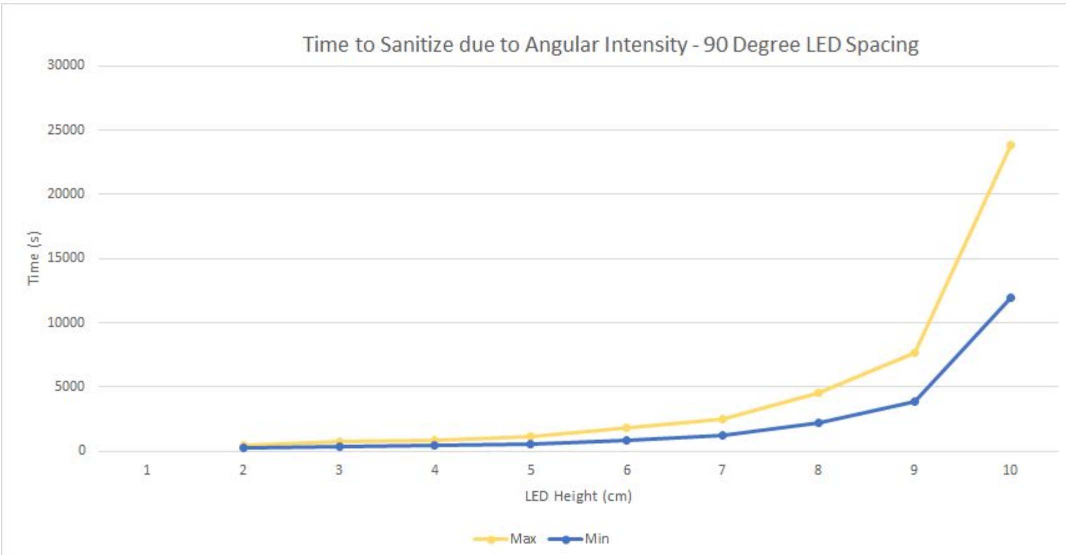


Figure 13: Time to Sanitize due to Angular Intensity - 90 LED Spacing

This greatly decreases the required time to sanitize due to the angular overlapping of the UV-C LED radiation. At 5 cm away from the target object the required time to sanitize is within the range of 588s to 1176s. This decreases the max time needed to sanitize to about 20 minutes. This is a large improvement from the max time with only three rows of LEDs but with the addition of the fourth row comes an increase in the required power to support the increased number in LEDs. This is a tradeoff which must be decided upon depending what the specific goals of the device are, a high-speed device or something which is smaller and more portable, because with increased power requirements the equipment needed to run that system becomes slightly bigger and less portable.

3.3 Power System Design Requirements

In the design of the device, there are a number of different subsystems that need access to power. Although the sanitization chamber is of utmost concern when designing the power system, there are other equally important operations that must be assessed.

3.3.1 Sanitization Chamber Power Requirements

In the design of the sanitization chamber, the parts that consume power are the UV-C LEDs. The UV-C LEDs are the GD Klaran parts from Crystal IS. From the datasheet, it was found that the parts ran on 8.45V and 350mA during typical use. From this, the following power calculation could be performed for a singular device.

$$P(\text{single LED}) = V*I = 2.9575W$$

(EQ. 8)

As can be seen from above, the LEDs tend to run off of around 3W as single units. However, it was decided that 10 LEDs would be used for the chamber. In the chamber design, the architecture of how the LEDs are connected together whether in series, parallel or both effects the power budget in different ways.

When the LEDs are all implemented in series, each of the LEDs requires 8.45V and since they are in series require 84.5V. 84.5V is was too high of a voltage to expect from a portable solution for a sanitizer. This realization pushes the design to a parallel connection implementation as shown in the figure below.

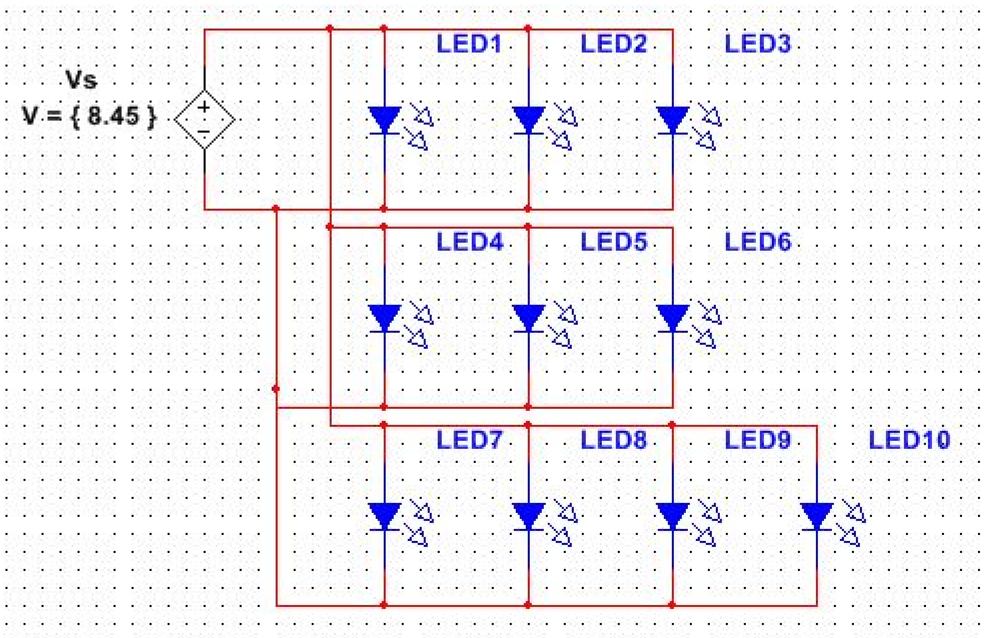


Figure 13: LED Connection Schematic

Connecting the devices in parallel allows for the implementation of a minimum voltage of 8.45 volts which can easily be provided by a solar charged system. Although the low voltage requirement allows for system portability, the design ends up draining power through a high current requirement. As all the LEDs are in parallel, the theoretical max currents can be added up to ten times 350mA or 3.5A.

$$P(10\text{ LEDs in Parallel}) = V*(3.5A) = 29.575W$$

(EQ. 9)

As can be seen above, the sanitization chamber requires almost 30W to operate properly if max current is pulled through the system. Although this can be seen as a problem, the parallel architecture was still chosen for the system. This design was chosen because of the important of maintaining the portability of the overall system as well as theoretically only needing to run the chamber in short intervals. This architecture allows for less battery weight in the overall design, improving the general carry ability.

3.3.2 Arduino Control Power Requirements

The Arduino Uno has a number of different options for power. The first is using the USB connection to power the board. Through this option, there is a 500mA fuse which limits the total power to 2.5W at 5V. However, this requires using a USB connection and eliminates the ability to use an outside DC power supply.

When using an outside DC power supply through its respective connector, the Arduino board can handle anywhere between 6 and 20 VDC. However, the Arduino recommended input voltage is between 7 and 12V to prevent the board from becoming unstable or the voltage regulator overheating respectively. With this in mind, it would be ideal to run the Arduino off of the same 8.45V source that the LEDs require. With a max current pull of around 800mA, the Arduino would at absolute most, require 6.76W. However, the Arduino will only be operating with a few of the many I/O pins, drastically reducing the used power from the maximum. Below is how the Arduino will be implemented in our design.

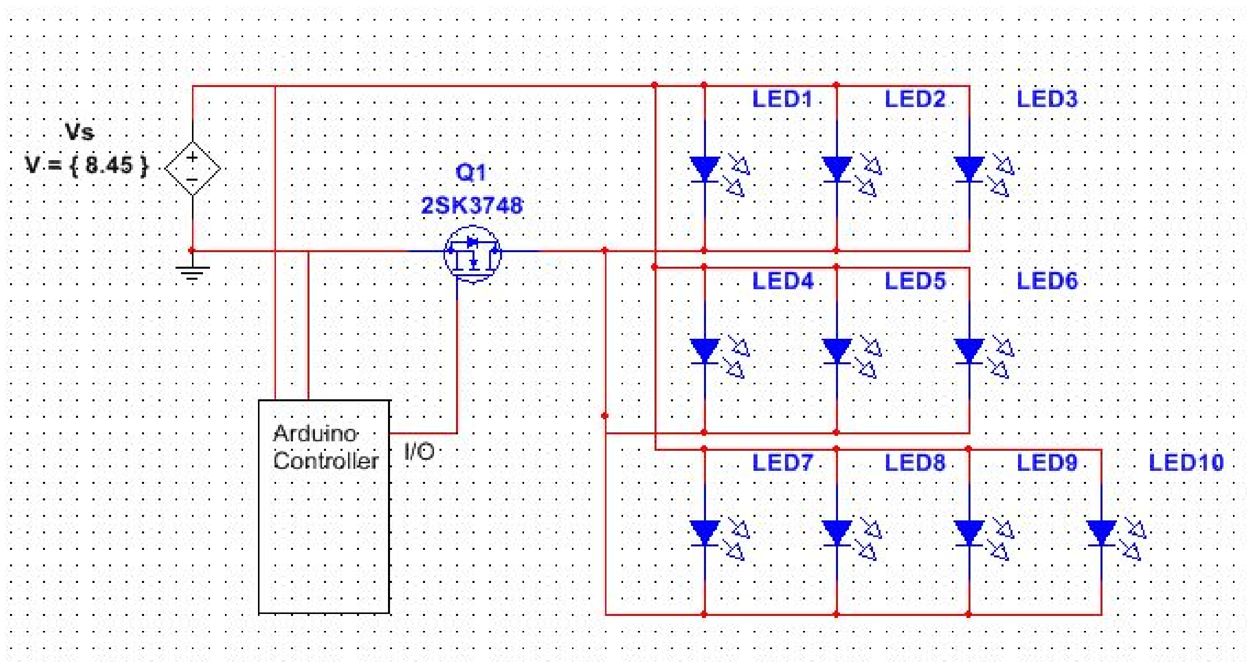


Figure 14: Arduino Control System Power Schematic

As can be seen above, the Arduino will run off of the same voltage source as the LEDs. The Arduino will also be controlling the LEDs through applying a voltage to a transistor. Along with controlling the LEDs, the Arduino will interface with a number of switches and a LED display.

3.3.3 Battery Requirements

In order to supply enough power to the device, batteries with sufficient capacity are a necessity. In choosing the batteries, this maximum current needed is used in the following calculation to tell how long the batteries will last.

$$\text{Time (hrs)} = \text{capacity rating (mAh)} / \text{maximum current (mA)}$$

(EQ. 10)

The capacity of a battery is often recorded in mAh which is milliamp hours. This measurement describes how many mA can be pulled from the battery to empty it in an hour. With this measurement, batteries of multiple capacities can be chosen from given knowledge of the maximum current the system will pull. From adding up the currents for the sanitization chamber and the Arduino, it can be seen that at a maximum, the system will pull around 4.3A or 4300mA and at a minimum around 800mA.

The batteries chosen for this project were 12VDC and rated at 6000mAh. For this application, two batteries were combined in parallel for a total capacity of 12000mAh. Using the equation above, the following table was completed for using one battery or two and how long they would last on minimum and maximum current pull.

# of 6000mAh Batteries	Imax = 4.3A = 4300mA	Imin = 0.8A = 800mA
1	1.39 hours or 83.7 minutes	7.5hrs or 450 minutes
2	2.79 hours or 167.4 minutes	15hrs or 900 minutes

Figure 15: Battery Capacity Calculations

With these calculations, the single 6000mAh battery was considered sufficient for the design. With a rated battery capacity time of 83.7 minutes at consistent max power use and a rated battery capacity time of 450 minutes at minimum power use, the battery is sufficient for multiple quick uses as was designed. As previously mentioned, cutting weight for the sake of portability is important in our design. Therefore, the single battery option was determined to be the most effective for our design goals.

3.3.4 DC-DC Converters

In order to transform the chosen battery voltage to voltage that is usable by the LEDs and Arduino, two DC-DC converters were chosen. The converters chosen were variable input/output converters with input contacts, output contacts and a dial to vary the output voltage. The battery power is applied to each of the converters in parallel, resulting in 8.5V to the LED array and 5V to the arduino. This product is made by HJ Garden and can handle the current and voltage requirements of our design.

3.4 Solar Panel Requirements and Configurations

In this section, the topic of solar panel choice and configurations will be discussed. Factors such as portability, time to charge and more will be compared against our design requirements to choose a Panel of desired characteristics.

3.4.1 Solar Panel Power Requirements

When investigating the power requirements of the solar panel, there are a number of factors in panel choice. With a battery capacity chosen from the previous section of around 12000mAh, a time to charge can then be calculated. However, these calculations are based around charging the batteries to 100% which is not always recommended.

$$T(\text{time to charge}) = (\text{Battery Capacity (mAh)} \times \text{Battery Voltage (V)}) / \text{Solar Panel Power (W)}$$

(EQ. 11)

As can be seen above, the time to charge in hours can then be calculated in regards to the listed solar panel and battery specifications. Below is a table that takes this equation and generates a time to charge for a number of different solar panel wattages.

Panel Power (W):	10	15	20	25	30	35	40
Time to Charge (hrs):	7.20	4.80	3.60	2.88	2.40	2.06	1.80

Figure 16: Time to charge for solar panel power ratings

As can be seen above, it would take a 20W solar panel 3.60 hours to charge the device to full capacity. This panel was chosen as a good middle ground choice for the design. Along with this, the panel would also be able to easily charge the batteries during the day, even allowing for a number of uses while charging.

3.4.2 Solar Panel Configuration

For this design, the solar panel has a number of different configuration types. As mentioned previously, solar panel performance can be improved by the use of solar panel tracking systems. However, most systems tend to be bulky and power intensive, questioning the practical use of a solar panel tracking system on a small design like this. Because of the overall portability challenge that is presented by the needs of the project, a stationary panel that is placed 90 degrees straight upwards was decided on.

3.5 Arduino Control System

The Arduino Control System is what brings the whole project together. The control system peripherals consist of two buttons and an indicator LED. The first button is not connected to the Arduino at all. This button is the power button and is wired across the battery power. This button is included to save power and to not have the device constantly draining saved battery capacity.

The second button is wired to the Arduino through digital pin 5. This pin is read in the code and assigned to a bool variable named chamberonoff. If this variable is LOW, i.e. the button is pushed in, the Arduino will push digital pin 7 to HIGH, applying 5V to the gate of the LED control MOSFET. This turns on the LED array. If the variable is read as HIGH, the Arduino will pull digital pin 7 to LOW, closing the transistor and turning the LEDs off. These actions can be seen below in the following chart.

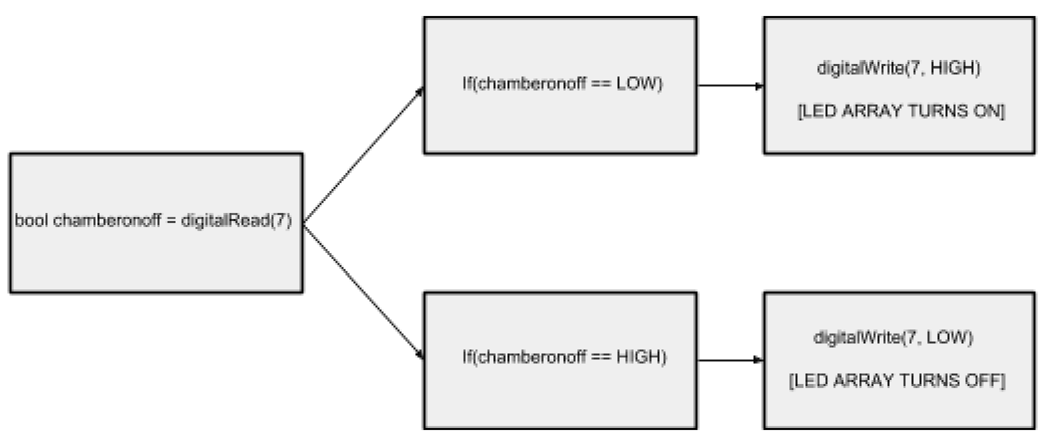


Figure 17: Arduino Transistor Control

The final peripheral in the control system is the indicator LED. The indicator LED is used to show the user what state the system is in. The system has four states and four different colors on the indicator LED for each. The first state is the “ready” state. This state is when the chamber is prepped and ready for a sanitization cycle. If the user presses the second button during this state, the upper branch of the previous figure executes. This causes the LED to turn red which is acknowledging the beginning of the “sanitation” state. This state is when the UV-C array is on and sanitizing the devices. After the cycle time completes, the light then switches to a yellow/white color which denotes the beginning of the “cooldown” state. This state is important to give the system time to cool down before the next use. During this time, the second button will not turn the LEDs back on. Once the cooldown delay is complete, the Arduino will either switch back to the “ready” state or switch to the “hold” state which is indicated by a blue light. This state is included to prevent the system from starting again immediately if the button was left in the on position. Once the button is returned to the off position, the state will change back to the “ready” state and be ready for the next cycle. The following table describes each system state.

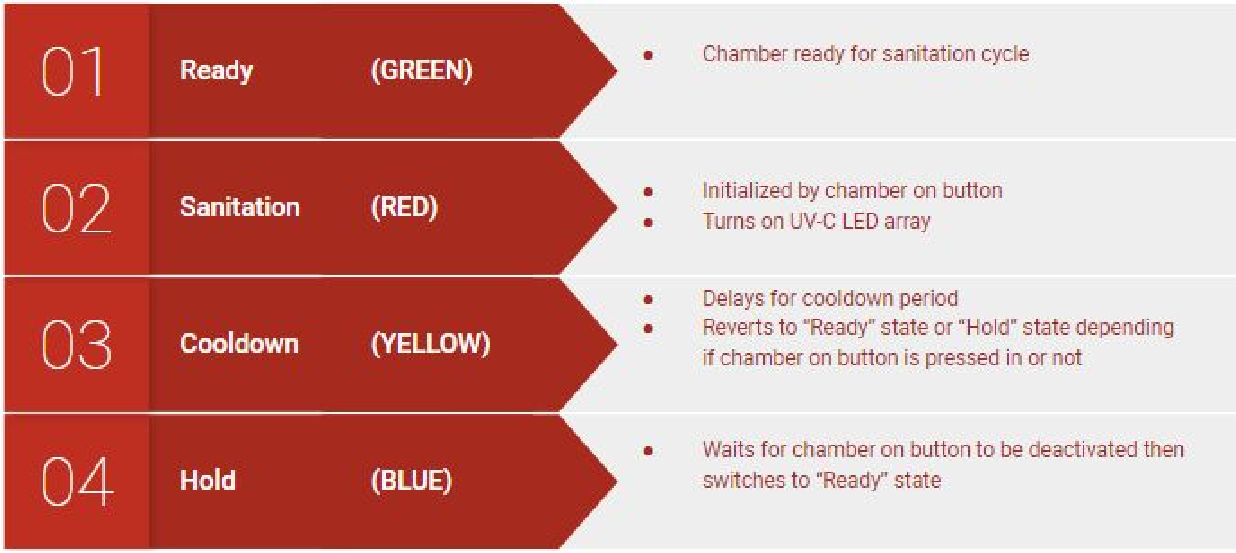


Figure 18: System state descriptions

With the implementation of these states, the Arduino serves to make the device user operable and simple for anyone to use. The entirety of the Arduino code used can be found in appendix B.

3.5 Case Design

3.5.1 Sanitation Chamber

To decide the size of the sanitation chamber we must first analyze how portable we want

the device to be. For this design we envisioned a device that was similar in size to a small backpack, making it very portable. This means, we wanted a cylindrical chamber that was about 1 foot long or 30.5 cm. Due to this size, with about 4 cm of dead space at each end, we had to divide our LEDs evenly within the remaining 22 cm. For this project we have 10 UV-C LEDs, so we decided to place them in three rows of 3 LEDs, each row spaced 120 degrees apart around the cylindrical chamber. With one LED centered in the middle of the chamber the other two were placed evenly between the center LED and the end of the sanitation region giving 5.5 cm between each LED.

From this knowledge we then had to decide the size of the cylinder. Assuming the object being sanitized will be on a small shelf directly in the middle of the tube, we could determine the distance between the LEDs and object using our analysis in Section 3.2.1. In Figure 7 it can be seen that when the distance between the LEDs is at 5.5 cm, the required height from the object must be no higher than 5.5 cm. With this we know that the diameter of the chamber has to be about 11 cm across. What we decided upon is a 4 inch or 10.16 cm tube.

Next we looked into securing the LEDs to the chamber itself. We did this by drilling holes through the tube itself and using a dremel to create pockets on the outside of the chamber for the LEDs and PCB board they are connected to. These pockets allowed for accurate placement of the LEDs as well as making it so only the LEDs protrude into the chamber. From here we designed a cradle what can hold this tube and protect all the component of the LED and power systems. Shown in the figure below is the Sanitation chamber.

Figure 19: Open Sanitation Chamber

3.5.2 Sanitation Chamber Cradle

To protect the chamber and the LED hardware attached to the tube we created a wooden cradle. This cradle provides at a minimum about 2 inches of space which protects the protruding LED heatsinks. This cradle became about 12 inches tall, 8 inches wide, and about 11 inches long. The 12 inch height of the cradle gave the top and bottom of the tube about 3.5 inches of space while the width of 8 inches allowed for 2 inches of space on the sides.

Figure 20: Sanitation Chamber Cradle - Front View

4.0 Conclusions

4.1 Effectiveness of Sanitization

Using UV-C LEDs to sanitize equipment is feasible as long as the correct design is implemented. As explained in Section 3, the speed of sanitation can be decreased by adding LEDs, making them closer together both linearly and angularly. When the LEDs are placed in a proper orientation to allow for full sanitation coverage the target object is able to become sanitized in a certain length of time. This system however may encounter some problems.

Within the chamber, the device will be sitting on a wire shelf which holds it in the center of the sanitation chamber. The points of the device which are covered by the shelf with not be sanitized. This then requires sanitizing the device moving it so the hidden portions are visible then complete the sanitation process for a second time. This problem inevitably can double your sanitation time unless another system such as a motor powered rotation system is implemented or there are materials created which are clear and do not degrade or refract the power intensity of the LED.

4.2 Power Feasibility

When it comes to the feasibility of the product in terms of power, our design on average has a 30 minute run time. When compared to the 6000mAh batteries chosen for this design, the product can run continuously for approximately 84 minutes. When the 30 minute run time is chosen, the product can only run for two complete cycles. This shows the problems within the current design rest on the current LED technology. As previously mentioned, another way to improve the time of each cycle would be to add more LEDS, but this also increases the cost and power demand for the total device. As the device does operate in its current state, the design could be improved upon with further evolution of UV-C technology.

4.3 Portability

When referring back to our original project goals, portability was of major concern. We found that the test design we created was easily carryable and small in size. This design can be carried with one hand. The solar panel is also separate from the case, having the ability to be carried separately also by one hand. The separability of the overall design and low weight of each object allows for simple movement of the parts and reassembly at the site of use. The following figure shows the test design that was created.

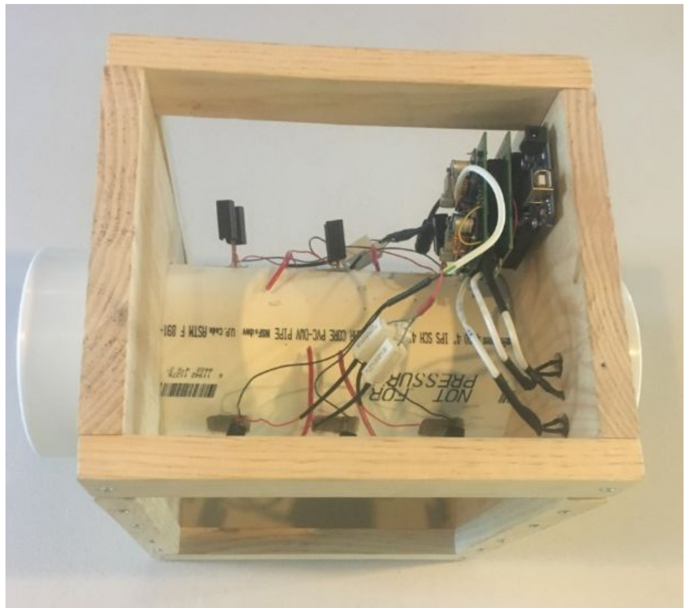


Figure 21: Test Design Build of Chamber Box

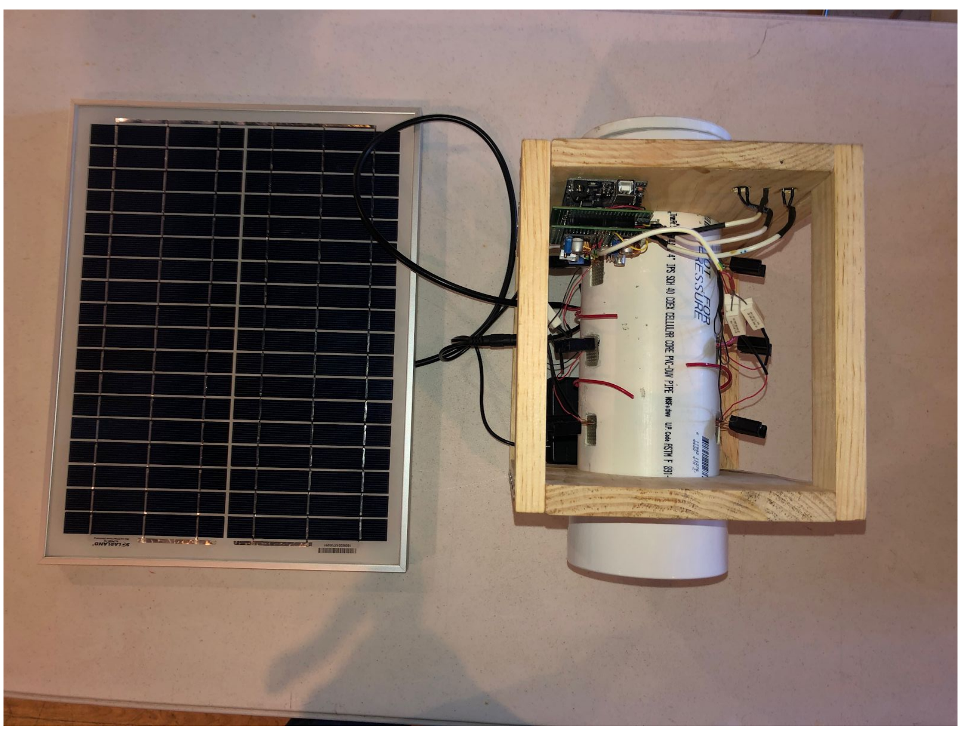


Figure 22: Device with Solar Panel

4.4 Cost/Unit and Marketability

When constructing this product, the following table shows an approximate cost per item. This list was generated from building the test design as well as speaking with LED manufacturers for healthcare grade LED costs.

PART DESCRIPTION	COST/ITEM	# OF ITEMS	TOTAL COST
Solar Panel Power Controller	$16.00	1	$16.00
12V Batteries	$30.00	1	$30.00
20W Solar Panel	$35.00	1	$35.00
DC-DC Converter	$7.00	2	$14.00
IRLB8721 MOSFET	$0.85	1	$0.85
CIS265-35R-SMGD-M (LEDS)	$62.10	9	$558.90
Arduino Uno	$20.00	1	$20.00
Addressable RGB LED	$1.50	1	$1.50
Clip-on Heatsinks	$1.00	9	$9.00
PVC Pipe	$8.00	1	$8.00
Pipe End Caps	$2.35	2	$4.70
Misc Wire / Material	$20.00	xxx	$20.00
Total Cost / Item			$759.05

Figure 23: Cost per Unit

As can be seen above, the cost per unit is estimated at around $760 dollars. About 74% of the total cost is shown to come from only the LEDs. When it comes to price, a manufactured device with these parts would run high in cost. This makes the affordability of the device difficult for small and remote health centers. For the device to be able to reach remote health programs as designed, the cost of UV-C technology has to become more affordable.

5.0 Recommendations

5.1 Heat Dispersion

During testing, it was found that the UV-C LEDs output a large amount of heat during extended use. The use of a cooldown period after each use would be beneficial to the product’s longevity and effectiveness. With the use of the on board arduino, the system could be coded to prevent the chamber from activating for a certain amount of time. This code would be enabled after every use to prevent the user from damaging the product from overuse.

During this cooldown time, the arduino could also turn on a set of fans that moves air from the outside across each row of heatsinks. To assist in the cooldown process, ventilation could be incorporated into the case design, making it easy for the fans to create continuous airflow across the heatsinks. As well as effective airflow and ventilation, the heat sinks used on this product should be made specifically for this application.

In the current design, the heatsinks are there to prevent malfunction during controlled testing of the test system. The newly designed heatsinks should be multi-fanned to increase its surface area and mass while minimizing the space the chamber takes up. As will be discussed in the next section, the heatsink design must also incorporate the ability to be mounted to the heat pad on the LED.

5.2 LED PCB Boards

As the PCB boards used in our test design were made from general test boards, the final design must allow for a number of characteristics. As shown below in Figure 22 from the LED datasheet, the LEDs used have three separate pads on their reverse side. These pads are the positive contact, the negative contact and a heat pad which is to be connected to a heatsink.

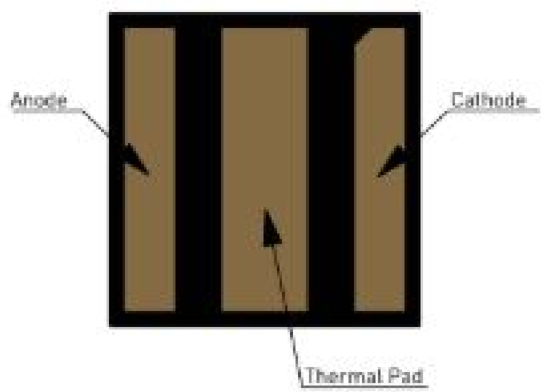


Figure 24: LED Electrical Contacts Diagram

(Appendix B)

As the LEDs are small, the mountability of the LEDs is the first characteristic required. As the contacts are small and close together, the positive and negative contacts must have small enough PCB-side connections that extend off into wire connections. The PCB board must also have space for a heatsink connection as talked about in the previous section. This connection must either be a conductive strip that goes all the way through the board or an empty space over the heat pad to feed the desired heatsink through. Finally, the PCB board must be easily mountable. Two holes on either side of the board would be optimal for secure mounting points to the chamber. These could be matched with pre milled mounting spots on the chamber to allow for ease of assembly and maintenance. The following pair of figures depicts the LED measurements and a visualization of what the desired PCB board would look like following the LED pad measurements.

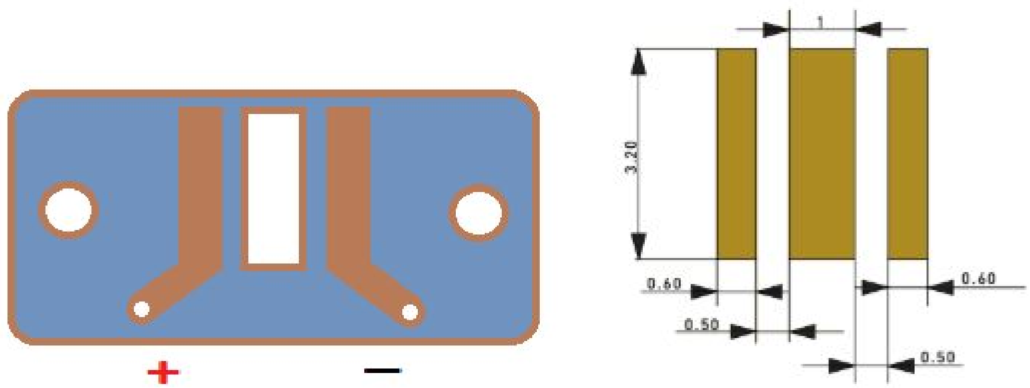


Figure 25: Recommended PCB board design and LED contact pad measurements (a-b)

(From Appendix B)

Bibliography

Arduino - Home. (n.d.). Retrieved from https://www.arduino.cc/

Bilenko, Y.; Shturm, I.; Bilenko, O.; Shatalov, M.; Gaska, R. New UV Technology for point-of-use water disinfection. In Proceedings of the 2010 Clean Technology Conference, Anaheim, CA, USA, 21–24 June 2010; pp. 339–342.

CDC. (2018, February 27). Guideline for Disinfection and Sterilization in Healthcare Facilities (2008) - Steam Sterilization. Retrieved October 1, 2018, from https://www.cdc.gov/infectioncontrol/guidelines/disinfection/sterilization/steam.html

Clordisys. (n.d.). Ultraviolet Light Disinfection Data Sheet. Retrieved September 11, 2018, from http://www.clordisys.com/pdfs/misc/UV Data Sheet.pdf

Connolly, E. (2018, May 29). Reconsider UV-C LED lifetime for disinfection based on development decisions (MAGAZINE). Retrieved September 10, 2018, from https://www.ledsmagazine.com/articles/print/volume-15/issue-5/features/developer-forum/reconsider-uv-c-led-lifetime-for-disinfection-based-on-development-decisions.html

Finkiel, M. (2013, July 4). Sterilization by Heat. Retrieved September 3, 2018, from https://tuttnauer.com/blog/sterilization-heat

Harris, T.R.; Pagan, J.; Batoni, P.; Stokes, R. Optical and fluidic co-design of a UV-LED water disinfection chamber. In Proceedings of the Electrochemical Society Meeting, Seattle, WA, USA, May 2012.

Hessling, M., Gross, A., Hoenes, K., Rath, M., Stangl, F., Tritschler, H., & Sift, M. (2016, January 27). Efficient Disinfection of Tap and Surface Water with Single High Power 285 nm LED and Square Quartz Tube. Retrieved September 11, 2018, from https://www.mdpi.com/2304-6732/3/1/7/htm

M.H.M Sidek. (2015, March 19). GPS based portable dual-axis solar tracking system using astronomical equation. Retrieved from https://ieeexplore.ieee.org/document/7062450

Solar Flexrack. (2015, November 12). What's the Difference Between Single and Dual Axis Trackers? Retrieved September 3, 2018, from http://solarflexrack.com/whats-the-difference-between-single-and-dual-axis-trackers/

The Sun as a Source of Energy - Part 3: Calculating Solar Angles. (n.d.). Retrieved September 3, 2018, from https://www.itacanet.org/the-sun-as-a-source-of-energy/part-3-calculating-solar-angles/

Breeze, Paul. (2014). Power Generation Technologies (2nd Edition) - 13.6.2 Solar Photovoltaic Costs. Elsevier. pp. 259-286. Retrieved from https://app.knovel.com/hotlink/pdf/id:kt00U84F62/power-generation-technologies/solar-photovoltaic-costs

Appendix

Appendix A - Ultraviolet Light Exposure Dosage

The degree of inactivation by ultraviolet radiation is directly related to the UV dose applied. The UV dose is the product of UV intensity [I] (expressed as energy per unit surface area) and exposure time [T]. Therefore: DOSE = I x T

UV Dose (mJ/cm^2) Needed For a Given Log Reduction

	Log Reduction						
	1	2	3	4	5	6	Reference
Spore							
Bacillus anthracis spores - Anthrax spores	24.32	46.2					Light Sources Inc. 2014
Bacillus magaterium sp. (spores)	2.73	5.2					Light Sources Inc. 2014
Bacillus subtilis ATCC6633	24	35	47	79			Mamane-Gravetz and Linden 2004
Bacillus subtilis WN626	0.4	0.9	1.3	2			Marshall et al., 2003
Bacillus subtilis spores	11.6	22.0					Light Sources Inc. 2014
Bacterium							
Aeromonas salmonicida	1.5	2.7	3.1	5.9			Liltved and Landfald 1996
Aeromonas hydrophila ATCC7966	1.1	2.6	3.9	5	6.7	8.6	Wilson et al. 1992
Bacillus anthracis - Anthrax	4.52	8.7					Light Sources Inc. 2014
Bacillus magaterium sp. (veg.)	1.3	2.5					Light Sources Inc. 2014
Bacillus paratyphusus	3.2	6.1					Light Sources Inc. 2014
Bacillus subtilis	5.8	11.0					Light Sources Inc. 2014
Campylobacter jejuni ATCC 43429	1.6	3.4	4	4.6	5.9		Wilson et al. 1992
Citrobacter diversus	5	7	9	11.5	13		Giese and Darby 2000
Citrobacter freundii	5	9	13				Giese and Darby 2000
Clostridium tetani	13.0	22.0					Light Sources Inc. 2014
Corynebacterium diphtheriae	3.37	6.51					Light Sources Inc. 2014

UV Dose (mJ/cm²) Needed For a Given Log Reduction

	Log Reduction						
	1	2	3	4	5	6	Reference
Ebertelia typhosa	2.14	4.1					Light Sources Inc. 2014V
Escherichia coli O157:H7 CCUG 29193	3.5	4.7	5.5	7			Sommer et al. 2000
Escherichia coli O157:H7 CCUG 29197	2.5	3	4.6	5	5.5		Sommer et al. 2000
Escherichia coli O157:H7 CCUG 29199	0.4	0.7	1	1.1	1.3	1.4	Sommer et al. 2000
Escherichia coli O157:H7 ATCC 43894	1.5	2.8	4.1	5.6	6.8		Wilson et al. 1992
Escherichia coli	3.0	6.6					Light Sources Inc. 2014
Escherichia coli ATCC 11229	7	8	9	11	12		Hoyer 1998
Escherichia coli ATCC 11303	4	6	9	10	13	15	Wu et al. 2005
Escherichia coli ATCC 25922	6	6.5	7	8	9	10	Sommer et al. 1998
Escherichia coli K-12 IFO3301	2.2	4.4	6.7	8.9	11.0		Oguma et al. 2004
Escherichia coli O157:H7	<2	<2	2.5	4	8	17	Yaun et al. 2003
Halobacterium elongate ATCC33173	0.4	0.7	1				Martin et al. 2000
Halobacterium salinarum ATCC43214	12	15	17.5	20			Martin et al. 2000
Klebsiella pneumoniae	12	15	17.5	20			Giese and Darby 2000
Klebsiella terrigena ATCC33257	4.6	6.7	8.9	11			Wilson et al. 1992
Legionella pneumophila ATCC33152	1.9	3.8	5.8	7.7	9.6		Oguma et al. 2004
Legionella pneumophila ATCC 43660	3.1	5	6.9	9.4			Wilson et al. 1992
Legionella pneumophila ATCC33152	1.6	3.2	4.8	6.4	8.0		Oguma et al. 2004
Leptospiracanicola - Infectious Jaundice	3.15	6.0					Light Sources Inc. 2014
Micrococcus candidus	6.05	12.3					Light Sources Inc. 2014
Micrococcus sphaeroides	1.0	15.4					Light Sources Inc. 2014
Mycobacterium tuberculosis	6.2	10.0					Light Sources Inc. 2014
Neisseria catarrhalis	4.4	8.5					Light Sources Inc. 2014
Phytomonas tumefaciens	4.4	8.0					Light Sources Inc. 2014
Proteus vulgaris	3.0	6.6					Light Sources Inc. 2014
Pseudomonas stutzeri	100	150	195	230			Joux et al. 1999
Pseudomonas aeruginosa	5.5	10.5					Light Sources Inc. 2014
Pseudomonas fluorescens	3.5	6.6					Light Sources Inc. 2014
Salmonela paratyphi - Enteric fever	3.2	6.1					Light Sources Inc. 2014
Salmonella anatum (from human feces)	7.5	12	15				Tosa and Hirata 1998
Salmonella derby (from human feces)	3.5	7.5					Tosa and Hirata 1998
Salmonella enteritidis (from human feces)	5	7	9	10			Tosa and Hirata 1998
Salmonella infantis (from human feces)	2	4	6				Tosa and Hirata 1998
Salmonella spp.	<2	2	3.5	7	14	29	Yaun et al. 2003
Salmonella typhi ATCC 19430	1.8	4.8	6.4	8.2			Wilson et al. 1992
Salmonella typhi ATCC 6539	2.7	4.1	5.5	7.1	8.5		Chang et al. 1985
Salmonella typhimurium (from human feces)	2	3.5	5	9			Tosa and Hirata 1998
Salmonella typhimurium	50	100	175	210	250		Joux et al. 1999
Salmonella enteritidis	4.0	7.6					Light Sources Inc. 2014
Salmonella typhimurium	8.0	15.2					Light Sources Inc. 2014
Salmonella typhosa - Typhoid fever	2.15	4.1					Light Sources Inc. 2014
Sarcina lutea	19.7	26.4					Light Sources Inc. 2014
Serratia marcescens	2.42	6.16					Light Sources Inc. 2014
Shigella dysenteriae ATCC29027	0.5	1.2	2	3	4	5.1	Wilson et al. 1992
Shigella dyseteriae - Dysentery	2.2	4.2					Light Sources Inc. 2014
Shigella flexneri - Dysentery	1.7	3.4					Light Sources Inc. 2014
Shigella paradysenteriae	1.68	3.4					Light Sources Inc. 2014
Shigella sonnei ATCC9290	3.2	4.9	6.5	8.2			Chang et al. 1985
Spirillum rubrum	4.4	6.16					Light Sources Inc. 2014

UV Dose (mJ/cm²) Needed For a Given Log Reduction								
		Log Reduction						
Virus	**Host**	**1**	**2**	**3**	**4**	**5**	**6**	
Adenovirus type 15	A549 cell line (ATCC CCL-185)	40	80	122	165	210		Thompson et al. 2003
Adenovirus type 2	A549 cell line	20	45	80	110			Shin et al. 2005
Adenovirus type 2	Human lung cell line	35	55	75	100			Ballester and Malley 2004
Adenovirus type 2	PLC / PRF / 5 cell line	40	78	119	160	195	235	Gerba et al. 2002
Adenovirus type 40	PLC / PRF / 5 cell line	55	105	155				ston-Enriquez et al. 2003
Adenovirus type 41	PLC / PRF / 5 cell line	23.6	ND	ND	111.8			Meng and Gerba 1996
B40-8 (Phage)	B. Fragilis	11	17	23	29	35	41	Sommer et al. 2001
Bacteriopfage - E. Coli	N/A	2.6	6.6					Light Sources Inc. 2014
Calicivirus canine	MDCK cell line	7	15	22	30	36		Husman et al. 2004
Calicivirus feline	CRFK cell line	5	15	23	30	39		ston-Enriquez et al. 2003
Coxsackievirus B3	BGM cell line	8	16	24.5	32.5			Gerba et al. 2002
Coxsackievirus B5	Buffalo Green Monkey cell line	6.9	13.7	20.6				Battigelli et al. 1993
Coxsackievirus B5	BGM cell line	9.5	18	27	36			Gerba et al. 2002
Echovirus I	BGM cell line	8	16.5	25	33			Gerba et al. 2002
Echovirus II	BGM cell line	7	14	20.5	28			Gerba et al. 2002
Echovirus I	BGM cell line	8	16.5	25	33			Gerba et al. 2002
Echovirus II	BGM cell line	7	14	20.5	28			Gerba et al. 2002
Hepatitis A	HAV/HFS/GBM	5.5	9.8	15	21			Wiedenmann et al.
Hepatitis A HM175	FRhK-4 cell	5.1	13.7	22	29.6			Wilson et al. 1992
Hepatitis A HM175	FRhK-4 cell	4.1	8.2	12.3	16.4			Battigelli et al. 1993
Infectious Hepatitis	N/A	5.8	8.0					Light Sources Inc. 2014
Influenza	N/A	3.4	6.6					Light Sources Inc. 2014
MS2 (Phage)	Salmonella typhimurium WG49	16.3	35	57	83	114	152	Nieuwstad and Havelaar
MS2 (Phage)	*E. coli* ATCC 15597	20	42	70	98	133		Lazarova and Savoye 2004
MS2 (Phage)	*E. coli* HS(pFamp)R		45	75	100	125	155	Thompson et al. 2003
MS2 ATCC 15977-B1 (Phage)	*E. coli* ATCC 15977–B1	15.9	34	52	71	90	109	Wilson et al. 1992
MS2 DSM 5694 (Phage)	*E. coli* NCIB 9481	4	16	38	68	110		Wiedenmann et al. 1993
MS2 NCIMB 10108 (Phage)	*Salmonella typhimurium* WG49	12.1	30.1					Tree et al. 1997
PHI X 174 (Phage)	*E. coli* C3000	2.1	4.2	6.4	8.5	10.6	12.7	Battigelli et al. 1993
PHI X 174 (Phage)	*E. coli* WG 5	3	5	7.5	10	12.5	15	Sommer et al. 2001
Poliovirus - Poliomyelitis	N/A	3.15	6.6					Light Sources Inc. 2014
Poliovirus 1	BGM cell line	5	11	18	27			Tree et al. 2005
Poliovirus 1	CaCo2 cell-line (ATCC HTB37)	7	17	28	37			Thompson et al. 2003

Appendix B - LED Specifications

KLARAN

DATASHEET

Klaran® GD Series UVC LEDs




PURE DISINFECTION

Klaran GD Series UVC LEDs provide effective chemical, odor and mercury free disinfection.

COMPACT SIZE

Klaran GD Series UVC LEDs easily integrate into a wide range of disinfection applications — from single LED designs up to multiple LED arrays for wide area disinfection.

REDUCED LIFETIME COSTS

With highly reliable optical output from Klaran GD Series UVC LEDs, manufacturers can extend the service life of disinfection products and reduce maintenance cost.

Product Nomenclature

Klaran LEDs are binned by peak wavelength and total power output (P_t).

Part Number	Peak Wavelength	Total Optical Power Output at 350mA	
		Min	Max
265 Series			
KL265-35P-SM-GD	260 nm - 270 nm	15 mW	20 mW
KL265-35Q-SM-GD	260 nm - 270 nm	20 mW	25 mW
KL265-35R-SM-GD	260 nm - 270 nm	25 mW	30 mW
275 Series			
KL275-35P-SM-GD	270 nm - 280 nm	15 mW	20 mW
KL275-35Q-SM-GD	270 nm - 280 nm	20 mW	25 mW
KL275-35R-SM-GD	270 nm - 280 nm	25 mW	30 mW

LED Characteristics

Characteristic	Unit	Typical	Max
Viewing angle[1]	degrees	105	
Forward voltage at 350 mA at T_s 35 °C[2]	V	8.45	10
Thermal resistance, junction-to-case at T_s 35 °C	°C/W	10	
Power dissipation at 350 mA at T_s 35 °C	W	2.96	3.50
Recommended Forward Current	mA	350	400

NOTES:

1. Viewing angle is twice of half-value angle. A half-value angle is the angle between axial direction and direction in which the light intensity value is half of the axial intensity.

2. T_s is defined as the temperature at the solder point. See Crystal IS AN010 for more information

Absolute Maximum Ratings

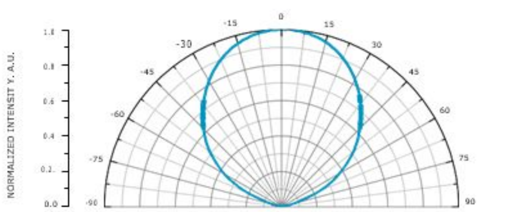
Characteristic	Unit	Min	Max
Forward current	mA	*	400
Reverse voltage	V		-5
Operating case temperature range	°C	-10	55
Storage temperature	°C	-40	100
Junction temperature	°C		115

** Note: Crystal IS recommends operating LEDs at a current greater than 10% of the noted operating maximum current to stabilize the LED characteristics.*

Typical Radiation Pattern

Klaran LEDs have a nominal viewing angle of 105°.

TYPICAL RADIATION PATTERN

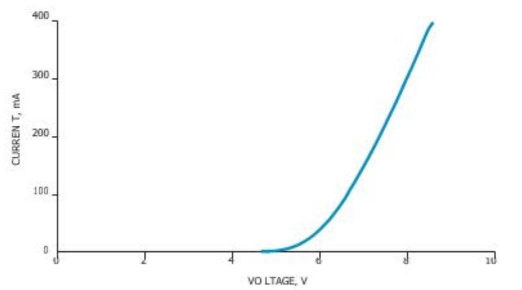


Test Conditions: I (CW) = 100 mA.
CW = Continuous Wave Mode

Typical Electrical Characteristics

The typical forward voltage is less than 10 V at an operating current of 350 mA.

ELECTRICAL CHARACTERISTICS

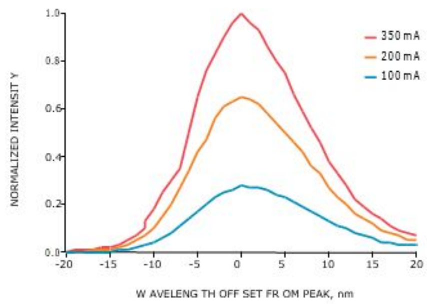


Test Conditions: Solder temperature (T_s) = 35 °C
Pulse mode operation from 1 mA to 350 mA

Typical Spectral Characteristics Over Current

The plot below shows the typical spectral emission curve for Klaran LEDs.

SPECTRUM OVER CURRENT

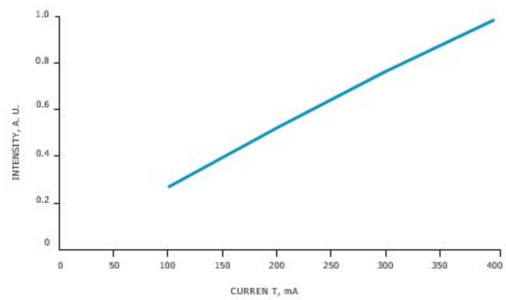


Test Conditions: Solder temperature (T_s) = 35 °C
Pulse mode operation

Typical Light Output Characteristics Over Current

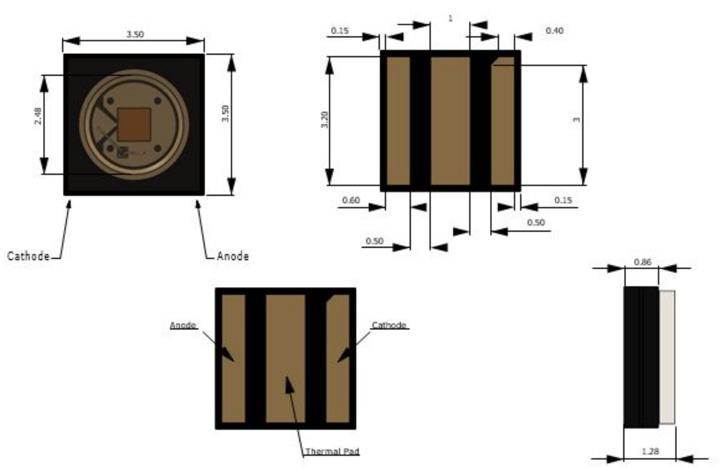
The plot below shows the typical variation in light output with forward current.

LIGHT OUTPUT OVER CURRENT

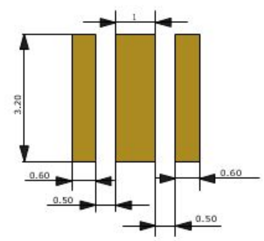


Test Conditions: Solder temperature (T_s) = 35 °C
Pulse mode operation

Mechanical Dimensions



SOLDER PATTERN

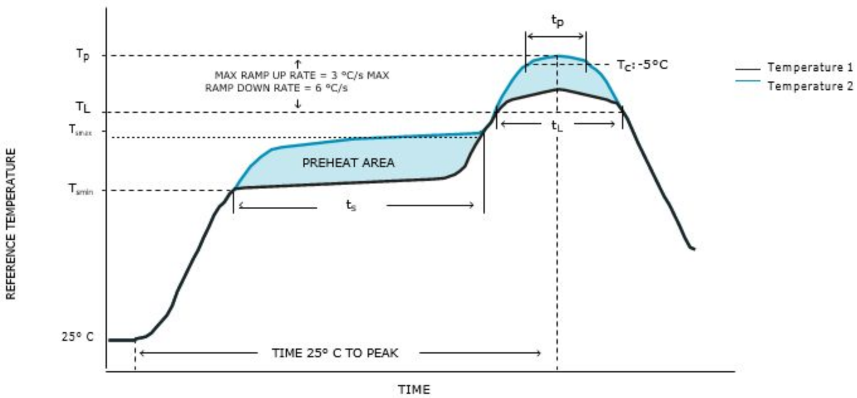


All dimensions are in millimeters. Unless noted otherwise, all dimensions have a tolerance of ± 0.05 mm.

Recommended Soldering Guidelines

The recommended solder reflow profile for Klaran UVC LEDs follows the JEDEC standard J-STD-020D.
Hand soldering is not recommended for these devices.

FIGURE 1



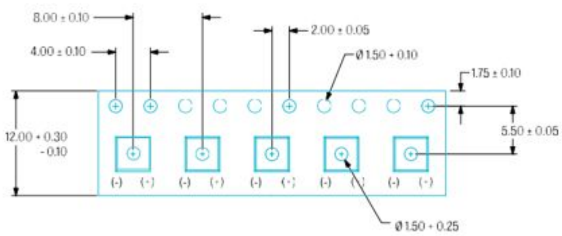
Guidelines

Profile Feature	Pb-Free Assembly
Preheat/Soak	
> Temperature Min(T_{smin})	150 °C
> Temperature Max(T_{smax})	200 °C
> Maximum Time (t_s) from T_{smin} to T_{smax}	60~120 seconds
Ramp-up rate (T_L to T_P)	3 °C/second max.
Liquidous Temperature (T_L)	217 °C
Time (t_L) maintained above T_L	60~150 seconds
Maximum peak package body temperature (T_P)	260 °C
Time (t_P) within 5 °C of the specified temperature (T_C)	30 seconds
Ramp-down rate (T_P to T_L)	6 °C/second max.
Maximum Time 25 °C to peak temperature	8 minutes max.

Reel Packaging Specification

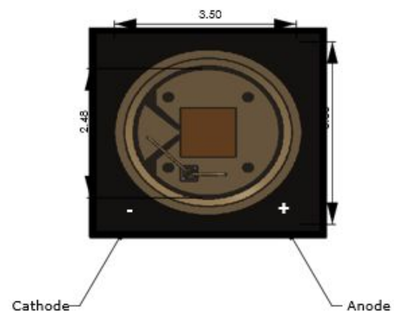
Klaran UVC LEDs are packed in tape and reel for machine manufacturing.

TAPE DIMENSIONS



All measurements are in millimeters (mm).

LED POSITION IN TAPE



Devices are placed with the cathode to the left so the polarity direction is cathode to anode.

REEL INFORMATION

Each reel includes a leader and trailer section that is not loaded with LEDs.

Handling Precautions

LEDs are sensitive to static electricity. When handling, proper ESD protection is required, including:

> Eliminating static charge

> Using grounded wriststrap, ESD footwear, clothes, and floors

> Grounded workstation and tools.

Eye Safety Guidelines

During operation, the LED emits high intensity ultraviolet (UV) light, which is harmful to skin and eyes. UV light is hazardous to skin and may cause cancer. Avoid exposure to UV light when LED is operational. Precautions must be taken to avoid looking directly at the UV light without the use of UV light protective glasses. Do not look directly at the front of the LED or at the LED's lens when LED is operational.

Attach the following warning labels on products/systems that use UV LEDs.

Compliance

The levels of environmentally sensitive, persistent biologically toxic (PBT), persistent organic pollutants (POP), or otherwise restricted materials in this product are below the maximum concentration values (also referred to as the threshold limits) permitted for such substances, or are used in an exempted application, in accordance with EU Directive 2011/65/EU on the restriction of the use of certain hazardous substances in electrical and electronic equipment (RoHS), as adopted by EU member states on January 2, 2013.

Appendix C - Arduino Control System Code

```
/*
 * Solar Powered Instrument Sanitization Device - MQP
 * Written by: Ryan Kent
 *
 * Description:
 * The following code is used to run a sanitization chamber system for medical instruments.
 * The system has a few connected peripheral devices such as:
 * 1 Button, 1 LED and 1 MOSFET Transistor.
 */

/* LED setup */
#include <Adafruit_NeoPixel.h>
#ifdef __AVR__
#include <avr/power.h>
#endif

#define PIN        5
#define NUMPIXELS     1

Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
int delayval = 500;                  /* delay for half a second */

/* Pin setup */
void setup() {
  /*pinMode(3, OUTPUT);              /* LED */
  pinMode(2, INPUT);                /* On Button */
  pinMode(7, OUTPUT);               /* MOSFET */
  pixels.begin();                   /* This initializes the NeoPixel library */
}

/* Main looping code */
void loop() {
  bool chamberonoff;
  chamberonoff = digitalRead(2);
  indicator(1);                /* Change LED for ready case */
  leds(chamberonoff);
}

/* chamber cycle */
```

```
void leds(int start) {
  if(start == LOW){
    digitalWrite(7, HIGH);              /* Activate chamber */
    indicator(3);                       /* Changes LED color for ON */
    delay(10000);                       /* Chamber run time */
    digitalWrite(7, LOW);               /* Turn off chamber */
    cooldown();
  }
}

/* cooldown process below */
void cooldown(){
  bool chamberonoff;                    /* Declare button bool */
  indicator(2);                         /* Change LED for cooldown case */
                                        /* Recommended cooling fan code belongs here */
  delay(10000);                         /* Chamber cooldown time */
  chamberonoff = digitalRead(2);
  while(chamberonoff == LOW){           /* While the chamber on button is on */
    chamberonoff = digitalRead(2);
    indicator(4);                       /* Change LED for case where button needs to be released */
  }
}


/* LED indicators */
void indicator(int color){
  if(color == 1){                       /* setting color to green */
      for(int i=0;i<NUMPIXELS;i++){
        pixels.setPixelColor(i, pixels.Color(0,255,0));
        pixels.show();.
    }
  }
  if(color == 2){                       /* setting color to yellow */
      for(int i=0;i<NUMPIXELS;i++){
        pixels.setPixelColor(i, pixels.Color(250,234,35));
        pixels.show();
    }
  }
  if(color == 3){                       /* setting color to red */
      for(int i=0;i<NUMPIXELS;i++){
        pixels.setPixelColor(i, pixels.Color(255,0,0));
        pixels.show();
    }
```

```
  }
  if(color == 4){                        /* setting color to blue */
      for(int i=0;i<NUMPIXELS;i++){
        pixels.setPixelColor(i, pixels.Color(0,0,255));
        pixels.show();
    }
  }
}
```