Solar Powered LED Blinds

By:
Austin Estes
Kerr Oliva

TA: Katherine O’Kane

4 May 2016
Project No. 28
Abstract

This document explains our design for solar powered LED blinds. The product consists of solar panels attached to the blades of the blinds, which charge a rechargeable battery pack housed at the top of the blinds. The rechargeable battery powers an electric motor, which turns the blinds, and an array of RGB, (red, green, and blue), LEDs. By using solar panels, our project is entirely self-sufficient.
Table of Contents

Contents

1.0 Introduction ................................................................................................................................. 1
2.0 Design ........................................................................................................................................... 2
   2.1 Solar Panel Array ......................................................................................................................... 2
   2.2 Charge Controller Circuit & Rechargeable Battery ................................................................. 2
   2.3 Microcontroller ......................................................................................................................... 3
   2.4 Motor Driver Circuit .................................................................................................................. 4
   2.5 RGB LED Circuit ...................................................................................................................... 4
3.0 Design Verification ....................................................................................................................... 6
   3.1 Solar Panel Array ......................................................................................................................... 6
   3.2 Charge Controller Circuit & Rechargeable Battery ................................................................. 6
   3.3 Microcontroller ......................................................................................................................... 7
   3.4 Motor Driver Circuit .................................................................................................................. 8
   3.5 RGB LED Circuit ...................................................................................................................... 8
4.0 Costs ............................................................................................................................................ 10
   4.1 Parts .......................................................................................................................................... 10
   4.2 Labor .......................................................................................................................................... 11
   4.3 Total Cost ................................................................................................................................ 11
5.0 Conclusion .................................................................................................................................. 12
   5.1 Accomplishments ..................................................................................................................... 12
   5.2 Uncertainties ............................................................................................................................. 12
   5.3 Ethical Considerations ............................................................................................................... 12
   5.4 Future Work .............................................................................................................................. 13

References .......................................................................................................................................... 14

Appendix A  Requirements & Verifications Table ........................................................................... 15
Appendix B  Power Block Diagram ................................................................................................. 18
Appendix C  Signal Block Diagram ................................................................................................. 19
Appendix D  Microcontroller Programming Flowchart ................................................................. 20
Appendix E  LT3652 Charge Controller Circuit ............................................................................ 21
<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix F</td>
<td>SN754410 H Bridge Chip</td>
<td>22</td>
</tr>
<tr>
<td>Appendix G</td>
<td>LTC3619B Current Limiter Circuit</td>
<td>23</td>
</tr>
<tr>
<td>Appendix H</td>
<td>RGB LED Circuit Schematic</td>
<td>24</td>
</tr>
</tbody>
</table>
1.0 Introduction

Recently, many people have been rightfully concerned about the environment and our effect on it due to pollution. In order to make using green energy more consumer friendly and easier to install, we designed solar powered LED blinds. This way, instead of having the blinds only block the sun’s light, the solar panels will also convert the solar energy to electricity. This electricity is stored in a rechargeable battery, which can be used to power RGB LEDs at night to add an accent color to a room.

By having our product on a set of blinds, the difficulty of the installation is only as difficult as installing a new set of blinds for a window. On the other hand, installing solar panels for home use can be expensive, when considering the cost of higher wattage solar panels and the cost of professional installation. This means that our product could easily be installed in several rooms without the homeowner having to do an expensive retrofit.

Since our system is self-sufficient, the solar panels produce all of the power that is stored in the rechargeable battery for use later by the motor during the day or the LEDs later at night. Our power block diagram also shows the essential subsystems between the main components, such as the charge controller circuit, the motor driver circuit, and the transistor current mirror circuit (See Appendix B). In conjunction with the power flow through our project, there are also various signals that are sent to different parts in order for portions of the circuit to be activated (See Appendix C). A microcontroller reads the voltage after the photoresistor and uses the reading to determine how much to turn the blinds with the electric motor, if the automation switch is flipped. Similarly, if the LED switch is on, the microcontroller will activate the RGB LEDs.
2.0 Design

2.1 Solar Panel Array

The solar panel array acts as the power generator of our entire project. Its sole purpose is to output power harnessed from the sun, which is then modulated through a charge controller circuit in order to charge a rechargeable, lithium-ion battery. The first iteration of the solar panel array consisted of twenty-seven solar panels. These panels were configured in series of three with nine parallel lines. With each individual panel rated for 6 V and 70 mA, each parallel line was rated for 18 V and 70 mA. Using Equation (1), we can see that the solar panels outputs 11.34 W in ideal conditions, that is, in full sun.

\[ P = IV = (0.63)(18) = 11.34 \text{ W} \]  

The motivation behind this design was to produce enough current to charge the 6600 mAh lithium-ion battery within the limited ten hours of sunlight in Champaign, IL which is the average minimum hours of daylight. [9] Using Equation (2), we can see that the solar panels would charge the battery in 10.48 hours for this design.

\[ \text{Charging Time} = \frac{\text{Battery Capacity (Ah)}}{\text{Charging Current (A)}} = \frac{6600 \text{ mAh}}{0.63 \text{ A}} \times \frac{1 \text{ A}}{1000 \text{ mA}} = 10.48 \text{ hours} \]  

However, in the final design, a more cost effective design was implemented; it consisted of only twelve solar panels. Similar to the original idea, the panels were configured in series of three but with only four parallel lines. The new panels that were used were rated for 6 V and 166mA. Consequently, this resulted in a power output of 11.952 W.

2.2 Charge Controller Circuit & Rechargeable Battery

The charge controller utilized was an integrated circuit by Linear Technology; the chip used was the LT3652. As for the battery, we used a three-cell, 11.1 V lithium-ion battery with a capacity of 6600 mAh. The LT3652 was chosen as our charging circuit because of its capability of charging lithium-ion batteries and its ability to accommodate solar panels as its power source. When programming the chip, the charging characteristics of a lithium-ion battery were taken into consideration. To properly charge a lithium-ion battery, there are multiple charging stages: constant current stage, constant voltage stage, and charge termination. [8] During the constant current stage, the current being delivered to the battery should be within the range of 0.2C A to 1.0C A, where C is the capacity of the battery. Upon completion of the constant current stage,
each cell should reach a voltage of 4.2 V. The battery is then held at this voltage until charge termination occurs; charge termination occurs when a charge current of 0.02C A has been reached. The LT3652 chip implements all of these stages through programming.

The nominal voltage for a lithium-ion cell is 3.70 V. [7] Thus, the nominal voltage for the battery is 11.1V. When charging, the battery voltage will reach 12.6 V. The LT3652 however was programmed to have a battery float voltage of 12.3 V, that is, each cell will charge to a voltage of 4.1 V (See Appendix E). This was done to account for any voltage deviation that could occur when charging; it provides a 0.3 V cushion to prevent exceeding 12.6 V. A voltage larger than 12.6 V could damage the battery and result in combustion. The charging current was programmed to output 0.2C A which equates to 1.32 A for our battery. This was done because the standard charging rate for the lithium-ion battery was 1.3 A. Programming the charging current to be larger would definitely speed up charging time but would also decrease the life of the battery due to increased heat.

2.3 Microcontroller

The microcontroller used was an ATmega328P chip from an Arduino Uno board. Using only the chip and not the entire board on our PCB reduced the power consumption, extending the life of our lithium-ion battery. The microcontroller takes three inputs, the first from a switch that activates the automatic blind positioning by activating the electric motor. The second input is from the switch that activates the LED signal, which is a square wave. Lastly, the third input is from a photoresistor position on the blinds near the solar panels.

When the motor switch is triggered, the microcontroller begins to execute our algorithm to find the local maximum for light intensity (See Appendix D). To begin, the microcontroller’s algorithm turns the blinds in one direction until the voltage values read after the photoresistor begin to decrease. At this point, the microcontroller begins to check the photoresistor readings in the opposite direction, stopping once the voltage readings begin to decrease again. Once this occurs, the microcontroller knows it has found a local maximum and will wait for fifteen minutes before looping through the algorithm again to find the sun’s new position.

If the motor switch is not turned on, then the microcontroller checks if the LED switch is active. If it is, then the microcontroller begins to send a square wave with a frequency of 60 Hz to the switch transistor, which provides power to all of the LEDs. We chose to use a square wave of frequency 60 Hz in order to reduce the amount of power being drawn by the LED circuit, opposed to having a constant high current running through the circuit. Using a frequency of 60 Hz also means that the RGB LEDs still seem as though they are running continuously, and not flashing or pulsing.
2.4 Motor Driver Circuit

The motor driver circuit contains three main components, an H bridge, a current limiter, and DC motor. The first component of the motor driver circuit is the SN754410 H bridge chip. The H bridge contains two different power inputs, one for the chip, and a second that provides the power that is sent to the motor. The H bridge takes three signals; one enable bit, and then two select bits that select the direction of the motor. 1A selects 1Y as the output of power for the motor, while 2A selects 2Y (See Appendix F). Whichever direction is selected, power will then be sent through the corresponding output. This way, the microcontroller is able to control which direction the motor will turn.

We placed a current limiter between the output from the H bridge and the input of the electric motor as a safety precaution. In the case of the motor stalling when trying to turn the blinds, the motor will begin to draw more current than it is rated for, which could damage the motor and also lead to a fire hazard. The current limiter was also from Linear Technology, it was a LTC3619B IC (See Appendix G). We chose the resistors and capacitors in order to limit the current to 400 mA, which was the highest current rated for our motor.

The last component of the motor driver circuit is the DC motor itself. Even though the motor was selected was a simple DC motor, it came with gearing attached that turned the axis of rotation ninety degrees and geared the motor for torque instead of speed. This was beneficial since it allows the motor to sit flush with the top of the blinds while allowing it to be attached to the pulley system that turns the blinds.

2.5 RGB LED Circuit

The RGB LED array consisted of 10 RGB LEDs. Each LED was rated for an operating current of 60 mA; 20 mA was required for red, green, and blue light individually. In terms of the voltage required by the LEDs, the red lead required 2.0 V while the green and blue leads both required 3.2 V. The actual RGB LED circuit can be broken down into three subsections: a switch-transistor, an operational-amplifier, and current mirrors. The idea behind our design was to have the microcontroller output a 5 V signal upon enabling the LED circuit through the use of a switch. The output of the microcontroller would then be amplified by the op-amp to 9.5 V. The voltage output of the op-amp would then be applied to the base of the switch-transistor which would then allow the flow of power through the transistor to the current mirrors.

In order for our circuit to operate, we biased the voltage at the collector terminal of the transistor to also be 9.5 V. The reasoning behind this was so that the transistor would behave like a diode-connected transistor when the 9.5 V was applied to the base. The actual biasing was done by adding a resistance of 1.333 Ω at the collector terminal. This resistance was determined by
knowing 1.2 A was needed to power all colors of the ten RGB LEDs. Equation (3) verifies that
the 1.333 Ω draws 1.2 A of current which is the current flowing into the collector terminal. $V_{\text{bat}}$
is the battery voltage, $V_c$ is the collector voltage, and $R_c$ is the resistor.

$$I_c = \frac{V_{\text{bat}} - V_c}{R_c} = \frac{11.1V - 9.5V}{1.333\Omega} = 1.2\text{ A} \quad (3)$$

As a result of the transistor being diode-connected, the voltage at the emitter terminal
would be 8.8 V to reflect the 0.7 V drop across the transistor. At the emitter terminal of the
switch-transistor, we have the current mirrors which drive the RGB LEDs. The current mirror
works by having one biasing transistor and a mirror transistor. The same type of transistor should
be used for both the biasing and mirror transistor so that they have the same device properties. In
terms of actual operation, the biasing transistor is set to draw some specified current and the
mirror will draw the same amount of current. Thus, three current mirrors are needed to control
the current for the red, green, and blue leads of the LEDs. The biasing transistor was diode-
connected to allow current to flow through. An initial resistor was added in series to the biasing
transistor to actually draw the required current 0.2 A. Equation (4) shows how the resistor value
was derived for each of the current mirrors.

$$R_{\text{bias}} = \frac{V_{\text{ST,E}} - V_{\text{diode}}}{I_{\text{bias}}} \quad (4)$$

In Equation (4), $V_{\text{ST,E}}$ is the emitter voltage of the switch-transistor. $V_{\text{diode}}$ is the voltage
across the diode-connected transistor in the current mirror and $I_{\text{bias}}$ is the current we want running
through the biasing transistor. Plugging in our values results in a resistor value of 40.5 Ω as
shown in Equation (5).

$$R_{\text{bias}} = \frac{8.8V - 0.7V}{0.2A} = 40.5\Omega \quad (5)$$

In addition to the initial resistor in each of the current mirrors, a 500 Ω potentiometer
rated for 25 W was added in series. This potentiometer allowed for the variation of current draw
in the biasing transistor. Thus, the user could determine what light color they could produce with
the RGB LEDs. Refer to Appendix H for the full RGB LED circuit schematic.
3.0 Design Verification

The following sections describe our testing and verification procedure for all of our subsystems. Refer to Appendix A for our requirements and verifications table.

3.1 Solar Panel Array

The testing undergone by our solar panel array simply consisted of measuring the open-circuit voltage of the panels. First, the voltage each individual panel was measured to ensure that each panel functioned. The solar panels were then configured in series of three and the open-circuit voltage across the three panels were measured in both full sun and light overcast conditions. Measurements were taken in light overcast conditions to see what kind of effect it had on the output power of the solar panels. Under a light overcast, solar panels might output about half as much power as they would in full sun. The measured open-circuit voltages along with the percent change between the two conditions are tabulated in Table 1.

Table 1. Measured open-circuit voltages of solar panels.

<table>
<thead>
<tr>
<th>Solar Panels</th>
<th>Full Sun</th>
<th>Light Overcast</th>
<th>Percent Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Series 1</td>
<td>18.50 V</td>
<td>15.21 V</td>
<td>17.78 %</td>
</tr>
<tr>
<td>Series 2</td>
<td>19.28 V</td>
<td>16.57 V</td>
<td>14.06 %</td>
</tr>
<tr>
<td>Series 3</td>
<td>19.15 V</td>
<td>16.24 V</td>
<td>15.20 %</td>
</tr>
<tr>
<td>Series 4</td>
<td>18.75 V</td>
<td>15.43 V</td>
<td>17.71 %</td>
</tr>
<tr>
<td>Average</td>
<td>18.92 V</td>
<td>15.86 V</td>
<td>16.17 %</td>
</tr>
</tbody>
</table>

3.2 Charge Controller Circuit & Rechargeable Battery

Ideally, the charge controller circuit should take the power outputted by the solar panel array and modulate it to the programmed battery float voltage of 12.3 V and the maximum charging current of 1.32 A. We were able to simulate this process using Linear Technology’s simulation program. As for actual charge controller circuit testing, we were unable to complete our verification process.
3.3 Microcontroller

The microcontroller was tested by reading the voltage output from the pins of the ATmega328P. Logical high signals are conventionally 5 V, which is what the microcontroller was specified to output. We wanted to be sure that the output of the microcontroller would be what we expected, since we boosted the signal from 5 V to 9.5 V using an op-amp for the base of the switch transistor. When we tested the microcontroller, the voltage value being output was 4.8 V, which gives an error of 4% compared to 5 V.

Secondly, we wanted to ensure that the microcontroller would trigger signals at the correct time that was programmed. For example, when the microcontroller activated the LEDs, it sent a square wave with a frequency of 60 Hz. This means the voltage output of the microcontroller to LEDs changed from low to high or vice versa every 8 ms. Using an oscilloscope, we observed that the microcontroller did give us the correct frequency of square wave to trigger the LEDs so that the user would not be able to tell that the LEDs were flashing. The oscilloscope reading can be seen in Figure 1, which shows that the microcontroller was able to trigger the switch transistor correctly. The microcontroller’s voltage is the yellow signal in graph, and the switch transistors voltage is the green signal. One can also observe that the transistor does activate when the microcontroller’s signal is high, verifying further that our LED circuit worked as we had designed.

Figure 1. An oscilloscope reading of the microcontroller’s voltage output, (yellow), in comparison with the voltage from the switch transistor, (green).
3.4 Motor Driver Circuit

In order to test our motor driver circuit, we had two separate testing phases. First, we coded a test program for the microcontroller that was meant to verify that the microcontroller’s signals to the H bridge would be able to control the direction of the DC motor. The test program’s simple algorithm would make the microcontroller output a logical high signal to input 1A of the H bridge for one second, and then a logical high signal to input 2A (See Appendix F). By using the test program, were able to determine that the microcontroller could successfully control the motor and which direction it spins.

Next, the second phase of testing the motor driver circuit was to demonstrate that the microcontroller would be able to read the voltage after the photoresistor attached to the blinds. Then, using the photoresistor reading, the microcontroller should direct the blinds until it finds a local maximum of light intensity. During this phase of testing, we were unable to show that with different light intensities on the photoresistor, the microcontroller would turn the motor a different amount of intervals. This could be due to the microcontroller not correctly reading the voltage after the photoresistor or the algorithm for turning the blinds containing a bug in the code.

3.5 RGB LED Circuit

Initially, all the RGB LEDs were tested individually to ensure proper functionality. This was done by applying 5 V at the anode of the LED and placing resistors in series with each of the leads so that there was a 20 mA current draw for each color. Once it was verified that each LED was operational, the RGB LED circuit was tested. Many measurements were taken including switch-transistor terminal voltages and currents running through the current mirrors. As stated before, we wanted the switch-transistor to be diode connected. Thus, the voltages at collector and base terminal should both be 9.5 V and cause the emitter terminal voltage to be 8.8 V. Table 2 holds the expected voltages, the measured voltages, as well as the percent error between them.

<table>
<thead>
<tr>
<th>Transistor Terminal</th>
<th>Theoretical Voltage</th>
<th>Measured Voltage</th>
<th>Percent Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base</td>
<td>9.5 V</td>
<td>9.32 V</td>
<td>1.89 %</td>
</tr>
<tr>
<td>Collector</td>
<td>9.5 V</td>
<td>9.44 V</td>
<td>0.63 %</td>
</tr>
<tr>
<td>Emitter</td>
<td>8.8 V</td>
<td>7.95 V</td>
<td>9.66 %</td>
</tr>
</tbody>
</table>

Table 2. Switch-transistor voltages.
Table 3 holds the measured currents running through each of the biasing transistors within the current mirror.

**Table 3. Currents running through biasing transistor.**

<table>
<thead>
<tr>
<th>Biasing Transistor</th>
<th>Measured Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>0.172 A</td>
</tr>
<tr>
<td>Green</td>
<td>0.172 A</td>
</tr>
<tr>
<td>Blue</td>
<td>0.173 A</td>
</tr>
</tbody>
</table>
### 4.0 Costs

#### 4.1 Parts

Table 4. The parts used for this project and their prices.

<table>
<thead>
<tr>
<th>Item/Service</th>
<th>Model Number/Name</th>
<th>Vendor</th>
<th>Unit Cost</th>
<th>Quantity</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solar Panel</td>
<td>6V 166mA Solar Cell</td>
<td>EachBuyer</td>
<td>$3.29</td>
<td>12</td>
<td>$39.48</td>
</tr>
<tr>
<td>Battery</td>
<td>Tenergy 18650 11.1V 6600mAh Rechargeable Battery Pack</td>
<td>All-Battery</td>
<td>$63.59</td>
<td>1</td>
<td>$63.59</td>
</tr>
<tr>
<td>Microcontroller</td>
<td>Arduino Uno - R3</td>
<td>Sparkfun</td>
<td>$24.95</td>
<td>1</td>
<td>$24.95</td>
</tr>
<tr>
<td>RGB LEDs</td>
<td>LED - RGB Clear Common Anode</td>
<td>Sparkfun</td>
<td>$1.95</td>
<td>30</td>
<td>$58.50</td>
</tr>
<tr>
<td>Motor</td>
<td>Solarbotics Gear Motor 3 - GM3 224:1 90 Degree Shaft</td>
<td>Solarbotics</td>
<td>$7.00</td>
<td>1</td>
<td>$7.00</td>
</tr>
<tr>
<td>Blinds</td>
<td>White Faux Wood Blind</td>
<td>Home Depot</td>
<td>$19.77</td>
<td>1</td>
<td>$19.77</td>
</tr>
<tr>
<td>H-bridge Chip</td>
<td>SN754410</td>
<td>Sparkfun</td>
<td>$2.35</td>
<td>2</td>
<td>$4.70</td>
</tr>
<tr>
<td>Voltage Regulator</td>
<td>L7805 5 V Regulator</td>
<td>Sparkfun</td>
<td>$0.95</td>
<td>2</td>
<td>$1.90</td>
</tr>
<tr>
<td>Charge Controller</td>
<td>LT3652</td>
<td>Linear Technology</td>
<td>$0</td>
<td>2</td>
<td>$0</td>
</tr>
<tr>
<td>Current Limiter</td>
<td>LTC3619B</td>
<td>Linear Technology</td>
<td>$0</td>
<td>2</td>
<td>$0</td>
</tr>
<tr>
<td>Potentiometer</td>
<td>25W 500 Ohm High Power Wirewound Potentiometer</td>
<td>Electronics-Salon</td>
<td>$13.00</td>
<td>3</td>
<td>$39.00</td>
</tr>
<tr>
<td>Wood (Frame)</td>
<td>1” by 3”, 8’ long</td>
<td>Home Depot</td>
<td>$3.50</td>
<td>3</td>
<td>$10.50</td>
</tr>
</tbody>
</table>

Various resistors and capacitors used were free from the ECE parts shop.
4.2 Labor

Table 5. The cost of labor for this project.

<table>
<thead>
<tr>
<th>Name</th>
<th>Hourly Rate</th>
<th>Total Hours Invested = 10 Weeks x 15 Hours/Week</th>
<th>Total = Hourly Rate x Total Hours Invested x 2.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austin</td>
<td>$31.00</td>
<td>150</td>
<td>$11,625.00</td>
</tr>
<tr>
<td>Kerr</td>
<td>$31.00</td>
<td>150</td>
<td>$11,625.00</td>
</tr>
<tr>
<td>Total</td>
<td>-</td>
<td>300</td>
<td>$23,250.00</td>
</tr>
</tbody>
</table>

4.3 Total Cost

Table 6. The total cost for labor, parts, and the entire project.

<table>
<thead>
<tr>
<th>Category</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labor</td>
<td>$23,250.00</td>
</tr>
<tr>
<td>Parts</td>
<td>$269.39</td>
</tr>
<tr>
<td>Grand Total</td>
<td>$23,519.39</td>
</tr>
</tbody>
</table>
5.0 Conclusion

5.1 Accomplishments

At the end of the semester, we were able to present a fully functional RGB LED circuit which operated as expected in conjunction with our microcontroller. We were able to implement our intended design where the user would be able to activate when they wanted to turn on the RGB LEDs through a switch. Additionally, the user was able to alter the color of the RGB LED to their preferred color through the use of potentiometers. The motor driver circuit had some successes as well in that it was able to rotate upon enabling it. However, the incorporation of light sampling to determine how much it should rotate was left uncompleted.

5.2 Uncertainties

Some of the uncertainties we encountered dealt with circuit components. In the case of our switch-transistor, there was a significant discrepancy between our theoretical and measured value for the emitter terminal. The lower voltage could be due to the load (current mirrors) and imperfections within those components or imperfections within the switch-transistor itself. Several of our other debugging issues revolved around our microcontroller. On certain occasions, the ATmega328P would be unresponsive and required reprogramming. More specifically, the integration of the photoresistor with the microcontroller for light sampling presented uncertainties. The code was thoroughly looked over and modified slightly in an attempt to fix it. After numerous debugging attempts, the issue remained unsolved.

5.3 Ethical Considerations

This project follows the IEEE Code of Ethics by following these points:

1) To accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment
3) To be honest and realistic in stating claims or estimates based on available data
5) To improve the understanding of technology; its appropriate application, and potential consequences
7) To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others
9) To avoid injuring others, their property, reputation, or employment by false or malicious action
10) To assist colleagues and co-workers in their professional development and to support them in following this code of ethics [4]

By following these points, we hope to increase the quality of life of the user of our project. Additionally, taking and giving helpful criticism is an important part of not only this class, but a part of being an engineer. Another important part of being an engineer is to be truthful when presenting data found and progress made on one’s current project. This is important not only to one’s own code of ethics, but could also lead to an unsafe design that could harm others at a later time. This leads into one of the most important points, which is making the design safe for use. This includes cutting corners and cost in such a way that the end product may not be entirely safe. An example of this in our project is our motor driver circuit, which contains a current limiter. We added the current limiter to our design in the case that the DC motor stalls, meaning it would begin to draw more current than it is rated for. It only takes 0.1 A of current to kill, and our battery has the capability to supply 5.7 A. [1] Our battery also poses as a potential fire hazard, which is why we configured the charge controller in a way to offer safe charging of the battery.

5.4 Future Work

In the future, we would need to debug the microcontroller completely. An approach we would take would be completely rewriting the algorithm for the automation of the blinds which uses photoreistor as its input receiver. Once the microcontroller correctly took inputs from the photoreistor and correctly signaled the motor to rotate, we would then attach the photoreistor to the blind panel themselves. The motor would also be attached to the blind pulley system to adjust the blind angle. The charging circuit subsystem would also need to be worked on. That would require us to configure the circuit and test it to determine if it can safely charge our lithium-ion battery. As for an improvement to our RGB LED circuit, we would want to use a lower voltage battery with sufficient current to supply the load in order to minimize resistive losses.
References


## Appendix A  Requirements & Verifications Table

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Verifications</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Solar Panels: 10 Points</strong></td>
<td></td>
</tr>
<tr>
<td>1. Since our whole project is powered by an array of solar panels, it is vital that in full sun our 3 by 4 array is able to produce 18 V and 0.664 A in order to satisfy the 12.3 V battery float voltage. The voltage will decrease and the current will increase after going through the charge controller in order to charge the battery. A voltage not exceeding 13.05 V will result in the charge controller not operating.</td>
<td>1. We can test each individual solar panel in full sun and measure, using the multimeter, to check if it can produce the rated 6 V and 0.166 A. We will test three solar panels in series to verify the 18 V potential and 0.166 A current. We will also need to test that our array can produce our calculated values within 1-3%.</td>
</tr>
<tr>
<td><strong>Charge Controller: 10 Points</strong></td>
<td></td>
</tr>
<tr>
<td>1. The charge controller should be able to take the solar panels’ varying output, (maximum of 18 V and 0.664 A), and utilize maximum power point tracking to charge the li-ion battery. The battery float voltage and maximum charge current should be 12.3 V and 1.32 A, respectively.</td>
<td>1. This can be tested by setting up the charge control circuit and emulating the output of the solar panels, then measuring the output of the charge control circuit. We want the voltage of the charge controller to be 12.3 V ± 0.3 V, since that is the recommended and safe charging voltage for the battery.</td>
</tr>
<tr>
<td><strong>Li-ion Battery: 8 Points</strong></td>
<td></td>
</tr>
</tbody>
</table>
| 1. Supply voltage and current to the microcontroller, RGB LED array, and motor to run all three simultaneously. (Max of 11.1 V and 5.7 A)  
2. Hold charge while not in use.  
3. The battery must be able to fully charge over the course of a sunny day at the least. | 1. Measure voltage and current, using a multimeter, of battery at varying levels of charge in order to test if it is within 1-3% of its rated values of 11.1 V and 5.7 A.  
2. After fully charging the battery, we will measure the battery’s charge level over time, taking readings at determined intervals.  
3. We will imitate the power the battery would be receiving from the solar panel/charge controller combination |
and measure if the battery is fully charged after 9 hours, which is the least amount of daylight the solar panels have access to.

### Microcontroller: 7 Points

1. Be able to read the correct signals, from 0 to 5 V, from the user inputs and light data and also send the correct signals to the motor and the RGB LEDs.
2. Supply a regulated 5 V to the components that require it.
3. The software of the microcontroller is able to successfully activate the RGB LEDs and the electric motor.

1. This can be done with a testing circuit, where when a switch is flipped, the microcontroller should read it as on, and then turn an LED on by sending a signal. This way we can test both input and output of the microcontroller at once.
2. This can be tested by simply measuring the voltage across the microcontroller’s I/O pin and ground. It should read 5 V ± 0.25 V because it is simply acting as a trigger to the switch transistors.
3. The software can be tested by using a breadboard and simulating signals to the microcontroller, measuring the voltage output of specific pins.

### Electric Motor: 5 Points

1. The electric motor should be able to apply torque to turn the blinds itself, without going to the stall current of 0.4 A.
2. The H-bridge should allow the microcontroller to choose which direction to turn the motor, in order to allow the microcontroller to turn the blinds in either direction.
3. The H-bridge should be able to power the motor without reaching harmful temperatures over 85°C. At that point, the H-bridge precedes to thermal shutdown.

1. Either the force the motor can apply is measured with varying current levels, or we will attach the motor to the blinds first and physically test what current will slowly turn the blinds.
2. This can be tested by physically choosing the control inputs of the H-bridge and then either measuring the output or attaching the motor and observing how it spins.
3. While having the H-bridge power the motor at full capacity, measure the temperature of the ground pins, since those are act as the heatsinks for the H-bridge.

### RGB LED Circuit: 10 Points
1. We want the LEDs to all have approximately the same luminosity, at least to the human eye. This way the light will be distributed across the room evenly.
2. The LEDs should be able to produce different colors
3. Output a variable current, controlled by a potentiometer, to the parallel lines
4. Output a current of 0.6 A to parallel lines when potentiometer set at 0 Ω

1. Each color will be tested individually using the highest recommended current (20 mA per lead) in order to view the maximum amount of light the LED will display.
2. All colors will be tested simultaneously at various currents (60 mA current draw).
3. With the LED circuit set in place and voltage & current being supplied adjust potentiometer to some value. Measure the reference current in the BJT and the current running through all parallel lines.
4. Set the potentiometer to 0 Ω. Measure that 0.6 A runs through the BJT and all parallel lines.
5. Set the potentiometer to 1 kΩ. Measure that 8 mA runs through the BJT. The LEDs should not be lit.
Appendix B

Power Block Diagram

Solar Panels → Charge Controller → Li-ion Battery → Voltage Regulation → Transistor Current Mirror (BJT) → RGB LEDs

Voltage Regulation → Motor Driver → Electric Motor → Microcontroller
Appendix C  Signal Block Diagram

- Photoresistor
- Microcontroller
- User Inputs
- Motor Driver
- Transistor Current Mirror (BJT)
Appendix D  
Microcontroller Programming Flowchart

- Automate Motor Signal
  - LED Signal
    - Flash LEDs
      - Wait for 15 minutes while checking if the Automate Motor Signal has changed state
  - Turn Blinds Using Motor One Direction
    - Check if light is increasing or decreasing
    - Increasing: Continue turning
    - Decreasing: Stop turning and check other direction
  - Turn Blinds In Other Direction
    - Check if light is increasing or decreasing
    - Increasing: Continue turning
    - Decreasing: Stop turning, local max has been found
- High/True
- Low/False
Appendix E  LT3652 Charge Controller Circuit
Appendix F

SN754410 H Bridge Chip
Appendix G  LTC3619B Current Limiter Circuit
Appendix H  RGB LED Circuit Schematic