ECE 445

SENIOR DESIGN LABORATORY

FINAL REPORT

Automatic Bike Light

Team #56

PESANDI GUNASEKERA
(pesandi2@illinois.edu)
MAGDALENE NOFTZ
(noftz2@illinois.edu)
NATHANAEL SALAZAR
(nsala6@illinois.edu)

TA: Chihun Song

May 6, 2026

Abstract

As over a thousand Americans are killed every year in bicycle motor accidents [1], there remains a need for better safety measures to mitigate bicycle collisions with motor vehicles. This paper describes our senior design project in which our team designed an Automatic Bike Light System. Our bike includes a light sensor, LiDAR, accelerometer, and an indicator light to keep a bicyclist more visible and aware to improve their safety on the road.

Contents

1 Introduction **1**

2 Design **2**

2.1 Design Introduction 2
2.1.1 Power Subsystems 3
2.1.2 Sensing and Data Processing Subsystems 3
2.1.3 User-Interface Subsystems 4
2.1.4 Visual Aid 4
2.1.5 High-Level Requirements 5
2.2 Design Procedure 5
2.2.1 Power Subsystems 5
2.2.2 Sensing and Data Processing Subsystems 5
2.2.3 User-Interface Subsystems 7
2.3 Design Details 7
2.3.1 LED PCBs 7
2.3.2 Front and Rear Power PCBs 9
2.3.3 Control PCB 9

3 Design Verification **11**

4 Cost **13**

4.1 Parts 13
4.2 Labor 13

5 Conclusion **14**

5.1 Accomplishments 14
5.2 Uncertainties 14
5.3 Ethical Considerations 14
5.4 Future Work 14

References **15**

Appendix A Abbreviations **16**

Appendix B Requirements and Verification **17**

Appendix C Circuit Schematics **20**

1 Introduction

Several issues exist within the current bicycle light market. Bicycle light systems often lack a method to automatically adjust between day and night riding, indicate turns outside of hand signals, or shut off the lights if a user forgets to turn them off manually. Any of these factors present inconveniences and potentially create safety concerns for the bicyclist.

To improve the safety of bicyclists on the road, we created a bike light system that addresses these market issues. Our visibility lights (front and rear) switch between rapid flashing and constant illumination based on the ambient light detected by a light sensor. We have an indicator light that flashes faster depending on the distance of a car approaching from behind the bicyclist. This distance is measured by a LiDAR sensor mounted to the back of the bike. We also have left and right turn signal lights that the user can control with buttons on the front of the bike. The lights turn off if the bike is stationary, which is detected using an accelerometer. There is also an on/off button that turns the entire system on and off.

2 Design

2.1 Design Introduction

The Automatic Bike Light System has various features on each of the modules to maximize the safety of the user. The front headlight and rear light have two settings: strobing and constant light. A light sensor measures the amount of ambient light in the surrounding environment to determine if it is day or night. If the sensor determines that it is day, the lights are set to the strobing setting, while if it is night, they are set to the constant light setting. A LiDAR sensor on the back of the bike alerts the user if a car is approaching within 30 feet by setting an indicator light to flash at the front of the bike. The indicator light has two settings: a slower flashing setting and a faster flashing setting. If the car is 7 - 30 feet away, the light is set to the slower flashing setting, while if the car is 0 - 7 feet away, the light is set to the faster flashing setting. Additionally, two push-buttons at the front of the bike control the rear turn signal lights to let drivers behind them know the user is turning. A third pushbutton at the front of the bike turns the whole system on and off. Finally, if the system is on and the bike has not moved for over a minute, as determined by an accelerometer sensor, the lights also turn off to conserve energy. The block diagram showing the subsystems in the design is shown below in Figure 1.

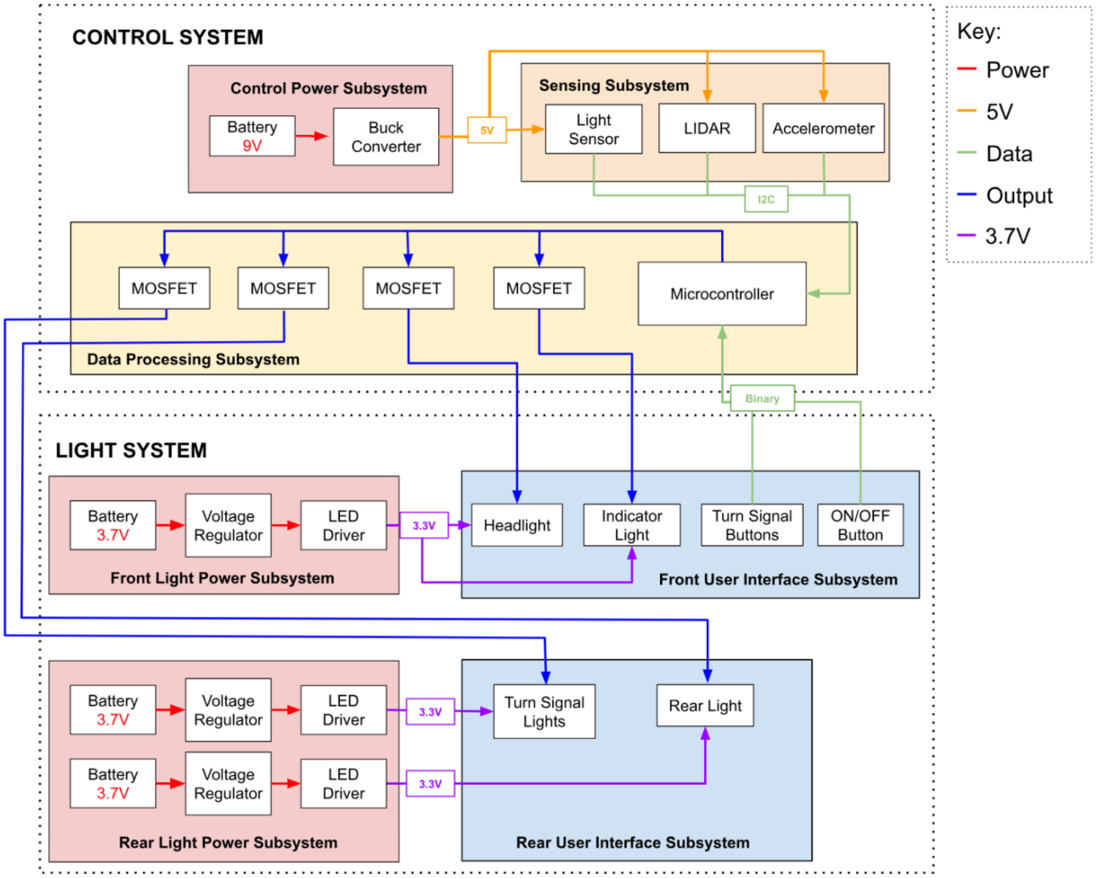

Figure 1: Block Diagram

2.1.1 Power Subsystems

Three power subsystems are used to power the bike light system. The Control Power Subsystem is powered by a 9 V battery and passed through a PWM controller chip that acts as both a buck converter and a voltage regulator to step down the voltage to a stable 5 V. The 5 V output is used to power the sensors and the microcontroller. The front and rear light power subsystems take input from a 3.7 V battery, which is then passed through a voltage regulator and LED driver resistor to produce a stable 3.3 V output. The 3.3 V output for each system is used to power the front, rear, and indicator lights.

Several changes were made to the power subsystems from the initial design. First, in the initial design, the Control Power Subsystem was used to power the sensors, microcontroller, front headlight, and indicator light with the 5 V output. But after it was determined that the lights only needed 3.3 V and would draw a substantial amount of current, this subsystem was split into the current Control Subsystem and Front Power Subsystem.

Second, in the initial design, the Control Power Subsystem contained a buck converter and a voltage regulator. However, upon testing our chosen buck converter, we discovered that it was not compatible with the rest of the circuit since its output voltage was negative. Therefore, the Control Power Subsystem was changed to contain only a PWM controller that was compatible with the rest of the circuit, which acted as both the buck converter and voltage regulator.

Third, the Rear Power Subsystem initially had only one 3.7 V battery powering the rear light as well as the left and right turn signals. However, after testing the physical circuit, it was determined that all three of the lights drew too much current for the singular 3.7 V battery. As such, an additional battery was added to the rear so that one battery could power the rear light while another one could power the turn signals.

2.1.2 Sensing and Data Processing Subsystems

The Sensing Subsystem contains the light sensor, accelerometer, and LiDAR sensors. The data output from each of the sensors is sent through I2C protocol to the microcontroller in the Data Processing Subsystem. Then, the microcontroller is programmed to turn each light to the correct setting based on the output data. The microcontroller sends the signal to turn each light on or off through a MOSFET so that the lights could each be powered by the 3.7 V batteries and not the 9 V battery. The only light not controlled by a MOSFET is the indicator light.

Only one change was made to the sensing and data processing subsystems from the initial design. In the initial design, the indicator light signal from the microcontroller was sent through a MOSFET. However, after testing, we found that the MOSFET interfered with the stability of the system, so the MOSFET was removed, the microcontroller output was connected directly to the light, and the indicator light was powered by the 5 V signal instead of the Front Power Subsystem.

2.1.3 User-Interface Subsystems

The two User-Interface Subsystems contain all of the lights and buttons. The lights remain off until the on/off button is pressed. While this button is pressed, the microcontroller controls the setting of the front, rear, and indicator lights based on the sensor outputs. The two turn signal buttons control each of the turn signal lights, and when one of these buttons is pressed, the microcontroller sets the corresponding turn signal light to flashing. The only change made to these subsystems from the initial design is that the indicator light is powered by the 5 V output, when initially it was powered by the Front Power Subsystem, as explained in the previous section.

2.1.4 Visual Aid

Figure 2, shown below, displays a photograph of our final project with labels pointing to the most important features.

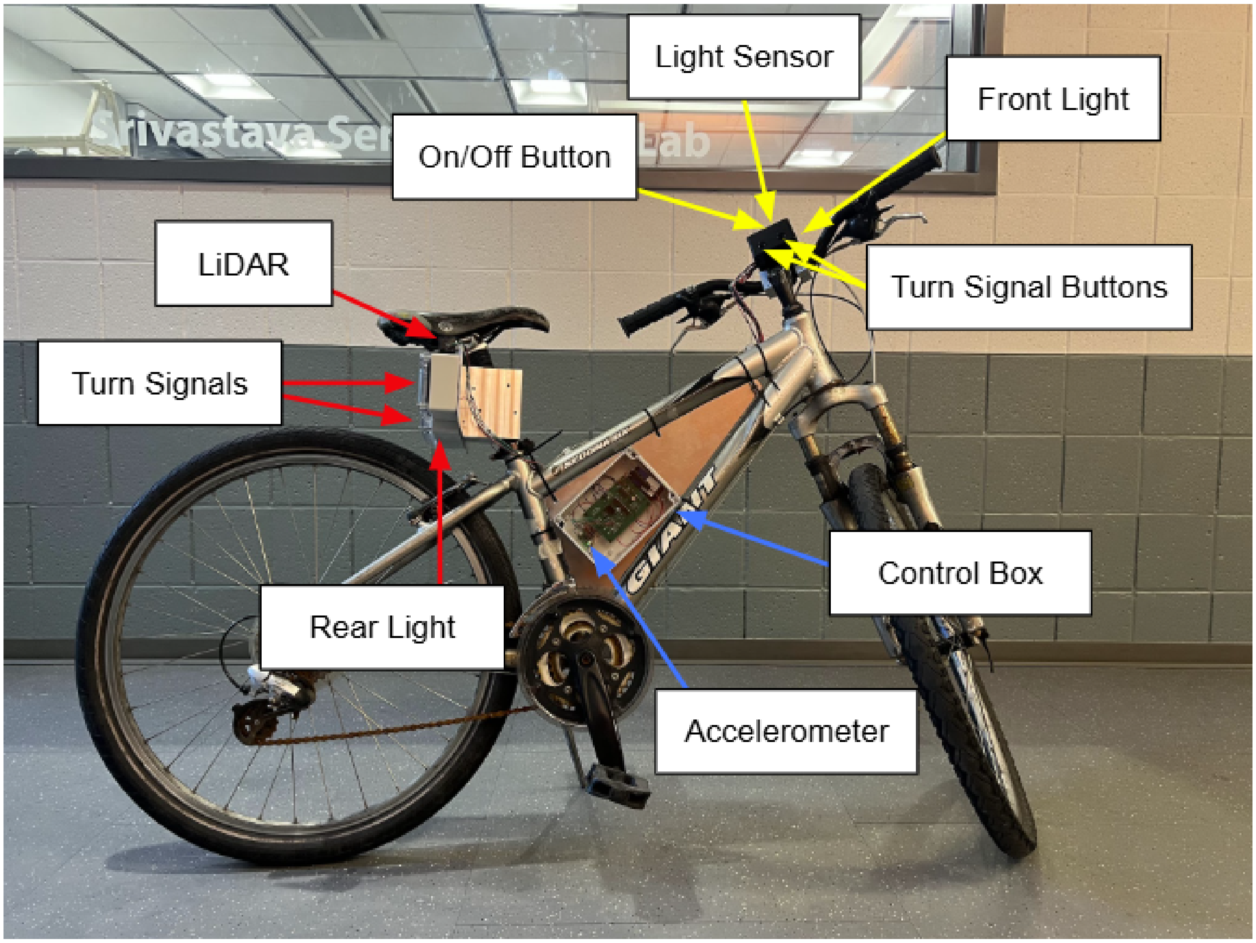

Figure 2: High-Level Diagram of Bike Light System

2.1.5 High-Level Requirements

For the Automatic Bike Light project to be considered a working solution to the problem statement described above, the following high-level requirements had to be met:

- Indicator light detects vehicle from a 30-foot distance
- Lights turn off after 1 minute of no movement
- Lights strobe/are constant depending on ambient light
- Turn signals work when pressed
- System turns off when on/off button is pressed

2.2 Design Procedure

2.2.1 Power Subsystems

One important aspect of our circuit design is voltage and current protection. Many of the components we used, such as the sensors, microcontroller, and LEDs, could be damaged during conditions of over-voltage or over-current. To protect the components from these conditions, we used voltage regulators and LED drivers.

The voltage regulator we used for the 3.7 V batteries is the LP2951 adjustable regulator. In the Front Light Power Subsystem and Rear Light Power Subsystem, the voltage regulator took the 3.7 V from the battery and outputted a steady 3.3 V, which protected the LEDs from sudden voltage spikes. The slight drop in voltage from the battery was not a problem for the overall operation of the circuit since the necessary forward voltage of the LEDs is around 3 V.

A separate voltage regulator for the 5 V output in the control power subsystem was not necessary since a TL494CN pulse width modulation (PWM) controller was configured as a buck converter and included a built-in voltage regulator. The PWM controller chip was used in place of a buck converter and voltage regulator for compactness and because it was available at no cost in the self-service shop.

The LED drivers we used for the LEDs were resistors. Since the LEDs are all low power (less than one watt), a simple resistor is enough to protect each LED from sudden current spikes. The value of the resistor depended on each LED, and we calculated each of the necessary resistor values using the voltage and current specifications of each LED, and the equation shown below:

$$R = \frac{V_s - V_f}{I_f} \tag{1}$$

Where V_s is the output of the voltage regulator, V_f is the forward voltage of the LED, and I_f is the current through the LED.

2.2.2 Sensing and Data Processing Subsystems

The sensing subsystem consists of the light sensor, accelerometer, and LiDAR. The light sensor we used to detect the amount of ambient light is the Adafruit SI1145 UV index/IR/Visible Sensor. The sensor is compatible with our ATMEGA328P-PU microcontroller, transmits data using I2C protocol (two-wire, short-distance, and low speed communication between a controller and its peripherals), and measures both artificial and natural light. The SI1145 sensor does not directly output the amount of ambient light (in units of luxes), but rather operates primarily by calculating the UV index. However, the sensor outputs 16-bit Analog-to-Digital conversion (ADC) counts [2]. The photodiodes on the sensor generate current when exposed to light, and the Analog-to-Digital converter converts it into a number based on the amount of light hitting the photodiodes. So, the higher the ADC count, the more ambient light. The ADC count from the SI1145 light sensor is used by the microcontroller to determine whether it is day or night, and what light level the bike headlight should be at. Testing was done to determine the threshold values for day and night and what ADC count ranges resulted in each bike light setting (strobing, brighter constant light, dimmer constant light).

The accelerometer we used to detect whether the bike is in motion or not is the Adafruit LIS3DH Triple-Axis Accelerometer. The sensor is compatible with our ATMEGA328P-PU microcontroller and could transmit data using both I2C and SPI protocol. We chose to use the I2C protocol due to the lower wire number requirements and for simplicity, so that all the sensors could connect to the same input on the microcontroller. The LIS3DH accelerometer operates by measuring the acceleration of the object it is mounted on. The sensor outputs the acceleration in the x, y, and z direction either as raw data or normalized data [3]. The normalized data are only important if calculations must be done with the results, and since for our project we only cared about whether there is a change in acceleration and not the actual values, we used the raw data. To determine whether the bike is stationary or in motion, the data was sent to the microcontroller, which calculated the change in acceleration between the current value and the previous value using the following equation:

$$\Delta = |x - x_{prev}| + |y - y_{prev}| + |z - z_{prev}| \tag{2}$$

This value was then compared to a threshold value, determined through testing of the accelerometer, that represents the lowest change in acceleration possible while in motion. If the delta value is less than this threshold, then the bike was determined to be stationary. If the bike remained stationary for at least one minute, then the microcontroller would turn off the system.

The proximity sensor we used to detect whether a car is approaching from behind the bike is the Garmin LIDAR Light V 4 LED. The LiDAR can detect an object within 30 feet with an accuracy of +/- 0.4 inches, requires 5 V for operation, and transmits data using I2C protocol [4]. The LIDAR sensor works by transmitting a laser beam, at a wavelength of 905nm so it is not visible to the human eye, and the beam reflects off of objects within range and returns to the sensor [5]. The sensor continuously sends the laser beam, receives return signals, and stores the strength of the return signals in its memory. If there is a peak in the return signal strength that emerges from above a threshold (determined by typical noise), then the sensor will determine that there is an object within range and use the information and the speed of light to calculate the distance of the object.

We mounted the LiDAR sensor on the back of the bike facing directly backwards. The sensor outputs a non-zero distance value if it determines there is an object within 30 feet behind the bike. This information is sent to the microcontroller which then turns on the indicator light at the front of the bike if a non-zero distance value is received.

The microcontroller we chose to use was the ATMEGA328P-PU chip. We chose this microcontroller because we were not using any SPI or wireless communication, and this chip mostly handles I2C protocol. The chip had low power consumption and could be programmed to handle the functionality of our project.

2.2.3 User-Interface Subsystems

In the Front User-Interface Subsystem, the front headlight of the bike is set to a strobing setting during the day and a constant light setting at night. A small rear-alert light on the Front Light Module is connected to the Rear Control Subsystem and turns on when the LiDAR detects an approaching object within 30 feet behind the bike. Turn signal buttons (one for right and one for left) on the back of the Front Light Module are used to indicate the direction the cyclist intends to turn and sends this information to the Rear Control Subsystem.

The front user interface subsystem consists of the front headlight, the rear-alert light, the turn-signal button, and the on/off button. The front headlight is powered by the Front Power Subsystem and is controlled by a MOSFET which receives its gate signal from the microcontroller. To be considered functional, the headlight must be set to a strobing setting during the day and a constant light brightness settings at night. The rear-alert light is a small indicator light on the Front Light Module that is powered and controlled by the microcontroller. To be considered functional, the rear-alert light must be on when an approaching object behind the bike is within 30 feet and must be off otherwise. The turn-signal buttons, one for left-turn and one for right-turn, are located on the back of the Front Light Module and send information on the direction that the cyclist is turning to the microcontroller. To be considered functional, when the left-turn button is pushed, the button must send the signal to the Front Control Subsystem that the cyclist is turning left. When the right-turn button is pushed, it must send the signal that the cyclist is turning left. If the button is pushed a second time, it must send the signal to the Front Control Subsystem that the turn signal should be reset.

2.3 Design Details

2.3.1 LED PCBs

Our design uses three different colors of LEDs: white for the front light, red for the rear light, and PC amber for the turn signal lights. The specifications for the LEDs that we chose are shown in Table 1:

Table 1: LED Specifications

	White LED	Red LED	Orange LED
Power Class	0.2 W	0.5 W	0.5 W
Test Temperature	$25°C$	$25°C$	$25°C$
Test Current	55 mA	140 mA	140mA
Typical Forward Voltage	2.67 V	2.25 V	2.96 V
Typical Flux	34.1 - 32.3 lm	27.2 lm	61 lm
Typical Efficiency	232 - 220 LPW	86 LPW	147 LPW
Maximum Current	240 mA	250 mA	24 0mA

All of the LEDs in our system are powered by 3.7 V batteries that are stepped down by a voltage regulator to a steady 3.3 V. Each LED was placed in series with a current-limiting resistor, which was determined using Equation 1. The calculations we used to determine the specific resistances and the corresponding lumen outputs are shown below. First the required resistor for each LED was found with the equation. The resistances were rounded to the nearest value, then the current and power recalculated. The actual power values were multiplied by the luminous flux ratings to determine the lumen output of each LED.

White LED:

$$R_{white} = \frac{3.3V - 2.67V}{0.055A} = 11.45\Omega$$

$$R_{white,approx} = 12\Omega$$

$$I_{actual} = \frac{3.3V - 2.67V}{12\Omega} = 0.0525A = 52.5mA$$

$$P_{actual} = (2.67V)(0.0525A) = 0.140W$$

$$Lumens = 0.140(232 - 220) = 32.5 - 30.8lm$$

$$Lumens = 32.5 - 30.8lm$$

Red LED:

$$R_{red} = \frac{3.3V - 2.25V}{0.14A} = 7.5\Omega$$

$$R_{red,approx} = 8\Omega$$

$$I_{actual} = \frac{3.3V - 2.25V}{8\Omega} = 0.131A$$

$$P_{actual} = (2.25V)(0.131A) = 0.295W$$

$$Lumens = 0.295(86) = 25.37lm$$

Orange LED:

$$R_{orange} = \frac{3.3V - 2.96V}{0.14A} = 2.43\Omega$$

$$R_{orange,approx} = 3\Omega$$

$$I_{actual} = \frac{3.3V - 2.96V}{3\Omega} = 0.113A$$

$$P_{actual} = (2.96V)(0.113A) = 0.335W$$

$$Lumens = 0.335(147) = 49.31lm$$

2.3.2 Front and Rear Power PCBs

An aspect of the design of this project that poses a risk to successful completion is the lifespan of the battery. The battery must be able to power all of the required circuit components and last for a reasonable amount of time. Below are the battery life times for the front (Battery 1), rear light (Battery 2), and turn signal (Battery 3) batteries. We chose to use a rechargeable 3.7 V battery because it provided more mAh allowing for a longer battery life and because the batteries were rechargeable for cost efficiency. Batteries 2 and 3 were initially a singular battery but due to the current draw required for the LEDs we separate the rear visibility light and the turn signal to be powered from two separate batteries.

Battery 1 and 2: 3.7 V battery power rating = (3.7 V) (3000 mAh) = 11.1 Wh

- Battery 1 (3.7 V):
  - Headlight: (P_{actual} = 0.140 W) (2 lights) = 0.28 W
  - Battery 1 total power: 0.28 W
  - Using a 11.1 Wh battery → minimum of 39.64 hours of power
- Battery 2 (3.7 V):
  - Red rear light: (P_{actual} = 0.295 W) (2 lights) = 0.59 W
  - Battery 2 total power = 0.59 W
  - Using a 11.1 Wh battery → minimum of 18.81 hours of power
- Battery 3 (3.7 V):
  - Turn signals: (0.335) (4 lights) = 1.34 W
  - Battery 2 total power = 1.34 W
  - Using a 11.1 Wh battery → minimum of 8.28 hours of power

2.3.3 Control PCB

On our control board, we chose to use a 9 V battery because most of our sensors and our microcontroller needed 5 V for operation. As with the Front and Rear Power PCBs, the 9 V battery must be able to power all of the required sensors and circuit components to support operation of the system without needing excessive replacement. The following calculations show the estimated lifetime of the battery we used in our control system.

Battery 4: 9 V battery power rating = 5 Wh

- Battery 4 (9 V):
  - LiDAR sensor power: (100 mA)(5 V) = 0.5 W
  - Light sensor power: ($100mA_{max}$)(5 V) = 0.5 W
  - Accelerometer: ($100mA_{max}$)(5V) = 0.5 W
  - Microcontroller: (11 mA)(5 V) = 0.055 W
  - Rear-alert light: (15 mA)(5 V) = 0.075 W
  - Battery 4 total power: 1.63 W
  - Using a 5 Wh battery$\rightarrow$ minimum of 3.067 hours of power

Originally the light sensor and LiDAR were designed to be powered by the front and rear batteries respectively, which extended the battery life of Battery 4. However, by the final project, the light sensor and LiDAR were changed to be powered by the 9 V battery.

3 Design Verification

The full requirements and verification for the Control Power, Sensing (light, accelerometer, and LiDAR), Data Processing, Front and Rear Power, and User-Interface Subsystems can be found in Appendix B: Tables 4-9.

The detailed requirements of determining the threshold values for each of our sensors is described in Table 6 of Appendix B. As shown below in Table 2, we tested our light sensor in different environments to get the raw ADC light counts and to help us determine the ADC count ranges for day and night. After testing, we determined that we would use the threshold of 270, with counts less 270 being night and counts greater than or equal to 270 being day.

Table 2: Light Sensor Data

Testing Environment	**ADC Count Range**
Inside/Dark	150-270
Cloudy Day	270-400
Sunny Day	400-1500

As shown in Table 3, we tested our accelerometer sensor both while the bike was stationary and while it was moving at a walking pace. The micro-controller was programmed to calculate the delta values, explained above in Section 2.2.2, which were used to determine the threshold values for the accelerometer. After testing, we determined that we would use the threshold value of one, with values under one indicating that the bike is stationary and values above one indicating that the bike is in motion.

Table 3: Accelermter Sensor Data

Testing Environment	**ADC Count Range**
Stationary	<1
Moving	2-10

As shown in Table 4, we tested our LiDAR using a sheet of aluminum foil that represents a car. We started as close to the bike as possible and ensured that the indicator light flashed at the appropriate frequency. Then we moved further back and ensured that the flash frequency matched our distance from the bike. After approximately 31.5 feet, as measured with a tape measure, the indicator light turned off.

Table 4: LiDAR Data

Indicator Light Flash Frequency	**Distance**
10 ms	0-7 ft
300 ms	7-31.5 ft
Off	>31.5 ft

The requirements and verifications tables in Appendix B (Tables 6, 9, and 10) details the testing of the power subsystems to ensure the correct voltage output, as well as the testing of the functionality for the lights and buttons in the system.

4 Cost

4.1 Parts

The cost of each component necessary to complete the Automatic Bike Light project is shown below in Table 5.

Table 5: Itemized list of Components and Costs

Part Name	**Part Number**	**Manufacturer**	**Unit Cost ($)**	**Quantity**	**Actual Cost ($)**
LiDAR Connector	GHR-06V-S	JST	$0.13	2	$0.26
LiDAR Terminal	SSHL-002T-P0.2	JST	$0.07	12	$0.86
Accelerometer Sensor	1528-1516-ND	Adafruit	$4.95	1	$4.95
Buck Regulator	MAX636ACPA	Maxim	$0.01	3	$0
Voltage Regulator	LP2951	Texas Instruments	$0.60	2	$0
Microcontroller	ATMEGA328P-PU	Microchip Technology	$2.89	2	$0
ISP Header	AVR-ISP-6	Adafruit	$0.85	3	$2.55
USB ISP Programmer	N/A	Unknown	$0 (borrowing)	1	$0
MOSFET	2SK3703-1E	Onsemi	$0.34	4	$0
9V Battery - Alkaline	EN22	Energizer	$3.04	2	$0
DC Female Barrel Jack	N/A	Unknown	$0	1	$3.29
9V Battery Clip	233	Keystone Electronics	$0.84	2	$1.68
3000mAh Battery	N/A	MakerFocus	$0	4	$29.99
On/off/turn button	IP66	DMWD	$2.80	4	$13.99
Front Lights (White LEDs)	JK2835AWT-P-B57 EB0000-N0000001	Cree LED	$0.10	20	$2.00
Rear Lights (Red LEDs)	MLESRD-A1-0000-000W01	Cree LED	$0.88	10	$8.80
Turn Signal Lights (Orange LEDs)	JE2835APA-N-0001 A0000-N0000001	Cree LED	$0.21	20	$4.20
Indicator Light	5102H5-5	VCC	$3.89	2	$7.78
Programmer reverse current protection	MBR0520	Micro Commercial Components	$0.23	5	$1.15
LiDAR	4441	Garmin	$59.95	1	$59.95
Machine Shop Supplies			$50		$50
Total			**$131.78**		**$191.45**

4.2 Labor

Labor cost:

$$Cost = (14 \text{ weeks}) \times (30 \text{ hours/week}) \times (\$42\text{/hour}) \times (2.5 \text{ overhead}) = \$44,100 \tag{3}$$

The 30 hours/week includes labor from the machine shop and team members.

5 Conclusion

5.1 Accomplishments

We accomplished everything we set out to do in our design document: turn signals, front and rear lights that would strobe/be constant based on ambient light, a method to turn off the lights when the bike was stationary, and a LiDAR detection system. The only places we fell short of our design was in our LiDAR range and our battery light. For our LiDAR, we used a different sensor than initially intended that had a smaller range. For our battery life, we put more components than initially anticipated on our 9 V battery, thus shortening our battery life from the intended 4.3 hours to only 3 hours. Aside from these shortcomings, we successfully learned how to design a technical project from start to finish. We also learned how to use KiCad to design a PCB and how to select the appropriate components for a project.

5.2 Uncertainties

Our LiDAR data reliability changes based on ambient light and the surface used to reflect the laser from the sensor. For our project we chose to use aluminum foil to represent a car, but an actual car would be less reflective and therefore have a shorter range than the 31.5 ft range obtained using aluminum foil.

5.3 Ethical Considerations

We must consider important safety concerns about possibly blinding drivers and pedestrians, as well as possible repercussions if certain features, such as the LiDAR and the turn signals, malfunction. We followed the IEEE code of ethics [6] in all aspects of our design to prevent any circuitry issues. Additionally, our lights do strobe and may cause issues for individuals with epilepsy. To mitigate the potential damage, the IEEE standard 1789-2015 for handling high-brightness LEDs to reduce the health risk to viewers was closely studied and followed [7].

5.4 Future Work

For future work beyond the scope of this semester, students could work on making this system wireless for a more compact design, weatherproofing the system, placing components more efficiently on the various batteries, and replacing the LiDAR sensor with a RADAR sensor to improve the range of vehicle detection.

References

[1] National Highway Traffic Safety Administration, *Bicycle Safety*, Accessed: 2026-05-05, 2018. [Online]. Available: https://www.nhtsa.gov/road-safety/bicycle-safety.

[2] Silicon Laboratories, *Si1145/46/47 Proximity/UV/Ambient Light Sensor*, Accessed: 2026-05-05, 2022. [Online]. Available: https://www.silabs.com/documents/public/data-sheets/Si1145-46-47.pdf.

[3] Adafruit Industries, *Adafruit LIS3DH Triple-Axis Accelerometer Breakout*, Accessed: 2026-05-05, 2024. [Online]. Available: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-lis3dh-triple-axis-accelerometer-breakout.pdf.

[4] Microchip Technology Inc., *ATmega48A/PA/88A/PA/168A/PA/328/P*, 2020.

[5] Garmin, *LIDAR-Lite V4 LED: Operation Manual and Technical Specifications*, Accessed: 2026-05-05, 2020. [Online]. Available: https://static.garmin.com/pumac/LIDAR-Lite%20LED%20v4%20Instructions_EN-US.pdf.

[6] IEEE, *IEEE Code of Ethics*, Accessed: 2026-05-05. [Online]. Available: https://ewh.ieee.org/cmte/substations/posted_documents/ieee_codeofethics.pdf.

[7] Institute of Electrical and Electronics Engineers, *IEEE Recommended Practices for Modulating Current in High-Brightness LEDs for Mitigating Health Risks to Viewers*, Accessed: 2026-05-05, 2015. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7118618&tag=1.

Appendix A Abbreviations

Unit or Term	Symbol or Abbreviation
Light detection and ranging	LiDAR
Pulse width modulation	PWM
Pulse width modulation	PWM
Light emitting diode	LED
Inter-integrated circuit	I2C
Metal-Oxide-Semiconductor Field-Effect Transistor	MOSFET
Ultriviolet	UV
Infrared	IR
Analog-to-Digital Conversion	ADC
Serial Peripheral Interface	SPI
Phosphor-Converted	PC
Printed Circuit Board	PCB
Institute of Electrical and Electronics Engineers	IEEE
Direct current	DC
Volt	V
Watt	W
Degrees Celsius	$^\circ C$
Milli-amp	mA
Amp	A
Lumen	lm
Lumens per watt	LPW
Ohm	Ω
Watt-hour	Wh
feet	ft
milli-seconds	ms

Appendix B Requirements and Verification

Table 6: Control Power Requirements and Verification

Requirements	**Verification**
The buck converter steps down the 9V battery to 5V +/- 3%	• Connect the DC power supply, set to 9V, to the input of the buck converter. Use a multimeter to check that the output of the buck converter is 5V. • Monitor the multimeter readings for a period of one minute, and check that the readings are all within +/- 3% of 5V. • Replace the DC power supply with a 9V battery and repeat the first two steps.

Table 7: Sensing Subsystem Requirements and Verification

Requirements	**Verification**
Light sensor correctly determines it is day when: ADC ≥ 270 And determines it is night when: ADC count 270	• Determine an appropriate threshold value (270) using the light sensor and arduino. • On the breadboard, connect the light sensor to the rest of the circuit. Observe the sensor ADC readings and the output light setting during day/night
Accelerometer determines if bike is moving if: $\Delta \geq 1$ And determines the bike is off if: Δ 1	• Calculate an appropriate Δthreshold value (1) using the accelerometer and arduino. • On the breadboard, connect the accelerometer to the rest of the circuit and mount on the bike and observe the threshold values
LiDAR sends a signal to the microcontroller if an object is approaching behind the bike within 30 ft.	• Confirm that the LIDAR reads the correct distance when using an arduino uno. • Move the object incrementally further away up to 30 feet and confirm that the LiDAR operates at a 30-foot distance.

Table 8: Data Processing Requirements and Verification

Requirements	**Verification**
Turn signal buttons send correct signal to microcontroller	• Press the left turn signal button to the “on” position. Use a multimeter to confirm that the input pin on the microcontroller is pulled high. • Press the left turn signal to the “off” position. Use a multimeter to confirm that the input pin on the microcontroller is pulled low. • Repeat the tests for the right turn signal button.
On/Off Button sends correct signal to the microcontroller	• Press the on/off button to the “on” position. Use a multimeter to confirm that the input pin on the microcontroller is pulled high. • Press the on/off button to the “off” position. Use a multimeter to confirm that the input pin on the microcontroller is pulled low.

Table 9: Front and Rear Power Requirements and Verification

Requirements	**Verification**
Output of voltage regulator is 3.3 +/- 3%	• Start with breadboarding components and use a multimeter to ensure 3.3V output. • Monitor the multimeter readings for a period of one minute, and check that the readings are all within +/- 3% of 3.3V. • Repeat tests with components soldered onto PCB.

Table 10: User Interface Requirements and Verification

Requirements	**Verification**
Rear approaching alert light lights up when LiDAR gives signal	• Hold aluminum foil next to LiDAR and ensure indicator light blinks as the person walks backwards. • As a person walks backwards an indicator light should flash slower. • There should be three modes of operation. Fast flashing, slow flashing, and off. • At approximately 30 feet light should turn off
Front head light and rear light correctly chooses between strobing/constant light depending on day/night	• Based on the signal sent from the microcontroller (measured through multimeter) the front head light and rear light strobes or has constant light.
Lights turn off after a minute of no acceleration	• When the system is on, wait for a minute with no motion and confirm that all lights turn off.
Turn-Signal buttons turn the turn signal lights on and off	• Press the left indicator button and confirm that the left turn-signal light turns on. Press the button again and determine that the light turns off. • Press the right indicator button and confirm that the right turn-signal button turns on. Press the button again and d1etermine that the light turns off.

Appendix C Circuit Schematics

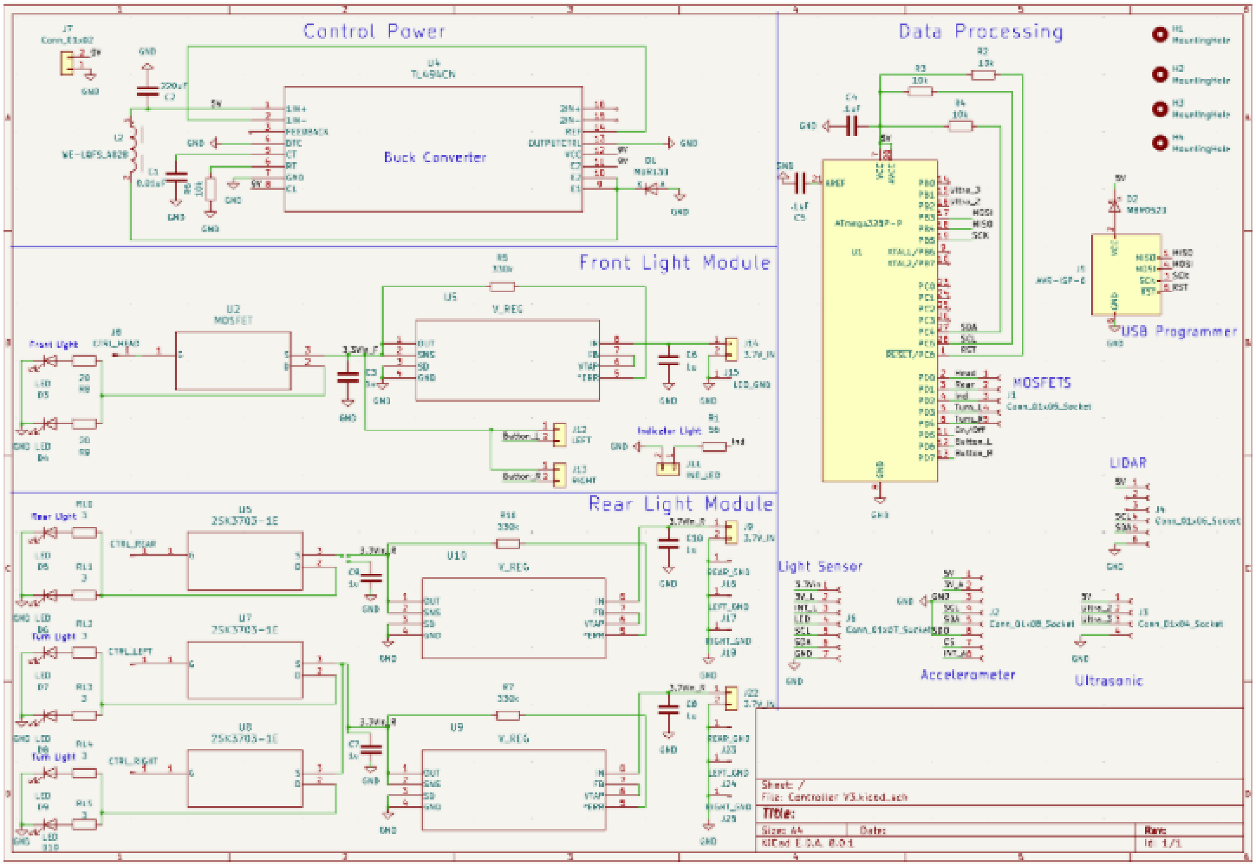

Figure 3: Overall KiCAD Schematic

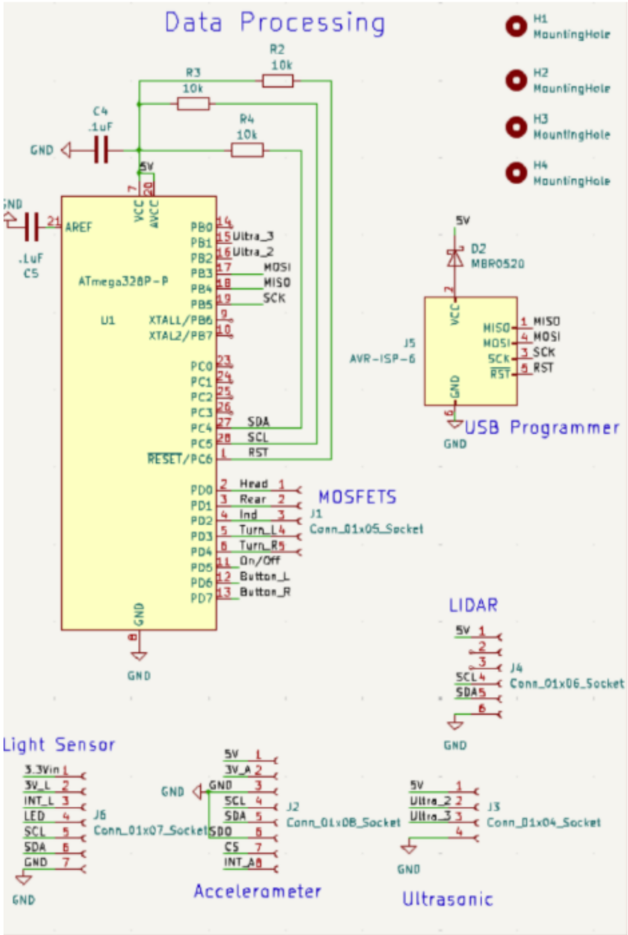

Figure 4: KiCad Schematic for Data Processing PCB

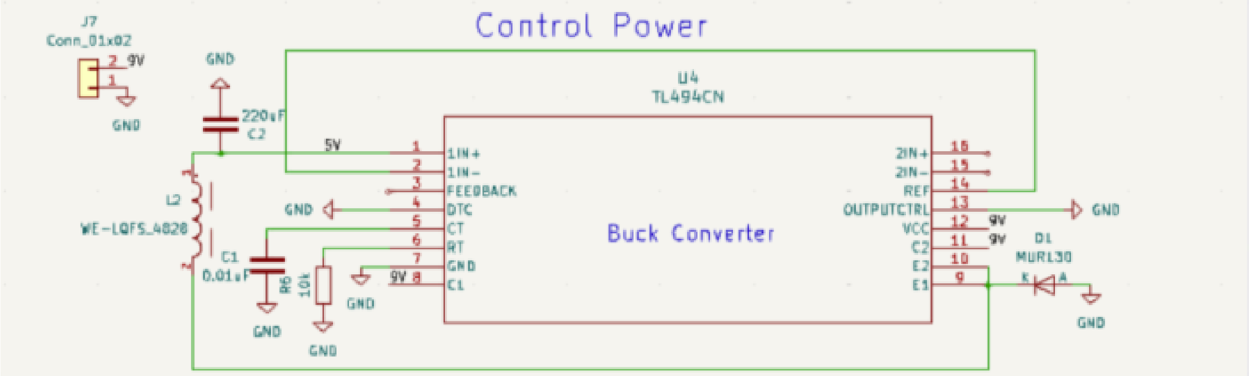

Figure 5: KiCad Schematic for Control Power Subsystem

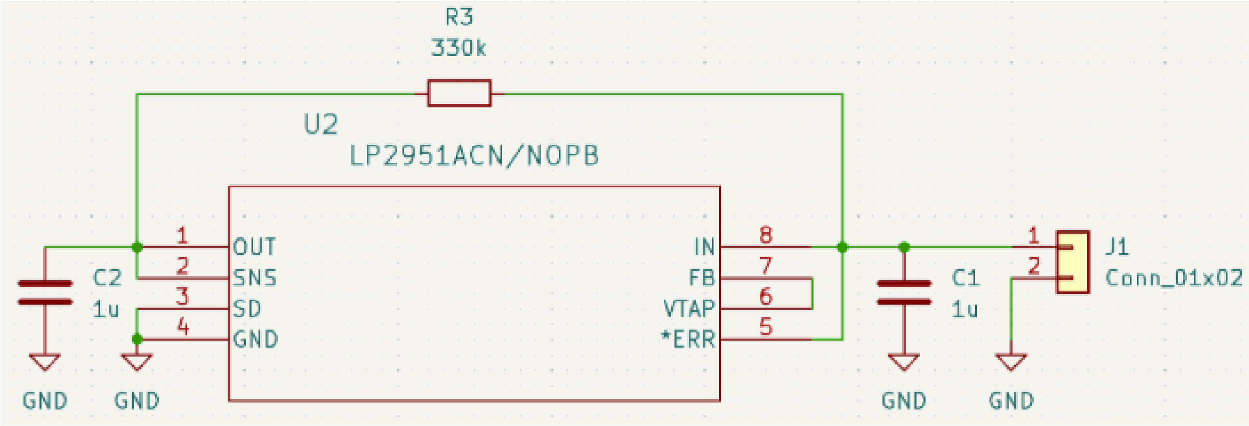

Figure 6: KiCad Schematic for Front/Rear Light PCB

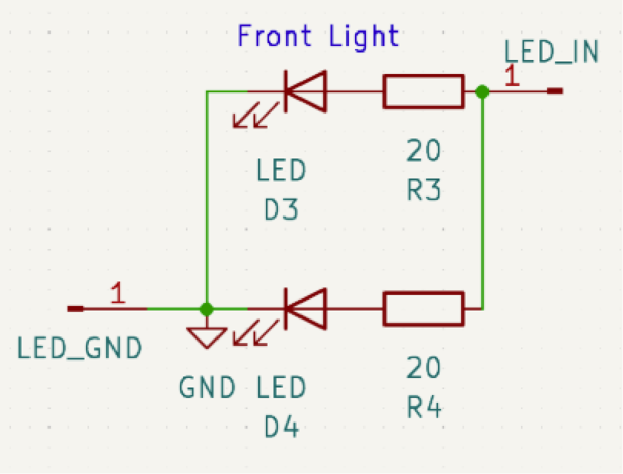

Figure 7: KiCad Schematic for Front Light LED PCB

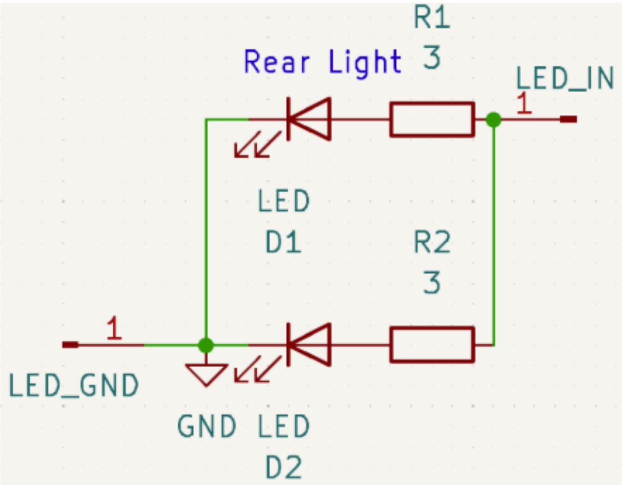

Figure 8: KiCad Schematic for Rear Light LED PCB

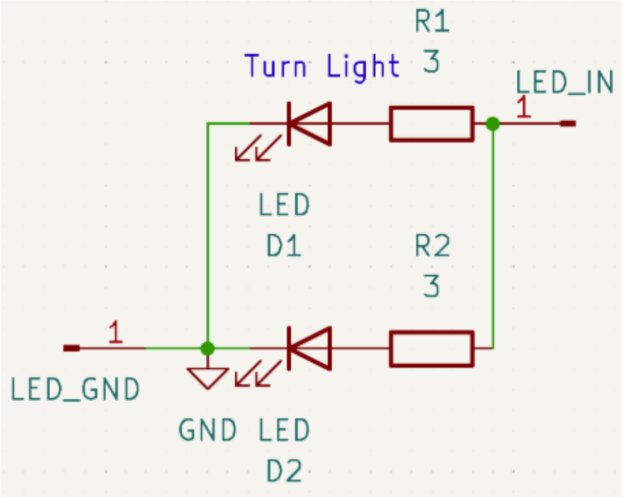

Figure 9: KiCad Schematic for Turn Light LED PCB