DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Smart Crosswalk Dynamic Lighting System

Design Document

Submitted to:

Randolph F. Follett, Ph.D., P.E.
ECE 4512 (or 4522): Senior Design I (or II)
Department of Electrical and Computer Engineering
413 Hardy Road, Box 9571
Mississippi State University
Mississippi State, Mississippi 39762

SCDLS

walk smarter.

Prepared by:

Emily Dance (ed307)
George Hilliard (gh403)
Preston Stinson (pcs90)
Lee Sargent (lhs51)
Hagan Walker (hdw65)
Faculty Advisor: Dr. Robert Reese
Department of Electrical and Computer Engineering
Mississippi State University
413 Hardy Road, Box 9571
Mississippi State, Mississippi 39762
Tel: 662-325-3154, Fax: 662-325-3670

Acronyms

ASTM American Society for Testing and Materials

CAVS Center for Advanced Vehicular Systems

CDC Center for Disease Control and Prevention

CID Current Interrupt Device

CPU Central Processing Unit

DUT Device Under Test

GPIO General Purpose Input/Output

I^2C Inter-Integrated Circuit

IC Integrated Circuit

IP Ingress Protection

IR Infrared

LED Light-Emitting Diode

LEDs Light-Emitting Diodes

MPPT Maximum Power-Point Tracking

PIR Passive Infrared

QFN Quad-Flat No-Leads

RGB Red, Green, and Blue

RTOS Real-Time Operating System

SCDLS Smart Crosswalk Dynamic Lighting System

SSOP Shrink Small Outline Package

Executive Summary

While in crosswalks, pedestrians are exposed to an increased risk of being involved in life threatening collisions with motor vehicles. Despite the numerous deaths associated with using crosswalks each year, they have remained relatively unchanged for decades. In order to help combat collisions between pedestrians and motor vehicles in crosswalks, the Smart Crosswalk Dynamic Lighting System (SCDLS) was developed. SCDLS consists of modules attached to a crosswalk, as shown in Figure 0.1. These modules illuminate the interior and exterior of the crosswalk when a pedestrian enters the crosswalk, thus dynamically alerting drivers to the presence of the pedestrian. They also collect vehicle traffic metrics for analysis by the systems owner.

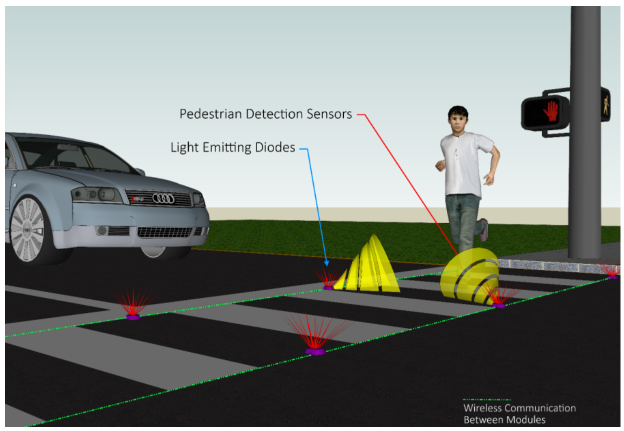

Figure 0.1: SCDLS System Overview

The main constraints in the design of SCDLS were ease-of-use and power consumption. The modules are able to be mounted to the surface of a road without the need for modifications to the road, aside from drilling mounting holes. The modules are also powered by only solar energy for up to five years. Since the modules will be installed on a roadway, they can withstand the compressive load of an automobile while remaining waterproof. Lastly, the module communicates wirelessly between modules at a miniumum distance of 10 meters.

Each module consists of a microcontroller, sensors, a battery, a solar panel, wireless modules, and Light-Emitting Diodes (LEDs) encased in an aluminum housing. A relatively large solar panel was chosen in order to sustain the modules for a minimum of five years. The system makes use of passive infrared sensors to detect pedestrians. The sensors were chosen for their accuracy and low power consumption. A magnetometer is used to detect vehicles so that municipalities can analyze traffic patterns. Two wireless radios provide Wi-Fi (to upload traffic statistics to an Internet server) and low-power wireless communication (to allow individual nodes to communicate). Finally, multicolor LEDs were chosen for use in the modules because they allow the devices to display multicolor patterns, improving their ability to garner drivers' attention.

SCDLS improves on other similar products by making installation and maintenance simple while also adding data gathering capabilities that no other products of this size offer. Future iterations of SCDLS will reduce the modules footprint and power consumption. Widespread adoption of SCDLS should reduce the number of collisions between motor vehicles and pedestrians in crosswalks, while also giving municipalities improved information about road usage.

Contents

1 Problem

1.1 Historical Introduction

The crosswalk is one of the most commonplace structures on city streets and has been for centuries. Paved roadways have been in existence since 4000 B.C.E, and crosswalks followed in 79 C.E. [1–3]. In 1868, the first illuminated crosswalk symbol was designed, but it required constant manual operation of a gas-lamp [4]. Even though crosswalk lighting technology has progressed since then, the first pedestrian fatality due to an automobile accident occurred in 1896 as a woman crossed a London street, and the number of crosswalk-related fatalities has increased every year [5–7]. More than 4000 pedestrians were killed in traffic accidents every year between 2003 and 2012, most of whom were hit at night [6].

Furthermore, most current traffic laws do not have a firm stance when it comes to the right of way of pedestrians and motorists in crosswalks [8]. Laws that regulate crosswalks without traffic control devices, called "uncontrolled crosswalks," are few, especially considering how many people use them daily. Shockingly, only "[n]ine states and the District of Columbia require motorists to stop when approaching a pedestrian in an uncontrolled crosswalk" [8]. Other states require motorists to stop only when the pedestrian is in the motorists same lane, while some states only require the motorist to yield [8]. Even though the Federal Highway Administration is in the process of implementing more programs to bring awareness to pedestrian friendly walking environments, uncontrolled crosswalks still present a danger of accidents to many people [9].

The need for dynamic safety systems in crosswalks is clear. Motorists need to be aware of pedestrians, and research has shown that pedestrians and motorists are more alert and observant while using a dynamically flashing crosswalk than while using a traditional crosswalk [10]. This increased alertness due to adding dynamic lighting is likely to reduce the number of pedestrian-related injuries and fatalities that occur in crosswalks.

1.2 Market and Competitive Product Analysis

SCDLS is the only smart crosswalk system available that requires no direct pedestrian interaction and combines both a smart crosswalk and a traffic monitoring system into one unit. Most commercially available crosswalk illumination systems are either always on, push button activated, or require the installation of obtrusive posts around the crosswalk to sense pedestrians and activate the lighting system. Furthermore, the vast majority of these other systems require significant alterations to the road during installation, such as running wires under the road from an external distribution box to the system. These alterations, though effective, can be very costly and cause substantial disruption to the normal use of the road.

In addition to the high cost of the installation, most of these systems cost significant amounts of money to even obtain. For example, a company named LightGuard Systems offers a smart crosswalk unit costing $19,885 for a two-lane pedestrian system that blinks for a pre-set amount of time. Their system requires the pedestrian to push a button to activate the system, which is a step that can be easily forgotten, making the device marginally more useful than an uncontrolled crosswalk [11]. Finally, few systems offer smart capabilities, such as data gathering. This capability is instead often seen as a separate product altogether. One such popular traffic monitoring system often used by municipalities is the MetroCount 5600, which costs $1307. SCDLS has all of the functionality of the MetroCount system, but wirelessly transmits traffic information without the need for onsite personnel [12, 13]. Finally, the MetroCount system and other competition products require unsightly external hardware that must be locked to nearby trees and acts as a tripping hazard.

The primary consumer of lighted crosswalks and traffic monitoring systems are municipalities and road maintenance companies. As such, SCDLS will be marketed towards the same clientele. Given the reduced cost and ease of installation of SCDLS compared to systems already available on the market, municipalities may be interested in implementing SCDLS in a small number of crosswalks as a case study to determine if they would be interested in wide scale adoption.

1.3 Concise Problem Statement

SCDLS seeks to prevent collisions between vehicles and pedestrians in crosswalks by lighting crosswalks in a way that alerts both motorists and pedestrians automatically. This system is an improvement over the competitors solutions mentioned previously because of its lower cost, relative ease of installation, and traffic statistics collection capabilities. The systems various sensors enable it to detect pedestrians in the crosswalk and alert pedestrians and vehicles of potential collisions. The units themselves are easy to install, requiring no destructive modifications to the road.

SCDLS is a wirelessly networked system of modules that are mounted across the road on the outer sides of the crosswalk. The modules are solar powered, eliminating the need to cut into the road to route power to them. The modules function as a single system that can detect pedestrians and alert drivers. They can also effectively measure and record both pedestrian and vehicle traffic usage for analysis by administrators.

A typical SCDLS module consists of the following: a solar panel for power harvesting, a lithium ion battery to store the power produced by the solar panel, a 2.4 GHz wireless radio for mesh networking between modules, sensors to determine if pedestrians are actively using the crosswalk, and LEDs to alert pedestrians and drivers of hazards using varying blink patterns. These components allow the devices to effectively detect pedestrians and vehicles and provide the information to other units on the crosswalk as well as to transportation authorities. In short, SCDLS is a dynamic road information mesh network that will help prevent crosswalk collisions and, ultimately, save lives.

1.4 Implications of Success

The intended result of implementing SCDLS in crosswalks is to reduce the number of pedestrian fatalities and injuries resulting from motor vehicle collisions. The benefits include the prevention of death or injury due to reduced collisions, as well as gathering data for use by municipalities. Reductions in pedestrian fatalities and injuries will cut back on the annual $99 billion spent on automotive crashes, as indicated in a 2010 Center for Disease Control and Prevention (CDC) study [14]. Along with economic advantages, the use of the SCDLS will improve driver engagement with the road, which could in turn reduce the number of crashes resulting from distracted driving.

In addition, since SCDLS can gather and transmit traffic analytics through the use of a magnetometer and a connection to a wireless network, municipalities can use this functionality to improve their road infrastructure. The aggregate data from all intersections in a city could help analyze pedestrian and vehicle traffic, reducing congestion and optimizing traffic routing. According to the traffic analysis company INRIX, traffic congestion has an annual economic cost of upward of $120 billion, which is “expected to increase ... to $186 billion by 2030” [15]. SCDLS could help cities reduce these costs.

Large scale adoption of SCDLS could also have effects on the market for crosswalk and pedestrian safety devices. Competing crosswalk lighting devices may begin to include some of the smart features used by SCDLS, such as pedestrian sensors and traffic data gathering capabilities. Additionally, the design of SCDLS modules allows for the possibility for these to be adapted for other applications to warn drivers of road hazards. However, the primary result of large scale adoption of SCDLS will still be the protection of human life.

2 Design Requirements/Constraints

In order to protect pedestrian lives, our team has developed SCDLS. By using Light-Emitting Diode (LED) lighting, this system could prevent pedestrian fatalities due to motorist/pedestrian collisions that occur in crosswalks. SCDLS consists of multiple modules installed on a crosswalk that will light a path for pedestrians while alerting motorists of the pedestrian by using flashing lights. There are two different types of modules used in SCDLS: nodes and hubs. The nodes are the basic makeup of the system and consist of a battery, a wireless module, a microcontroller, a solar panel, LED driver, and LEDs. Hubs are nodes with the addition of pedestrian sensors and Wi-Fi modules and are installed along the edges of the crosswalk. The hubs will communicate with the nodes and other nodes to light the nodes and record traffic data. The hubs will have a minimum range requirement for detection of pedestrians and vehicles. Based on the limits of available sensor technologies, SCDLS is intended to be used on crosswalks consisting of two or more lanes. In designing, developing, and producing SCDLS, there are technical and practical design constraints that both give SCDLS functionality and consumer acceptability.

2.1 Technical Design Constraints

The technical constraints outlined in Table 2.1 are important for the reliable and robust function of the device.

2.1.1 Power Sustainability

Power sustainability is important for the function of the modules because the expected life of a module is five years. If the battery voltage or state of charge drops too low, the battery could be damaged and the product will be unable to function properly. Since the team cannot wait five years to prove the robustness of this system, the power sustainability will be modeled based on initial data gathered from the system and component sizes will be adjusted accordingly. In order to properly model this system, data will be gathered about the cumulative energy input to the

Table 2.1: Technical Design Constraints

Name	Description
Power Sustainability	The solar panel should power the battery and other circuitry for at least 5 years.
Ingress Protection	The hubs and nodes must meet the solid and liquid ingress protection as defined by IP-68.
Compressive Strength	The hubs and nodes must be able to withstand a 6000lb compressive load as defined by ASTM-D4280-12.
Transmission Distance	The hubs must be able to communicate with the other hubs/modules at a distance of 10 meters.
Software Functionality	The system must be able to upload traffic metrics to a server when the hubs are in range of a Wi-Fi network.
Pedestrian Detection	The hubs must be able to detect a pedestrian at a distance of 16 feet or less.

battery from the solar panel and the cumulative power output from the battery. Considering the unpredictable nature of solar power generation, data must be gathered that is representative of a typical years sunlight. This data will be used in coordination with battery state-of-charge data and a worst case scenario of pedestrian traffic to determine if energy stored within the battery will ever fall below the low voltage shutoff point within five years. In order to account for the varying levels of sunlight in the areas in which this product could possibly be used, a safety factor will be added that corresponds to predicted sun exposure and cloud cover for areas that SCDLS would be installed.

2.1.2 Ingress Protection

Due to the varying conditions that a can affect a modern road, SCDLS will need to be protected against adverse environmental conditions. In the event that SCDLS is installed in a dusty area or on a road that has poor drainage the system needs to be robust. The Ingress Protection (IP) level selected for this project is IP-68. This level of protection is defined by the manufacturer, and in this case indicates that the modules are completely dust tight and can be immersed in up to 1 meter of water for 48 hours.

2.1.3 Compressive Strength

SCDLS will be installed on roads that consistently see automotive traffic, so they need to be designed to withstand the impact of an automobiles tire. This product will be designed in accordance with the American Society for Testing and Materials (ASTM) standard for Standard Specification for Extended Life Type, Non-plowable, Raised Retro-reflective Pavement Markers, D4280-12. This standard outlines tests that can be performed to validate the strength and anticipated durability of devices applied to road surfaces. This standard is generally intended for purely reflective road markings, but SCDLS is identical in application and loads faced. This standard requires placing the Device Under Test (DUT) in a compression testing apparatus that consists of 13 mm (0.5 in) thick steel plates that are larger than the DUT on the top and bottom and applying a 6000 lb load at a rate of 0.1 inches per minute. Before performing the test, SCDLS modules will be conditioned at 23.0 °C for 4 hours prior to conducting the test. The test will be considered successful if the unit does not break or deform more than 3.3 mm as defined by D4280-12 [16]. This testing will be performed at the Center for Advanced Vehicular Systems (CAVS) or at Future Labs, LLC.

2.1.4 Transmission Distance

For acceptable performance of the crosswalk the modules need to be able to communicate at a distance of at least 10 meters, allowing all nodes to communicate directly with the main hub. This is significant because a two-lane road can be up to 24 feet wide [17]. This gives our team a safety margin of 9 feet in the event of adverse weather conditions or interference due to other sources. This will be tested utilizing the wireless modules and a metal enclosure similar to one that will be used in the final product.

2.1.5 Software Functionality

For the product to be successful it must be able to upload encrypted traffic metrics when a trusted Wi-Fi network is in range. This will be supported by software by having a systematic method for connecting to a Wi-Fi network, finding a server, and uploading traffic statistics. The system would connect to a trusted Wi-Fi network on a scheduled basis to avoid depleting the battery.

2.2 Practical Design Constraints

The following design constraints are less technical in nature than the ones given previously and are outlined in Table 2.2. These constraints focus on the systems market appeal, overall safety, and cost. Therefore, they are equally critical to the success of the SCDLS system.

2.2.1 Cost (Economic)

The SCDLS system is designed to compete in a market occupied by other traffic safety and analysis systems, several of which are detailed in the Problem Statement section above. Therefore, the system must be substantially less expensive, in terms of both its installation and upkeep costs, and of its purchase price.

Many competing products require the installer to modify the road by cutting trenches for power cables or installing sensor towers. SCDLS should require no such modifications. The only installation the modules should require is affixing them to the road with bolts, in addition to a one-time software configuration. This constraint should ensure that SCDLS is easy to deploy in nearly all crosswalks.

Table 2.2: Practical Design Constraints

Type	Name	Description
Economic	Cost	The expected retail price for this system is $3000. This is based on a projected parts cost of $1000 (for a two-lane system). The system must be able to be installed without modifying existing roads.
Sustainability	Reliability	The system should be affixed to the roadway in a manner where it will not become detached under normal use cases.
Manufacturability	Case Material	The case must be made out of a material that is easy to machine and cost effective to use.
Health and Safety	Safety	The device must be designed to prevent tripping of pedestrians. Additionally, the devices must be designed to prevent the likelihood of automotive tire damage.
Sustainability	Maintenance	If a module is added or replaced, new modules must be able to be integrated into the existing device network, requiring no changes to the existing nodes.

2.2.2 Reliability (Sustainability)

The SCDLS system is designed as a long-term, low-maintenance solution for crosswalk lighting. To that end, the system should last a reasonable amount of time before requiring replacement. Because the system is highly modular, failure of one component should affect the operation of other components as little as possible. For example, if one of the hubs fails, the other hubs must continue to detect pedestrians to the best of their ability.

Individual modules should also be designed in such a way that they will last several years on the road. There are limiting factors to the devices longevity: the batteries in each module will gradually lose capacity, and abuse from vehicles will wear out the waterproofing and physical case. Therefore, considering the relatively low cost of the SCDLS system and the low maintenance costs, individual modules should have a lifespan of at least five years and can be replaced on an as-needed basis after that date.

2.2.3 Case Material (Manufacturability)

Because of the mechanical restrictions outlined in the Technical Design Constraints section, the case must be manufactured out of a sturdy material that is capable of withstanding years of abuse by automobiles and weather. In addition, the material must be affordable enough to meet the target retail price given above. Because of the nature of its application, the SCDLS system must operate for extended periods of time in potentially wet environments. Although the case will not be affected by water, the internal components must remain sealed from moisture. Therefore, the open edges on the modules (for example, where the solar panel meets the case) will also need to be sealed. This seal must remain intact throughout the expected life of the product.

2.2.4 Safety (Health and Safety)

Because the system will be installed in an area with high foot traffic, the product must not be a tripping hazard. Pedestrians, especially distracted ones, will not be paying much attention to the edges of the crosswalk. The modules must be sized and shaped in a way that minimizes the tripping hazard the modules pose to pedestrians.

The modules will also be subjected to vehicle traffic. Any obstruction on the road, including SCDLS modules, could potentially puncture vehicle tires. Because drivers cannot easily steer around the modules, the modules must also be designed to minimize the danger they pose to vehicles tires.

2.2.5 Maintenance (Sustainability)

One of the SCDLS systems goals is to minimize the expenses associated with making crosswalks safer. To this end, the system should not require active maintenance to modules that have been installed. Furthermore, the system must be highly modular so that the failure of one device should not require the replacement of the other devices. Instead, the replacement device should be able to begin communicating with the other modules. Therefore, in the event that a module fails, a faulty device should be able to be replaced without having to replace or modify the other working modules.

3 Approach

SCDLS is designed to aid pedestrians who are using crosswalks by actively lighting the crosswalk and making it visible to approaching vehicles. By dynamically changing the status of the crosswalk, this system has the capability to save numerous lives by gaining the attention of approaching motorists with the use of LEDs.

3.1 System Overview

SCDLS consists of several subsystems in order to actively track pedestrians as they enter and exit the crosswalk and dynamically trigger the LED lighting to alert oncoming traffic. The team has gone to great lengths to keep power consumption to a minimum and ensure that the system as a whole will stay functional for an average of 5 years.

Figure 3.1 shows an overview of the various subsystems that comprise a SCDLS module. The modules are powered by solar power and are intelligently controlled via an on-board microcontroller in order to further conserve power.

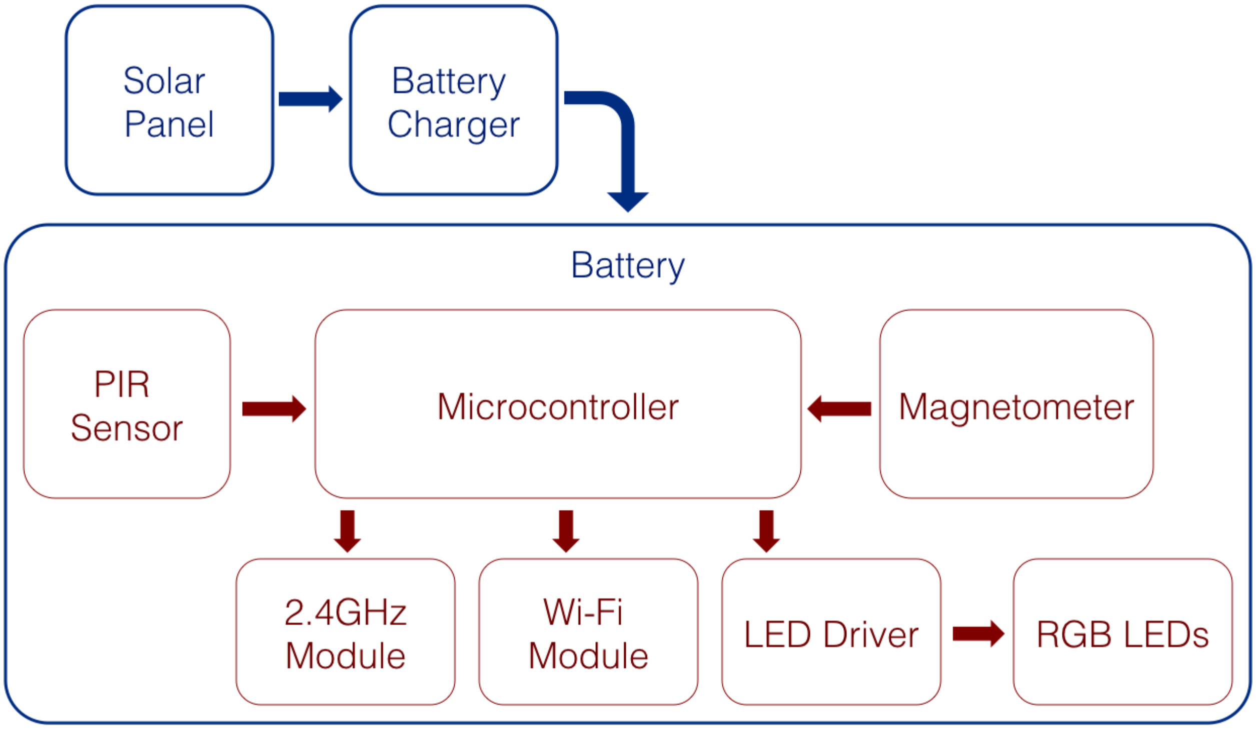

Figure 3.1: SCDLS System Overview

3.2 Hardware

3.2.1 Microcontroller

The microcontroller in each SCDLS device is required to have several features. First, it needs to provide sufficient computational power to handle the load generated by events from both local and remote sensors. Second, it needs to consume very little power in order to remain in the device's power budget. Third, the device needs to be able to drive the numerous other components. Finally, the microcontroller must be low cost.

For these reasons, we are using the Atmel ATmega 328P microcontroller. This device has very low power consumption and has a large library of software available for it, including the Arduino stack. It is also fairly inexpensive, while providing sufficient processing power.

3.2.2 Pedestrian Detection

One of the major advantages of SCDLS over competitors' offerings is the ability to detect pedestrians accurately without user interaction. In order to achieve this, the team considered a number of various sensors before choosing Passive Infrared (PIR) technology as the sensor type. Other considerations included ultrasonic sensors and Infrared (IR) distance sensors. There were many considerations associated with each sensor type. Before expanding on the background theory of each sensor, Table 3.1 gives an overview of the major characteristics of each of the sensor types previously mentioned.

Table 3.1: Considerations of Different Sensor Types

Technology	**Maximum Distance**	**Power Consumption**	**Weather Considerations**	**Field of View**
PIR	16 feet	30 µA	Easily weatherproofed	22° conical
Ultrasonic	6.5 feet	15 mA	Easily weatherproofed	10° conical
IR	6.2 feet	40 mA	Susceptible to Humidity	2° conical

Based on the information above, the team decided to eliminate IR sensors from further consideration based on their extremely small field of view. The small field of view would mean that an array of IR sensors would be needed in order to accurately detect when a pedestrian enters or exits the crosswalk. Also, IR sensors do not fare well in standard weather conditions. The PIR and ultrasonic sensors accomplish similar tasks in different ways. To start, PIR sensors fundamentally work by detecting changes in amounts of infrared radiation produced by objects. Usually the module consists of a comparator and a Fresnel lens. The infrared radiation values change significantly as individuals move in front of the module. Through the use of the Fresnel lens, these changes can be compared to the sensor's previous value to see if movement has been detected.

In comparison, ultrasonic sensors work by detecting sound waves. The module we were considering was unique in that it basically acts as both a speaker and a microphone. The sensor first emits a precise high frequency sound. Then, the sensor listens for a response back to the emitted sound and uses the time for the response to occur and determine the distance from the object.

Both modules had the following pros and cons. The ultrasonic sensors could give us distance data. However, the data was inherently noisy and would need to be filtered using fairly advanced techniques in order to obtain usable data. In comparison, the PIRs were one-third of the size but only recorded if an object had passed by. The team chose to use PIR sensors because they are more accurate than ultrasonic sensors and produce data that does not require as much filtering. Also, the detection threshold for the PIR sensors is greater than the ultrasonic sensors and, thus, false positives would be reduced. Finally, the PIR sensors are more energy efficient than the ultrasonic sensors.

The normal width of a crosswalk is approximately six feet, however, in larger cities, this is oftentimes doubled. This presented the unique challenge of finding sensors that could work at distances of up to 14 feet, while still being small enough to physically fit in the modules. The PIR is the only sensor that met this requirement. In addition, with a standby current of 1A, they were the obvious choice for our power and life-of-use constraints.

3.2.3 Vehicle Detection

Although PIR sensors work very well for pedestrian detection, our tests have shown that they are not effective at detecting vehicles. Because of the potentially high speeds of vehicles and their relative direction to the crosswalk, the PIRs do not accurately detect vehicular traffic. To counter this, the team decided to add another component to the hubs, a magnetometer. Fundamentally, a magnetometer works by detecting changes in the magnetic field around itself. Because all vehicles contain large quantities of metal, when a vehicle crosses a magnetometer, it detects a dramatic change in the magnetic field around it. The microcontroller can then use an algorithm to determine when a car passes over the hub using the data produced by the magnetometer.

3.2.4 Light Emitting Diodes

A number of LEDs were considered before the team decided to use common anode Red, Green, and Blue (RGB) LEDs. In conjunction with the team's constant current LED driver, these modules will allow the team to display alerts to drivers in an array of colors at a varying brightness. This was an important consideration in preventing inhibitions in the driver's field of vision due to exceedingly bright light. The team chose not to use single color LEDs because different states and countries have different laws on what colors are permitted for use on roadways. By using RGB LEDs, the color can be set and changed after installation to choose the best color light for the setting.

3.2.5 LED Drivers

The two options for driving the LEDs on the modules are either using the General Purpose Input/Output (GPIO) pins on the microcontroller or using an LED driver Integrated Circuit (IC). Using the GPIO pins is unsuited for our application because it would require placing current limiting resistors in series with the pins of the LEDs. The current limiting resistors would consume power unnecessarily, which is undesirable since one of the system's main constraints is power consumption. Constant current LED drivers contain hardware to regulate the current through each LED. As such, constant current LED drivers do not require the use of current limiting resistors, and they also ensure that all of the LEDs receive the same amount of current while preventing the LEDs from drawing more than their maximum rated current. Therefore, the team decided to use constant current LED drivers.

The characteristics used to evaluate the different LED drivers were the following: minimum supply voltage, communication bus, number of outputs, package, dimming capabilities, constant current capabilities, and cost. Originally, the chosen minimum supply voltage was 2.7V, however, later in the design, the minimum supply voltage was increased 3.3 V due to factors that will be discussed later. The team chose Inter-Integrated Circuit (I^2C) due to the ease of use using the Arduino Wire library and the team's familiarity with it. Since each module will use four RGB LEDs, the minimum number of outputs required on the LED driver is twelve because each RGB LED has three cathodes that must be connected to the driver. However, since I^2C supports multiple devices on the same bus, multiple LED drivers could be used to bring the total number of outputs to twelve, if the drivers have enough addressing pins to support the required number of drivers on the I^2C bus. The preferred package for the LED driver is Shrink Small Outline Package (SSOP), or an SSOP variant, due to the packages' small sizes and ease of soldering. However, an LED driver in a Quad-Flat No-Leads (QFN) package is acceptable even though it is harder to solder than SSOP. The LED driver needs dimming capabilities in order to reduce power consumption by the LEDs and to allow control of the brightness to prevent unnecessary distractions to drivers or pedestrians. As discussed previously, the LED driver needs constant current capabilities. Finally, cost was considered as a final discriminating factor.

The following is a table showing the characteristics of four different LED drivers considered for use.

Table 3.2: Comparison of LED Drivers

Manufacturer	**Part #**	**Minimum Supply Voltage**	**Outputs**	**Package**	**Constant Current**	**Cost (USD)**
Maxim Integrated	MAX8647	2.7V	6	Thin QFN	Yes	$5.46
NXP Semiconductors	PCA9532	2.7V	16	TSSOP, HVQFN	No	$2.65
ISSI	IS31FL3218	2.7V	18	SOP, QFN	Yes	—
Texas Instruments	TLC59116	3.0V	16	TSSOP, VQFN	Yes	$3.15

The Maxim LED driver was the initial candidate for use in the system. However, the Maxim LED driver has a hardcoded I^2C address, and, thus, only one LED driver can be used on the I^2C bus. Also, the Maxim LED driver was one of the most expensive drivers found. The NXP LED driver would be suitable for the system if it were constant current. Since the NXP LED driver is not constant current, it was not chosen for use in the system. The ISSI LED driver met all of the requirements for the system. However, Digi-Key does not sell the ISSI LED driver, and Mouser only sells the ISSI LED driver in multiples of 550 units. The team was unsuccessful at finding a supplier for the ISSI LED driver. Finding LED drivers that met the 2.7V minimum supply voltage requirement proved to be extremely difficult, so the team decided to increase the supply voltage of the system to 3.3 V. Given the increased supply voltage, the Texas Instruments LED driver was chosen because it met or exceeded all of the requirements for the LED driver while reasonably priced.

3.2.6 Energy Management

A high-level overview of the SCDLS energy pipeline is given in Figure 3.2.

The team estimated power consumption based on data gathered from the individual components in the system to determine what amount of power would be necessary. The worst-case calculations for a node and a hub are located below. The calculations assume that the module is located in Atlanta, Georgia. This assumption was made because SCDLS is not capable of withstanding an impact from a snowplow or other large machinery used for clearing snow form a road. The modules will only illuminate at night because it is not feasible for the LEDs to compete with the light from the sun during the daytime, while still relying solely on solar power. Additionally, the duty cycles

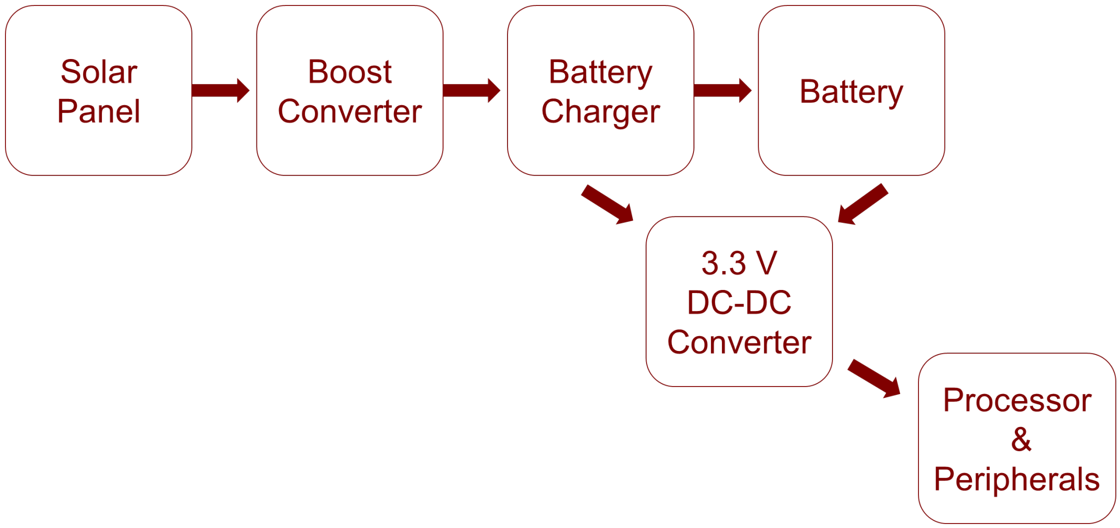

Figure 3.2: Power Architecture for SCDLS Module

are team-estimated values that will depend highly on the traffic of a specific crosswalk. For ease of power calculation, the primary voltage is 3.3 V because it is compatible with all components, which will be explained further in this document.

3.2.7 Node Energy Requirement Calculations

$$\text{Energy required} = 24\text{ hours} - \#\text{ Hours of sunlight} \cdot \text{Average active power consumption}$$

$$\begin{aligned}
\text{Average Active Power Consumption} &= \frac{\text{NRF Avg Power}}{\frac{\text{NRF Duty Cycle}}{100\%}} + \frac{\text{Microcontroller Avg Power}}{\frac{\text{Microcontroller Duty Cycle}}{100\%}} \\
&\quad + \frac{\text{LED Driver Avg Power}}{\frac{\text{LED Driver Duty Cycle}}{100\%}} \\
&= \frac{13\,\text{mA} \cdot 3.1\,\text{V}}{\frac{10\%}{100\%}} + \frac{4.3\,\text{mA} \cdot 3.1\,\text{V}}{\frac{5\%}{100\%}} + \frac{10\,\text{mA} \cdot 3.1\,\text{V}}{\frac{5\%}{100\%}} \\
&= 6.24\,\text{mW}
\end{aligned}$$

$$\text{Energy required} = (24\,\text{h} - 9.92\,\text{h}) \cdot 6.24\,\text{mW} = 316\,\text{J}\,\text{d}^{-1}\ [18]$$

3.2.8 Hub Energy Requirement Calculations

$$\text{Energy required} = 24\text{ hours} - \#\text{ Hours of sunlight} \cdot \text{Average active power consumption}$$

$$
\begin{aligned}
\text{Average Active Power Consumption} &= \frac{\text{NRF Avg Power}}{\frac{\text{NRF Duty Cycle}}{100\%}} + \frac{\text{Wi-Fi Avg Power}}{\frac{\text{Wi-Fi Duty Cycle}}{100\%}} \\
&+ \frac{\text{Microcontroller Avg Power}}{\frac{\text{Microcontroller Duty Cycle}}{100\%}} + \frac{\text{PIR Sensor Avg Power}}{\frac{\text{PIR Sensor Duty Cycle}}{100\%}} \\
&+ \frac{\text{LED Driver Avg Power}}{\frac{\text{LED Driver Duty Cycle}}{100\%}} + \frac{\text{Magnetometer Avg Power}}{\frac{\text{Magnetometer Duty Cycle}}{100\%}} \\
&= \frac{13\,\text{mA} \cdot 3.1\,\text{V}}{\frac{30\%}{100\%}} + \frac{197\,\text{mA} \cdot 3.1\,\text{V}}{\frac{0.35\%}{100\%}} + \frac{4.3\,\text{mA} \cdot 3.1\,\text{V}}{\frac{30\%}{100\%}} \\
&+ \frac{0.17\,\text{mA} \cdot 3.1\,\text{V}}{\frac{20\%}{100\%}} + \frac{10\,\text{mA} \cdot 3.1\,\text{V}}{\frac{20\%}{100\%}} + \frac{0.1\,\text{mA} \cdot 3.1\,\text{V}}{\frac{20\%}{100\%}} \\
&= 62.4\,\text{mW}
\end{aligned}
$$

$$
\text{Energy required} = (24\,\text{h} - 9.92\,\text{h}) \cdot 62.4\,\text{mW} = 1365.8\,\text{J}\,\text{d}^{-1}
$$

Multiple methods of power generation were considered before solar power was selected. Comparisons of the options can be seen in Table 3.3. Based on this analysis, the solar panel was a standout choice for powering SCDLS modules. Solar panels are also used by other manufacturers for similar applications.

Table 3.3: Power Sources for SCDLS Module [19]

Generation Technology	Solar	Thermoelectric	Acoustic	Piezoelectric	RF
Power output	15mW/cc	40 µW/cc	960nW/cc	330 µW/cc	116 µW/cc

Based on the power analyses that are outlined above, it was clear that solar power was the only feasible option for supplying power to this application. Therefore, multiple solar panels were compared. In order to account for any unforeseen issues, and to have a functional prototype before completing all necessary power optimizations, a safety factor above 4 was desired. The two primary requirements for selecting a solar panel were that it meets the desired power output with a safety factor and is a highly durable panel. For that reason, a 5V, 250mA peak, epoxy-covered solar panel was chosen. Figure 3.3 shows the power output from the solar panel during a sunny February day in Starkville, Mississippi.

When integrated using the MATLAB `trapz` function, this data yields a single day power output of $6614\,\text{J}\,\text{d}^{-1}$, which gives our team a safety factor of 4.98 for the hub and a safety factor of 13.4 for the node on that particular day. While this is not representative of the worst possible day, the team plans to use a lithium ion battery to allow the system to operate for extended periods without sunlight. When evaluating different battery options, the primary evaluation factors were the use of protected cells, energy density, packaging efficiency, and temperature requirements. Protected cells are standard battery cells with an added Current Interrupt Device (CID) that will disconnect the battery terminals from the battery in the event of an overcharge, undercharge, or overcurrent.

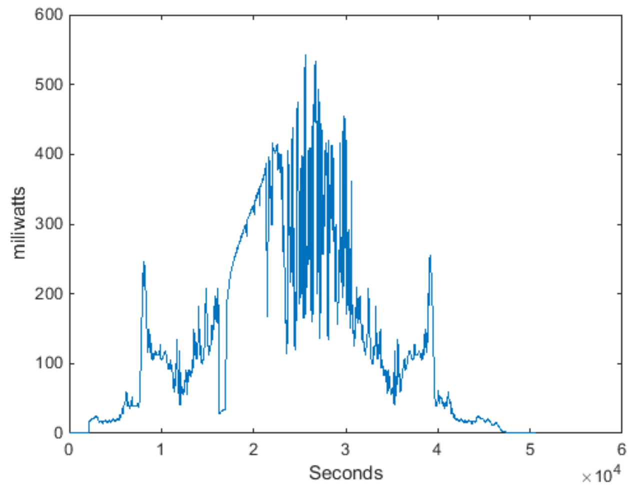

Figure 3.3: Solar Panel Power Output

This is necessary for product safety and reliability. Based on Figure 3.4, it was clear that the Li-Ion were the most energy-dense batteries and were capable of meeting the temperature requirements of $-10\,°\mathrm{C}$ to $50\,°\mathrm{C}$ [20]. In order to prevent thermal issues with the lithium battery a textured silicone rubber-insulating pad will be added between the road surface and the module. This pad will increase the friction between the module, and road while having a thermal conductivity of $0.14\,\mathrm{W\,mK^{-1}}$ [21, 22]. A few packages were considered, including the popular 18650 or 14500, but the most efficient package for our system is a rectangular prism due to the design of the casing. As such, a 3500 mA h lithium-ion Samsung Galaxy Note 2 battery was selected because of its flat, rectangular shape. Additionally, the Galaxy Note 2 battery has built-in cell protection and is manufactured by a large experienced manufacturer [20, 23].

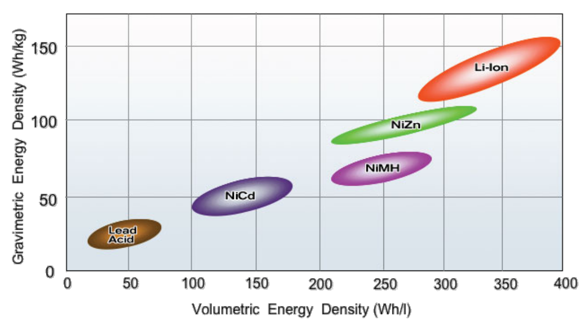

Figure 3.4: Energy Density of Different Battery Technologies [24]

The team wanted a fully charged battery to be able to power a module for 30 days without receiving any energy from the solar panel. A 3.3 V switching regulator will be used that has an efficiency above 90% at our desired output voltage under our expected current draw. Since the voltage regulation circuit is 90% efficient, the amount of time the battery can power a module is as follows:

$$\text{Hub Battery Capacity} = \frac{90\% \cdot 3500\,\mathrm{mA\,h} \cdot 3.7\,\mathrm{V} \cdot 3.6\,\mathrm{J/mWh}}{1328\,\mathrm{J\,d^{-1}}} = 32\,\mathrm{d}$$

In order to transfer the energy from the solar panel to the battery, an energy conversion circuit is necessary. Our requirements for this application include a minimum efficiency of 90% and compatibility with the previously selected solar panel. Some possible solutions to this problem were integrated circuits, a custom made circuit, or a premade solar panel charger circuit board by Adafruit. The custom-made circuit designs, with discrete components, soon exceeded the complexity that the team felt comfortable with, at no benefit. After further investigation, it was determined that the Adafruit board was less efficient, more expensive, and used more board space than the team could spare. As a result, integrated circuits were the front-runner. Multiple ICs were evaluated, but the ST Microelectronics SPV1040 and L6924D shined ahead of the pack due to the high quality documentation and a reference design that was closely applicable to our design. This reference design also came with test data indicating its performance in applications similar to SCDLS's [25].

Once energy is stored in the battery, it needs to be converted to a consistent voltage that is compatible with the microcontroller and all other peripherals. 3.3 V was chosen because it is between the minimum and maximum input voltage for all of the components chosen. The team set several constraints on the conversion of energy from the battery to power the other components: compatibility with input and output voltages, sufficient current handling capacity, and 85% efficiency under typical load conditions. Possible solutions meeting these requirements included linear regulators, low dropout regulators, and switching regulators, but after some analysis it was soon apparent that only a switching regulator would meet the efficiency requirement. In order to find an integrated circuit with these requirements a parametric search was performed that only displayed components meeting these requirements and the lowest cost component was chosen. This component is the TPS62240 switching regulator from Texas Instruments. This synchronous step down DC-DC switching regulator design has been implemented following the reference design included in the documentation for that part.

3.3 Software

3.3.1 Implementation Details

The SCDLS software stack is written in C++. This choice was made for a few reasons. First, much of the Arduino libraries are written in C++. Using C++ allows us to utilize these libraries with little to no additional effort. Because we utilize these libraries, our code's interaction with basic hardware features of the Central Processing Unit (CPU) is greatly simplified. C++ also has modern language features that make designing reusable software very easy. Of course, many of these features, such as run-time type information, are too resource-intensive to be used on a microcontroller, so we carefully considered each feature of the language to use. Done correctly, C++ is a great system to work in.

All the subsystems in our firmware are implemented as C++ classes. This approach provides encapsulation and facilitates asynchronous algorithms, which are crucial to maintaining the responsiveness of the system. The only downside to this approach is that there is a small amount of runtime overhead because instance pointers must be passed to all member functions consuming a hardware register. However, this would have to be done anyway for many of the subsystems because they maintain state between scheduler events.

3.3.2 Subsystems

Our project incorporates various software subsystems to meet our constraints and to provide a maintainable codebase. These subsystems include the scheduler, the device drivers, and the command layer.

Scheduler

The scheduler is responsible for keeping the system's time and executing events at the appropriate point. Other software modules communicate with the scheduler to schedule events. The scheduler is also ultimately responsible for managing the power saving modes on the microcontroller (when this functionality is implemented). The scheduler will awaken the CPU only when tasks are scheduled to be executed, leaving the device in a low-power state at all other times.

Modules wishing to use the scheduler are required to inherit from a C++ abstract base class, which allows the scheduler to call their callback function at the appropriate time without knowing exactly what system it is invoking. Although there is a small runtime cost associated with virtual function calls, it is quite small, even on embedded platforms such as ours. Furthermore, much of the Arduino stack already makes extensive use of virtual calls. We feel that any small runtime cost is easily offset by the ease of implementation. The primary alternative to the use of virtual functions is passing pointers to raw functions; however, because most modules maintain state, the scheduler would need to pass them an instance pointer anyway. Furthermore, code involving pointers to functions is generally less readable and maintainable than code that uses proper inheritance.

The scheduler relies on cooperative multitasking between modules and does not make any strong real-time guarantees. Therefore, it is the responsibility of the subsystems' authors to ensure that the subsystems do not block the CPU while waiting for external events. Although it would be advantageous to have a Real-Time Operating System (RTOS) from a responsiveness perspective, the very limited system resources of the microcontroller prohibit an advanced task-switching framework. Therefore, the cooperative scheduler provides a good compromise between a full RTOS and no multitasking framework whatsoever.

Device Drivers

The device drivers are the subsystems which communicate directly with the hardware. These modules are responsible for handling input from sensors, sending data over the wireless links, and turning on and off the lights. These systems forward incoming events to the command layer. The implementation details for most of these driver systems are fairly straightforward.

Network Driver Subsystem

In contrast to most of our device drivers, the networking subsystem is fairly complex. It provides several features beyond those provided by the nRF24L01+ radio hardware. The radio uses a simple data-link protocol called ShockBurst. A ShockBurst frame consists of a five-byte address, a 32-byte payload, and a CRC16 checksum. The radio can be configured to receive frames addressed to up to five addresses. Our system uses the first four bytes of the address as a "network ID." Each crosswalk system will have a unique network ID assigned by the system administrators at installation time. This allows multiple crosswalks to be installed in close proximity without interfering with each other's communications. The last byte of the address is the node ID; each module assigns itself a random ID when it boots. ID collisions are handled in the command protocol, which will be discussed later.

Offset	0x00	0x01	0x03
Data	Frame Type	Frame Data Size	Packet ID

Figure 3.5: Network Subsystem Large Frame Header

Our network subsystem is a combined network and transport layer that is implemented on top of the radio's data link layer. It allows nodes to send messages to any other node, as discussed above. It also allows nodes to send packets of arbitrary size, not limited to the 32-byte ShockBurst frame size. Large packets, which do not fit into a single frame, are broken into pieces and sent sequentially. The receiving node(s) reassemble the packets upon delivery. This model incurs an overhead of one byte on all frames sent and an additional overhead of three bytes per frame on large packets. This overhead consists of frame information, such as the packet ID, the relative frame number, and the total number of bytes in the packet. Because the nodes could potentially be transmitting relatively large data, such as sending stored traffic statistics to the master node, this overhead is rather small and is acceptable.

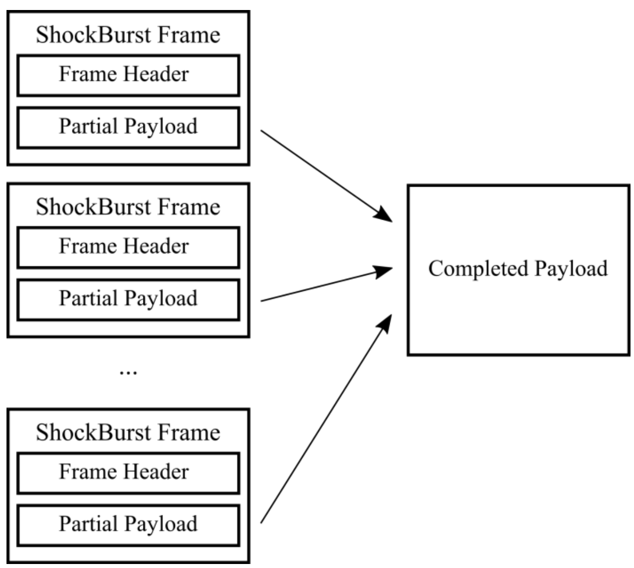

Figure 3.6: Large Payload Reassembly

The last main feature provided by the networking subsystem is broadcast packets. In addition to sending packets to other individual modules in the network, modules can broadcast a packet that will be received and processed by all nodes. This functionality is used by the hubs to broadcast sensor data, which is received and interpreted simultaneously by all nodes. The networking subsystem uses the multiple-address functionality of the radio IC to not only receive packets addressed to its specific node ID, but also packets addressed to a reserved "broadcast ID" (currently `0x00`). This design eliminates the need for each hub to keep track of every other node in the network and manually send copies of sensor data to each of them, thereby reducing power consumption and network congestion.

Command Layer

The command layer is responsible for taking inputs from the sensors and performing actions appropriately. The primary actions performed by the command layer are dispatching commands to other subsystems; due to the asynchronous nature of our software stack, little CPU time is spent in the command layer itself.

Because no sensor is perfect, there will be some false readings. One of the functions of the command layer is to filter out sensor noise and to only turn on the lights when it is highly probable that there are pedestrians present. There are several options for filtering algorithms for the PIR sensors we chose. The algorithm we are currently using is a simple threshold detection algorithm where two sensors must be triggered within a preset number of seconds, currently five seconds. This algorithm is simple but should prevent occasional false positives caused by random errors.

The flowchart in Figure 3.7 gives an overview of the actions performed by the command layer. Note that only the hubs have onboard sensors; therefore, the shaded grey section of the flowchart is only implemented on hubs.

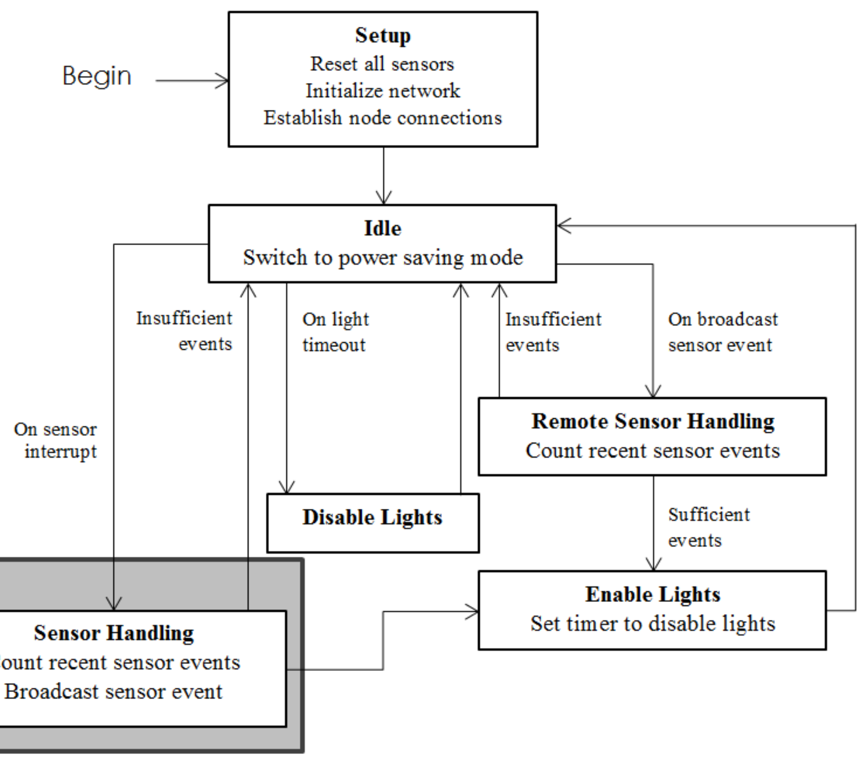

Figure 3.7: Command Layer Logic Flowchart

The modules spend most of their time in a power saving "idle" mode. Because all the subsystems are activated by external events or on a timed basis, the command layer does not have to spend most of its CPU time in a busy-waiting loop (that is, executing no-operation instructions while waiting for an event). When an event occurs, the subsystem's handler runs. Then, if necessary, the handler invokes the appropriate routine in the command layer. The command layer will handle the sensor event and dispatch a command to the light driver, if necessary. It will also schedule a lights-off event so that the CPU does not busy-wait for the lights to turn off. The scheduler will then wake the processor and return control to the command layer at the appropriate time.

3.3.3 Use Cases

Using the above high-level flowchart, several user interactions are possible. Note that, in each case, the goal of the system is to provide as much lighting to the crosswalk as possible. The ideal "sunny day" user interaction is shown in Figure 3.8.

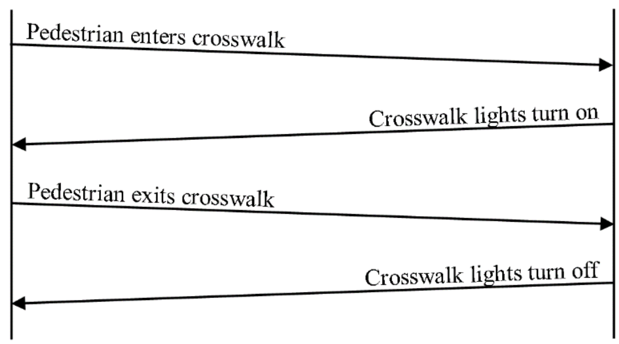

Figure 3.8: Sunny Day User Interaction

This is a very simple ideal interaction; this is by design. SCDLS should work with no direct interaction from its primary users. Of course, this diagram is from the perspective of the user; from the standpoint of the system, this interaction gets somewhat more complicated — see Figure 3.9.

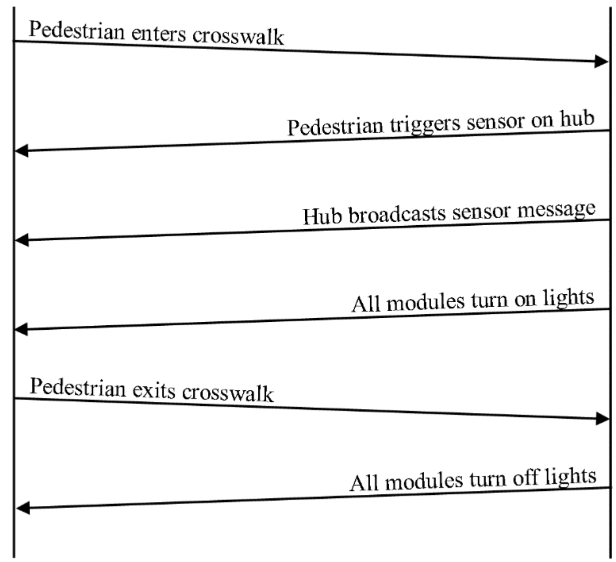

Figure 3.9: Sunny Day System Interaction

A worse-case scenario is when there is interference on the 2.4 GHz spectrum. In this scenario, not all nodes may receive the broadcasts. Even though these messages may not be delivered, the nodes that do receive the message turn on their lights, providing some protection to the pedestrian. Although this may look unusual to users, we decided that this degraded operation provides some protection to pedestrians and is better than total system failure. Figure 3.10 shows the events for this degraded operation.

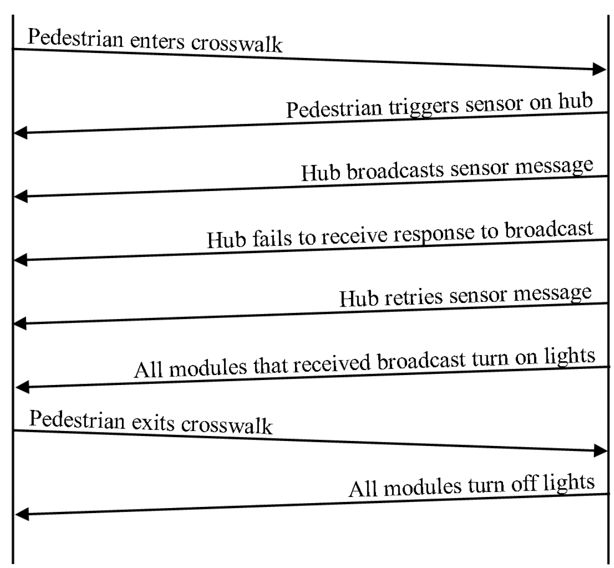

Figure 3.10: Rainy Day System Interaction

3.4 Packaged Case

Images of the final packaging of a SCDLS module can be seen in Figure 3.11 and Figure 3.12. From bottom to top, the following components are visible: The battery, the main circuit board and charger board, and the LED wiring. Not shown in Figure 3.11 is the solar panel, which goes on top.

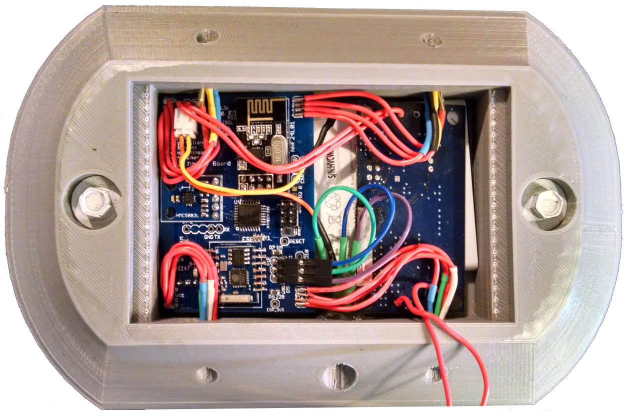

Figure 3.11: Top-Down View of Packaged Device

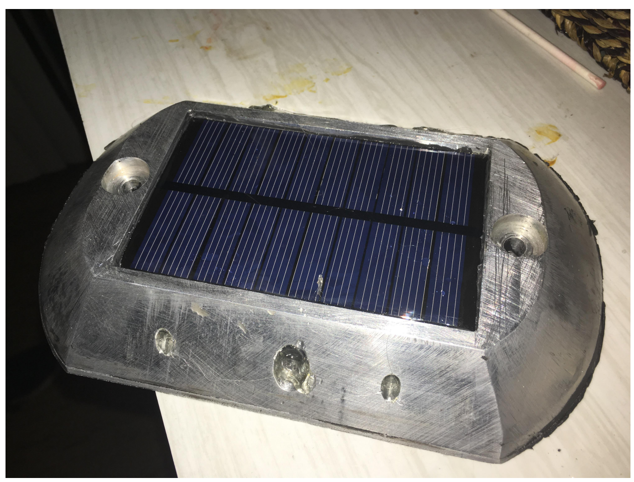

Figure 3.12: Completed Unit

4 Evaluation

This section explains the testing and evaluation of hardware and software subsystems of the SCDLS modules. The subsystems were tested during the building and evaluation phases of the project to ensure that the modules would work as expected upon completion of the build.

Table 4.1 contains the technical constraints tested as part of the evaluation of the subsystems.

The following sections detail the testing and validation for each of the subsystems listed in Table 4.1.

Table 4.1: Evaluation Technical Constraints

Name	Test Description
Module to Module Communication	Modules must be able to communicate wirelessly with one another at a distance of 10 meters.
Solar Generation	The solar panel must provide over 1800 joules of energy on average per day.
Battery Life	The system must consume less than 1600 joules of energy on average per day.
Wi-Fi Communication	Module must be able to upload traffic analytics to a remote web server.
Motion Detection	This component must be able to accurately detect pedestrians at a distance of 16 feet or less.
Vehicle Detection	This component must be able to detect vehicles moving at a speed of at least 5 miles per hour (mph).
LED Driver	This component must be able to effectively power two or more LEDs and utilize the I^2C bus for communication with the microcontroller. All LEDs and consume less power than using resistors.
3.3V Regulator	This component must be able to output 3.3 V continuous to be used by other subsystems.

4.1 Solar panel charging circuit

A combination SPV1040 Maximum Power-Point Tracking (MPPT) boost converter and L6924 Li-Ion battery charger integrated circuits were used to efficiently convert solar power to usable energy in order to charge the battery. Pictured in Figure 4.1 is a test circuit showing the selected solar panel charging the selected battery. The schematic for this charging circuit, shown in Figure 4.2, is a reference design by ST Microelectronics. The orange and green meter is placed in series at the V_{bat} point designated by arrows, and the red and black meter is in parallel across capacitor C9.

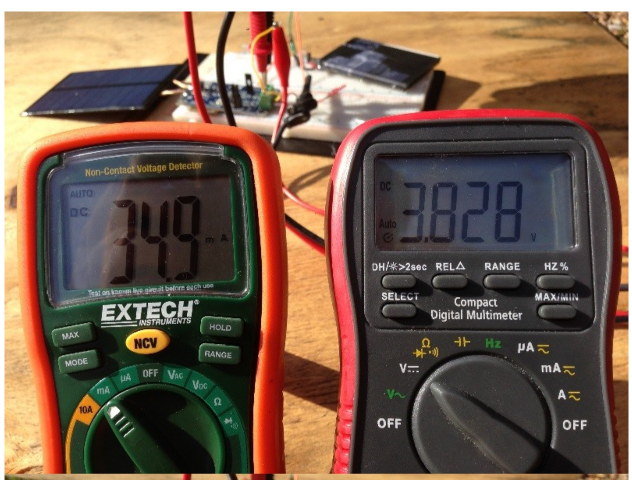

Figure 4.1: Picture of the solar panel charger test

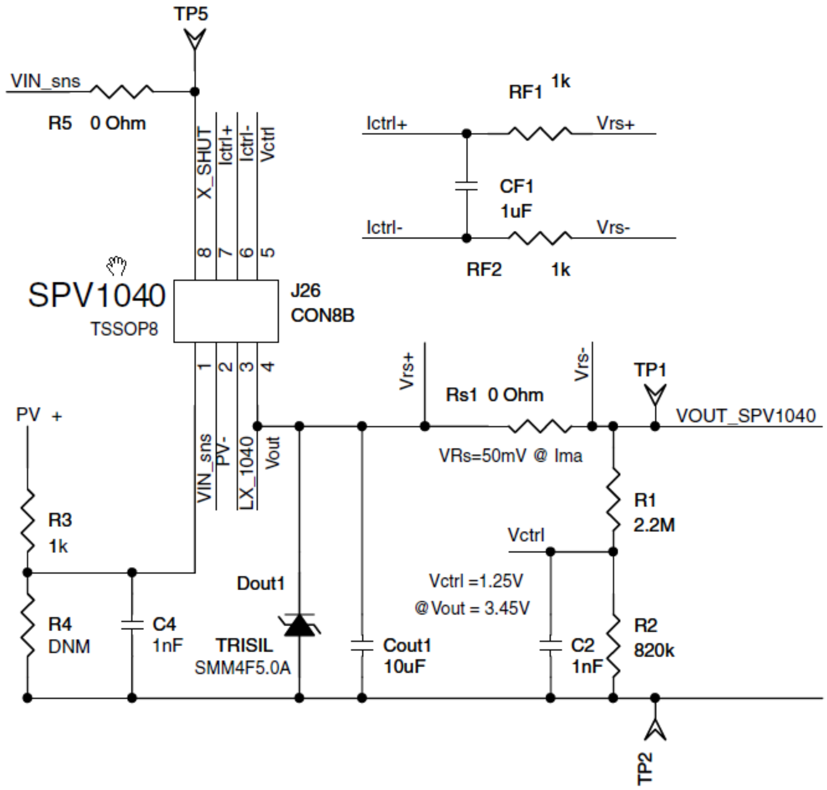

Figure 4.2: Schematic of Solar panel charger boost converter circuit

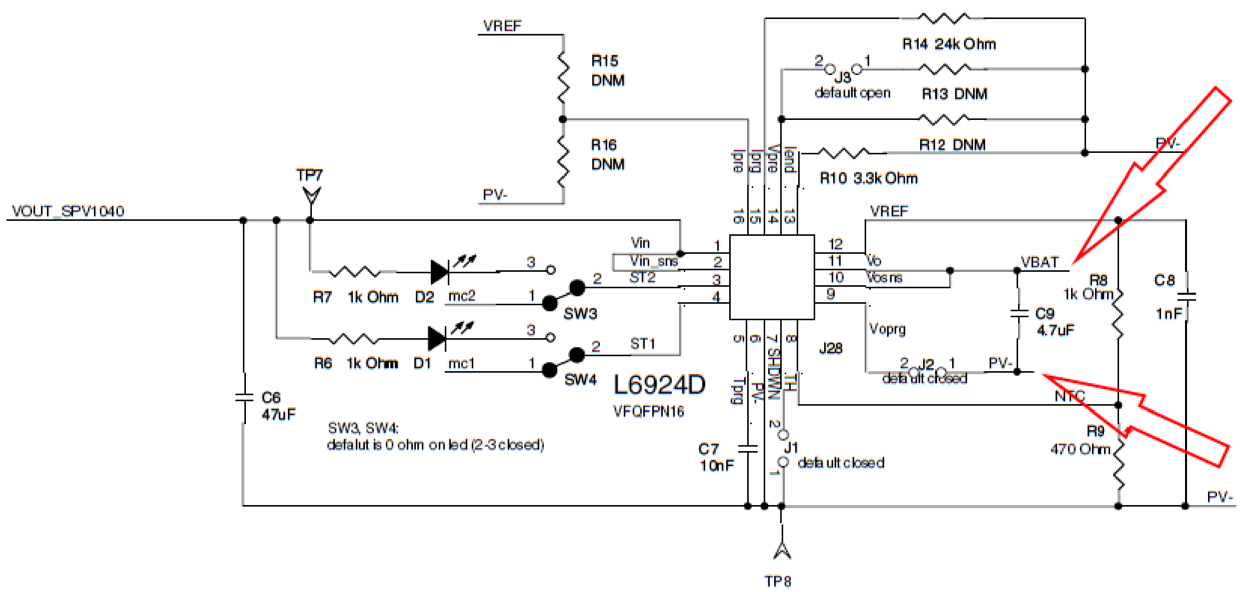

Figure 4.3: Schematic of battery charger circuit

The solar panel charging circuit does function as intended, and an analysis of energy harvested in this circuit is performed in section 4.2.

4.2 Solar Generation

For this test, the solar panels were soldered to an ST Microelectronics SPV1040 chip including MPPT. The test included leaving a panel outside for a number of hours while actively recording the voltage and current fluctuations over time. By choosing a suitably sized panel, the output power was deemed more than suitable for powering our devices. A MATLAB generated trace of $I_{output} \cdot V_{bat}$ is displayed in Figure 4.4, and when the MATLAB `trapz` trapezoidal integration function is applied, a result of approximately 6.6 kJ is found. This data was generated from a data logging function on a Fluke 199C Scopemeter. A DC current probe was placed around the V_{bat} wire connecting the lithium battery charger circuit displayed in Figure 4.4 to the battery. The battery was discharged to 3.7 V before the start of the test to ensure that the battery would not reach full charge during the test.

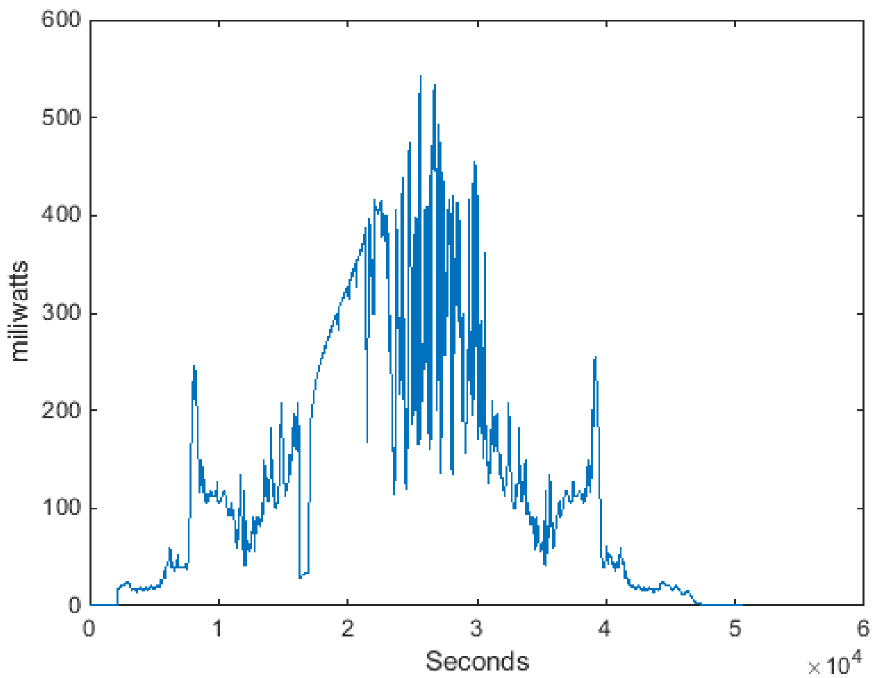

Figure 4.4: Solar energy transferred from panel to battery

4.3 Battery Life

Battery life is dependent on the ability for the solar panel to recharge the energy that was depleted during operation of the unit. As a result, the unit must meet certain power requirements to function appropriately and reliably for the life of the unit. In Table 4.2 the current consumption for the individual components is found from the accompanying datasheet, and also measured with an Amprobe AM 220 multimeter. This analysis was completed using a bus voltage of 3.3V for power calculation. In order to test these subsystems under normal operating circumstances the multimeter was inserted between the components and their voltage source. For this test it was assumed that no power was being transmitted over the signal lines connected to these various components

Table 4.2: Power Consumption Data

Component	Duty Cycle	Datasheet Current (mA)	Datasheet power (mA)	Average datasheet Power (mW)	Measured Current (mA)	Measured power (mA)	Average Measured power (mA)	% Difference
NRF	0.3	8.9	29.37	8.81	9.3	30.69	9.21	4.49
ESP 8266	0.0035	197	650.1	2.28	91	300.3	1.05	53.8
Micro	0.3	4.3	14.19	4.26	9.7	32.01	9.60	125.5
PIR	0.2	0.17	0.561	0.112	0.22	0.726	0.145	29.4
LED	0.2	15	49.5	9.9	15.3	50.49	10.1	2
Magnetometer	0.2	0.1	0.33	0.066	0.11	0.363	0.072	10
EEPROM	0.01	2	6.6	0.066	1.7	5.61	0.056	15
				=			=	
Energy (Joules)				1292			1532	18.6

In general, the datasheet current measurements were relatively close to the values that were later measured. However, a two major discrepancies exist between the expected power and the measured power, the microcontroller and the ESP 8266 Wi-Fi chip. The microcontroller uses substantially more power than was expected and this can be explained by our team's use of the Arduino Nano board instead of using just the ATmega 328P processor. In the Arduino Nano board there is an additional chip for USB communication that we are not using, and there are two status LEDs that are unnecessary in our final application. The ESP 8266 uses less power in its steady state operation phase likely as a result of the low amount of interference between modules while testing was ongoing. The power consumption of the ESP 8266 should be reevaluated once final enclosures are made.

4.4 2.4GHz Communication

Nordic Semiconductor's nRF24L01+ radio transceiver IC is ideal for module-to-module communication because of its low power usage and relatively high throughput. The nRF24L01+ uses the SPI bus for communication with the microcontroller. A test circuit to verify the nRF24L01+'s output is shown in Figure 4.5.

To test that the subsystem is working correctly, a logic analyzer was connected to the SPI bus. A transaction was captured and decoded, and the expected payload bytes were searched for in the byte stream. These bytes were found and are displayed in Figure 4.6. The bytes on MISO, from left to right, are: packet ID, payload length, message type, and sensor value.

The code that generates these bytes is as follows:

```
struct AnnounceSensors {
    const static CommandByte commandByte = 2;
    bool detected;
};
template<typename CommandType>
struct Message {
```

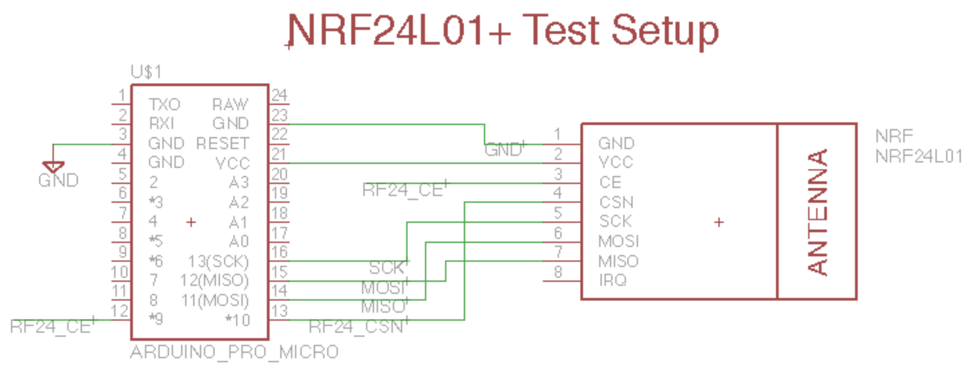

Figure 4.5: SPI Communication Test Circuit for the nRF24L01+ Module Communication Network.

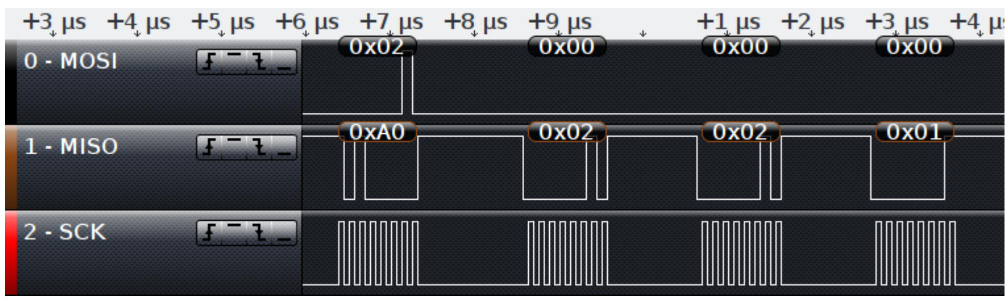

Figure 4.6: Sample SPI communication between nRF24L01+ radio and microcontroller

```
    uint8_t commandType = CommandType::commandByte;
    CommandType command;
};
```

A `Message<AnnounceSensors>` was sent with `detected = 1`, which compiles to the bytes `{0x02, 0x01}`.

To test the range of these devices, the distance between two devices was increased until they could no longer communicate. The devices were said to communicate if stimulating the sensor of one unit caused the LEDs on the other unit to illuminate. Figure 4.7 shows the maximum range achieved using this method, which was in excess of 40 meters. It also shows the devices functioning correctly at 10 meters. This performance meets our constraint.

4.5 Wi-Fi Communication

Expressif's ESP8266 SOC allows for interaction with Wi-Fi networks, allowing our devices to upload stored traffic analytics to a remote server with ease. To ensure that this subsystem is functional, a logic analyzer was connected to observe the UART bus traffic between the ESP8266 and the microcontroller. The test circuit can be seen in Figure 4.8.

A sample of the captured byte stream is shown in Figure 4.9. This sequence shows the microcontroller using an AT modem command to put the ESP8266 in Wi-Fi client station mode. The ESP8266

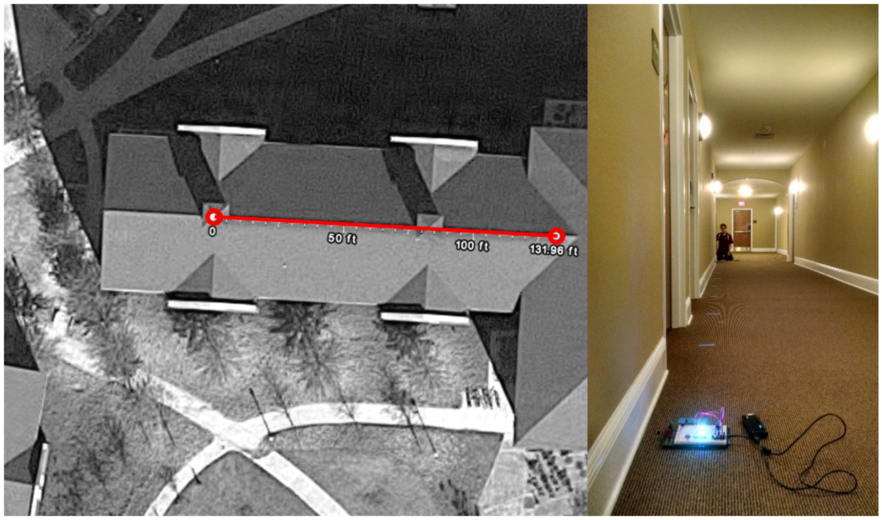

Figure 4.7: Maximum range of nRF24L01+ radios

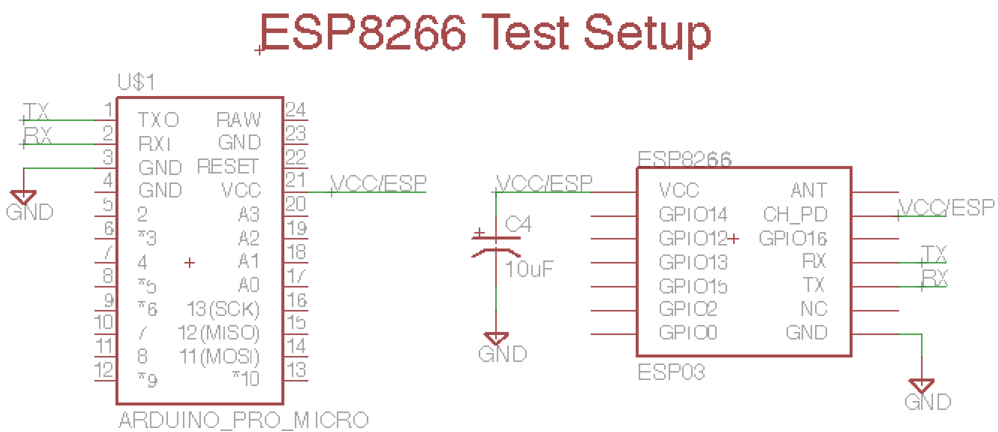

Figure 4.8: Serial Communication Test Circuit for the ESP8266 Wi-Fi Module.

responds with an OK, confirming that the device is functioning correctly.

The server side component must also be tested. To test this, a dummy traffic record was uploaded to the server. This record is shown in Table 4.3.

Table 4.3: Test traffic record

Offset	**0x00**	**0x01**	**0x05**
Payload	0x00	0x0018F387	0x23
Description	Record Type (car)	Timestamp (1970-01-19 16:13:27)	Speed (32 mph)

The server parsed and stored the data in a database, which was then dumped. The data matches that which was originally encoded, showing that the server is working. The database dump is as follows:

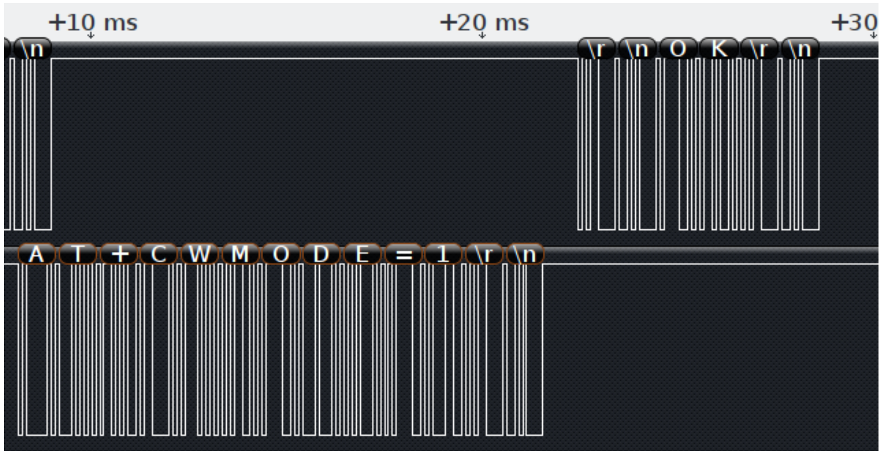

Figure 4.9: Sample UART communication between microcontroller and ESP8266

```
upload_time|network_id|node_id|timestamp                  |id|speed
           |          |       |1970-01-19 16:13:27.000000|1 |35
```

Note that several fields are not populated, because the server software has been designed to hold several fields that are not present in the current version of the device software. This is normal.

4.6 PIR Sensors

In our application the PIR is extremely important because this sensor is used to detect pedestrians as they enter and exit the crosswalk. A regular crosswalk has a width of eight feet, although in several metropolitan areas, this is doubled because of the amount of regular foot traffic. For that reason, the PIR sensors must detect pedestrians at a minimum distance of 16 feet. For this test, grid lines were measured out in a controlled area and team members walked along the grid lines in order to simulate a pedestrian using a crosswalk. Figure 4.10 displays some of the grid lines used for testing.

A grid line was placed every foot from the PIR sensor for twenty feet. Two computers were used to display the output of the PIR sensor. To begin the test, a team member walked along the grid line twenty feet from the PIR sensor in order to simulate a pedestrian. The team member then repeated this procedure at each grid line until reaching the PIR sensor. The observations from the tests are shown in Table 4.4.

From the data in Table 4.4, it is obvious that the PIR sensor meets the minimum detection distance of 16 feet. The testing regiment was repeated five additional times in order to validate the data.

Because PIR sensors work off of a comparator and Fresnel lenses, there usually tends to be a fluctuation in the output values. This fluctuation of the output can be seen in Figure 4.11.

The data was recorded through an Arduino's serial monitor and a Java application called *Processing* was utilized to take this live data output and translate it into the graph shown previously. The accuracy and relatively low false detection rate of the PIR sensor allows the microcontroller to simply receive an interrupt to wake from sleep. Instead of needing to continuously sample the

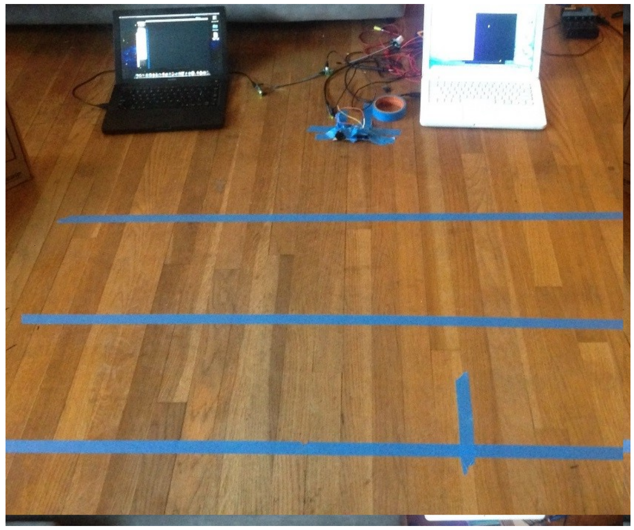

Figure 4.10: Test setup for PIR distance validation

Table 4.4: PIR Distance Validation Data

Distance	Pass/Fail
20 Feet	Fail
18 Feet	Pass
16 Feet	Pass
14 Feet	Pass
12 Feet	Pass
10 Feet	Pass
8 Feet	Pass
6 Feet	Pass
4 Feet	Pass
2 Feet	Pass

output of the PIR sensor, the sensor's output can simply trigger an interrupt on the microcontroller, thus reducing the processing load on the microcontroller as well as overall power consumption. A test circuit for the PIR sensor can be found in Figure 4.12, and the output is plotted using the method previously mentioned.

4.7 Magnetometer

A magnetometer is installed in the devices to detect vehicular traffic. The large, moving body of metal creates a fluctuation in the ambient magnetic field that can be detected. The device needs to be able to accurately detect traffic moving at speeds of at least 5 mph. The magnetometer uses the I^2C bus for communication, and the test circuit is shown in Figure 4.13.

To verify this functionality, a test device was placed in the road, then vehicles were driven over the device (taking care not to crush the device with the vehicle's tires). The device sampled

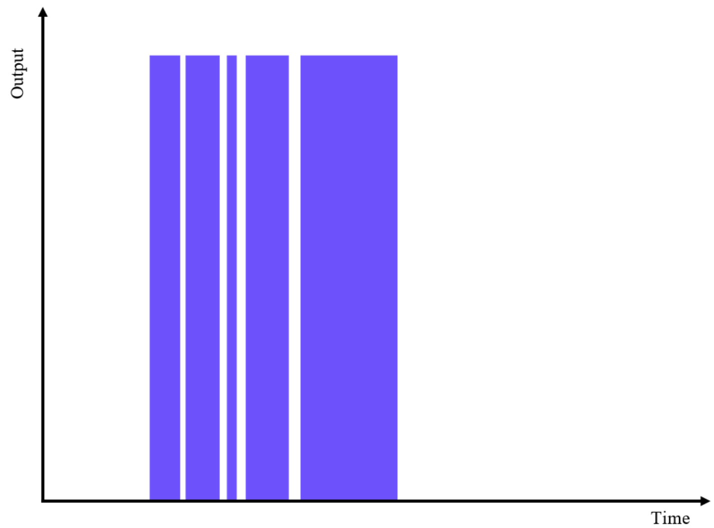

Figure 4.11: Digital Output of PIR Sensor

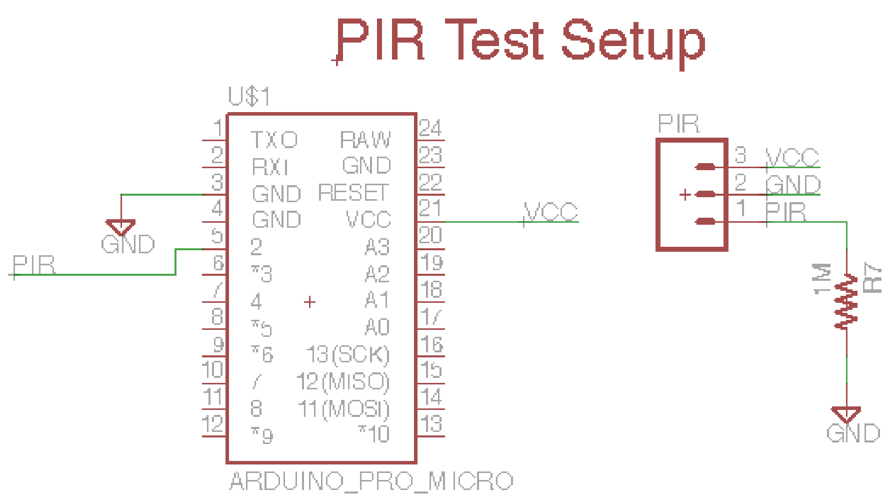

Figure 4.12: Digital Test Circuit for the Panasonic PIR Sensor.

the magnetic field every 50 milliseconds, then processed the data through the vehicle-detection algorithm. The results were captured and graphed, as shown in Figure 4.14. In this graph, the x axis is in the units of samples (which occur every 50ms), and the y axis is in arbitrary units read from the magnetometer. It is easy to see that the three colored lines, which correspond to the feeds from the magnetometer, begin to fluctuate at about sample 20. The algorithm's Boolean output is represented by the black line; a high value corresponds to "vehicle present," and a low value corresponds to "vehicle absent." The algorithm detects the vehicle very accurately.

Because the flux in magnetic field increases with vehicle speed, this test demonstrates that the device can meet its requirement of 5 mph minimum detected vehicle speed.

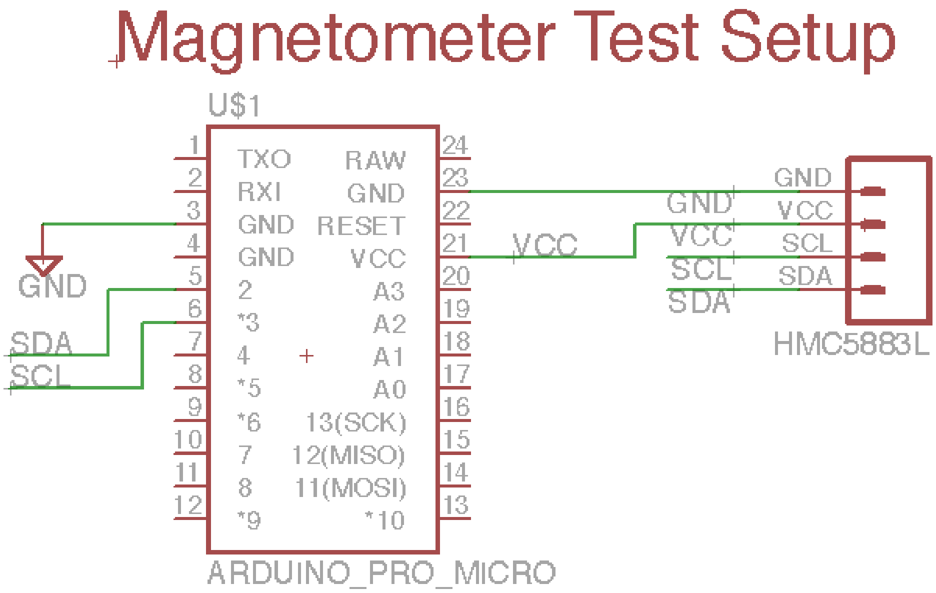

Figure 4.13: I2C Test Circuit of HMC5883L Magnetometer.

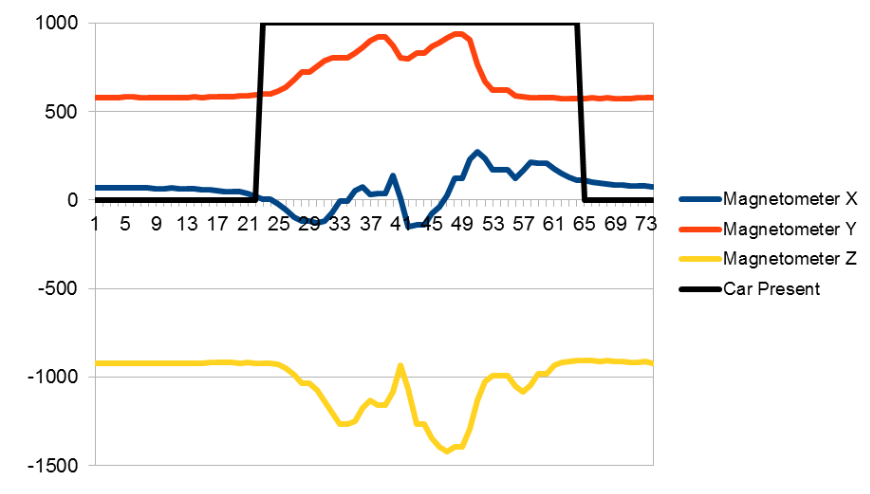

Figure 4.14: Sample test data from 5 mph vehicle drive-over

4.8 LED Drivers & LEDs

The microcontroller communicates with the LED driver using the I^2C serial bus. In order to show the I^2C communication between the microcontroller and the LED driver, a Bus Pirate multi-tool was used to sniff the I^2C bus. First, the microcontroller had a test program flashed to it that would initiate a series of test I^2C transactions with the LED driver. Then, the Bus Pirate was connected to the SCL and SDA lines. The Bus Pirate was then connected to a computer using USB. This test circuit can be seen in Figure 4.15.

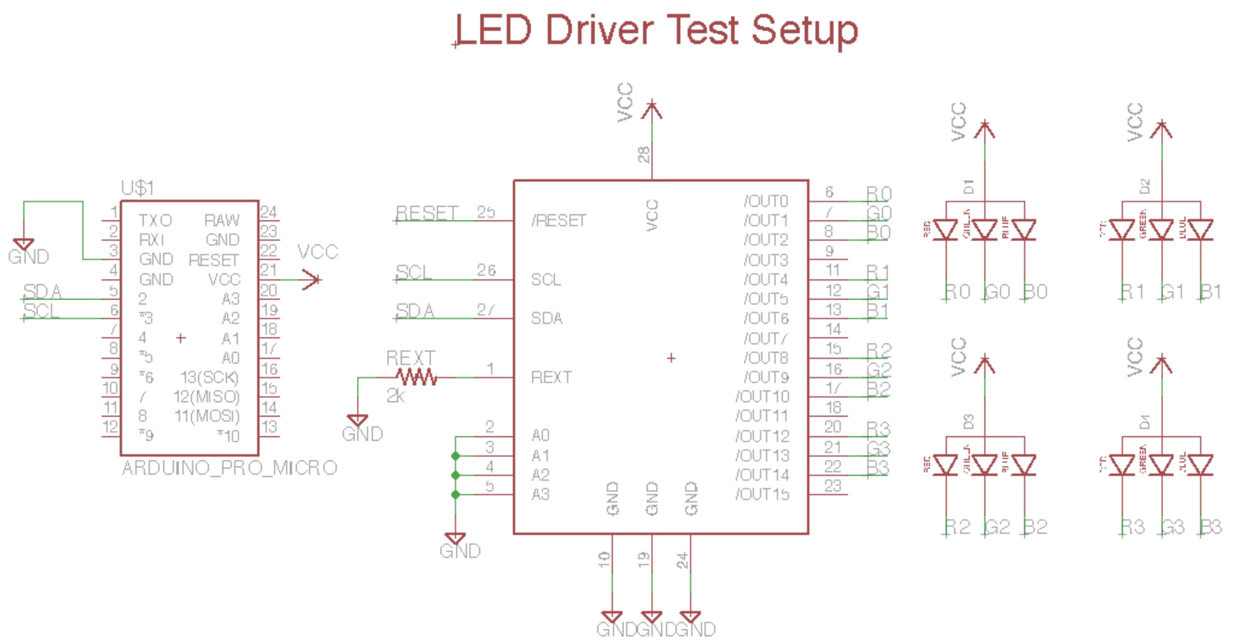

Figure 4.15: LED Driver Test Schematic

A proprietary program was used to communicate with the logic analyzer and start its I^{2}C bus sniffer. Finally, the microcontroller was reset so it would begin performing the test I^{2}C transactions. Three of the test transactions can be seen in Figure 4.16.

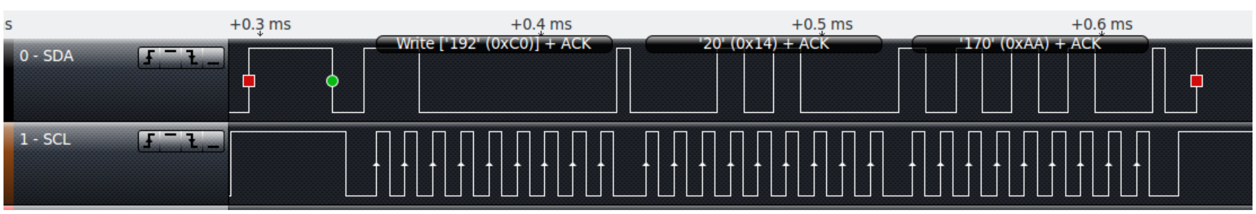

Figure 4.16: I^{2}C Transaction between the Microcontroller and LED Driver

The logic analyzer output shows a series of I^{2}C transactions used to set the value of one of the PWM registers on the LED driver. The first transaction initiates writing to the LED driver. The second transaction sets which register on the LED driver to write. The third transaction sets the PWM register to the test value. The PWM registers are used to control the dimming of each of the LED outputs. This set of test transaction was also used to test the RGB LEDs used in the system. Using this test transaction the RGB LEDs should be a violet color at maximum brightness. Figure 4.17 is a picture of the LEDs driven by the LED driver after completion of the I^{2}C test transactions.

It is difficult to tell the color of the LEDs in Figure 4.17. However, in person it was plain that the LEDs were violet. The test procedure was then repeated multiple times by setting the LEDs to the colors red, green, blue, yellow, and orange at different dimming values. The tests confirmed the operation of I^{2}C communications between the microcontroller and LED driver and the operation of

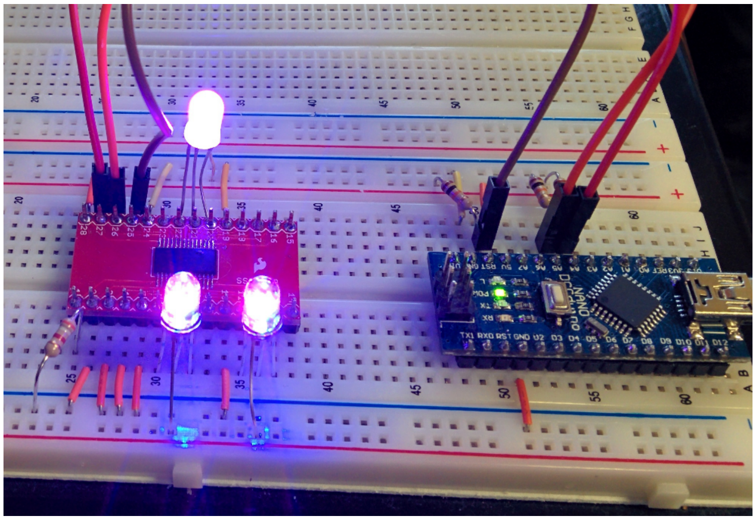

Figure 4.17: Picture of the Test of the RGB LEDs

the RGB LEDs used in the system.

4.9 3.3V Switching Regulator

The TPS62240 regulator was utilized to provide a constant voltage to all components and peripherals. The output voltage is selected based on an equation from this part's data sheet. The values of 680 kΩ for R1 and 150 kΩ for R2 were chosen to produce an ideal voltage of 3.32 V. The output was then tested and verified with a multimeter as shown in Figure 4.19. The schematic for this subsystem test is displayed in Figure 4.18.

4.10 System Test

To evaluate reliable operation of the system as a whole, a test crosswalk was built with two hubs and 4 nodes in their respective enclosures. When this model crosswalk was set up two tests were performed, one that validated that when an individual enters the crosswalk the lights flash in the desired pattern, and when an automobile passes over the node traffic data is collected.

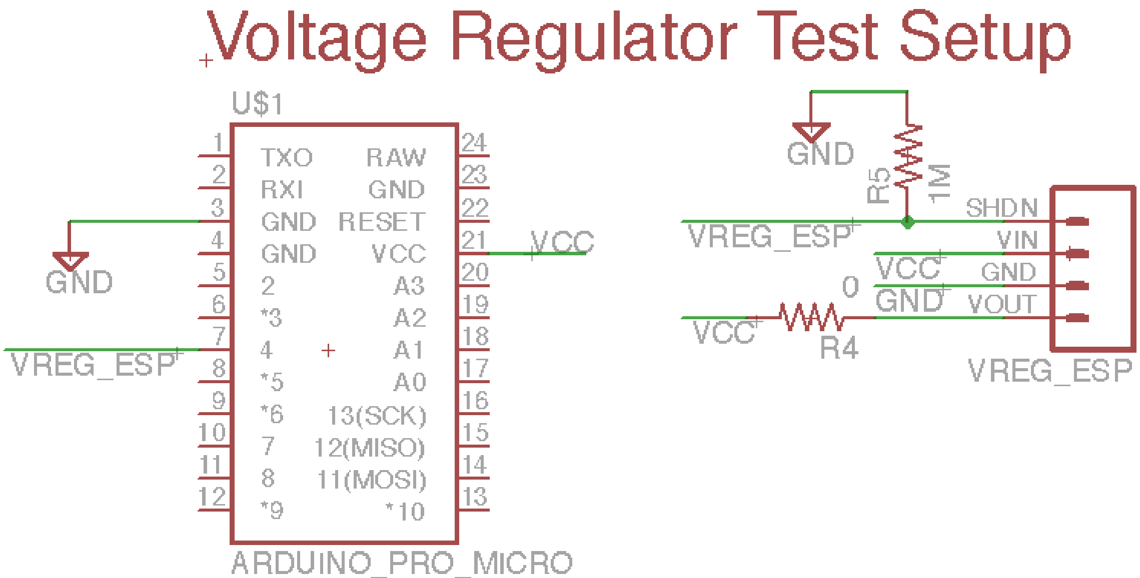

Figure 4.18: Schematic of test circuit for switching regulator

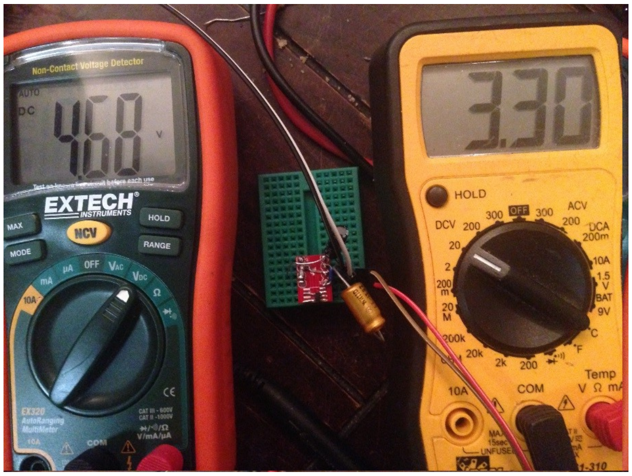

Figure 4.19: Schematic of Solar panel charger circuit

Figure 4.20: Full System Test

5 Summary and Future Work

The Smart Crosswalk Dynamic Lighting System is a crosswalk system that consists of pedestrian and vehicle detecting hubs that wirelessly communicate with nodes and other hubs to illuminate the interior and exterior of the crosswalk in order to alert oncoming vehicles about the pedestrian in the crosswalk. The hubs and nodes each contain four RGB LEDs in order to illuminate the crosswalk. The hubs maintain traffic metrics by using a magnetometer to detect vehicles and PIR sensors to detect pedestrians. The traffic metrics are then uploaded to a clients website at regular intervals.

In the first stage, the team began the design process optimistically about the time necessary to validate component selection for the modules. Most of the components that were researched worked as intended, but the initial decision to use ultrasonic sensors to detect pedestrians came with some disadvantages. The ultrasonic sensors were ideal in theory, but when tested, the output was very noisy, and after testing was performed our team realized that the variable transformer necessary for the ultrasonic sensor had an output voltage of over 100 VAC which presented some interference and safety issues. After the ultrasonic sensors proved to be less than optimal, the team decided to use PIR sensors. In retrospect, these were objectively a better option, they were smaller and an order of magnitude more energy efficient for the same task. Another problem the team faced was the design of the PCB for the solar panel charger. A PCB was designed for the solar panel charger in order to make more nodes, but the footprints of one of the ICs was so small that the team could not physically solder the IC to the PCB.

In the second stage, the PCB design resulted in reduced communication range due to the ground plane proximity to the communication antenna. In the future, we plan to research how our PCB and product packaging effect the communcation performance for our product. We plan to redesign our PCB once more with more consideration for the physics of radio communication.

6 Acknowledgements

The SCDLS team would like to acknowledge and thank Dr. Robert Reese for all of the guidance, insight, and encouragement he has given us this semester as our faculty advisor. The SCDLS team would also like to thank Dr. Randolph Follett for being our instructor for Senior Design I. The team looks forward to working with him next semester as well. The team acknowledges and thanks all the sponsors and donors that contributed components used during development of the prototype. A final thank you goes to the faculty and staff of the Electrical and Computer Engineering Department at Mississippi State University for helping and equipping each team member with the knowledge and skills required to make this project a success.

References

[1] A. Wallace-Hadrill, *Pompeii: Portents of disaster*, http://www.bbc.co.uk/history/ancient/romans/pompeii_portents_01.shtml, 29.

[2] M. Lays, "Ways of the world," *Primavera Press*, p. 401, 1992.

[3] N. Zeldes, *Pedestrian crossing in antiquity*, http://designblog.nzeldes.com/2010/12/pedestrian-crossings-in-antiquity/, 2010.

[4] *The man who gave us traffic lights*, http://www.bbc.co.uk/nottingham/content/articles/2009/07/16/john_peake_knight_traffic_lights_feature.shtml, 2009.

[5] A. McFarlane, *How the UK's first fatal car accident unfolded*, http://www.bbc.com/news/magazine-10987606, 2010.

[6] N. H. T. S. Administration, *Traffic safety facts 2012 data: Pedestrians*, http://www-nrd.nhtsa.dot.gov/Pubs/811888.pdf, 2014.

[7] V. from the Mirror, *The cruel capital: Two grim road firsts*, http://blackcablondon.net/tag/crystal-palace-park/page/2/, 2012.

[8] *Pedestrian crossing: 50 state summary*, http://www.ncsl.org/research/transportation/pedestrian-crossing-50-state-summary.aspx, 2014.

[9] *Bicycle and pedestrian safety*, http://www.ncsl.org/research/transportation/bicycle-and-pedestrian-safety.aspx, 2015.

[10] C. of San Hose, *Crosswalk enhancement comparison study*, http://www.lightguardsystems.com/wp-content/uploads/2014/11/Overhead-vs-IRWL-Study-Report-San-Jose.pdf, 7.

[11] L. Systems, "2 Ln Rd W-MSRP brochure,"

[12] T. of Lauderdale-By-The-Sea, *Agenda item request form*, http://lauderdalebythesea-fl.gov/Agendas_minutes/backup8-25-10/a15b.pdf, 2010.

[13] MetroCount, *Portable traffic survey*, http://metrocount.com/products/portable-traffic-survey/.

[14] C. for Disease Control and Prevention, *CDC study finds annual cost of motor vehicle crashes exceeds $99 billion*, http://www.cdc.gov/media/pressrel/2010/r100825.htm, 2010.

[15] INRIX, *Americans will waste $2.8 trillion on traffic by 2030 if gridlock persists*, http://www.inrix.com/press/americans-will-waste-2-8-trillion-on-traffic-by-2030-if-gridlock-persists/, 2014.

[16] A. International, *Standard specifications for extended life type, nonplowable, raised retroreflective pavement markers*, http://www.astm.org/Standards/D4280.htm, 2012.

[17] F. H. Administration, *Lane width*, http://safety.fhwa.dot.gov/geometric/pubs/mitigationstrategies/chapter3/3_lanewidth.cfm, 2014.

[18] W. Post, *Winter begins today: Five questions and answers about the solstice*, http://www.washingtonpost.com/blogs/capital-weather-gang/wp/2014/12/21/winter-begins-today-five-questions-and-answers-about-the-solstice/, 2014.

[19] N. E. M. C. of Excellence, *Energy harvesting from wireless sensor networks*, http://www.empf.org/empfasis/2011/January11/energy.html, 2011.

[20] I. Buchmann, *Charging at high and low temperatures*, http://batteryuniversity.com/learn/article/charging_at_high_and_low_temperatures.

[21] R. S. Roll, *Textured silicone rubber*, http://www.rubbersheetroll.com/silicone-rubber-textured.html, 2015.

[22] C. Lasance, *The thermal conductivity of rubbers/elastomers*, http://www.electronics-cooling.com/2001/11/the-thermal-conductivity-of-rubbers-elastomers/, 2001.

[23] *The anatomy of a protected LiIon battery*, http://www.lygte-info.dk/info/battery%20protection%20UK.html.

[24] A. Composites, *Nickel zinc (NiZn) batteries are a great cost-effective choice for many applications.* http://www.automotivecomposites.com/batteries.htm.

[25] S. Microelectronics, *Application note: Lithium-ion solar battery charger*, http://www.st.com/st-web-ui/static/active/en/resource/technical/document/application_note/DM00048561.pdf.