Engineering Design and Development

The Corner Cutters

Project Tiresias: A Blind Intersection Advance Warning Device

Lucas LaValva, Bridget Letson, Matthew Rosivack, Scott Wang

January 12th, 2018

Ms. Grunthaner and Dr. Eng

Project Tiresias

Table of Contents

Section #	Section Description	Page #
3	**Problem Statement**	3
	3.1 Problem Statement	3
	3.2 Justifications	3
4	**Design Brief**	5
5	**Project Management**	6
	5.1 Project Organization Chart	6
	5.2 Gantt Chart	7
6	**Current and Past Products**	8
	6.1 Patent Abstracts	8
	6.2 Analysis of Past Solutions	11
7	**Proposed Alternatives**	12
	7.1 Sketches of Proposed Solutions	12
	7.2 Evaluation of Proposed Solutions	18
	7.3 Selected Solution	18
8	**Prototype Planning and Construction**	20
	8.1 Mockup Model Construction	20
	8.2 Prototype Construction Plan	21

	8.3 Prototype Construction	21
	8.4: Technical Drawings, Electrical Schematics and Software Flowcharts	26
	8.5 Materials, Quantity, Suppliers and Costs	29
9	**Testing**	30
	9.1 Testing Procedure	30
	9.2 Testing Results	30
	9.3 Statiscal Analysis of Results	34
	9.4 Additional Comments	34
10	**Conclusion**	35
11	**Redesign**	36
12	**Acknowledgements**	37
13	**Appendices**	38
	13.1 Prototype Operation	38
	13.2 Source Code	39
	13.3 References	57
	13.4 Testing Locations	62
14	**Powerpoint**	63+

3: Problem Statement

3.1: Problem Statement

59,000 crashes in 2008 occurred at intersections with obstructed views of the road.

3.2: Justification

A report from the Nation Highway Traffic Safety Association (NHTSA) said that in total 787,236 crashes in 2008 occurred at intersections, of these 15.9% (125,022) occurred at uncontrolled intersections (Choi 2010) and 59,000 occurred at intersections with obstructed views (Choi 2010-). An additional 5.5% of these accidents at uncontrolled intersections happened because of a driver's misjudgment of gap or other's speed (Choi 2010). These accidents are highly frequent, having the highest relative ratio index, as seen below in the table, even when compared with distracted and illegal driving maneuvers (Choi 2010).

Table 1: Relative ratio of driver-attributed critical reason in intersection-related versus non-intersection-related crashes

Critical Reason	Relative ratio	Relative ratio index
Turned With Obstructed View	335.0	1
Inadequate Surveillance	6.1	2
Illegal Maneuver	4.1	3
False Assumption of Other's Action	3.8	4
Misjudgment of Gap or Other's Speed	3.1	5
Inattention	0.9	6
External Distraction	0.4	7
Internal Distraction	0.4	8
Critical Non-Performance Error	0.1	9
Performance Error	0.0	10
Too Fast for Conditions or Aggressive Driving	0.0	11

Data Source: NMVCCS 2005-2007

Figure 3.2.1: Screencap of NHTSA Statistics on Reasons for Crashes at Intersections, Crashes with Obstructed Views is Highlighted

In 2016, there were a total of 571 fatal crashes in New Jersey, resulting in 603 deaths (NJDOT 2017). According to the New Jersey Department of Transportation: Bureau of Safety Programs 2017 report, 8749 accident happened at uncontrolled intersections.The damages caused by the crashes can be seen in the figure below. Thirty-seven of the accidents were fatal.

Comparing this number to the total number of accidents in Monmouth County, there was only a difference of 4,000 accidents (NJDOT 2017).

ARDSMRPT10

New Jersey Department of Transportation
Bureau of Safety Programs
Crash Summary

August 9, 2017

Total Crashes At Unsignalized Intersections For State System Roads
(Excluding Toll Roads and Interstates)

For calendar year 2016

Total Crashes: 8749

Severity	Count	% of Total
Fatal	37	0.42%
Injury - Maj	30	0.34%
Injury - Mod	388	4.43%
Injury - Min	1880	21.49%
Property Damage	6414	73.31%

Intersection	Count	% of Total
At Signalized	0	0.00%
At Unsignalized	8749	100.00%
Between Intersections	0	0.00%
Railroad Crossing	0	0.00%

Figure 3.2.2: Accidents at NJ Unsignalized Intersections

These statistics show a clear need for a device that provides a warning to drivers about incoming traffic at uncontrolled intersections.

4: Design Brief

Problem Statement 59,000 crashes in 2008 occurred at intersections with obstructed views of the road.

Design Statement Design, build, and test an advance warning system that notifies if it is unsafe to exit or enter a blind intersection.

Client Municipalities in which a substantial number of intersection with obstructed views or areas with a high rate of accidents

Constraints

- Less expensive than current solutions ($500)
- Must be visible to, but not distract the driver
- Must fit all local, state, and federal traffic laws and regulations
- Must withstand wild animals, a variety of weather conditions, and stresses of the outdoors
- Must have safety measures against tampering and vandalism
- Near instantaneous response time
- Components must communicate wirelessly and automatically
- Backup power source in case of power outage

5: Project Management

5.1: Project Organization Chart

Table 5.1.1: Division of Responsibilities Between Team Members

Name	**Responsibilities**
Lucas	Construction of prototype, design, project coding and electrical design, purchasing parts
Bridget	Statistics, research, other documentation, outreach for testing, construction of prototype
Matt	Graphic designer, mock-up construction, construction of prototype
Scott	Project coordinator and designer, project coding and electrical design, construction of prototype, purchasing parts

5.2: Gantt Chart

ID	Task Name	Start	Finish
1	Design Brief	Fri 10/13/17	Fri 10/13/17
2	Brainstorming 3 Sketches Per Person	Fri 10/13/17	Wed 10/18/17
3	Decision Matrix	Wed 10/18/17	Fri 10/20/17
4	Work On Mockup Model	Fri 10/20/17	Fri 10/27/17
5	Submit Mockup Model	Fri 10/27/17	Fri 10/27/17
6	Design Review By Faculty	Fri 11/3/17	Fri 11/3/17
7	Design and Build Prototype	Fri 11/3/17	Fri 12/1/17
8	Testing of Prototype	Fri 12/1/17	Fri 12/22/17
9	Poster	Fri 12/22/17	Wed 1/3/18
10	Engineering Report	Fri 12/22/17	Fri 1/12/18
11	PowerPoint	Fri 12/22/17	Fri 1/19/18
12	Presentation and Demonstartion	Fri 1/19/18	Fri 1/19/18

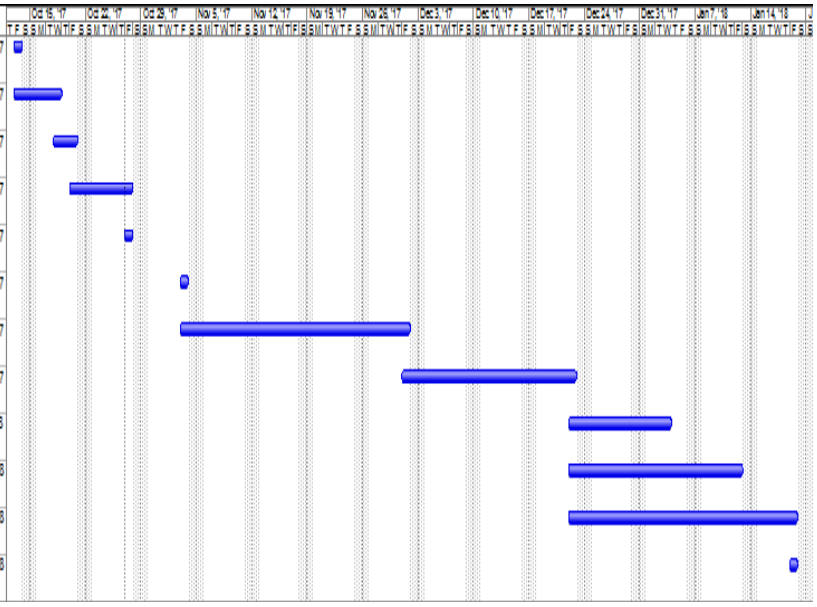


Figure 5.2.1: Gantt Chart

6: Current and Past Solutions

6.1: Patent Abstracts

Inventor Manfred Gillner wrote (1990):

> A traffic mirror comprises a spherically curved glass pane having a front surface, a rear surface, and an outer periphery, a metal reflective layer located on the inner surface of the glass pane, and a decorative frame located directly on the glass pane. The decorative frame is an enamel-like colored material which is baked onto the outer periphery of the glass pane. The traffic mirror further comprises a supporting structure for mounting the traffic mirror onto a vertical post. The supporting structure comprises at least two supporting struts which, are attached directly to the glass pane, for example, by bolts passing through bores in the glass pane itself. The glass pane is made from thermally prestressed single pane safety glass. The traffic mirror does away with the bulky frames used in prior art mirrors of this kind.

In his abstract Schondorf (2008) stated:

> A system, apparatus and method to automatically move a side mounted mirror on a vehicle from an inward position to an outward position in response to a signal from a detector indicative of an object outside an operator normal field of vision. The detector may be a blind spot detection system or a lane departure system, or any other system to detect vehicles in a blind spot. The system, apparatus and method further includes permitting an operator to manually actuate the movement of mirror on a vehicle to determine whether there are any objects in the operator's

normal field of vision. The systems and method may optionally include hysteresis during mirror movement."

6.2: Analysis of Past Solutions

Traffic mirrors are the most common solution to the problem that municipalities face with blind intersections. Unfortunately, they are quite expensive, as a thirty-two inch mirror can cost up to $490 (Cost 2017). While they do provide a clear view of the road on most days, the problems with these mirrors in adverse conditions are trifold. It may be harder to see an approaching car through the mirrors, or even the mirrors themselves, especially on days with poor weather conditions like rain or snow. In addition, the mirrors need to be constantly maintained to keep them free from dirt or frost, and they need to be replaced if the reflective surface is damaged (South Africa 2012). But the greatest weakness of traffic mirrors is that they still rely on human eyesight to function – drivers have to judge the distance and speed of an oncoming car from the distorted image of a convex mirror (South Africa 2012) using their own vision.

The combination of a road-mounted lidar (laser-rangefinder) and industrial-grade LED-signal makes a more viable solution to this problem. The lidar serves as the "eyes" of the system, while the LED signal notifies the driver of an oncoming car. The lidar measures the distance to the car by measuring the time a laser beam takes (Garmin 2016) to reflect off the approaching car. The main benefit of having separate mounts for the lidar and LED signal is that the sensor can be mounted out of the driver's line of sight, potentially hundreds of meters down the road. When multiple pulses are emitted in a short period of time, it is possible to accurately

gauge both the car's velocity and distance, something that cannot be done with the human eye. Meanwhile, on the signaling end, it is important to guarantee that the driver at the blind intersection knows about the oncoming car. This can be accomplished by something as distracting as a bright LED signal.

An alternative type of solution is an in-car blind spot detection, which can come in two types. One is a built-in blind spot detection system, which exists in newer case. Another type is a self-contained module that interfaces with sensors along the road. In both cases, these devices work by alerting the driver through a series of beeps or flashing LEDs (Schondorf 2008).

However, these in-car devices have their own set of problems. Built-in blind spot detection systems often come only in premium packages when buying a new car. In addition, these systems are better suited to warning the driver of emergencies such as hitting pedestrians at a crosswalk, rear-ending stopped cars in traffic, or touching a wall in a parking garage, not scanning along a blind corner for approaching cars (BLINDSPOT PROTECTION n.d.). While a possible workaround is to modify these systems to interface with road-based sensors, the variety of computer software systems on the market makes this feasible. Another concern is that in-car detection systems aren't as effective as intended. According to the Insurance Institute for Highway Safety (IIHS), people who do have access to these safety systems may, in fact, overlook the warnings that it gives off, even ignoring their mirrors where the feedback is often given. For a variety of reason, most people will not be able to benefit from the full potential of in-car blind spot detection systems. An accessible device that is compatible with all vehicles is needed in order to make the roads safer for all.

From an engineering standpoint, it is clear that the logistics and design requirements both favor a road-mounted signaling system, which can work for anyone on the road and does not suffer from the drawbacks of either traffic mirrors or in-car systems.

7: Proposed Alternatives

7.1: Sketches of Proposed Solutions

Bridget's Sketches

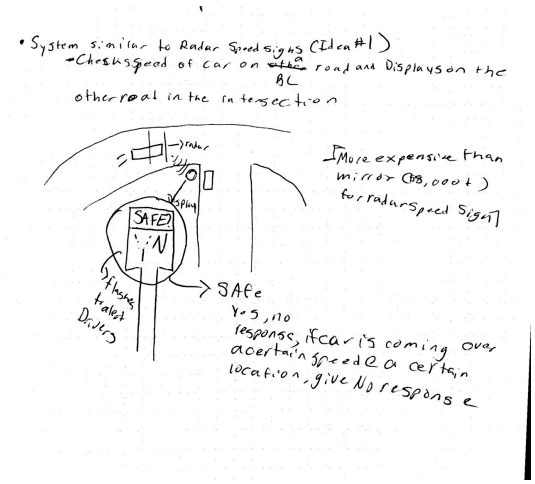


Figure 7.1.1: Bridget's Brainstorming Sketches 1

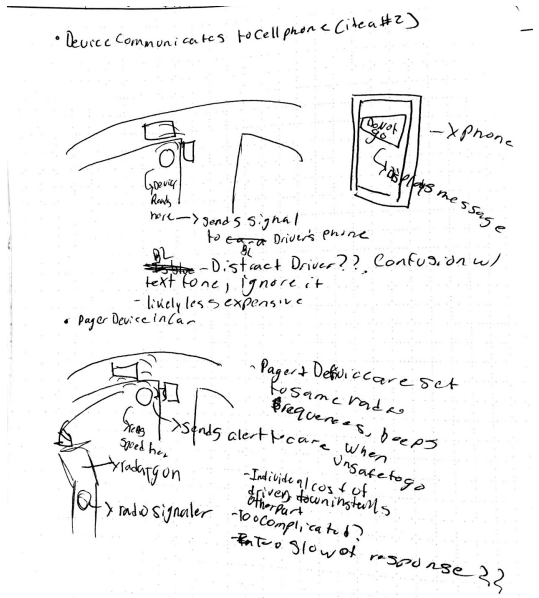


Figure 7.1.2: Bridget's Brainstorming Sketches 2

Description of Bridget's Sketches

The first sketch is of a an LED street sign. This device would communicated with some kind of speed gun and tell the driver if it is safe to enter the intersection by returning a "Y" for yes or an "N" for no. The second idea was a wireless communication with a radar gun and cell phone through some kind of application. The app would communicated with the speed gun and alert the driver if they are going to enter an unsafe intersection, The third was a pager that would be housed in the car and would send out an alert if the intersection was unsafe, only to cars within the radar gun range.

Matthew's Sketches

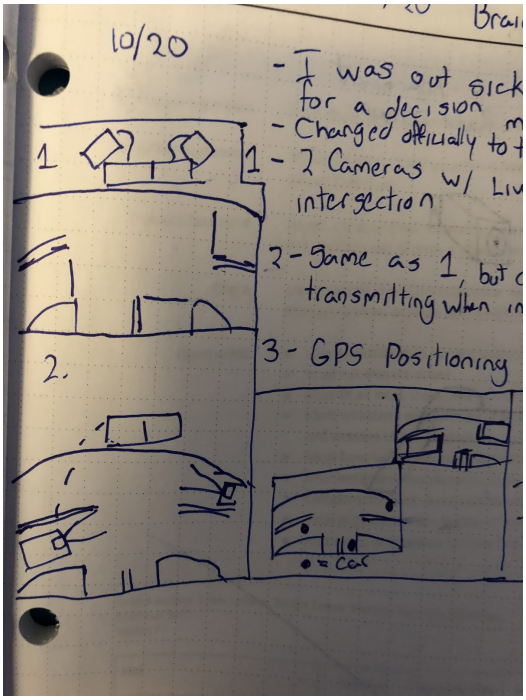


Figure 7.1.3: Matthew's Brainstorming Sketches

Description of Matthew's Sketches

Drawing 1 is a design utilizing two cameras attached to a display across from the driver. These displays display a live video feed transmitted from the camera. These show the driver a clear video of what is happening around the corner. This is similar to existing mirrors, but utilizes cameras to zoom in and show a larger picture of what is around the corner. Drawing 2 involves a similar design to drawing 1 but has cameras mounted on cars, which use bluetooth to transmit their video feed to the display screens. This allows other drivers to see if a car is near by the presence of video, implying there is a car in range. It also lets you see what they see to show how far they are and allowing the driver to determine whether it is safe to go. Drawing 3 involves utilizing GPS features inside of cars to determine the position of cars. There is a video display with an overview map, showing the intersection and surrounding bends. Then, if a car's GPS position is on one of those roads, it will put a dot on the map showing there is a car. This would show the driver the location of all other cars and they can see how fast the car's dot is moving which would allow them to know all their surroundings no matter how far away from the intersection they are.

Luke's Sketches

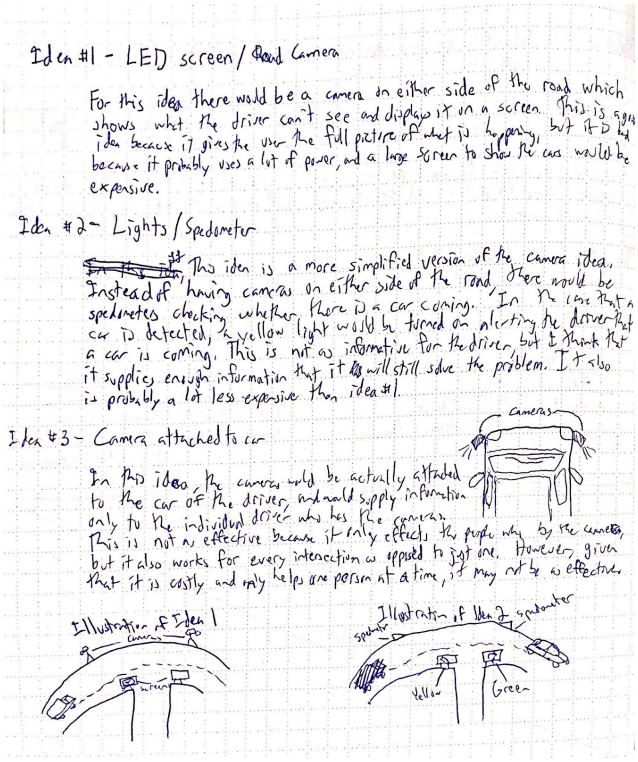


Figure 7.1.4: Luke's Brainstorming Sketches

Description of Luke's Sketches

Idea one is centralized around the idea that there would be a camera on either side of the road, filming cars that are passing by. There would be a large TV screen on either side of the road showing people who are stopped at the intersection who is on either side. This design is inefficient because it is much more expensive than many of the other designs, and it would require a constant power source supplying a large amount of power.

Idea two is similar to idea one, except instead of cameras and TV screens it uses speedometers and LED lights. In this design both sides of the road would be tested for cars and two lights would display whether it was safe to turn left or right. This is the closest to the actual design we used, but it is not perfect.

The third idea places cameras on the cars instead of putting them on the street. There would be a small screen inside the car that displayed the input of the cameras. This design is useful because it can be used at every intersection, but it also must be purchased by each individual user instead of being inserted while the roads are being made. This design would also have to be very weatherproof, as it would be attached to the car at all times.

Scott's Sketches

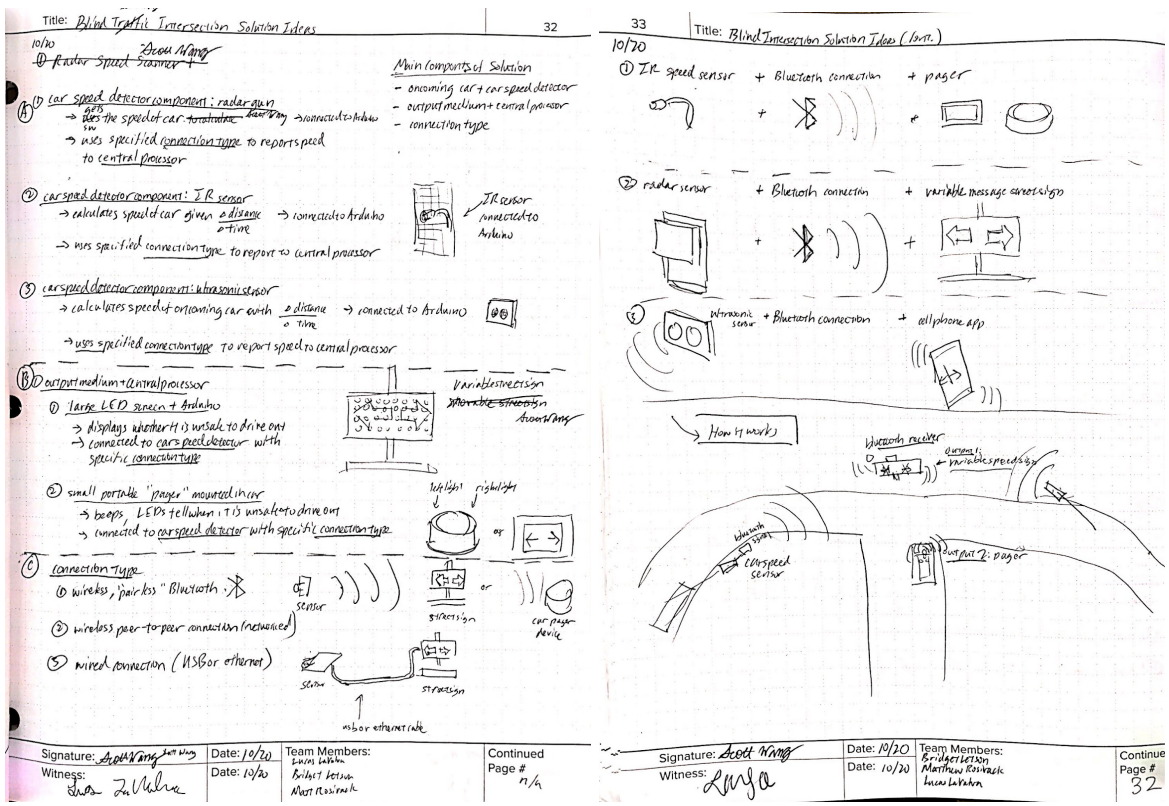


Figure 7.1.5: Scott's Brainstorming Sketches

Description of Scott's Drawings

There are three distinct solutions above. The first idea is a combination of an IR sensor and an in-car pager device that can be mounted on the dashboard of a car, connected to each other by a Bluetooth connection. The street-mounted IR sensor would read the speed of the

oncoming car to and report it to the pager device through a Bluetooth connection, and the pager would beep and light up, notifying the driver at the blind section that a car is approaching.

The second idea is a combination of a Doppler radar sensor and a variable-message street sign, connected to each other by a Bluetooth connection. The street-mounted radar sensor would read the speed of the oncoming car to and report it to the street sign through a Bluetooth connection, and the LEDs on the street sign would light up as an arrow in the corresponding direction of the car, notifying the driver at the blind section that a car is approaching.

The third idea is a combination of a street-mounted ultrasonic sensor and a smartphone app, connected via a Bluetooth connection. The ultrasonic sensor would read the speed of the oncoming car to and report it to the app through a Bluetooth connection, and the phone will beep, notifying the driver at the blind section that another car is approaching.

However, the above solutions were designed to be modular; they can easily swap out different types of components, such as the type of speed sensor, the method of wireless communication between the two devices, and the method of feedback should complications arise from one specific combination of components.

7.2: Decision Matrix

Idea	Low Cost	Response Time	Legality	Durability	Ease of Building	Ability to Inform Driver of Danger	Total (out of 50)
Existing Solution (Convex Mirror)	4	10	10	9	9	4	46
Camera/LED screen	2	9	1	4	6	7	29
Radar sensor/Yellow arrows on variable message street sign	8	9	8	7	6	9	47
Cameras mounted on car hood	2	9	10	3	4	5	33
Radar Speed Signs	1	8	10	6	3	8	36
Pager/Radar Combo	4	7	8	6	5	9	39
Radar speed sensor + smartphone app + Bluetooth	6	9	6	6	6	7	40
IR speed sensor + pager device + Bluetooth	7	8	6	5	6	8	40
ultrasonic speed sensor + pager device + Bluetooth	6	7	7	6	6	7	39
Cameras pointed in both directions + Live video feed on TV screen	3	10	2	6	9	5	35
Previous idea but cameras are on car hood, trigger video feed via bluetooth	4	10	2	2	9	5	32
GPS positioning on TV screen, shows location of cars at the intersection.	6	10	5	8	8	3	40

(Key: 1 - Lowest Score, 10 - Highest Score)

Figure 7.2.1: Decision Matrix with Team Scores

7.3 Selected Solution

The design that was selected was a radar sensor that communicated with a sign with two arrows. The arrows would turn on on either side depending on the direction of the concoming car. The driver would see the sign as the came to an intersection. The biggest issues with most of the designs was cost, as items like cameras andpagers would be expensive to install. Another issue was durability; dash cameras could easily break if the car that they are mounted in is involved in a crash. The sixth criteria was the ability to inform the driver of danger. In the case, the device being a possible distraction to the driver was a major concern with designs like the

cell phone alert app. In the end, the valuation to the radar sensor and sign idea was the most practical to build and had the most potential to be more effective than the existing solution of traffic mirrors.

8: Prototype Planning and Construction

8.1 Mockup Construction

Figure 8.1.1: Mock up Model

While building the mock up model, it was determined that the large screen, with rows of LEDs forming two X's would be too difficult to construct. Therefore, the prototype would be constructed with a single LED on a much smaller screen that would light up when a car was detected; this would also make it easier for the screen to stand on its on. It was also found that making two elongated boxes to support the radar and the LED screen would be ineffective and instead decided to use wood beams to construct the prototype.

8.2: Prototype Construction Plan

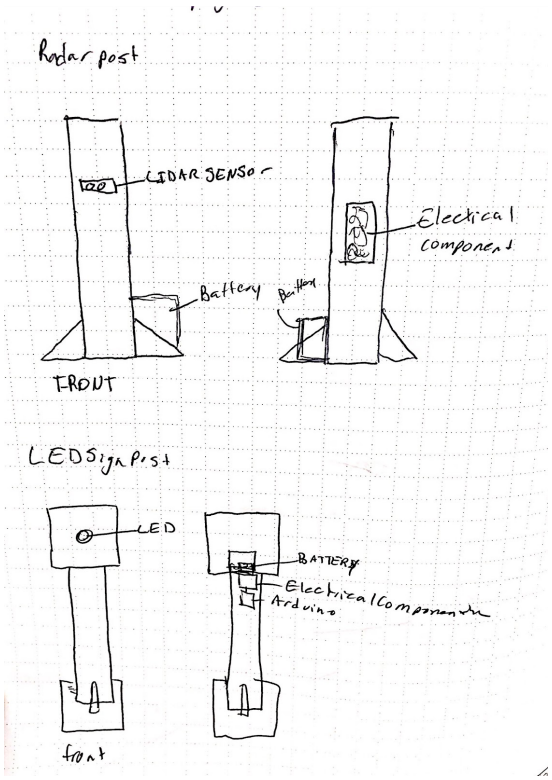


Figure 8.2.1: Sketch of Planned Construction

8.3: Prototype Construction

NOTE: These instructions are for the final prototype construction-

1. Obtain all materials (see materials list for details).
2. If needed, measure and cut wood to dimensions listed.
3. Using provided schematics as a guide, put the electronic components together.
    a. Code for these components are provided.
4. Using screws and a drill, attach each support beam to the center of its base.

5. Drill the triangular supports to the wider sides of the beam, also drill into base.
6. Assemble two boxes of the same size.
    a. One of the boxes must have two holes 2 ½ inches from the top of the box, 1ich apart and 3/4 inches in diameter
    b. Also include two smaller holes (about ⅛ diameter) in the top of the other box, so that the LED's wiring can go through
7. Attach boxes using drill and 2" screws
    a. Top of the lidar sensor box is ¾ of the beam length away from the base on the radar post
    b. Top of the warning light box is level with top of beam on LED post
8. Mount the electronic components to the box
    a. Attach lidar sensor to the back of the box's lid, make sure it is lined up with the holes
    b. Mount Arduino and bluetooth modules to the back of the box.
    c. Connect LED to wiring and drill it down
9. Attach 6V battery to lidar sensor electronics, place in box
10. Attach 9V battery to LED Arduino, place in box
    a. Attach 12V battery to LED
11. Stick adhesive velcro strips onto the open edge of the box
    a. Close box
12. Run tests to make sure assembly is functional
    a. Check for errors in code/electronics

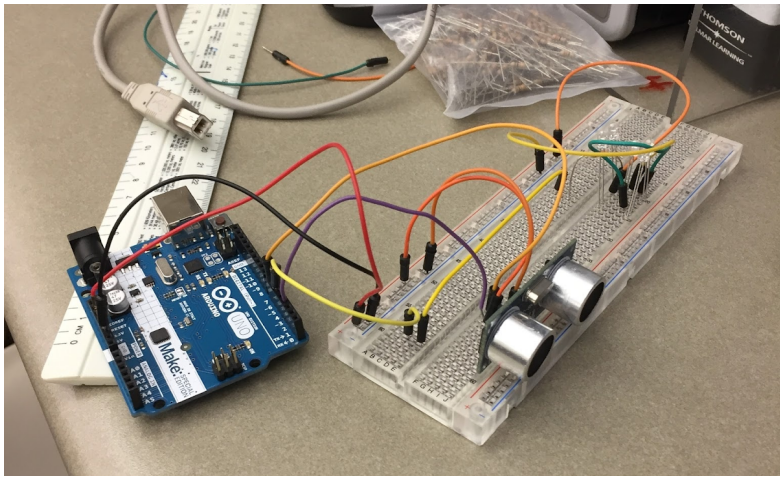


Figure 8.3.1: Construction and Testing of Electrical Components

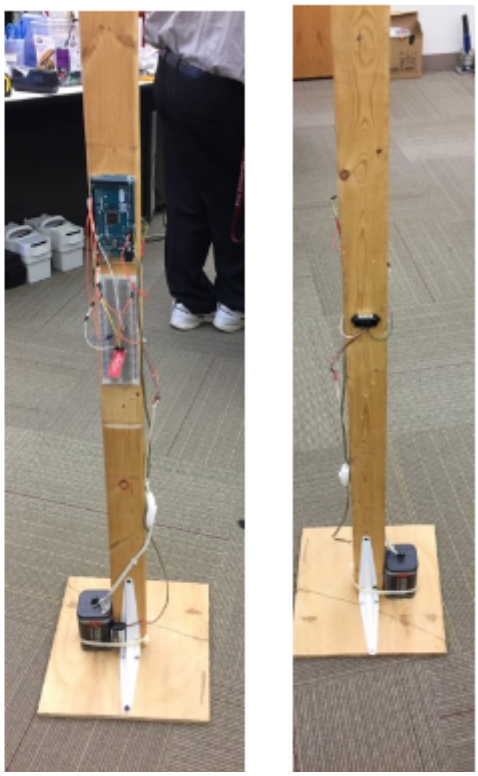


Figure 8.3.3: Front and back view of speed gun component of prototype

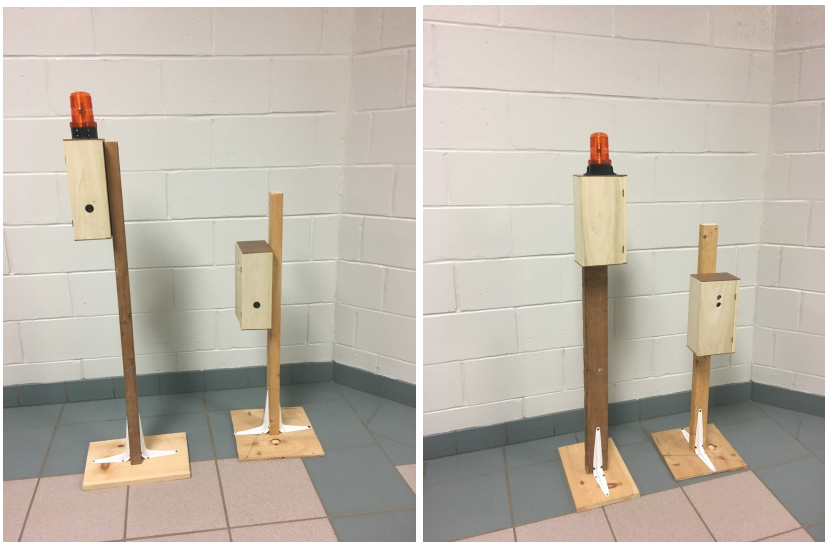


Figure 8.3.2: Side and front view of final prototype

Figure 8.3.4: Inside view of final prototype

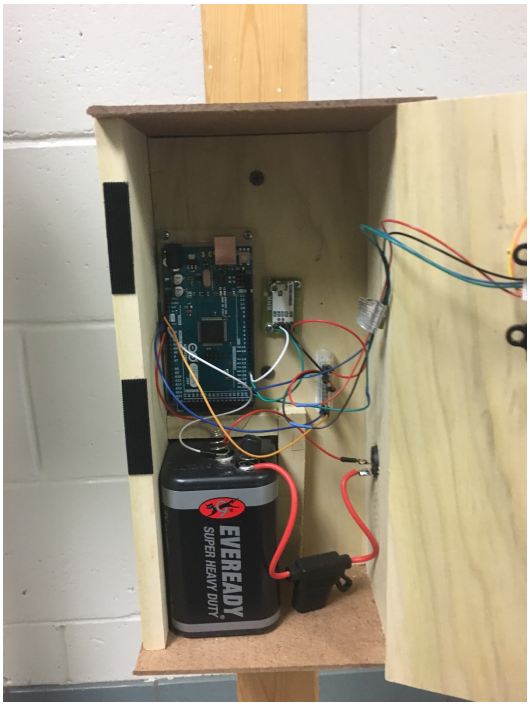


Figure 8.3.5: Detail view of radar gun electronics

Figure 8.3.6: Detail view of LED sign electronics

8.4: Technical Drawings, Electrical Schematics and Software Flowcharts

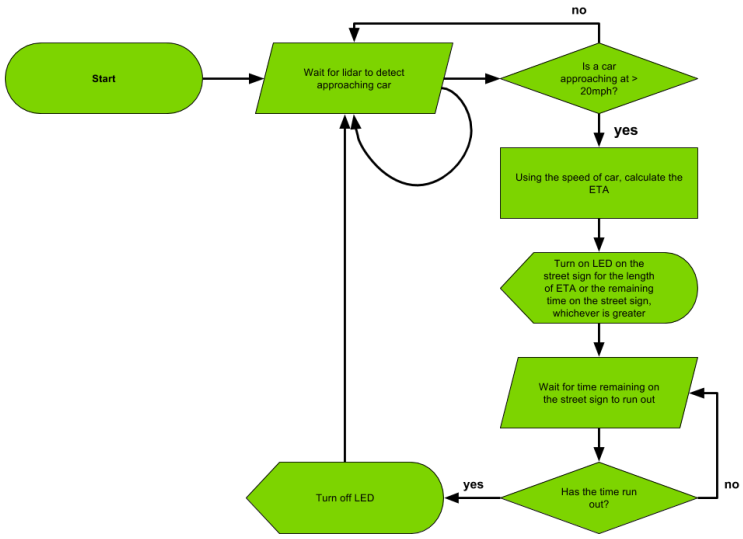


Figure 8.4.1: Flowchart of software

This code will be written with the built-in Arduino programming language. When the programs on both devices are initiated, the lidar sensor will turn on and start continuously searching for moving objects. When a car is detected, a sensor will be used to calculate the speed of an oncoming car. If this car is approaching at greater than 20 mph, the speed of the car will be wirelessly reported to the street signal and evaluated to determine the estimated arrival time of the car. If the arrival time is greater than the time remaining on the street signal's timer, then the arrival time is added to the time remaining that the LED will stay on. Once the timer has finished, the light will turn off.

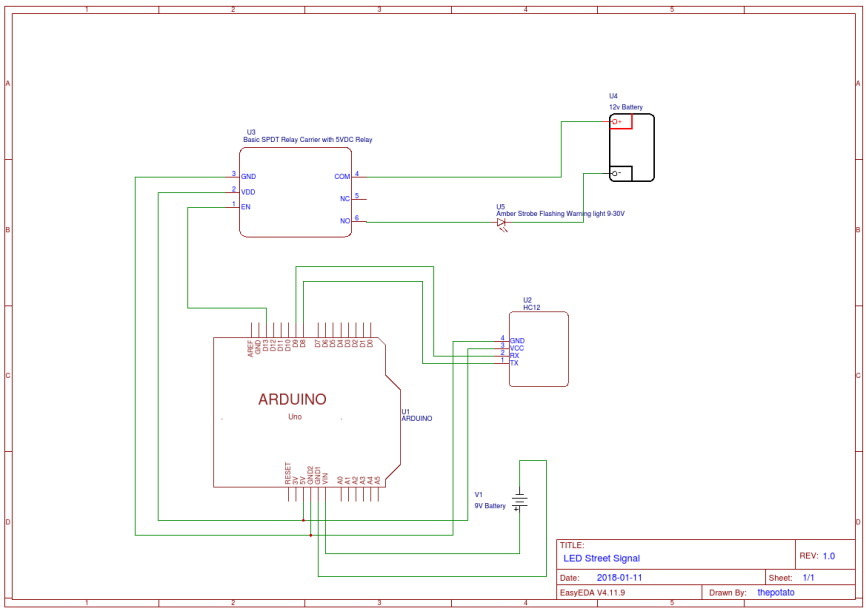


Figure 8.4.2: Schematic of LED screen electronics

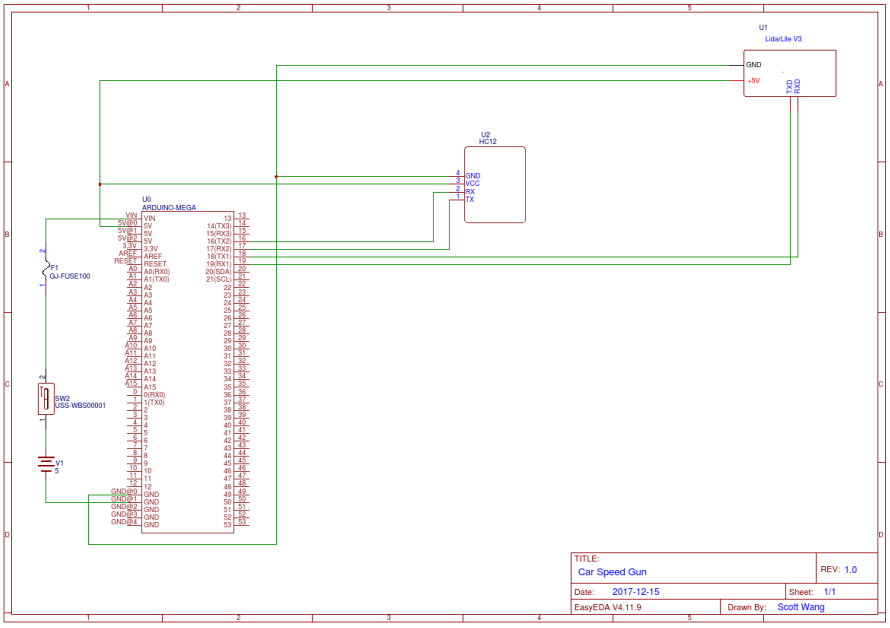
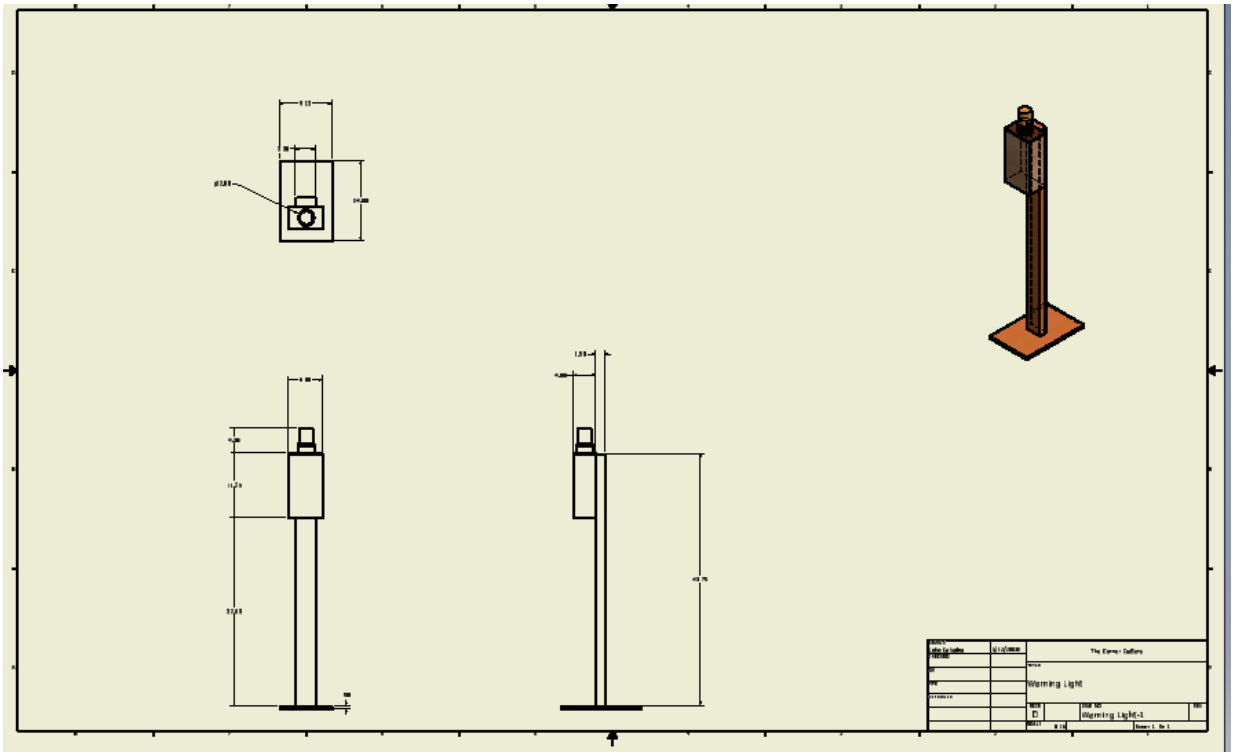
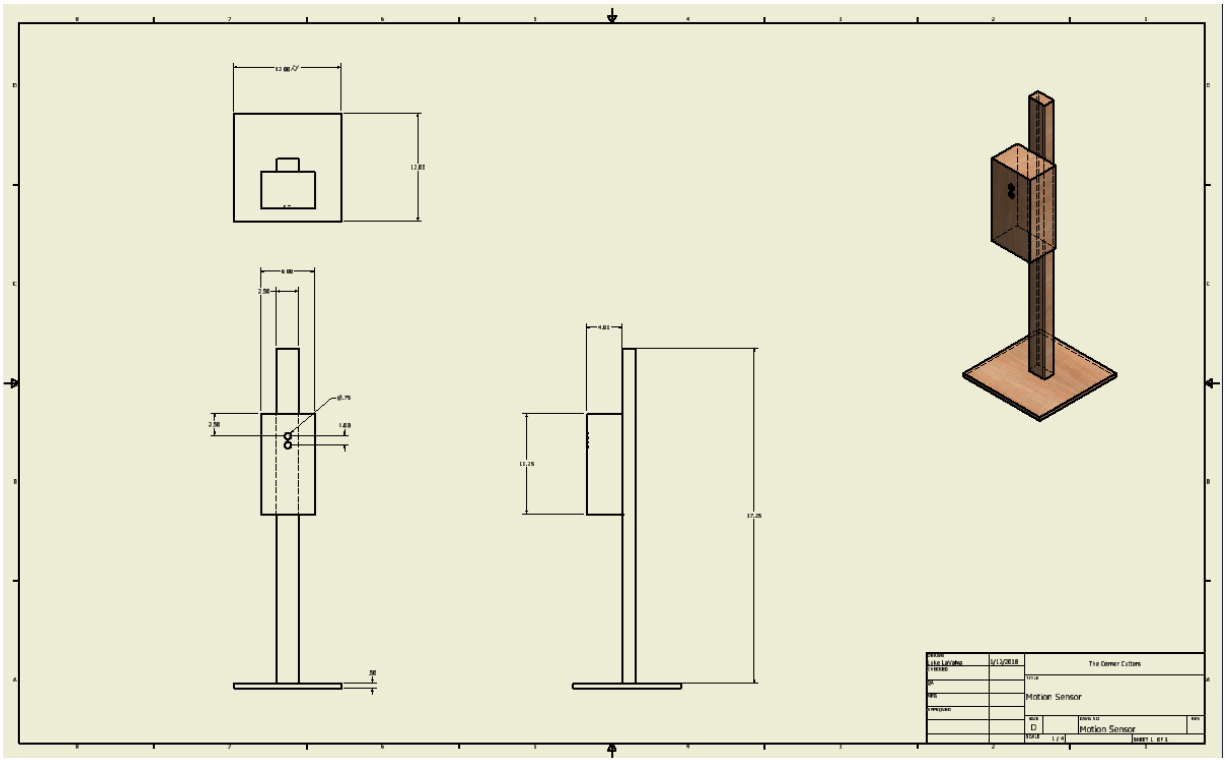


Figure 8.4.4: Schematic of radar gun electronics



8.4.5: Inventor drawing of LED post with dimensions



8.4.6: Inventor drawing of radar gun post with dimensions

8.5 Materials, Quantity, Suppliers and Cost

Table 8.5.1: Materials List

Material	**Quantity**	**Supplier**	**Price**
LidarLite V3 SENSORS	1X	LidarLite – RobotShop	$119.67
ARDUINO Mega	1X	Amazon	$45.95
Arduino Uno	1X	HTHS Tech Lab	–
WOODEN BASES, 12” x 12” AND 14” x 6 ½”	2X	HTHS TECH LAB	–
Solderless Breadboard	1X	HTHS TECH LAB	–
PCB Breadboard	1X	HTHS TECH LAB	–
Solder	1X	HTHS TECH LAB	–
WOODEN POST, 1” x 4” x 43.25”	1X	Scott’s Basement	–
WOODEN POST, 1” x 4” x 37.25”	1X	Scott’s Basement	–
Brackets	4X	Scott’s Basement	–
LARGE LED Signal	1X	GPPOWER – Amazon	$13.99
WOOD FOR BOX CONSTRUCTION, 10” x 8”	8X	Home Depot	$ 7.34
SCREWS	24X	HTHS TECH LAB	–
Door Hinge Sets	2X	Lowe’s	$1.87
Jump Wire	use as seen fit	RM 185	–
Fuse, Fuse Holder	1X	Ficbox – Amazon	$10.69
Switches	2X	U.S. Solid – Amazon	$5.95
5V Relay Module	1X	Tolako – Amazon	$5.80
HC-12 Wireless Modules	2X	HiLetgo – Amazon	$7.99
12v Battery	1X	IEI Electronics	$24.33
6v Battery	1X	IEI Electronics	$4.99
9v Battery	1X	HTHS Tech Lab	–
Total			**$248.54**

9: Testing

9.1 Testing Procedure

1. Set up device according to operation instructions.
2. Switch on both devices.
3. Drive a car past the sensor towards the blind intersection at a constant 30 mph.
4. Observe and record whether the street sign successfully turns on the moment the car passes the car speed sensor, and turns off at a reasonable time after the car passes (around the same time it takes to arrive at the blind intersection)
5. Repeat steps 2-3 three more times.
6. Repeat step 10 for 35 mph and 40 mph.

9.2 Testing Results and Photos

Figure 9.2.1: Testing of prototype (1)

Figure 9.2.2: Testing of prototype (2)

Figure 9.2.3: Testing of prototype (3)

Figure 9.2.4: Testing of prototype (4)

Table 9.2.1: Raw Data Table of Testing Results

Trial	Result	Speed
1	Failure	30
2	Failure	30
3	Failure	35
4	Failure	35
5	Failure	40
6	Failure	40
7	Success	30
8	Success	30
9	Success	35
10	Success	35
11	Success	40
12	Success	40

Table 9.2.2: Processed Data of Test Results

Mean	6
Variance	3
Standard Deviation	1.73
Wald CI Test - Mean	0.7829

NOTE: For the processed data table successes and failures were counted (six successes and six failures). Values are in regard to the number of successful trials.

9.3 Statistical Analysis of Results

To measure the success rate of the device, the number of successful trials were counted and then were divided over the total number of trials; resulting in a success rate of 50%. An Adjusted Wald Confidence Interval Completion rate test was used to help determine the significance of the data. Results of the test are described as significant if they fall between 2.00 and 11.26 (Lane n.d.), since the value returned by the test was 0.7829, the data is not significant.Due to the circumstances of testing, the device successfully operated in 50% of the trails. However, this can be explained by the fact that the sensor was damaged during the first half of the testing. For the remaining 6 trials, the device experienced a 100% success rate with a replacement lidar.

9.4 Additional Comments

The devices were tested outdoors at 11˚ F. While the actual recording iPhone encountered problems in the cold, spontaneously freezing and shutting down, the devices proved to continue functioning.

In addition, the wireless transceivers, stated to have a range greater than one kilometer, were tested out at ranges greater than 100 m. This was ample range for the purposes of this device.

10: Conclusion

10.1 Discussion of Results

Although the device failed 50% of the testing trials, the lidar sensor had actually been broken during the tests. It was decided that the same trails would be repeated, just to make sure that the failures were due to the broken sensor. The second set of trials were all successes. The device met most of the criteria that were set forth in the design brief. It is cheaper than present solutions, costing only about $250 in total; this is less than half of the estimated cost for a high quality traffic mirror (Cost 2017). It follows local, but not state or federal laws; the prototype that was built was designed for local roads. The initial prototype was not weatherproof at all. Boxes were added to cover the electronics and the batteries. There were, however, no measures taken against tampering. The response time was very quick, under seconds, as can be seen in the prototype testing videos. The two components (the posts) do communicate wirelessly, but no backup power source was added. While not perfect, the prototype overall functioned very well.

11: Redesign

One of the biggest concerns with the prototype was the noisiness of the lidar. While the software-side of the system addressed this issue somewhat with methods of statistical analysis, the car speed gun gave false positives about once a minute, incorrectly predicting the presence of an oncoming car. Major effort should be made in the redesign to completely eliminate this issue, since false positives can potentially be dangerous as well.

Another concern is the protection of the delicate electronic components inside the devices. For redesign, more weatherproofing would be ideal. The current prototype is made of wood, but materials like plastic would better project the electrical components of the design. Along the same vein, a future redesign would require changing the velcro of the hinge doors with a secure lock that only authorized technicians are able to access. This would prevent any possibility of unauthorized people or wildlife, which occurs often in more rural areas, tampering with the operation of the device.

In addition, a more stable, permanent method of securing the post to the ground is preferred. Currently, superstorm-strength winds may blow over the devices, especially on sloping roads. Eventually, it would be best to make the posts are permanent roadside installations, so that they are further tamper and breakage proof.

A final concern is the question of a backup power source. Both devices are powered by large batteries that are expected to last days, if not weeks. While it is easy to swap the batteries, an alternative power source is necessary if they ever run dry. One possibility could be a direct electrical connection to municipal power lines; another possibility is using solar panels as a power source.

12: Acknowledgements

12.1 Aknowledgements

The design of a blind intersection warning system was not possible without the help a several individuals.

First, this project relied heavily on the resources of the HTHS tech lab, ranging from jumper wires to Arduino boards to slabs of wood.

Second, this project would not be possible without Scott's father, who spent long nights helping Scott solder together components and constructing the wooden cases for the two stands.

Finally, this project would not possible without the guidance and physical assistance of Ms. Grunthaner and Dr. Eng.

13: Appendices

13.1 Product Operation

1. Place the traffic signal at the blind intersection and the car speed sensor within 300m along the road into the direction of oncoming traffic.
2. Measure the distance between the two devices. Measure the angle between the direction of the lidar sensor and the length of the road. It is advisable that the distance is between 50m and 300m and that the angle is between 5˚ and 25˚.
3. Open the trapdoor on the car speed sensor and hook up the Arduino inside to a computer.
4. Open the Arduino IDE and open an new serial monitor.
5. Save the angle between the direction of the lidar sensor and the length of the road into the Arduino's EEPROM memory by typing "<aX>" where X is in degrees.
6. Save the distance between the car speed gun and the street signal into the Arduino's EEPROM memory by typing "<dX>" where X is in centimeters and rounded to the nearest meter.

13.2 Source Code

Street Signal Code

EDD_Car_Speed_Gun_2_finalfinal

```
#include <Wire.h>
#include <LIDARLite.h>
#include <SoftwareSerial.h>

// Using supplied cable:
// - Black = GND (connected to GND)
// - Red = 5V (4.5 - 6.0V) (connected to Vin on Teensy 3.5, or 5V on Arduino)
// - White = TFmini RX (aka. connect to microcontroller TX, pin1 on Teensy 3.5)
// - Green = TFmini TX (aka. connect to microcontroller RX, pin0 on Teensy 3.5)

// NOTE: for this sketch you need a microcontroller with additional
// serial ports beyond the one connected to the USB cable
// This includes Arduino MEGA (use Serial1), Teensy (3.x)
// (use one of the available HW Serial connections)

boolean debugDist = false;

boolean debugCurrentVelocity = false;

boolean debug2 = false;

boolean debug3 = false;

// debugs velocity
boolean debugVelocity = false;

boolean debugVelocities = false;

int distance = 0;
int newDist = 0;
int oldDist = 0;
int deltaDist = 0;
float startTime = 0;
```

Figure 13.2.1: Speed Gun Code

```
35 unsigned long oldTime = 0;
36 unsigned long deltaTime = 0;
37 float deltaTimeSeconds = 0;
38
39 float currentVelocity = 0;
40 float filteredVelocity = 0;
41 int filteredVelocityInt = 0;
42
43 boolean firstFrame = true;
44
45
46 // Globals
47 LIDARLite lidarLite;
48 int cal_cnt = 0;
49
50
51 // Serial2 (17, 16) is for Wireless.
52
53 // 12 is the minimum number of data points :D
54 const int numVelocities = 3;
55 const int lastIndex = numVelocities - 1;
56 float velocities[numVelocities];
57
58 // tracking the indices of velocities
59 int indexNum = 0;
60 boolean onLastIndexOnce = false;
61 boolean onLastIndexMoreThanOnce = false;
62
63 const int numStrings = 400;
64 String velStrings[numStrings];
65
66
67 void readLidar(){
68     // At the beginning of every 100 readings,
```

Figure 13.2.2: Speed Gun Code 2

```
    // take a measurement with receiver bias correction
    if (cal_cnt == 0) {
        distance = lidarLite.distance();      // With bias correction
    } else {
        distance = lidarLite.distance(false); // Without bias correction
    }

    // Increment reading counter
    cal_cnt++;
    cal_cnt = cal_cnt % 100;

}

void getDistTimeVel() {
    // update distance
    readLidar();

    // you can only have a "previous distance"
    // after the 2nd iteration of the loop
    if (!firstFrame) {
        // get delta T from Tfinal - Tinitial
        oldDist = newDist;
    }
    newDist = distance;
    deltaDist = newDist - oldDist;

    // convert microseconds into seconds
    deltaTime = (micros() - oldTime);
    deltaTimeSeconds = deltaTime/1000000.0;

    // just to avoid dividing by 0
    deltaTimeSeconds = (float) max(deltaTimeSeconds, 0.00001);

    currentVelocity = deltaDist/deltaTimeSeconds;
```

Figure 13.2.3: Car Speed Gun Code 3

```
    if (debugCurrentVelocity) {
//      if (currentVelocity > 5000000) {
            String s = "cv:" + String((currentVelocity > 0) ?
            "positive" : "negative") +
            "\tdeltaD: " + String(deltaDist) + "\tdeltaT: " +
            String(deltaTimeSeconds);
            Serial.println(s);
//      }

    }
    if (debugDist) {
        Serial.println(distance);
    }

}

// add the current velocity to the new velocities buffer
void logVelocity() {
    // only log velocities if the car is approaching (v < 0)
    // use reasonable values
    if (currentVelocity < 5000 && currentVelocity > -4000) {
        if (indexNum != lastIndex) {
            velocities[indexNum] = currentVelocity;
        }

        if (indexNum == lastIndex) {
            onLastIndexOnce = true;
        }

        if (indexNum == lastIndex && onLastIndexOnce) {
            onLastIndexMoreThanOnce = true;
        }

```

Figure 13.2.4: Car Speed Gun Code 4

```
        if (onLastIndexMoreThanOnce) {
            // shift every velocity index to the left
            for (int i = 1; i < numVelocities; i++) {
                velocities[i - 1] = velocities[i];
            }

            velocities[lastIndex] = currentVelocity;
        }

        // update the frame status ONLY if a velocity was logged
        updateIndexStatus();
    }

}

float getMeanVelocity() {
    float mean = 0;
    for (int i = 0; i < numVelocities; i++) {
        mean += velocities[i];
    }
    mean = mean/10;
    return mean;
}

float getStdDev() {
    float x_bar = getMeanVelocity();

    float RSS = 0;

    for (int i = 0; i < numVelocities; i++) {
        RSS += pow(velocities[i] - x_bar, 2);
    }

    return pow(RSS/(numVelocities + 1), 0.5);
```

Figure 13.2.5: Car Speed Gun Code 5

```
}

// keep track of the current velocity index
void updateIndexStatus() {

    if (indexNum < numVelocities - 1) {
        indexNum += 1;
    }
}

float ttt = 0;
int nnn = 0;

void getFilteredVelocity() {
    float total = 0;
    float mean = getMeanVelocity();
    float stdev = getStdDev();
    int numbersUsed = 0;


    // i < numVelocities is acceptable because array initially looks like:
    // {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
    for (int i = 0; i < numVelocities; i++) {
        // only take the mean of the velocities that are within
        // 2 standard deviations from the
        if (abs(velocities[i] - mean) < 2 * stdev) {
            total += velocities[i];
            numbersUsed += 1;
        }
    }
    if (numbersUsed == 0) {
        numbersUsed = 1;
    }

```

Figure 13.2.6: Car Speed Gun Code 6

```
    ttt = total;
    nnn = numbersUsed;
    filteredVelocity = total / numbersUsed;
    filteredVelocityInt = (int) round(filteredVelocity);
}


unsigned int frames = 0;


void setup() {
    Serial.begin(9600); // Serial output through USB to computer

    // Set configuration to default and I2C to 400 kHz
    lidarLite.begin(0, true);
    // Change this number to try out alternate configurations
    lidarLite.configure(0);

    // begin wireless
    Serial2.begin(9600);

    for (int i = 0; i < numStrings; i++) {
        velStrings[i] = String(i * 10);
    }

    delay(100); // Give a little time for things to start


}


boolean debug = true;

void loop() {
```

Figure 13.2.7: Car Speed Gun Code 7

```
    readLidar();

    getDistTimeVel();

    logVelocity();

    getFilteredVelocity();


    firstFrame = false;

    oldTime = micros();

    if (debugVelocity) {
        Serial.println(filteredVelocityInt);
    }

    if (debugVelocities) {
        for (int i = 0; i < numVelocities; i++) {
            Serial.print(String(velocities[i]) + ", ");

        }
        Serial.println();
    }

    int rounded = abs(((int) filteredVelocity)/10);
    String reportedValue;
    String message = "<1>";
    // 0 to 399
    if (rounded > 0 && rounded < numStrings) {
        reportedValue = velStrings[rounded];
        message = "<" + reportedValue + ">";
    }
```

Figure 13.2.8: Car Speed Gun Code 8

```
    // if car travels between -20 and -70 mph
  if (-3129.28 < filteredVelocity && filteredVelocity < -1000) {

        Serial2.println(message);

    }
    Serial.println(message);

    ////////////
    //////////////////////////// Serial Debug Stuff
    // If stuff was typed in the serial monitor



    frames += 1;
    delay(10);
}
```

Figure 13.2.9: Car Speed Gun Code 9

Street Signal Code

Final_Sign_Processor

```
#include <EEPROM.h>
#include <SoftwareSerial.h>


boolean debugBT = true;
boolean debugProcess = false;
boolean debugTimes = false;

SoftwareSerial bluetooth(8, 9);


int ledPin = 13;

unsigned long oldTime = 0;
unsigned long deltaTime = 0;
float deltaTimeSeconds = 0;
float latestCarTime = 0;


float defaultAngle = 0.0;
float defaultDistance = 8000.0;

// 5000 cm = 50 m
float mountingDistance = 7500.0;
float angle = 12.0;

float readAngle = 0.0;
float readDistance = 0.0;

boolean safe = true;

// Variables used for incoming data


```

Figure 13.2.10: LED Street Signal Code 1

```
35 const byte maxDataLength = 20;          // maxDataLength is the maximum length
36 // allowed for received data.
37 char receivedChars[maxDataLength+1] ;
38 boolean newData = false;               // newData is used to determine if
39 // there is a new command
40
41
42 void setup() {
43     // put your setup code here, to run once:
44     Serial.begin(9600);
45     bluetooth.begin(9600);
46     delay(100);
47
48     readAngle = EEPROM.read(0);
49
50     // problem: for values > 128
51     // try converting to meters first
52     readDistance = EEPROM.read(1);
53
54     if (readAngle != readAngle) {
55         // NaN
56         Serial.println("NaN");
57     } else {
58         Serial.println("not NaN");
59         angle = readAngle;
60     }
61
62     if (readDistance != readDistance) {
63         Serial.println("NaN");
64     } else {
65         Serial.println("not NaN");
66         mountingDistance = readDistance * 100;
67     }
68
```

Figure 13.2.11: LED Street Signal Code 2

```
    pinMode(ledPin, OUTPUT);
    digitalWrite(ledPin, LOW);
}


void processCommand() {
    newData = false;

    String msg = String(receivedChars);
    // msg = msg.substring(1, msg.length() - 1);

    // convert to int
    float velocityPrime = (float) msg.toInt();

    // we need to divide by cos(angle)
    float newVelocity = velocityPrime/cos(angle * 3.14159/180.0);

    // assuming a mounting distance of 50 m
    float eta = mountingDistance/newVelocity;

    if (debugProcess) {
        Serial.print("message: ");
        Serial.println(msg);
        Serial.print("v\': ");
        Serial.println(velocityPrime);
        Serial.print("v: ");
        Serial.println(newVelocity);
        Serial.print("eta: ");
        Serial.println(eta);
    }

    if (eta > latestCarTime) {
        latestCarTime = eta;
    } else {
```

Figure 13.2.12: LED Street Signal Code 13

```
        latestCarTime = latestCarTime + 0.5;
    }


}

void recvWithStartEndMarkers2() {
    static boolean recvInProgress = false;
    static byte ndx = 0;
    char startMarker = '<';
    char endMarker = '>';

    if (bluetooth.available() > 0) {
        char rc = bluetooth.read();

        // if currently receiving
        if (recvInProgress == true) {
            if (rc != endMarker) {
                if (ndx < maxDataLength) {
                    receivedChars[ndx] = rc; ndx++;
                }
            } else {
                receivedChars[ndx] = '\0'; // terminate the string
                recvInProgress = false;
                ndx = 0;
                newData = true;
            }
        // if not already started receiving, and
        // the message begins with a start marker
        } else if (rc == startMarker) {
            recvInProgress = true;
        }
    }

```

Figure 13.2.13: LED Street Signal Code 14

```
}

void recvWithStartEndMarkers() {
    static boolean recvInProgress = false;
    static byte ndx = 0;
    char startMarker = '<';
    char endMarker = '>';


    while (bluetooth.available() > 0) {
        char rc = bluetooth.read();

        if (debugBT) {
            Serial.print(rc);
        }

        // if currently receiving
        if (recvInProgress == true) {

            // if not the end marker yet
            if (rc != endMarker) {
                if (ndx < maxDataLength) {
                    receivedChars[ndx] = rc;
                    ndx++;
                }
            } else {
                // if it is the end marker
                receivedChars[ndx] = '\0'; // terminate the string
                recvInProgress = false;
                ndx = 0;
                newData = true;
            }
        // if not already started receiving, and
        // the message begins with a start marker
```

Figure 13.2.14: LED Street Signal Code 15

```
        } else if (rc == startMarker) {
            recvInProgress = true;
        }
    }
}

void updateTimes() {
    deltaTime = (micros() - oldTime);

    deltaTimeSeconds = deltaTime/1000000.0;

    latestCarTime -= deltaTimeSeconds;

    if (latestCarTime < 0) {
        latestCarTime = 0;
    }

}


void updateSafeToGo() {
    // as long as there is not any car that will arrive within 3 secs,
    // you are safe
    safe = true;

    if (latestCarTime > 0) {
        safe = false;
    }

}

void updateLEDs() {
    // if not safe, light up
    if (!safe) {
```

Figure 13.2.15: LED Street Signal Code 16

```
        digitalWrite(ledPin, HIGH);
    } else {
        // if safe, turn off
        digitalWrite(ledPin, LOW);
    }

}

void receiveCommands() {

    char commandChars[maxDataLength+1];

    static boolean commandInProgress = false;
    static byte i = 0;
    char angleMarker = 'a';
    char distanceMarker = 'd';
    char getAngleMarker = 'g';
    char getDistanceMarker = 'h';
    char endMarker = '>';


    while (Serial.available() > 0) {
        char c = Serial.read();

        // if currently receiving
        if (commandInProgress == true) {

            // if not the end marker yet
            if (c != endMarker) {
                if (i < maxDataLength) {
                    commandChars[i] = c;
                    i++;
                } else {
                    // give up
```

Figure 13.2.16: LED Street Signal Code 17

```
                    commandChars[i] = '\0'; // terminate the string
                    commandInProgress = false;
                    i = 0;
                }
            } else {
                // if it is the end marker
                commandChars[i] = '\0'; // terminate the string
                commandInProgress = false;
                i = 0;

                String msg = String(commandChars);

                // should also break
                if (commandChars[0] == angleMarker) {
                    msg = msg.substring(1, msg.length() - 0);
                    angle = (float) msg.toInt();
                    Serial.println("set angle to: " + String(angle));
                    EEPROM.write(0, angle);
                    Serial.println(EEPROM.read(0));
                } else if (commandChars[0] == distanceMarker) {
                    msg = msg.substring(1, msg.length() - 0);
                    mountingDistance = (float) msg.toInt();
                    Serial.println("set dist to: " + String(mountingDistance));
                    EEPROM.write(1, (int) round(mountingDistance/100));
                    Serial.println(EEPROM.read(1) * 100);
                } else if (commandChars[0] == getAngleMarker) {
                    Serial.println("angle value: " + String(angle));
                    Serial.println("EEPROM value: " + String(EEPROM.read(0)));
                } else if (commandChars[0] == getDistanceMarker) {
                    Serial.println("dist value: " + String(mountingDistance));
                    Serial.print("EEPROM value:\t\t");
                    Serial.println(EEPROM.read(1) * 100);
                }
            }
```

Figure 13.2.17: LED Street Signal Code 18

```
        // if not already started receiving, and
        // the message begins with a start marker
        } else if (c == angleMarker || c == distanceMarker || c ==
        getAngleMarker ||c == getDistanceMarker) {
            commandInProgress = true;
            commandChars[i] = c;
            i++;
        }
    }
}



void loop() {
    // check to see if we have received any new commands
    receiveCommands();

    recvWithStartEndMarkers();

    if (newData) {
        processCommand();
    }

```

Figure 13.2.18: LED Street Signal Code 19

```
    updateTimes();

    updateSafeToGo();

    updateLEDs();


    if (debugTimes) {
        Serial.println(latestCarTime);
    }

    oldTime = micros();

    // let's run at 100hz
    delayMicroseconds(10000);
}
```

Figure 13.2.19: LED Street Signal Code 20

13.3 References

AliExpress. (n.d.). Retrieved November 01, 2017, from https://www.aliexpress.com/cheap/cheap-radar-sensor.html.

Association for Safe International Road Travel: Road Crash Statistics. (n.d.). Retrieved November 17, 2017, from http://asirt.org/initiatives/informing-road-users/road-safety-facts/road-crash-statistics.

Arduino Bluetooth Basic Tutorial. (n.d.). Retrived October 20, 2017 from https://create.arduino.cc/projecthub/user206876468/arduino-bluetooth-basic-tutorial-d8b737.

B., & E. (2011, April 13). Can Class 1 Bluetooth communicate with class 2 at 100m if one way transmission i - [Solved] - Wireless Networking. Retrieved January 11, 2018, from http://www.tomshardware.com/forum/33051-43-class-bluetooth-communicate-class-100m-transmission

BLINDSPOT PROTECTION - GREATER SITUATIONAL AWARENESS FOR DRIVERS. (n.d.). Retrieved January 12, 2018, from http://www.exeros-technologies.com/products/blindspot-protection

Bluetooth Power Classes. (2008, January 11). Retrieved November 01, 2017, from http://bluetoothinsight.blogspot.com/2008/01/bluetooth-power-classes.html.

Bogan, C. (2015, March 19). Adaptive Display's Digital Signage Blog. Retrieved November 11, 2017, from http://www.adaptivedisplays.com/blog/bid/361881/Understanding-Sign-Code-Regulations-for-LED-Sign-Permits

Choi, E., PhD. (2010, September). National Highway Traffic Safety Administration (United States of America , U.S. Department of Transportation, National Highway Traffic Safety Administration). Retrieved October 20, 2017, from https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/811366.

Consumer Reports(2017, September 13). Cars With Advanced Safety Systems. Retrieved October 23, 2017, from https://www.consumerreports.org/car-safety/cars-with-advanced-safety-systems/.

Cost of 32" High Visibility Traffic Mirror. (n.d.). Retrieved October 25, 2017, from https://www.trafficmirror.com/high-visibility-traffic-mirrors/high-visibility-traffic-mirror-32-inch.

Eskrine, J. (n.d.). Implementation of Doppler Radar [Scholarly project]. In Michigan State University: College of Engineering. Retrieved October 29, 2017, from http://www.egr.msu.edu/classes/ece480/capstone/spring10/group05/documents/justin_app.pdf

Garmin.https://static.garmin.com/pumac/LIDAR_Lite_v3_Operation_Manual_and_Technical_Specifications.pdf [PDF]. (2016, September). Olathe : Garmin.

Gillner, M. Wlaschitsch, J. Nuding Deceased., W. 1990 Traffic mirror United States Vegla Vereinigte Glaswerke GmbH (Aachen, DE) 4978207 http://www.freepatentsonline.com/4978207.html

HC-SR501 PIR MOTION DETECTOR [PDF]. (n.d.). Marlin P. Jones & Assoc. Inc. Retrieved November 22, 2017, from https://www.mpja.com/download/31227sc.pdf

IIHS. (2017, December). General statistics. Retrieved January 11, 2018, from http://www.iihs.org/iihs/topics/t/general-statistics/fatalityfacts/state-by-state-overview

Intro to RF - EEs Talk Tech Electrical Engineering Podcast #21 [Broadcast]. (2017, November 9). Retrieved November 20, 2017, from https://www.youtube.com/watch?v=xZlpOXjaxTI&index=5&list=PLKI9SYFe_L-k7N1cdpQ0pnnOL9JY75pAZ

Jefferson, A. (2016, September 30). How Do Blind Spot Monitors Work? Retrieved January 11, 2018, from http://www.proctorcars.com/how-do-blind-spot-monitors-work

Lampton, C. (2012, April 17). How are cars making the blind spot less dangerous? Retrieved January 11, 2018, from https://auto.howstuffworks.com/car-driving-safety/safety-regulatory-devices/cars-making-blind-spot-less-dangerous1.htm

Lane, D. M. (n.d.). Significance Testing and Confidence Intervals. Retrieved January 12, 2018, from http://onlinestatbook.com/2/logic_of_hypothesis_testing/sign_conf.html

Limkin. (2013, August 9). Making the electronics for a $7 USD doppler motion sensor. Retrieved from http://www.limpkin.fr/index.php?post/2013/08/09/Making-the-electronics-for-a-%247-USD-doppler-motion-sensor

Mesriani Law Group. (n.d.). Accidents Caused by Dangerous Intersections. Retrieved November 22, 2017, from https://www.hg.org/article.asp?id=7652

Monmouth County Road Map [Map]. (2012, October 15). In Monmouth County. Retrieved November 3, 2017, from

https://co.monmouth.nj.us/Documents/24/MC%20Road%20Plan%20Map%20Oct%2015%202012.pdf.

Motor Vehicle Safety. (2017, October 13). Retrieved January 11, 2018, from https://www.cdc.gov/motorvehiclesafety/

New Jersey State Police. (2018, January 11). Fatal Accident Statistics 2018. Retrieved January 11, 2018, from http://www.nj.gov/lps/njsp/info/fatalacc/index.shtml

NJDOT. (n.d.). Intelligent Transportation Systems Mobility and Systems Engineering. Retrieved November 17, 2017, from http://www.nj.gov/transportation/eng/elec/ITS/product.shtm

NJDOT Traffic Regulations - Local Roads In Monmouth County. (n.d.)., www.state.nj.us/transportation/refdata/traffic_orders/speed/monmouth.shtm.

Office of the Attorney General: Highway Traffic Safety. (n.d.). Retrieved November 17, 2017, from http://www.nj.gov/oag/hts/vehicle-motorcycle.html

QUIROGA, T. (2010, February). How To: Adjust Your Mirrors to Avoid Blind Spots | Feature | Features. Retrieved January 11, 2018, from https://www.caranddriver.com/features/how-to-adjust-your-mirrors-to-avoid-blind-spots

Savage, J. W., PC. (2014, December 19). Blind Spot Car Accidents: 6 Facts to Know. Retrieved October 13, 2017, from https://savagelawyer.com/blind-spot-car-accidents-6-facts-know-pt-1

Schondorf, S. Y., & Schondorf, K. M. (2008). U.S. Patent No. US 20090244741 A1. Washington, DC: U.S. Patent and Trademark Office.

Smith, E. (n.d.). Statistics on Intersection Accidents. Retrieved October 21, 2017, from https://www.autoaccident.com/statistics-on-intersection-accidents.html.

South Africa, Department of Planning, Transport and Infrastructure. (2012, October 24). Retrieved from https://www.dpti.sa.gov.au/__data/assets/pdf_file/0018/40149/DOCS_AND_FILES-1585762-v4-Traffic_Management_-_AS1742_2_-_Convex_Traffic_Mirrors_-_Operational_Instruction_2_3.pdf

SparkFun. (n.d.). SparkFun Bluetooth Mate Gold. Retrieved October 25, 2017, from https://www.sparkfun.com/products/12580.

TRAFFIC SIGN DESIGN, PLACEMENT, AND APPLICATION GUIDELINES. (2017, March). Retrieved from https://mdotcf.state.mi.us/public/tands/Details_Web/mdot_signing_design_placement_application_guidelines.pdf'

United States of America , State of New Jersey, New Jersey Department of Transportation . (n.d.). State of New Jersey. Retrieved October 20, 2017, from http://www.state.nj.us/transportation/about/rules/documents/16-27-Current.pdf.

United States, State of New Jersey, Department of Transportation. (n.d.). NJDOT Crash Records: 2001 -2016. Retrieved November 17, 2017, from http://www.state.nj.us/transportation/refdata/accident/pdf/TotalCrashRecordsbyCounty.pdf

United States, State of New Jersey, Department of Transportation. (2017, August 9). Crash Records 2016: Crashes at Unsignalized Intersections . Retrieved November 15, 2017, from http://www.state.nj.us/transportation/refdata/accident/16/ARDSMRPT10.2016.pdf

United States, New York State Department of Maintenance and Operations , New York State Thruway Authority . (2011, May). New York State. Retrieved November 15, 2017, from http://www.thruway.ny.gov/commercial/forms/tap633.pdf

United States, U.S. Department of Transportation, Federal Highway Administration. (2009). Manual on Uniform Traffic Control Devices. Retrieved November 17, 2017, from https://mutcd.fhwa.dot.gov/htm/2009/part2/part2l.htm

Williams, A. (2017, May 24). Radar Sensors Put to the Test. Retrieved November 01, 2017, from https://hackaday.com/2017/05/24/radar-sensors-put-to-the-test/.

Vehicle Detection Sensors | Product Capabilities | Banner. (n.d.). Retrieved November 16, 2017, from https://www.bannerengineering.com/us/en/products/capabilities/vehicle-detection.html#flt/bec:capabilities|sensor-type|ultrasonic

Youngs, J. (2012, February 24). IIHS Reports on Emerging Safety Technologies. Retrieved December 1, 2017, from http://www.jdpower.com/cars/articles/safety-and-mpg/iihs-reports-emerging-safety-technologies

49 CFR 571.111 - Standard No. 111; Rear visibility. (n.d.). Retrieved January 11, 2018, from https://www.law.cornell.edu/cfr/text/49/571.111

13.4 Testing Location

The device was tested Georjean Drive, Holmdel, NJ 07733.

[POWERPOINT PRESENTATION GOES HERE]