

## Worcester Polytechnic Institute Digital WPI

---

Major Qualifying Projects (All Years)

Major Qualifying Projects

---

August 2011

# Integrated Automotive Safety System

Amanpreet Singh

*Worcester Polytechnic Institute*

Brennan James Caissie

*Worcester Polytechnic Institute*

James Lee MacDonald

*Worcester Polytechnic Institute*

Follow this and additional works at: <https://digitalcommons.wpi.edu/mqp-all>

---

### Repository Citation

Singh, A., Caissie, B. J., & MacDonald, J. L. (2011). *Integrated Automotive Safety System*. Retrieved from <https://digitalcommons.wpi.edu/mqp-all/1362>

This Unrestricted is brought to you for free and open access by the Major Qualifying Projects at Digital WPI. It has been accepted for inclusion in Major Qualifying Projects (All Years) by an authorized administrator of Digital WPI. For more information, please contact [digitalwpi@wpi.edu](mailto:digitalwpi@wpi.edu).



# Integrated Automotive Safety System

A Major Qualifying Project

Submitted to the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment for the

Degree of Bachelor of Science

**By**

Brennan Caissie

James MacDonald

Amanpreet Singh

August 23, 2011

Professor Robert C. Labonté, Principal Project Advisor

## Executive Summary

The purpose of the Integrated Automotive Safety System is to provide a cost-efficient means of outfitting older and low-end cars with the latest in safety appliances in the simplest way possible. Our design is an integration of several existing technologies to provide a crucial midpoint between cost and performance for low budget vehicle safety.

The project will focus mainly on the most crucial components of such an integrated system, such as the location-based services provided by cellular phone towers made possible by a GSM cellular modem. This board also allows for our system to make telephone calls or SMS text messages to alert authorities automatically in the instance of a traumatic event.

Emergency services already have the ability to track the location of an emergency call from a cell phone to about an area of a 300 foot radius, and will soon have the ability to accept text messages as alerts, making our system already ahead of its time.

In order to detect if a crash has occurred, we integrated an accelerometer to sense dramatic changes in acceleration on multiple axes. This way, not only will our system detect sudden acceleration from any direction, but also if the car has been rolled over as a result of the crash. By linking this information with the GSM board, we are then able to call the authorities within a short period of time after the crash and even include data relating to the force of the crash and vehicle orientation in a text message. This will allow emergency services to react in the quickest and most effective way possible to any automotive crash event.

Additionally, other peripheral devices were tested and included in our device, though they did not necessarily receive as much attention as the tracking system. These devices include a sonar-based proximity sensor, which can detect an oncoming vehicle from the front, back or side of the car and alert the driver to the danger through a visual or auditory warning.

Another device that we integrated was an LED light array, which could conversely alert oncoming drivers to their proximity to the host car, simply by flashing in a simple pattern to attract the operator's attention

early enough for corrective actions to be made. The system relies on communication with the sonar device to obtain a reading of proximity from other cars, and in turn switches the LEDs from their steady-burn state to a flashing pattern, warning oncoming drivers of danger.

As evident from the previously mentioned devices, our system relies on an assortment of sensors to provide top-level safety measures for consumers that, so far, only high-end automakers have been able to provide. Though our project is primarily a proof-of-concept on how such a system could be designed, we are certain that our project could be marketed and sold for a competitive price, and would potentially save many lives in the US and abroad, if implemented.

## Abstract

Most new cars today come with a host of advanced safety features including automated systems that assist the driver in maintaining control of the car and warning the driver of possible dangers. The problem, however, is that while these kinds of features greatly increase the safety of a car, they are prohibitively expensive and only available in new, high-end cars. The purpose of this MQP is to create a device that would integrate a series of safety features that could be installed in any car on the road at minimal cost to the consumer.

## Acknowledgments

We would like to thank WPI for giving us both the knowledge and resources to be able to carry out this project. We would also like to thank our project advisor, Professor Labonté, who made sure that we were always going in the right direction and made sure that we were able to get everything done on time. Without his guidance, we may have lost sight of the overall goal by focusing too much on the individual aspects of the project. With his help, we were able to make decisions that enabled us to complete the project in a timely manner, while still making sure that the final product was made to specifications.

We would like to thank Tom Angelotti from the ECE department for helping us set up a proper work environment and making sure that we had all the tools we needed to be able to complete this project.

We would like to give a special thanks to the WPI and Worcester police departments. They were very helpful in explaining how their systems currently work and what our system should be capable of doing.

Finally, we would like thank Analog Devices for their insight into how the accelerometers that we used work. Thanks to them, we were able to discern when a crucial piece of equipment was beyond repair and we were able use this knowledge to proceed with the project properly.



## Table of Contents

Executive Summary.....	2
Abstract.....	5
Acknowledgments.....	6
Table of Figures.....	11
Introduction.....	12
Product Requirements.....	15
Customer Needs.....	16
Competitive Value Analysis .....	17
Background Research .....	20
On Board Diagnostic Interface.....	20
Meeting with the Police Department.....	21
Design Approach .....	22
Component Selection.....	25
Microprocessor Base .....	25
GSM Module .....	27

Rangefinder .....	28
Accelerometer.....	30
Component Integration.....	32
Costs.....	33
Design Obstacles.....	35
Final Design Process and Results.....	39
Future Considerations.....	47
Conclusion.....	50
Bibliography .....	52
Works Cited.....	52
Appendix A - Code.....	54
Full Code.....	54
Appendix B - Weekly Summaries .....	58
Status Report #1.....	58
Status Report #2.....	60

Status Report #3.....63

Status Report #4.....68

Status Report #5.....72

Status Report #6.....75

Status Report #7.....77

Status Report #8.....79

Status Report #9.....81

## Table of Figures

Figure 1- Overall System Design.....	24
Figure 2 - Arduino Uno Board.....	27
Figure 3 - GSM Playground Board.....	28
Figure 4 - Ultrasonic Rangefinder .....	29
Figure 5 - Accelerometer .....	31
Figure 6 – Breakdown of Costs.....	33
Figure 7 - The final Product sans LED Strip.....	39
Figure 8 - LED Strips .....	39
Figure 9 - Stuck accelerometer position.....	42
Figure 10 - Axes directions as viewed using our accelerometer .....	44
Figure 11 - Rollover detection free-body diagram.....	45
Figure 12 - Overall System Design.....	66

## Introduction

The automotive industry of the 21st century is far from being defined by big muscle cars or flashy sports cars. In fact, the dominating features of most cars seen on the roads today revolve around safety systems and how well the car can withstand accidents without injuring the passengers within. Car makers today can easily build extremely safe automobiles for anyone looking to pay top dollar, with some features now even coming standard. ( Pietschmann, Herald – 4x4ABC) Safety is a top concern for automakers and will continue to be as long as cars remain on the road. (Brown, Robin – MotorTorque.net)

But what about cars made in previous generations? Cars made in a time when sporty features and hefty engines were more enticing than an airbag system or some anti-lock brakes. For these vehicles, a certain level of risk is assumed when driving on the roads, but not enough risk to necessitate the removal of these cars from the road. They are assumed to be “safe enough”, but when compared to the modern day Volvo and

Mercedes-Benz cars, with their multitude of sensors, On-Star™, and High-Intensity Discharge headlights, how safe can they really be?

Our goal for the Integrated Automotive Safety System is to provide a level playing field for all vehicles, regardless of age, when it comes to outfitting cars as well as possible for any risks one can face on the roads. These risks include rollovers, collisions, and non-responsive drivers after accidents, and lack of location information after an accident has occurred. These sorts of risks plague every driver in the US and abroad, but sadly only the newest vehicles provide protection from dangers such as these. Where does that leave the average teen driving a late 90s, early 2000s high-mileage car, or perhaps an elderly person driving the same car they've had for 40 years? These cars likely do not have sufficient safeguards for today's risks, but our project can remedy this.

The two key areas that our product needs to excel in, in order to be successful, are cost and adaptability. The device must not break the bank, but provide a level of safety at or above what other users expect from similar services and devices, such as those employed by On-Star™. The

other issue is the broad range of vehicles we wish to appeal to, which could span any number of years to modern day cars that haven't even rolled off the assembly line yet. A son driving an old 80s pickup truck should have the same safety net his father has driving a 2010 pickup truck and this is exactly what we look to achieve with this product. In essence, we hope to make every car on the road today as safe as possible for the lowest price possible.

## Product Requirements

- Must be able to operate at 12±2 volts.
- Must be able to operate between -10 and +100 degrees Fahrenheit.
- Must be able to accurately detect obstacles as far as 5 meters away.
- Must be able to detect events that could trigger a rollover and warn the driver about it.
- Must be able to provide feedback to the driver using LED lighting and sound.
- Must be able to determine if a crash has occurred.
- Must be able to make an emergency call in the case of a crash event.
- Must be able to determine and transmit the orientation of the car.



## Customer Needs

- Needs to be operated with the battery of the car.
- Needs to be very easy to set up with minimum wiring.
- Needs to be low cost.
- Needs to be able to fit into any vehicle (new or old).

## Competitive Value Analysis

In order for our product to succeed, it needs to be more appealing than existing products already on the market. This includes OnStar™, as well as other automotive safety systems such as built-to-order offerings from automakers, including modular products such as car alarms and reverse sensors or cameras.

As our system most directly competes with OnStar™, we will compare the two systems against each other to highlight the strengths and weaknesses of each product. This way, we can have an accurate assessment of a customer's considerations when looking to buy car safety devices.

First, let's look at the most important factor, which is price point. OnStar™ offers a subscription-based integrated system, which utilizes GPS technology and accelerometers to detect location and catastrophic events from within the control unit. This data is then transmitted over a cellular data connection, which also carries voice service for such things as

communication with emergency dispatchers, or simply asking an OnStar™ representative for directions to the nearest shopping mall.

For the basic service, which does not include turn-by-turn navigation, you can expect to pay around \$18.95/month (\$24.95 in Canada), which includes “Automatic Crash Response, Stolen Vehicle Assistance, Roadside Assistance, Remote Door Unlock, Remote Horn and Light Flashing, Red Button Emergency Services and OnStar™ Remote Vehicle Diagnostics ”. For \$10 more, you get the navigation services added in with the other features.

Our product offers a baseline layer of security, but offers less expensive coverage due to some key design moves. First, we opted to not include GPS in our unit, as cellular triangulation using the embedded GSM board offers plenty of accuracy for locating an accident site, down to within 300 feet of signal.

We also bypassed expensive subscription fees by using an unregistered SIM card for the GSM board, which can make toll-free

emergency calls anywhere in the continental United States. This gives us an economical advantage over OnStar, as the price of adding the OnStar device to a vehicle (whether integrated mirror model or embedded vehicle model) is upwards of 300 dollars, not including possible installation fees. Our device will need minimal installation work, requiring only a small amount of wiring to make the device operable.

Thus, with no cost for sustained subscription, and a low one-time cost for the customer up front, we offer an extremely affordable safety system for a wide variety of vehicles.

## Background Research

### On Board Diagnostic Interface

The Onboard Diagnostic interface is mandatory in all modern cars and is used to request data for diagnosing problems with the car. (California Environmental Protection Agency Air Resources Board) It can also provide data while running, such as speed, fuel usage, and engine speed. The codes for the data vary by manufacturer and by model, making a universal solution difficult. Complicating the matter is the secrecy associated with this proprietary information, as all manufacturers do not publicly document their OBD-II codes. Accordingly, we believe it is infeasible to incorporate into our system a method of communication with the vehicle using the OBD interface, and will rely on data obtained solely through our own device's sensors.

## Meeting with the Police Department

Since our system needs to be able to call the police during an emergency, we wanted to speak with them to find out what the logistics would be for such a call. The first thing that we learned, unfortunately, was that we would not be able to place automated calls from a cell phone-type device, as that is illegal in Massachusetts. We can, however, place a call to a local police station. Furthermore, we learned that besides the location of the crash, it is very helpful for the police to know the orientation of the vehicle.

We will be able to add the orientation data based on the readings from an accelerometer. As for the local calling, we can simply use a GPS module to find out what city the vehicle is in and call the local department from a database of numbers. Due to budget and time constraints, however, we will not implement this feature in our current project. We will include information on what needs to be done for future considerations.

## Design Approach

Our final product will be a series of integrated sensors and warning systems, but our design and build approach will be modular. That is, each sensor will be able to function independent of the others so that we can build and test one aspect of the project before working on the next one. This ensures that we will be able to interface all the parts more easily and that we don't have to think about the whole, more complex project when correcting for any errors we encounter.

Essentially, the build can be broken down into:

1. Design and build the first part.
2. Make sure that part is working perfectly.
3. Separately design the second part.
4. Make sure that part is working perfectly.
5. Make sure that the first two parts are able to work simultaneously.
6. Repeat for each of the next parts.

The Arduino board comes with its own development environment that allows for easy sensor reading and outputs. It can handle basic and complex math including basic arithmetic, bitwise operations, loops, interrupts, etc. It is based on C/C++, so we have a powerful development environment.

We have decided to use an Arduino board using an ATmega microcontroller. This setup makes available a GSM module that is able to place regular phone calls using a SIM card and 911 calls without one. In addition to this, it also contains a number of both analog and digital I/O ports which will be necessary for both the sensor readings and control of the running lights and warning systems. There is also plenty of documentation available to help us with this process.

The block diagram on the following page shows the overall top-level layout of the project design:



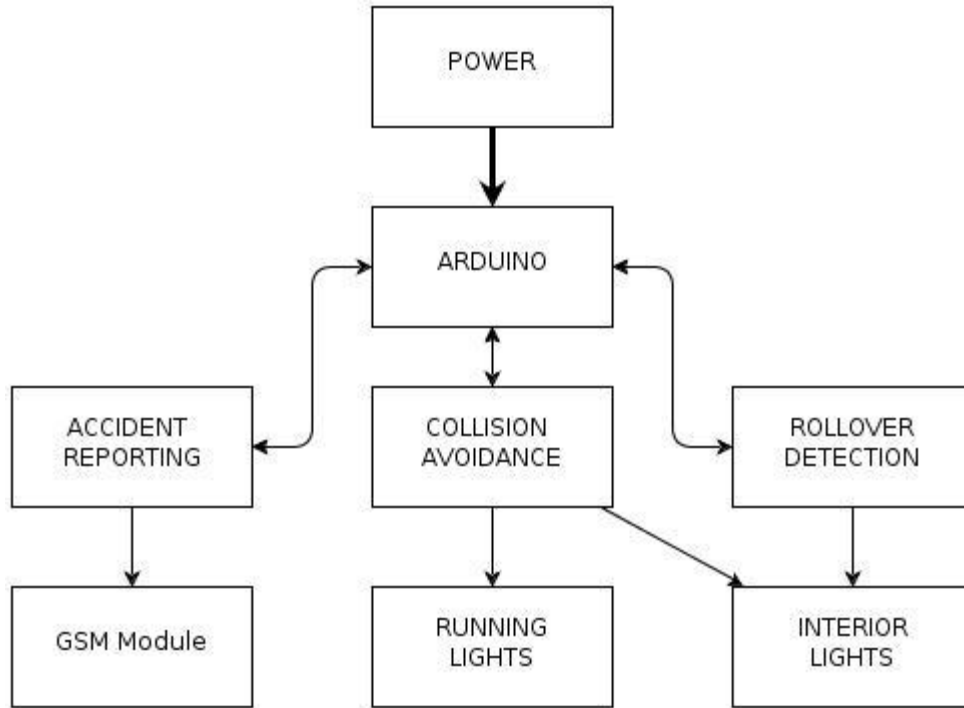


Figure 1- Overall System Design

Each block represents a different component of the project and the arrows represent the flow of information. For example, the running lights only receive information from the collision avoidance system, but the collision avoidance system both sends data to and receives data from the Arduino board.

## Component Selection

### Microprocessor Base

As the basis of this project is an integration of several existing technologies, we required a microprocessor capable of coordinating multiple sensors and output systems using both analog and digital communication. We reviewed a variety of possible microprocessors, including the MSP, Arduino, and PIC families. The PIC controller was quickly dismissed due to the difficulty of programming and lack of peripheral support. The MSP was a much more viable option, as it could be programmed using C and had more options for input and output.

Our final selection was the Arduino family, which were ideally suited to our project. The Arduino Uno development board chosen has many digital I/O ports, which supports the integration of several sensors with sufficient expandability for adding additional modules for a more comprehensive or more specialized future design. The programming environment was also a major consideration, as the Arduino environment

is simple enough to be used with little experience and programming in C suits our programming skills. Additionally, there are libraries available to support serial communication with digital peripherals, which eases the programming burden significantly. The development board also includes voltage regulators with accessible outputs for peripherals, which removed several components from the design.

Perhaps the greatest feature of the Arduino board was its ubiquity. There exists an extensive group of hobbyist users of these boards who make up a very large community and maintain very active support and development forums. Their popularity has led to a wide variety of peripheral boards which interface with the Arduino development board with no additional components. By pairing this Arduino board with a GSM board peripheral and software libraries, we were able to easily implement a fairly complex design with little difficulty.

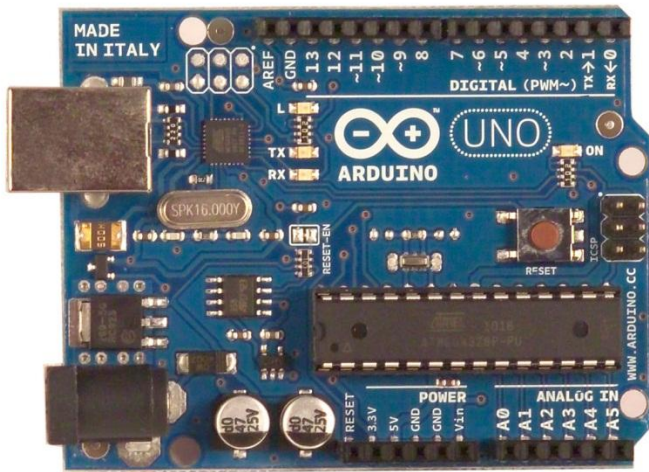


Figure 2 - Arduino Uno Board

## GSM Module

The crash reporting portion of this project required the use of the cellular telephone networks to convey information to emergency services. To accomplish this we needed a solution that would interface with our chosen microprocessor and be simple enough to implement with without prior experience with the hardware and software used for cellular telephony. Ultimately we were able to find an Arduino peripheral that could be used with minimum additional components and had a software library to handle all the communication functionality. This library includes

functions for placing and answering calls and sending and receiving text messages.

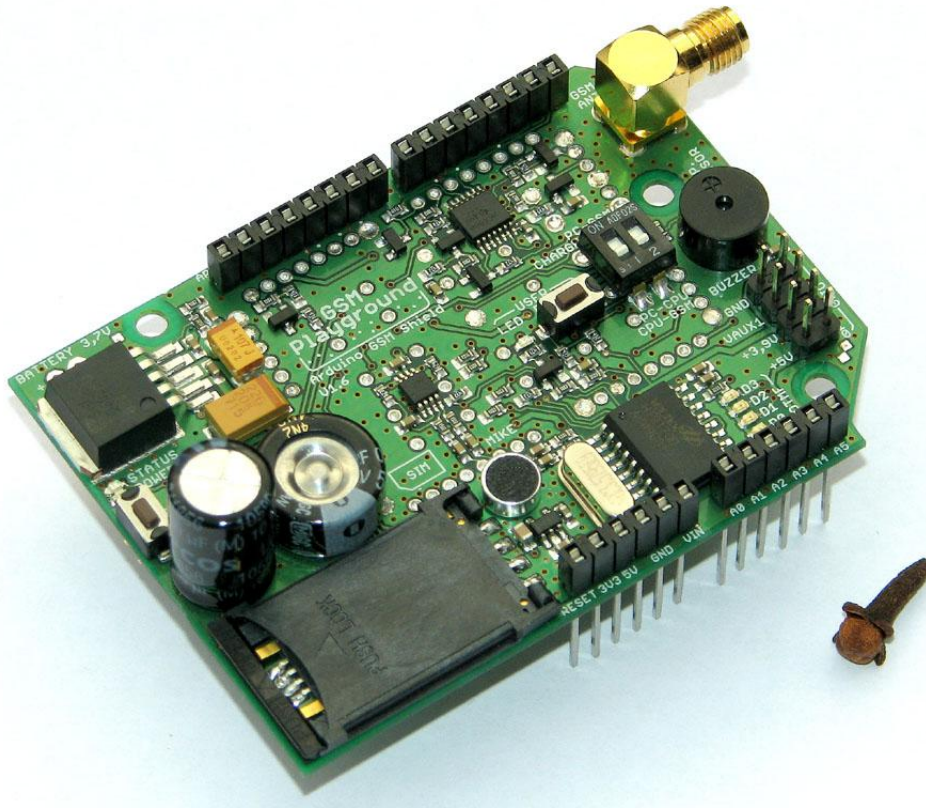


Figure 3 - GSM Playground Board

## Rangefinder

To perform the collision warning functions our device had a requirement to accurately measure distances between our sensor and a foreign object at ranges of up to several meters and at an angle which could provide coverage for a wide area. Additionally we sought a component

that would be easily interfaced with our development board and program. The Maxbotix line of acoustic rangefinders fit this requirement well and had been positively reviewed by a number of people. They operate at a voltage which is provided by our development board and has both analog and digital output modes. There are a range of models to support a variety of angles of observation.



Figure 4 - Ultrasonic Rangefinder

## Accelerometer

The crash and rollover detection functionality of our device required the use of an accelerometer. These two events occur solely on the XY plane, which is parallel to the ground, but detecting the final position of the vehicle requires a third axis of acceleration detection. Specifically, when the vehicle is flipped the X and Y axes will report identically to the case when the vehicle is correctly oriented. As this capability was specifically requested by the emergency services we interviewed, a three axis accelerometer was required. Our selection for this component was a device that could operate from our board's available supply voltages, communicate with our chosen microprocessor, and accurately measure acceleration values up to at least 3g. We initially selected a Bosch BMA180 three axis accelerometer for this role due to its serial interface and configurable resolution and measurement ranges, which allowed us the accuracy we required. Additionally the device could operate at one of the board's supply voltages and its I2C serial interface used only two microprocessor ports for communication. However, during testing and

initial construction, this part was damaged and a replacement was needed. We chose the Analog Devices ADXL345 accelerometer due to its similarity to the Bosch model. Both accelerometers had three axes with high resolution acceleration data and communicated over an I2C connection, allowing it to be implemented in our design with minimal adjustment.



Figure 5 - Accelerometer



## Component Integration

The focus of this project has been on integrating a number of existing technologies into a device which would provide novel functionality. This necessitates a design in which the incorporated sensor and output elements function well with each other. In selecting an Arduino development board for the base of the project, we were well equipped to meet this challenge. All the chosen peripheral components were able to operate on the supply voltages of the board and to interface with the microprocessor without additional components. In the case of the GSM module, the peripheral was specifically designed to work in tandem with the Arduino board. It is because of this interoperability that our design was able to be completed in the short time available to us and without including significant amounts of additional or specialized components or requiring more advanced programming or design knowledge.

## Costs

Part	Cost
Accelerometer	\$29.95
Arduino Board	\$26.95
GSM Shield	\$266.43
Untrasonic Rangefinder	\$27.95
LED Strips	Already Had, but this would generally be around \$10-\$15/foot
Batteries	Already Had, but this could be anywhere from \$6-\$16.
Antenna	\$7.95
Total Price with Shipping and Handling	\$363.64 (upwards of \$400 with all new parts)

Figure 6 – Breakdown of Costs

We had a combined budget from the WPI ECE department of \$375. We were able to stay within this budget by minimizing shipping costs and making sure we didn't already have spare parts before buying new ones. If we had to purchase the batteries and LED strips, we would have gone over the budget by only about \$20. To minimize shipping costs, we ordered

three of the parts from one vendor. We purchased the Arduino Board from a vendor that offered free shipping. Finally, the GSM shield was only available from one particular vendor that was based in Europe, so that drove up our costs a bit simply due to the shipping cost, but this this particular shield was the best match for our project in terms of ease of implementation.

## Design Obstacles

Our safety system has, from the beginning of the design process, incorporated a location-based tracking system to aid emergency services in the tracking of injured or lost motorists. Initially, this system was based upon tracking using GPS location, which would allow coordinates to be sent to emergency services through text or other data standard.

Implementing such a system would require not only a complex method of transmitting coordinate data to emergency services reliably, but also a way of parsing the data into usable information for dispatching life-saving services once received. In researching our options, we found that text transmission of GPS coordinates is not only an unnecessarily complicated procedure, but also very expensive. The new equipment needed to receive and view such information on the police side, and the circuitry required to send it from the car would be very costly, effectively destroying our major selling point against reigning auto safety king OnStar™. For our device to succeed in the market, and for it to not be just

another product with no further innovation than modular design, we would need to cut costs and complexity by a vast margin.

Through further investigation, a discovery was made that could save us not only hundreds on our budget, but also simplify the reporting system substantially. This discovery was in the form of “Location Based Services”.

Location Based Services are used by police and emergency services to locate cellular phone users when away from land line phones and in need of help. The technology involves triangulation of cellular signal from service towers in the area of the cell phone, thus giving responders a rough location by which to dispatch police, fire or EMS personnel. The technology has been around for many years, and has been improved upon substantially over its lifetime. What we were shocked to discover was the accuracy of which the triangulation had reached in that time. What used to be a mile-wide approximation has been drawn in to as few as 300 feet. In fact, a federal standard now mandates that location based services must locate a distressed person to an accuracy of within 300 meters within a time frame of 6 minutes. Clearly, this technology would play into our favor.

The technology already incorporated in our system allows for easy integration of Location Based Services, hereafter known as LBS, through the built-in GSM modem card. This card will be able to make the necessary call to 911 to connect to LBS regardless of subscribed services. This means a toll-free connection to emergency services with no intermediary, which means there will be no subscription fee for the end user, saving them money over competitors such as OnStar™.

Should the device be implemented as an in-vehicle system, the way we decided to integrate this technology into our crash detection system was to incorporate a speakerphone with the device so a hands free call could be made in the event of an accident. First, a short timer will need to expire as to alleviate any false alarms immediately following the detection of a traumatic event. This timer may be interrupted by a button or switch. After the expiration of time with no interrupt, a 911 call will be placed over speakerphone. From here, the motorist can either respond to the dispatcher verbally, or simply wait in silence for help to arrive. This way, even an unconscious victim can be helped in the event of an accident. In the case of

the device being mounted within the engine compartment, the timer, call cancellation, and voice interaction features will be bypassed in order to place a call immediately. With the 911 call triggering LBS, the location of the accident will be known within seconds, and certainly not longer than 6 minutes per federal standards. Finally, the 300-foot accuracy can help responders find the victim within a very manageable perimeter. Through implementation of already-existing safety technology, LBS will help our system deliver affordable protection to motorists with cars of all makes and models.

## Final Design Process and Results

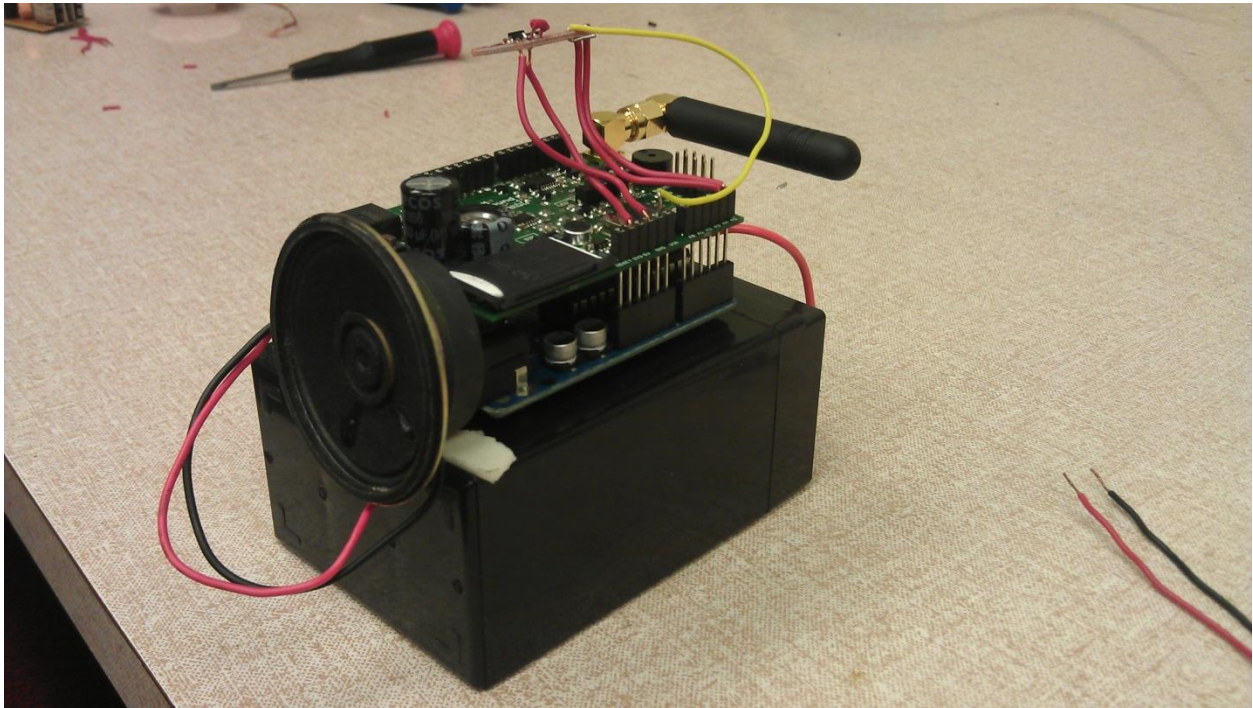


Figure 7 - The final Product sans LED Strip

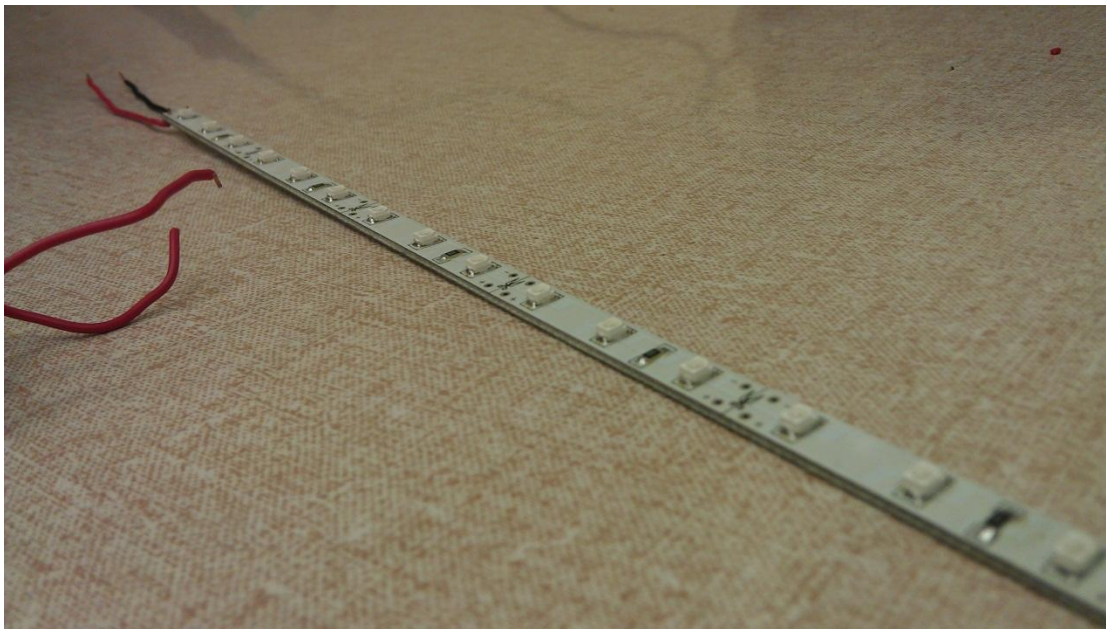


Figure 8 - LED Strips



Incorporating all the disparate components of our design into a single cohesive product was the major focus and challenge of our project; a task which was greatly eased by our selection of the Arduino development board for our base. As described earlier, our design methodology involved implementing and verifying the operation of a single module at a time, before gradually combining them.

The first component to arrive and the first to be designed and tested was the Arduino board itself and the basic input and output functions needed to measure analog signals and control the basic digital signals used for switching the light systems. These first steps in the project allowed us to become familiar with the Arduino board, the language and IDE used for programming the board, as well as debugging and overall program design. These tasks were accomplished fairly quickly and we soon moved on to working with the sensor modules.

The first sensor to be incorporated into the devices was the ultrasonic rangefinder. This device operated at the 5V that was available on the development board and had an analog output that ranged from 0V to the

input voltage. This behavior was ideally suited to the analog signal measurement of the Arduino, as it has an internal reference voltage of 5V, allowing us to measure over the complete range of the rangefinder's output with maximum accuracy. From the datasheet for the rangefinder we know that the device's analog output would correspond to  $\sim 9.8\text{mV/in}$ . The Arduino documentation describes the ADC as operating at approximately  $4.9\text{mV/LSB}$ . Based on these specifications we were able to obtain accurate distance information for use in implementing the collision warning functionality of our design.

The accelerometer was the next component to be integrated into the project. Our first IC, a Bosch BMA180, was connected to the Arduino ports designated for use with I2C serial interface devices. It was after considerable testing that we determined that the chip had been damaged when soldering wire leads to the device or had been defective upon arrival. This assessment was made after observing that the data along two axes was not responding to varying accelerations and appeared to be returning a value near the upper limit of the measurement range. This issue was

discussed with engineers at Analog Devices familiar with the operation of accelerometer ICs, who reported that due to the physical design of the IC a common failure was to have the devices become “stuck” at a maximum value, as illustrated below.

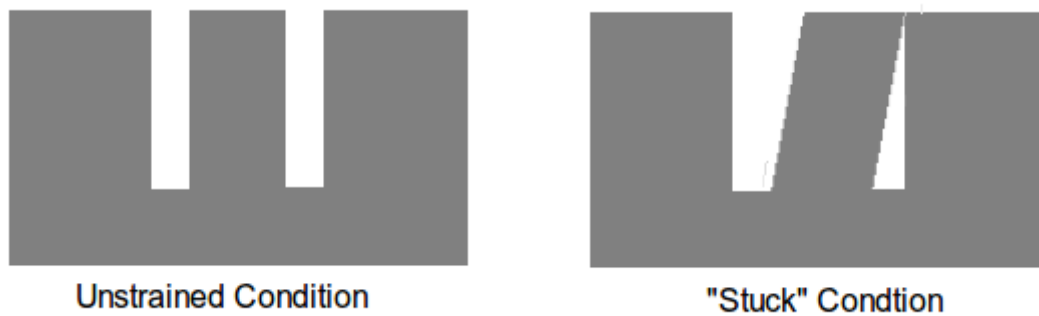


Figure 9 - Stuck accelerometer position

Another consequence of the physical design of the ICs was that a three-axis accelerometer would most likely be composed of two perpendicularly oriented two axis accelerometers whose data was collected and reported over a single interface. This was consistent with the mode of failure observed in our chip, as only two axes had failed, implying only one of the two components in the device had failed and the other continued to function. After confirming that it was not possible to recover the device’s functionality or proceed with a damaged device, we

researched a replacement. The Analog Devices ADXL345 was selected due to its similar feature set and we proceeded with the development of the device. Once successful configuration of registers and accurate data reporting had been confirmed, the algorithms for detecting crashes, rollover risk, and car orientation were developed. Our thresholds for determining crashes and rollover risk were based upon the assumption that a car would be unable to produce an acceleration value greater than 1.5g - 2g during normal operation. This was due to the coefficient of friction for standard vehicles being typically no greater than 1, meaning the acceleration of the vehicle under its own power could not exceed 1g on a flat surface. Thus our threshold for detecting a crash was determined to be an acceleration value of 2g in either the X or Y axes, perpendicular to the gravitational axis, to allow for the effects of differing road angles and the effects of variation in reported data due to noise.

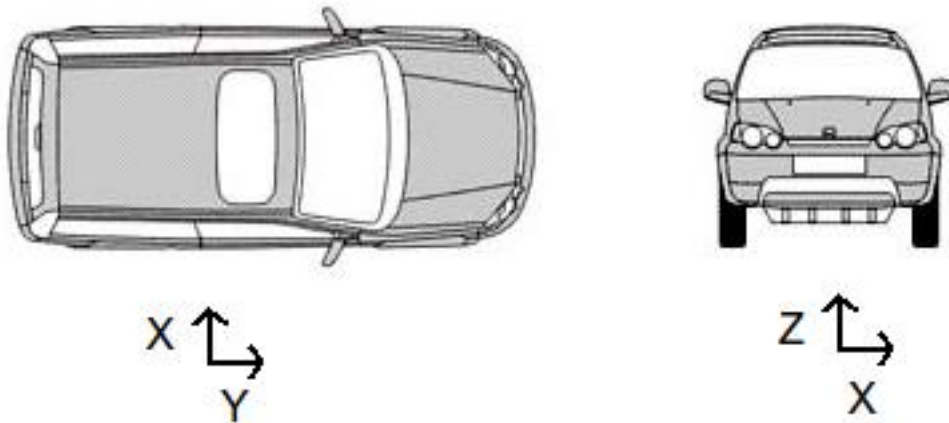


Figure 10 - Axes directions as viewed using our accelerometer

Our threshold for rollover danger was slightly more difficult to determine. The possibility of rollover was considered only along the axis corresponding to the width of the vehicle. A calculation of the acceleration along this axis needed to roll a car began with an assumption of infinite friction between the tires and road in this direction. This was justified because a rollover occurs when the wheels do not slide in the direction of force, but instead stick and act as a fulcrum about which the body of the car will rotate in response to a force oriented perpendicular to the side of the vehicle. In this case we simplify the situation to a point mass at the center of gravity of the vehicle and a pivot at the point where the outer edge of

the tire meets the road surface. Due to the differences in weight distribution across vehicles, it was more difficult to determine a threshold for a rollover warning.

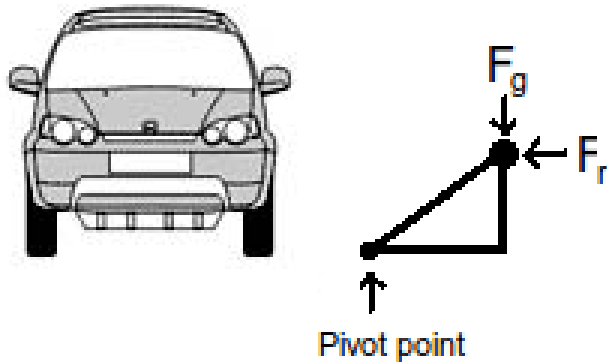


Figure 11 - Rollover detection free-body diagram

The orientation of the vehicle was also determined using accelerometer data. The calculation was not made until the magnitude of the acceleration dropped below approximately 1.5g, to allow for the vehicle to be settled after a collision. After this delay the acceleration vector was measured and the axis it was closest to was used to determine the orientation. If the measured vector was closest to the positive Z direction the vehicle was reported to be correctly oriented. A vector along the

negative Z direction was reported as a flipped vehicle, and along either X or Y direction as the vehicle being on its side.

Once this crash had been detected and the orientation determined, the cellular telephony module was enabled. After successfully connecting to a GSM network, the device has the capability of either placing a call or sending a text message. For evaluating the functionality of the device, it was configured to transmit a text message containing the orientation of the vehicle. This behavior was tested and verified for all orientation conditions.

## Future Considerations

As the scope of this project was limited to the crash detection and reporting and the rollover and collision warning systems, there were several additional functions necessarily excluded from the design. A more comprehensive device is easily created by adding modules to the base we have created, as the Arduino board is well suited to handling multiple sensor systems.

A major component excluded from our device was a GPS receiver. With this component our crash reporting could include exact locations for emergency services to be dispatched to, as well as allowing the device to contact the closest local dispatcher. This component was determined to not be crucial to the project as the Enhanced 911 service allows all cellular telephone calls to be traced by emergency services. With this capability the location of a reported crash could be determined without direct transmission of geographic coordinates. Therefore we were able to exclude this component from our project.



An additional component that was considered for the project was a relay placed in series with the fuel pump of the car, so that it could be disabled in the event of a crash. Ultimately we determined that implementing this functionality would require an unacceptable amount of interaction with the electrical systems of the car and in an area where an error in our product's operation would be of significant detriment to the user. However, if the user would like to add this functionality, it would be trivial to implement in our product itself. A simple relay operated from a digital output from the Arduino board would be sufficient to control the operation of the fuel pump; the difficulty and risk of implementation would lie nearly entirely in the manipulation of the wiring for the fuel pump and its power supply.

In regards to the capability of contacting emergency services, current law prevents making automated calls. However, future development is planned to enable these 911 services to handle both text messages and automated calls, enabling our device to be more easily implemented into the crash reporting infrastructure. This would allow either the GPS data

encoded in a text message and the tracking of an automated call methods of location determination to be used in a crash reporting system.

## Conclusion

At the beginning of our project, we sought to create a device that would revolutionize the way we protect our older vehicles and those who operate them. We looked to make an affordable protective device to appeal to a specific market; being those who drive older or cheaper vehicles that may not come with adequate safety features deemed desirable by the consumer. The process took many hours to research, prototype and configure, but we believe we've designed a unit that will have mass appeal to the automotive safety market.

By incorporating many popular features (rollover detection, impact detection, crash reporting, location services, emergency calling, sonar range finding and more), we have been able to create a product that will satisfy the insatiable hunger for safety demanded by American drivers. Due to our creative design which streamlines and customizes the whole system, we have also been able to create a system that, at its base, allows for extremely affordable implementation at all levels. This is crucial, as the people who need this most (i.e. those driving aging or cheaper vehicles) would

certainly be looking to save money while maintaining top-level safety in their car. Our product allows for such protection, at a price that competes with the competition by far.

# Bibliography

## Works Cited

Brown, Robin. "Europe Wants Brake Assist as Standard by 2009." *MotorTorque.com*. 10 Oct. 2007. Web. 21 May 2011. <<http://www.motortorque.com/news/auto-0710/europe-wants-brake-assist-as-standard-by-2009.asp>>.

"On-Board Diagnostic II (OBD II) Systems - Fact Sheet / FAQs." *Air Resources Board*. California Environmental Protection Agency, 28 Sept. 2009. Web. 10 May 2011. <<http://www.arb.ca.gov/msprog/obdprog/obdfaq.htm>>.

Parker, Akweli. "HowStuffWorks "How Brake Assist Works"" *HowStuffWorks "Auto"* Discovery, 30 Apr. 2009. Web. 23 May 2011. <<http://auto.howstuffworks.com/car-driving-safety/safety-regulatory-devices/brake-assist.htm>>.

Pietschmann, Herald. "How Mercedes-Benz BAS Works." *4x4ABC*. 1997-2007. Web. 20 May 2011. <[http://www.4x4abc.com/4WD101/BAS\\_how.html](http://www.4x4abc.com/4WD101/BAS_how.html)>.

Schagrin, Mike, John Harding, Gregory Davis, and Arthur Carter. "Research and Innovative Technology Administration (RITA) - United States Department of Transportation." *Research and Innovative Technology Administration (RITA) - Intelligent Transportation Systems (ITS)*. US Department of Transportation, 22 June 2011. Web. 23 May 2011. <<http://www.its.dot.gov/cicas/index.htm>>.

Swallow, Erica. "911 Will Soon Accept Texts, Photos, Video - CNN.com." *CNN.com* -

*Breaking News, U.S., World, Weather, Entertainment & Video News*. CNN, 11 Aug.

2011. Web. 13 Aug. 2011.

<[http://www.cnn.com/2011/TECH/mobile/08/11/text.911.mashable/index.html?eref=rss\\_topstories](http://www.cnn.com/2011/TECH/mobile/08/11/text.911.mashable/index.html?eref=rss_topstories)>.

# Appendix A - Code

## Full Code

```
#include <Wire.h>

#include <GSM.h>
#ifndef DEBUG_PRINT
  #error "!!! It is necessary to enable DEBUG_PRINT macro in the
  GSM.h !!!"
#endif

#define DEVICE (0x53)    //ADXL345 device address
#define TO_READ (6)     //num of bytes we are going to read
                        //each time (two bytes for each axis)

byte buff[TO_READ] ;    //6 bytes buffer for saving data read
                        //from the device
char str[512];          //string buffer to transform
                        //data before sending it to the serial port

int x, y, z, baseaccel;
float anglez;
int regAddress = 0x32;  //first axis-acceleration-data
                        //register on the ADXL345
int count = 0;
GSM gsm;

//Serial.print("made it through init");
void setup()
{
  gsm.InitSerLine(115200);
  gsm.TurnOn();
  gsm.SetSpeaker(1);
  gsm.CheckRegistration();

  delay(1000);
  Wire.begin();        // join i2c bus (address optional for
                        // master)
  // Serial.begin(115200); // start serial for output

  //Turning on the ADXL345
```

```

writeTo(DEVICE, 0x2D, 0);
writeTo(DEVICE, 0x2D, 16);
writeTo(DEVICE, 0x2D, 8);

readFrom(DEVICE, regAddress, TO_READ, buff); //read the
acceleration data from the ADXL345

//each axis reading comes in 10 bit resolution, ie 2 bytes.
Least Significat Byte first!!
//thus we are converting both bytes in to one int
x = (((int)buff[1]) << 8) | buff[0];
y = (((int)buff[3])<< 8) | buff[2];
z = (((int)buff[5]) << 8) | buff[4];
baseaccel = abs(x) + abs(y) + abs(z);
// Serial.print("made it through setup");
}

void loop()
{
if(count >= 100) {count=0;}
readFrom(DEVICE, regAddress, TO_READ, buff); //read the
acceleration data from the ADXL345

//each axis reading comes in 10 bit resolution, ie 2 bytes.
Least Significat Byte first!!
//thus we are converting both bytes in to one int
int x = (((int)buff[1]) << 8) | buff[0];
int y = (((int)buff[3])<< 8) | buff[2];
int z = (((int)buff[5]) << 8) | buff[4];
int accmag = abs(x) + abs(y) + abs(z);

float range = analogRead(A0) * 4.9 / 9.8;

//Object within ~5 feet
if(range < 60) {
if(count >= 50) {
DigitalWrite(9, HIGH)
}
if(count < 50) {
DigitalWrite(9, LOW)
}
}

if(accmag >= 2 * baseaccel){

```



```

while(accmag >= 2 * baseaccel){
    gsm.TurnOnLED();
    delay(10);
    gsm.TurnOffLED();
    delay(10);
    readFrom(DEVICE, regAddress, TO_READ, buff);
    int x = (((int)buff[1]) << 8) | buff[0];
    int y = (((int)buff[3]) << 8) | buff[2];
    int z = (((int)buff[5]) << 8) | buff[4];
    int accmag = abs(x) + abs(y) + abs(z);
}
char * orientation;
if((z > abs(x)) && (z > abs(y))){
    orientation = "CRASH DETECTED. CAR IS TOP UP";
}
if((z < abs(x)) && (z < abs(y))){
    orientation = "CRASH DETECTED. CAR IS FLIPPED";
}
if((abs(x) > abs(z)) || (abs(y) > abs(z))){
    orientation = "CRASH DETECTED. CAR IS ON SIDE";
}

while(!gsm.IsRegistered()){
    gsm.CheckRegistration();
    gsm.TurnOnLED();
    delay(50);
    gsm.TurnOffLED();
    delay(50);
}
gsm.SendSMS("9786975195", orientation);//strcat("CRASH
DETECTED. CAR IS: ", orientation);
    delay(10000);
    while(1){}
}
count++;
}

//----- Functions
//Writes val to address register on device
void writeTo(int device, byte address, byte val) {
    Wire.beginTransaction(device); //start transmission to device
    Wire.send(address);           // send register address
    Wire.send(val);               // send value to write
    Wire.endTransmission(); //end transmission
}

```

```
//reads num bytes starting from address register on device in to
buff array
void readFrom(int device, byte address, int num, byte buff[]) {
  Wire.beginTransmission(device); //start transmission to device
  Wire.send(address);           //sends address to read from
  Wire.endTransmission(); //end transmission

  Wire.beginTransmission(device); //start transmission to device
  Wire.requestFrom(device, num);   // request 6 bytes from
device

  int i = 0;
  while(Wire.available()) //device may send less than
requested (abnormal)
  {
    buff[i] = Wire.receive(); // receive a byte
    i++;
  }
  Wire.endTransmission(); //end transmission
}
```

## Appendix B - Weekly Summaries

### Status Report #1

May 16, 2011

#### Goals

The main goal for this week was to get the MQP started up. The first thing that needs to be done is registration for the project. We want to build a device that would add safety features (such as a rollover sensor) to a car. We first need to take a look at what kind of features are already available on the market to see what are the general things we want our system to be able to do. Then, we would need to figure out the specific features that we want to include in our system for this project.

#### Results

We have partial registration completed. There was an incorrect form available online for MQP registration that hindered progress. We should be able to get this done by next week's meeting.

For the features, we did some research and came to two important conclusions: 1) that we will not be able to make much use of the sensors already available on the cars and 2) that a lot of the safety features that we want to implement are already available for luxury cars. We will not be able to make use of the internal sensors of the car because car manufacturers are very secretive about access codes and it is very expensive to get a hold of them. The second point illustrates the need for this project. Manufacturers like Mercedes-Benz have features in their cars that enhance stability, assist braking, provide warnings for potential dangers, etc. The problem is that these features are limited to a small subset of cars; expensive, new, luxury cars have them, but most cars on the road do not. Therefore, we have decided that the primary goal of this project is to create a set of safety features that we will be able to easily integrate into any car, including those already on the road.

At this time, the one feature that has been agreed upon is a rollover sensor for a vehicle in motion. Our research showed that sensors are available that set off an alarm when a car tilts, but these sensors are

generally intended as anti-theft devices for parked cars. Our concern is for the safety of a moving vehicle. We will discuss addition features at next week's meeting.

## Status Report #2

May 23, 2011

### Goals

This week, we intend to finish up the registration process and decide on the specific set of features that we want to implement. We also want to work out how we want to carry out this project for the rest of the term.

### Results

Registration is almost done. The paperwork has been submitted online and we have registered with the ECE shop as well. We have also decided on a few features that we want to include in our system:

- Collision warning
- LED running lights

- 911 call on collision
- Tilt/rollover sensor (on a moving vehicle)
- Using LEDs as warning signals

Again, the idea is to be able to incorporate these features into any car on the road today. This would ensure safety for those who cannot afford luxury cars. The collision warning could be done using either infrared technology or SONAR. This would be beneficial for detecting obstacles while parking and also while driving on the road. The actual warning can be given to the driver through LEDs, sound, or both. The LEDs would also double as daytime running lights for vehicles that don't already have them.

For the 911 call feature, we at first thought that it would be too difficult to implement since we would need to somehow interface a cell phone with the car. Cars that already had a Bluetooth module are likely high-end cars that would not benefit from our system. We then realized that there are very cheap pre-paid phones that come with a limited amount of minutes, but can make 911 calls for free by law. If we can figure out how those prepaid phones work, we will be able to figure out how to implement a

cheaper and more accessible version on the OnStar feature available to a small number of cars on the market today.

For the tilt/rollover sensor, we will need to use a digital gyroscope and accelerometer. The trick is to make sure that going up a hill is not registered as the vehicle rolling over. Thus, we will likely need to use multiple modules placed throughout the car to interpret the results of the sensors correctly. These results could then be fed into the LED lights or a speaker.

## Status Report #3

June 2, 2011

### Goals

This week, we wanted to get started with a particular aspect of the project and flesh out some more details for it. We wanted to start thinking about the specific platform that we would use to bring everything together and decide which system to work on first.

### Results

We have decided to use an Arduino board using an Atmega microcontroller. This setup makes available a GSM module that can place regular phone calls using a SIM card and 911 calls without one. In addition to this, it also contains a number of both analog and digital I/O ports which will be necessary for both the sensor readings and control of the running lights and warning systems. There is also plenty of documentation available to help us with this process.



Our final product will be a series of integrated sensors and warning systems, but our design and build approach will be modular. That is, each sensor will be able to function independent of the others so that we can build and test one aspect of the project before working on the next one. This ensures that we will be able to interface all the parts more easily and that we don't have to think about the whole, more complex project when correcting for any errors we encounter.

Essentially, the build can be broken down into:

1. Design and build the first part.
2. Make sure that part is working perfectly.
3. Separately design the second part.
4. Make sure that part is working perfectly.
5. Make sure that the first two parts are able to work simultaneously.
6. Repeat for each of the next parts.

The Arduino board comes with its own development environment that allows for easy sensor reading and outputs. It can handle basic and

complex math including basic arithmetic, bitwise operations, loops, interrupts, etc. It is based on C/C++, so we have a powerful development environment.

We have decided to use an Arduino board using an ATmega microcontroller. This setup makes available a GSM module that is able to place regular phone calls using a SIM card and 911 calls without one. In addition to this, it also contains a number of both analog and digital I/O ports which will be necessary for both the sensor readings and control of the running lights and warning systems. There is also plenty of documentation available to help us with this process.

The block diagram on the following page shows the overall top-level layout of the project design:

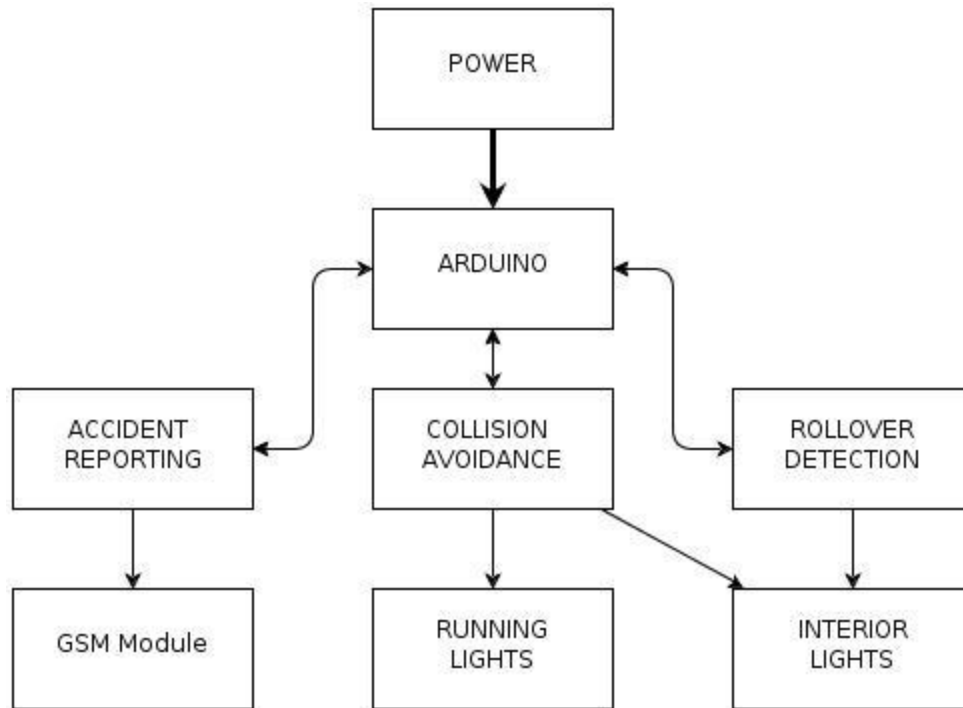


Figure 12 - Overall System Design

Once the board receives power, it can read information from and send information to the three big sensors: the rollover detection sensor, the collision avoidance system, and the accident reporting system. If it detects an accident, the GSM module is activated and an emergency call is made reporting the location of the accident to the authorities. If a possible collision is detected, both the running lights and interior lights are activated with a proper signal to warn both the driver and other drivers on the road. If a possible rollover is detected, the interior lights are activated with a different pattern to let the driver know.

We have decided to focus on getting the running lights working first because two of the sensors will use them. Then, we want to get the GSM module working with a SIM card. Obviously, we won't be able to place actual emergency calls just for testing. Once we do have it working with a SIM card, however, it will be trivial to switch it to make emergency calls. Once we have a prototype working with a SIM card, we will need to speak to the local authorities to both let know that this product exists so they can expect calls like this and also to ask them what is the best way for them to receive the information. We will also be able to get more information on how emergency calls are handled so that we can fine tune our product to make sure that it doesn't provide too much or too little information.

## Status Report #4

June 8, 2011

### Goals

This week, we intend to order most of the hardware needed and get started with building the device. We have already decided to use an Arduino UNO board and already have some LED strips (that serve as a warning signal) from previous projects. All that's left is the GSM shield (to make the emergency calls), the range finders (for collision detection), and accelerometers (for rollover and crash detection).

### Results

We have purchased all the parts and have tested the LEDs. We are currently waiting for some parts to be shipped (the accelerometer, the ultrasonic range finder, the GSM shield, and an antenna for the shield). We have already tested the PWM control for the LEDs and were able to vary the brightness and have the LEDs flash. This sets up the basic control that we need when a possible rollover or collision is detected.

For collision detection we have decided to use an ultrasonic range finder. After looking at various models, we decided to go with the Ultrasonic Range Finder - Maxbotix LV-EZ0. This model allows for detection over a large enough surface area that we will only need one in the front of a car and one in the back. Due to budget constraints, we have decided to only purchase one for now. For the scope of this project, a proof of concept will suffice and adding additional sensors would only be a monetary concern, not a technical one.

For the accelerometer, we found a three-axis model (Triple Axis Accelerometer Breakout - BMA180) that will allow us to monitor both rollover and accident conditions in one unit. We initially were going to use at least two separate 1-2 axis units, but this triple axis unit actually ends up being cheaper while still performing the functions that we need. It can handle detection of forces from 1-16g, which will suffice for detecting a car crash.

Finally, we have the expensive GSM shield called the GSM Playground. We selected this particular GSM module due to its capability

to make voice calls, simple Arduino interface, and extensive existing code base. Many modules we were able to find could only perform simple text message services and were thus considered unsuitable for our project, as we are doubtful of the ability to report emergencies with text messages currently. The module is an Arduino shield, meaning its profile is very similar to that of the Arduino board itself and all the connections for data and power are designed and positioned to connect directly to the microcontroller board without solder or additional wiring. The GSM draws power from the Arduino board and requires no additional connections, greatly simplifying the design of our project. Additionally, because the board was designed to interface with an Arduino development board and Arduino products have a large support community of hobbyist engineers, there is a large library of code for this Arduino and GSM implementation both from the developer of the GSM board and from customers who have published their work. The board also has several peripherals included, such as a temperature sensor, microphone, and audio amplifier, which may be useful for additional sensing and reporting to be incorporated into the

overall safety system. These factors make this particular GSM board an obvious choice for our project.

So we have the LED strips working, the new Arduino board is in (we were using a temporary one in the meantime), and we are waiting on the rest of the parts. Everything but the GSM shield should be here by today and the shield should be here by next week (since it is international shipping from EU). In addition to this, we spoke to a dispatch officer about the emergency calling feature to make sure that we set it up properly, but he was not allowed to give us any details. We have instead emailed another officer from the police department, who should be able to help us with some questions we have. We are still waiting to hear back from her.



## Status Report #5

June 16, 2011

### Goals

Our goal for this week was to have at least one sensor ready for demonstration. We will start out with the collision detection, then the rollover detection, and finally, the emergency calling. We have also set up a meeting with the Director of Emergency Communications from the Worcester Police Department to discuss the logistics of the 911 calling function.

### Results

We were able to get the collision detector working. It has a wide range and we were able to test it both at the lab bench and outside. It currently uses a speaker to send out a sound that gets higher pitched as an object gets closer to the sensor. We used a speaker to test out the functionality, but could easily incorporate blinking LEDs as warning signals. We were able to control the apparent frequency and the brightness

of the LED strips from last week, so the warning signal could be just the LEDs, just the speaker, or both. We will discuss which method is best during this week's meeting.

We were not able to get the accelerometer working during our meeting because we needed additional parts, but we were able to find the parts after the meeting and can continue to work on that for next time. We can use a voltage regulator and a car cigarette lighter adapter to mount the components internally. This would decrease the final product costs since we will only need to worry about protecting the collision detector from the elements.

We are still waiting for the GSM module, so we were not able to test it, but we were able to talk to the police and gain more information on how 911 calls are handled. The main problem we will have for this project is the fact that automated 911 from cell phones are not allowed in Massachusetts. Another piece of bad news is that we will not be able to solely rely on cellphone triangulation methods to pinpoint a location because it has too many factors for error involved. The police are able to

use this method in some cases, but the preferred method is to use GPS. The good news about this is that if we have a GPS, we could implement local emergency calling, which is allowed. A 911 call from a cell phone goes to the state police so we can't automate without getting into trouble, but if we have a GPS, we could use that to figure out which city a vehicle is in and call the local police with GPS coordinates instead. If we can demonstrate this functionality to the Director of Emergency Communications, he will be able to help us set something up with the state police so that we can test out a real 911 call. Even though we will not be able to use this in the final product, it will demonstrate that the product is fully working. We could then look into which states we are allowed to do this in and demonstrate the product there.

## Status Report #6

June 23, 2011

### Goals

For this week, we wanted to get another sensor working. Since the GSM shield is still not in, it will need to be the accelerometer. We were able to get a reading from the accelerometer previously, but only from one axis and we were not able to interpret that data correctly.

### Results

We were able to get a proper reading from the accelerometer. It can now read in data from all three axes and we can use that to determine both the speed and orientation of the car. Currently, we do not have it functioning to the point that we can get both the speed and orientation, but we have figured out a way to interpret the starting position of the car. Whether the car is on a hill or on a flat road, our system is able to take that starting point and detect how the orientation of the car changes. This will be used to figure out if the x-axis, y-axis, and z-axis, of the car or any

combination of the three axes is changing too quickly (i.e. to figure out if the car is in danger of rolling over, upside down, etc.). During our meeting with the police officer last week, we learned that this information about the current orientation of the car is very helpful in determining which kind of aid to send, so we definitely want to be able to use the accelerometer to get this part of the project working.

We have now also gotten hold of a voltage regulator and decided to use 12 volts as the base voltage for our device. The final product will be in a box containing all the circuitry and sensors, except for the range finder that will go under the hood of the car. The range finder will need to be on the outside of the car in order to detect obstacles. The power will be supplied from the cigarette lighter of the car receptacle, which provides 12 volts of power.

So, currently, we have the rangefinder working for a collision warning. We had hoped to get the accelerometer fully working for this week's meeting, but the code for it was a bit more complex than we had anticipated. The good news is that we were able to get reliable data from all

three axes by the end of our meeting, so we should be able to catch up in the next meeting.

Finally, the GSM shield has come in all the way from Europe and we can get going on the emergency call aspect of the project.

## Status Report #7

June 30, 2011

### Goals

We want to get our accelerometer finished up and get started on the GSM shield, which will be the most difficult part of the project.

### Results

We were able to get the accelerometer working to some degree, but think that we may have a defective part. We were able to get readings for all three axes and measure the tilt of the z-axis, but the readings from the x and y axes seem to be stuck. Looking on help forums for this part, it seems that this is a common issue. At first we thought it was just a bad

soldering job, but we redid that and verified that the connections are secure. At this point we think that either the chip on the board was damaged from the heat from the soldering process or that it was defective to begin with. In either case, we will need to get a new accelerometer.

We started work on the GSM shield as well. As expected, this part of the project is a bit difficult and so far we have been able to interface the shield with the board, but have not been able to make progress because the module is not picking up a signal.

Another aspect that we discussed during the meeting was that we would need a battery source to power the device, especially for the GSM shield since it should work in the event of a car crash. During a crash, we will not be able to rely on the power of the car, so we need to make sure that we have a physically separate unit that provides emergency power. In addition to this, we will need to use a voltage regulator even while using the power from the car because the available voltage can change dramatically, especially while starting up the car. If there are any large

spikes or drops in the available voltage or current, it might damage our device.

## Status Report #8

July 14, 2011

### Goals

For this week, we needed to get a new accelerometer since we've determined that the old one is broken. In addition to that, we will contact a student who worked on the rollover project to see if they have any insights for us. Finally, we wanted to make some more progress with the GSM board.

### Results

We were able to make significant progress on the GSM board. The main hurdle was to be able to make and receive calls. We were able to do that using terminal commands. The main issue we ran into was signal problems, but this was resolved as soon as we got outside the lab. We were able to both send a call and receive and pick up the calls. We were not able



to talk using the board, but that can be done relatively easily by adding a microphone. We hope to finish yjod by next week in addition to saving code on the device itself instead of using terminal commands.

We have purchased a new accelerometer and are waiting for it to arrive. We looked at various options, but decided on a similar breakout type model for the project because that is the easiest to interface with the Arduino board and development environment. Since we already have working code for it from previous weeks, we will simply need to make sure that it is able to function in conjunction with the GSM board and the collision detection system.

Finally, we are waiting to hear back from the students who previously worked on a rollover project.

## Status Report #9

July 28, 2011

### Goals

Our goal is to finish up working on the physical device. We are done with each individual component of the device. The new accelerometer has been installed and is working properly with the rest of the equipment. We also heard back from the students who worked on a rollover project previously.

### Results

The other group ended up using an accelerometer to detect the rollover, just like we are using. They also used a development environment instead of making everything from scratch. Their project, however, was much more focused on getting one component (rollover) to work very accurately on one particular model of car. Ours differs significantly since we intend to have our product function in any car.

The device itself is completed. All of the components work both separately and individually. The only wiring that needs to be done is to connect the device to the car battery and to run wiring for the rangefinder and LED strips, which use two wires each.

The device is now capable of placing a phone call and sending a text message to any number (including 911) with the orientation of the car if a large enough force is detected by the accelerometer. It also flashes LEDs if the car starts to tilt too quickly, and activates both LEDs and an alarm if something gets too close to the sensor. Since that physical device is completed, for the next few weeks, we will be fine-tuning the code to be as efficient as possible. We will also shift the focus on writing the final MQP report.