ECE 445

Final Report

Tire Tread Checking System

December 10th, 2014

TA: Samantha Knoll

Project No. 11

Byungchan Ryu
Jongho You
Rene Lamb

Table of Contents

1. Introduction

1.1 Statement of Purpose

The purpose of this project is to build a system that monitors the tread of a tire and alerts the driver when tread wear has reached an unsafe level. The system will be mounted above the wheel and use multiple sensors to measure the distance between the device and the outer and inner tire surface. Using these two distances, it will calculate the tread depth. This system will communicate wirelessly with the user’s Android phone to conveniently alert the car owner to the status of their tire and keep them safe.

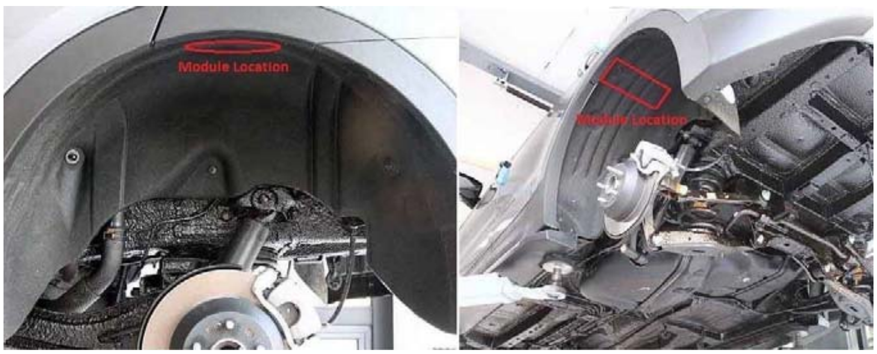

Figure 1: Module Location

Figure 2: Naming Tire Tread and Measuring Area of Tire

Figure 1 represents the mounting location for the system in the wheel well. Figure 2 represents the surface locations of the tire where measurements will be taken. Laser sensor will slide and takes measurements from the one side of the tire to the other side.

1.2 Objectives

1.2.1 Goals

- Design and build a system that monitors the tread of the tire with non-contact distance measurement sensors
- Design an Android application that interfaces the microcontroller and receives data and presents this to the user

1.2.2 Functions

- Measures and records the distance to the inner surface and the outer surface of the tire, and calculates the depth of the tire’s tread
- Make periodic (weekly or with a longer interval) measurements only when the vehicle is at a stop
- Alert user via Android app when the calculated tread is lower than the legal tread depth; 2/32 inches (1.59 mm)

1.2.3 Benefits

- Wireless communication provides a comfortable user interface
- Automatic alert system frees the user from having to worry or periodically check the tread
- Low-power consumption allows the system to have a long battery life

1.2.4 Features

- Plastic enclosure to protect the system from harsh environment.
- Shutter system to protect the sensors when closed
- Accelerometer detects when the vehicle is at a stop
- Sliding mechanism allows laser sensor to read the tire tread depth from one side of the tire to the other side.

2. Design

2.1 Block Diagrams & General Design Alternatives

In the initial design stage, we considered a number of different approaches to measure the tread depth of the tire. The basic idea was to use distance measuring sensors to measure the tread indirectly.

One idea was to use a combination of ultrasonic rangefinders and a laser rangefinder. Since some ultrasonic rangefinders have a resolution of 1mm, the idea was to use two ultrasonic rangefinders at each edge of the tire to measure the distance. Combined with the laser, the difference between the inner and outer surfaces can be calculated for the tread depth. The problem was that although the ultrasonic rangefinders are relatively cheap at $20 ~ 30 price tag, we could only afford a single laser rangefinder, meaning we would be able to measure the tread depth at only one location.

An alternative approach was using only the ultrasonic rangefinders with manual user input at the very first time the system is run. With this approach, the user would have to first measure and input the tread depth manually. From then, the ultrasonic rangefinder would monitor the distance from itself to the tire, and as the tire wears out, so would the distance from itself to the tire, meaning the tread depth has changed by such amount. We decided that this method is too indirect and prone to error, and also inadequate level of difficulty and completeness for the sake of Senior Design.

The approach we settled on was implementing a rail system with a single laser rangefinder for distance measurement. The rail system would mobilize the laser rangefinder to measure the distance at different locations on the tire, so that one laser rangefinder can allow us to calculate tread depth at multiple locations.

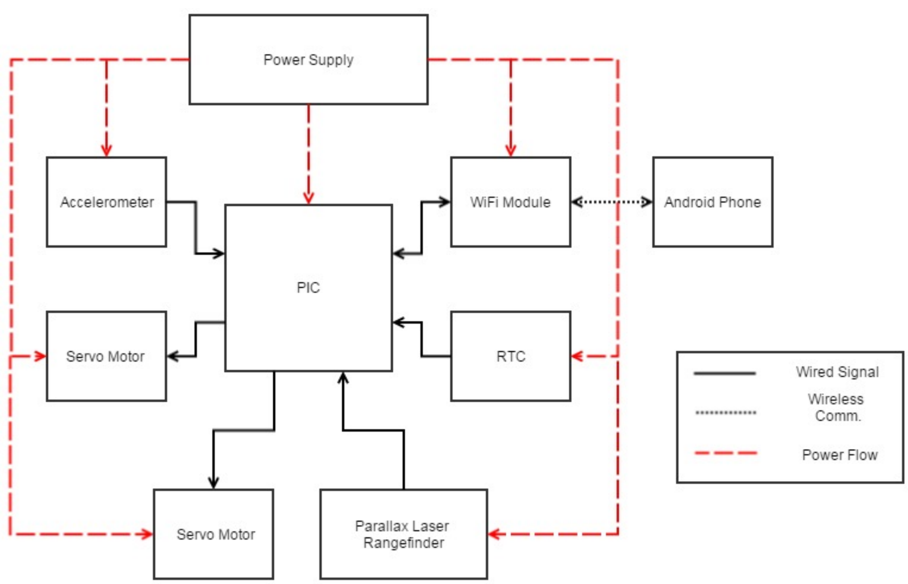

Figure 3 Top Level Block Diagram

Figure 3 is a high level modular view of the connections between all the electronic components within the system. The printed circuit board version is included in Appendix C. Arrows indicate the flow of data or power within the system. The microcontroller receives inputs from the Wi-Fi module and the laser rangefinder. Based upon the data collected at each measurement, the microcontroller controls the operation of the shutter servo motor and communicates with the user via their android phone. A separate power supply unit will provide the PIC with the power required. The microcontroller will also be programmed to control a servo responsible for the horizontal scanning of the laser from one end of the tire to the other.

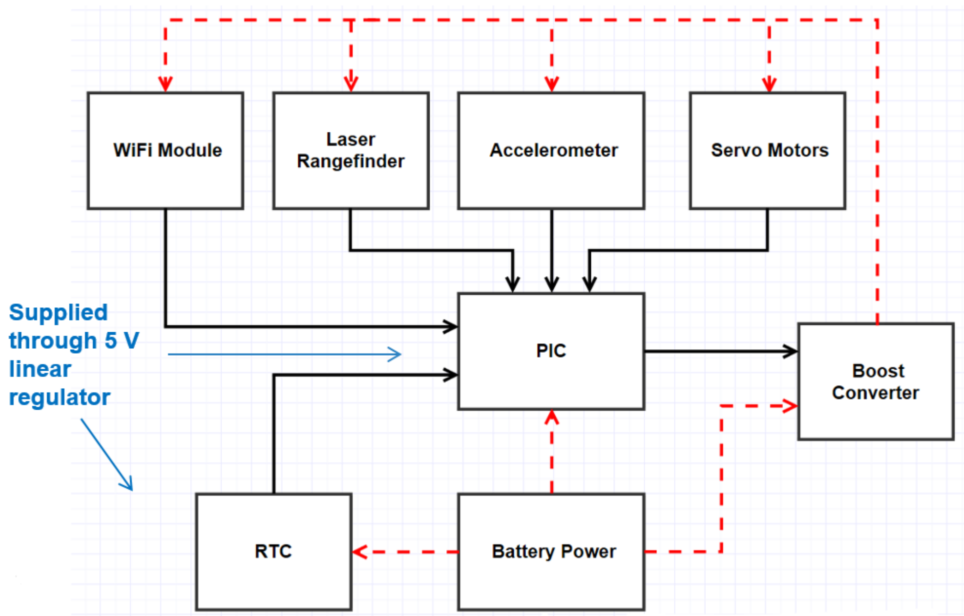

Figure 4: Power Supply Sub-Diagram

Originally, the system was designed so that the RTC and PIC would run directly off of 3 V from the battery power. The PIC that we had access to, however, was not the low voltage PIC required to run off the 3 V battery. Instead, the PIC required at least 4.5 Volts to operate. For the sake of testing and completing our proof of concept project, we opted to modify our power supply circuit and include a 5 V linear regulator within our design.

2.2 Block Descriptions

Parallax Laser Rangefinder

Laser rangefinders can detect smaller objects due to a small spot size of the laser beam. At 15cm the spot size diameter measured to be 3mm. The laser rangefinder will be used to measure the distance to the inner surface of the tire, so that the tread depth can be calculated.

The laser rangefinder we use is the Parallax Laser Rangefinder. The sensors run on 5V DC power and draw maximum 150 mA [3] during measuring process. It is connected to the microcontroller via its serial input, serial output ports and communicates through these ports. The spec range for this sensor is 15 ~ 122 cm with an average of 3% for error [3].

Wi-Fi Module

A Wi-Fi module is used for communication between the microcontroller and the user's android phone. With the module, tire depth data will be transmitted to the user who can then send a confirmation back to the microcontroller. The particular module we use is the RN171 WiFly module. This module runs on 3.3V connected directly to the 3.3V boost converter.

Power Supply

Tires can potentially last several years without needing to be changed. It is therefore important that the sensors power supply extend the battery life for as long as possible need not require frequent battery replacement. Additionally, PIC, laser rangefinders, servo motor, and Wi-Fi module, all require a relatively large amount of current when idle. Therefore, the tire tread sensor power supply must be able to source up to 1.036 A when required (See section 2.4 battery calculation). The battery supply consists of two 1.5 V AA batteries in series to produce approximately 3 V. These types of batteries have a relatively large charge rating (typically 3000 mAh for Li-ion) providing longer operation. As it can be seen in Figure 4, the battery pack directly supplies 3 V to the RTC and PIC. A 3.3 V and 5 V boost converter are used to supply the remaining devices. The battery pack in conjunction with boost converters was selected so as to minimize the amount of replacement batteries the end user would have to purchase when the batteries die. Boosting 3 V requires less standard batteries than stepping down the voltage from a value higher than 5 V (requiring at least 4 AA batteries).

Boost Converters:

A two boost converters where used to produce within 3% of 5V and 3.3 V to their respective devices with the capability to source up to 2.1 A. The 1230 converter is used to supply 5 V to the systems two servos and laser rangefinder. The 1232 boost converter is used to supply 3.3 V to the systems Accelerometer and Wi-Fi module. Each converter has a respective EN pin allowing voltage regulation when EN > .9 V [9]. Below .4 V, the converters enters shutdown mode, providing true circuit isolation between the battery and connected devices, thus reducing idle current draw. Supplied with 3 V these converters have an efficiency of approximately 90%. It is the high efficiency, potential 2.1 A output current, and circuit isolating ability that this particular boost converter was selected.

PIC Microcontroller:

The PIC16F877a microcontroller acts as the central hub of control and communication between the various components of the system. The PIC will be powered up once a week and take a depth sensor measurement. When not in use, the device operates in sleep mode, reducing current draw to 0.06 uA. Upon receiving an interrupt from the RTC, the device will begin the measurement process in Figure 11. Once a measurement is recorded the resulting laser data is processed by the microcontroller and the tread depth is calculated.

The PIC microcontroller attempts to trigger the converter enables once a week based upon an interrupt from the RTC if the vehicle is stationary at the time (the accelerometer does not detect movement). If it is not stationary, the PIC will wait an hour before attempting to enable again. If stationary, power is provided to the laser rangefinder, Wi-Fi module and servo motors. Once the distance measurements have been taken, the PIC disables the boost converter. The PIC and RTC draw relatively low current, allowing them to be always connected to the power source while maintain a reasonable device battery life.

Accelerometer:

The PIC should not be operated when the vehicle is moving. This is due to the risk of damage the tread depth sensor faces from debris. The accelerometer is used by the PIC microcontroller to determine whether not it is an appropriate moment to take the measurement. The ADXL345 accelerometer is a high resolution device, providing up to 4mg/LSB in the x, y, and z directions. Natural vibrations from a vehicle engine cause fluctuations in devices twos complement digital acceleration output values in the axis parallel to the engine. These fluctuations can be compared to fluctuations during an off vehicle to set an adequate “turn-on” threshold. A threshold value that has been surpassed lets the PIC know that the vehicle is on (Not safe to take a measurement).

Android Phone

The android phone is connected to the microcontroller via the Wi-Fi module. It will receive the tread data that the microcontroller has via Wi-Fi. Also it will have an application for this system which will provide a user interface that provides interaction with the microcontroller unit. The Wi-Fi module is connected to the microcontroller’s SPI network and the android phone will be able to access these ports on the terminal. The app will be built to separate the user from having to use the terminal directly.

Servo Motor

Arduino Analog 180 Micro Servo has three wires: Power (Red), Ground (Black), and Signal (Yellow). The power wire will be connected to the 5V pin, the ground wire will be connected to the ground pin, and the signal pin will be connected to a digital pin on the Arduino board. The servo motor will get a signal from the microcontroller as to whether it should close or open the sensor shutter. The shutter will be connected to the Servo Motor so that it slides and open when sensor needs to read the tread depth and close when the sensor needs.

Parallax Continuous Rotation Servo also has three wires: Power (Red), Ground (Black) and I/O (White). The power wire and I/O pin are typically connected to 5V pin. The biggest difference is that the continuous rotation servo is controlled by the signal with three-states: clockwise, counter-clockwise, and center. Unlike the regular servo motor, it can rotate in one direction as much as the user wants. Parallax Continuous Rotation Servo will control the movement of the laser rangefinder in the rail system by rotating.

Real Time Clock

The real time clock will provide the microcontroller with the external signal to wake up and begin measurement of the tires tread depth. The RTC provides a means for automatic measurements to be taken without the need of the user to interact with the tread depth sensor. An RTC will run directly off the battery pack. Additionally there exists a 3 V backup button cell battery port. This additional power supply will ensure that the measurement schedule of the sensor is not lost even when the primary supply power is low. The DS3234 RTC provides two convenient Alarms that can be programmed to generate a weekly interrupt. It is for this reason and the relatively low active current (700 uA) that this RTC was selected.

2.3 Test Setup

The ideal testing environment of our system is to fit the system in the wheel well of the car. However, our system would not function in the actual wheel well of the vehicle because adding the shutter and the rail system, we decreased the distance from our system to the tire, when the laser rangefinder has a minimum measuring distance of 15 cm.

This motivated us to build a setup do demonstrate that our system still works in model; a proof of concept. We implemented a shutter system and rail system into the test setup. Shutter system controls the shutter with servo motor in order to protect the sensor from hazardous conditions. It must stay open while rail system operates. Rail system controls the laser sensor along a slide, which is also controlled by another servo motor. The reason we built the rail system is so that we could use one laser sensor to take measurements at multiple spots.

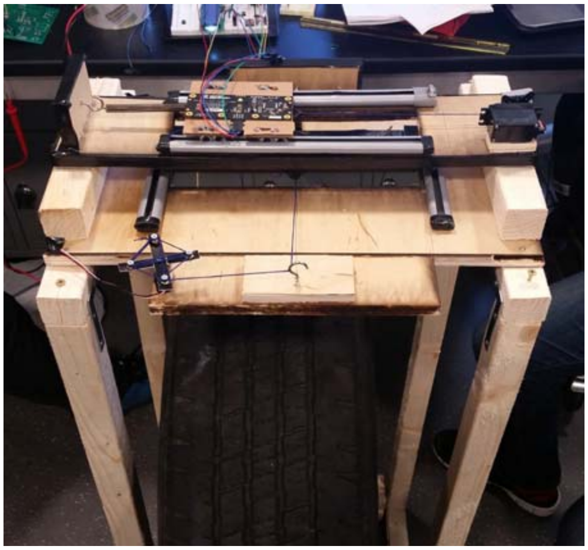

Figure 5: Final Test Setup

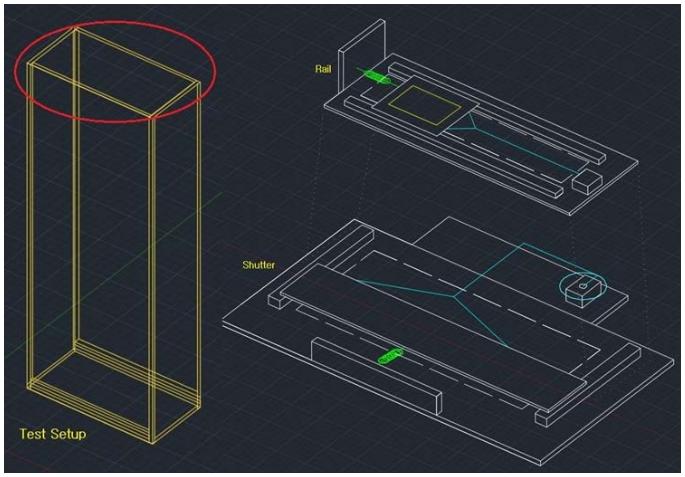

Figure 6: The Sketch of the Test Setup

Figure 5 and 6 are the final test setup we had for the demonstration. As you can see from the sketch in Figure 6, the rail system is placed on top of the shutter system and located in the red circled area of the test setup. The hyphen lines in both rail and shutter system indicate that the area is open and visible through. The strings (blue lines) are attached to the servo motors and control both the shutter and the laser sensor. The springs (green) pull both the shutter and the laser sensor back into the original position.

2.4 Equations, Simulations, General Circuits

Power Supply Battery Life Calculation

Higher current components such as the servos and laser rangefinder can drain a battery quickly if always powered on. The main power saving methodology behind our system is that tire tread depth measurement only needs to be taken once a week for a short period of time. This short period (<= 60 seconds) in which a measurement is being taken is called the active mode of the device. The period within the week in which the device is idle is called the sleep mode. The battery life calculation shown below was used throughout the design process to gauge the power savings and ultimately influenced component selection and circuit design. It should be noted that during actual testing of the system, current draw never exceeded 300 mA. However, for conservative calculations, we decided to use max current as the sum of total currents drawn from each device. These individual current where obtained from their respective device data sheets and are summarized in Table 1. After timing system sensor measurement, the measured time to take a full measurement was 60 seconds. At 20 MHz frequency operation and 5 V supply the PIC chip consumes 5.5 mA. In sleep mode it consumes .06 uA [7].

Table 1: Worst Case Current Draws of Each Block

Devices	Sleep Mode Currents (uA)	Active Mode Currents (mA)
PIC	.06	5.5
Boost Converter (x2)	3	.003
RTC	700	.700
Laser RF	0	150
Wi-Fi Module	0	180
Accelerometer	0	.140
Servos (x2)	0	700
Totals	703.06	1036.3

$$1\ Week\ (wk) = 604800\ seconds\ (s)$$
$$Active\ Mode\ time = 60s;\ Sleep\ mode\ time = 604800 - 60 = 604740s$$
$$Weekly\frac{mAs}{wk} = 1036.3mA * 60s + .70306mA * 604740s = 425168.5\frac{mAs}{wk} = \frac{118.1mAh}{wk}$$
$$Battery\ Life: \frac{3000mAh}{118.1mAh/wk} = 25.4\ weeks$$

Table 1 contains the maximum current draw of all the parts. The calculation of the battery life is included above.

3.3 V/ 5 V Boost Converter Calculations

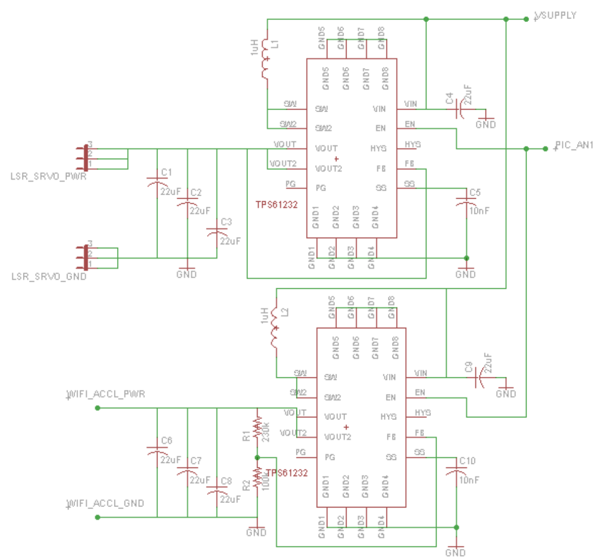

Figure 7: Boost Converter Schematic Diagram

Figure 7 shows the schematic of the two boost converters we are using. These boost converters provide boost regulation for input voltages between 2.3 V and 5.5 V. The output voltage is easily adjustable via external voltage divider. Over linear regulators, switching converters provide higher efficiency. According to the datasheet, at 3.0 V and the max expected current draw of 1036 mA. The efficiency is greater than 90% according to Figure 8 below.

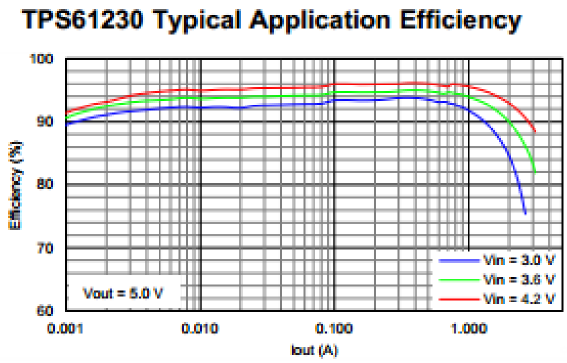

Figure 8: Boost Converter Efficiency Plot vs I_{Out} [9]

The switching of the internal gates is controlled internally and operates at a frequency of 2MHz The main external components of these boost converters are an inductor used to store energy when internal MOSFET is closed, the output capacitors used to store transferred energy and maintain a voltage on the load when the MOSFET is opened, and an additional input decoupling capacitor. Component selection was mainly based upon datasheet recommendation, however, these suggested values where validated by hand as well to ensure that the recommended values where appropriate for our design. Within Table 2 the calculated minimum required inductances and capacitances for each converter are shown.

In order to set the proper output voltage levels of the boost converters, the appropriate resistor values between V_{out} and FB (feedback voltage) and FB and ground had to be calculated. According to the TPS6123x data sheet, the output will be 5 V regulated when V_{out} is connected to FB. However, for producing the 3.3 V regulation the resistor values where calculated as follows:

$$Vout = Vfb * (1 + \frac{R1}{R2}) \qquad (1)$$

The data sheet also explains that the current through R2 should be at least 100 times larger than the FB pin leakage current so and R2 values should be kept at or below 100 kΩ. However, a higher R5 values increases efficiency. Choosing V_{fb} to be 1 V and R2 to be 100 kΩ we get the following:

$$3.3\,V = 1\,V * \left(1 + \frac{R1}{100k\Omega}\right) \rightarrow 2.3\,V = \frac{R1}{100k\Omega} \rightarrow R1 = 230k\Omega \qquad (2)$$

Additionally, an appropriate inductor value has to be selected for each regulator. The regulator is designed to work with 30% inductor current ripple (Δi_L) and the appropriate current ripple was calculated estimating that the maximum current draw from the system would be less than 1.8 A ($I_{out(max)}$). This is shown in equation 3.

$$\Delta iL = .3 * Iout(max) * Vout/Vin \qquad (3)$$

After ΔI_L has been determined the minimum inductor value to be used can be calculated knowing the nominal switching frequency (f_{sw} = 2 MHz) of the regulator and using the following equation.

$$L(min) = Vin * (Vout - Vin)/(\Delta iL * fsw * Vout) \qquad (4)$$

The results of these calculations for equations 1-4 are summarized in the table below.

Table 2: Boost Converter Calculation Results

Component/Value	3.3 V Regulator	5 V Regulator
R4 (kΩ)	230	N/A
R5 (kΩ)	100	N/A
$I_{out(max)}$ (A)	1.8	1.8
V_{in} (V)	3.0	3.0
V_{out} (V)	3.3	5.0
f_{sw} (MHz)	2.0	2.0
Δi_L (A)	.59	.9
L_{min} (uH)	.23	.6
C_{out} (uF)	1.6	2.76
D	.18	.46

The input capacitors (C1, C2, C5, C6) where selected according to the recommended capacitor values in the data sheet (22 uF). The output voltage is desired to remain within 3% of the V_{out} (ΔV_{out}). The following equations where used to determine the minimum circuit output capacitance (though ultimately the datasheet recommendation was used in the application).

$$Cout(min) = (Iout(max) * D)/(fsw * \Delta Vout) \qquad (5)$$

D is the duty cycle of the regulator which can be calculated using equation 6 and estimating the efficiency of the regulator (η) to be 90% [9] .

$$D = 1 - (Vin * \frac{\eta}{Vout}) \qquad (6)$$

These values for equations 5 and 6 are expressed in Table 2.

Spring Rate Calculation

For both of rail and shutter systems, the decision of the spring rate is also important. I have to make sure that servo motor is powerful enough to slide the laser sensor and the shutter block. In order to calculate the appropriate spring rate, I used this equation:

$$\tau \; of \; Servo \; Motor - 10\% \; of \; \tau \; of \; Servo \; Motor > k * \Delta x$$

τ = Torque (lbs/inch)
k = Spring Rate (lbs/inch)
Δx = the stretched distance of the spring

The reason why I subtracted 10% of torque of servo motor is that there will be a friction within the slide of rail.

Rail system
Torque of Parallax Continuous Rotation Servo is: 38oz/inch (≈2.375 lbs/inch)
Since the tire we tests has width of 245mm, the laser sensor will be able to slide about 10inchs long. Therefore, $\Delta x \approx 10\ inches$

Therefore,

$$2.375 lbs/inch - 0.2375 lbs/inch > k * 10 inches$$
$$k < 0.21375\ lbs/inch$$

So, I decided to use the spring rate of 0.10 (lbs/inch).
Decided to use two springs for better stability.

Shutter system
Torque of Arduino 180° Servo is: 1kg/cm (≈5.5997 lbs/inch)
Since the width of the laser rangefinder is 45mm, the shutter block only need to slide about maximum length of 2 inches. Therefore, $\Delta x \approx 2\ inches$
Therefore,

$$5.5997 lbs/inch - 0.55997 lbs/inch > k * 2\ inches$$
$$k < 2.5199\ lbs/inch$$

So, I decided to use the spring rate of ≈1.00 (lbs/inch).
Same as the rail system, decided to use two springs for better stability.

Unlike the Rail system, Arduino 180° Servo has rotation capability of 180°
Therefore, the minimum radius (r) of the rotation part of servo should be:

$$2\pi r * \frac{180°}{360°} \geq 2 inches$$
$$r \geq 0.6369\ inches$$

The minimum radius of the rotation part of Arduino 180° Servo must have equal or greater than 0.6369 inches (≈ 16.17mm)

Accelerometer Validation

Vehicle movement determination is important with our design because without it, measurement could be taken during time when sensors could be damaged by debris. Additionally, measurement during vehicle movement would be inaccurate due to the moving tire. The ADLX345 accelerometer provides a digital 16 bit twos complement representation of accelerations experienced in the x, y, and z direction with a 4mg/LSB resolution. The idea behind the accelerometer is that it is used to detect vehicle vibrations that occur when a vehicle is on. The accelerometer must be sensitive enough to determine if a vehicle is on but not moving vs when the vehicle is off. This is necessary or else an accidental reading might be initiated if the vehicle temporarily stops (e.g. stop light). The validation in our design choice was made based upon test data on an actual vehicle. An Arduino Uno was connected to the accelerometer mounted to the hood of the vehicle.

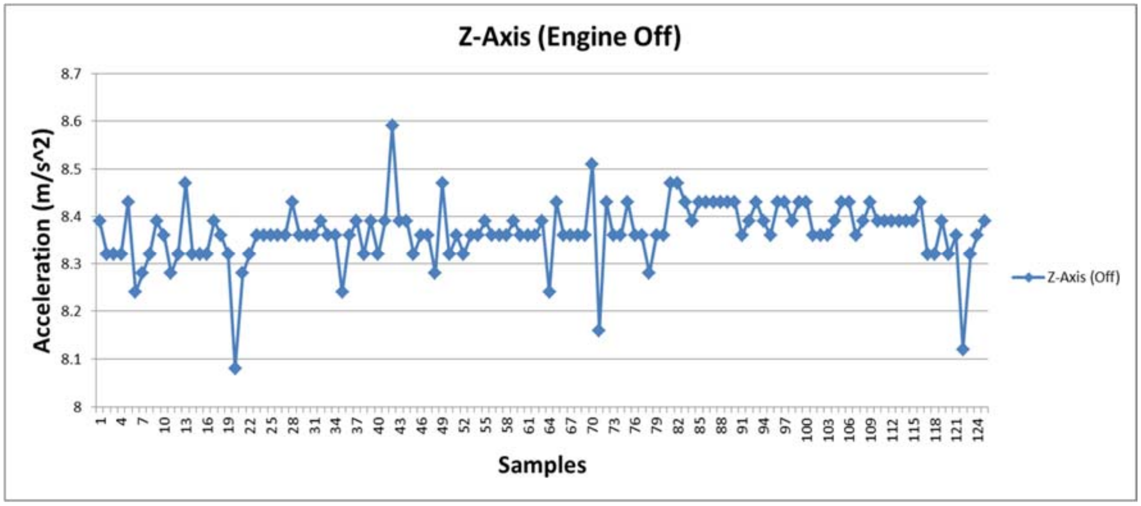

Figure 9: Accelerometer Readings with Vehicle Engine Off

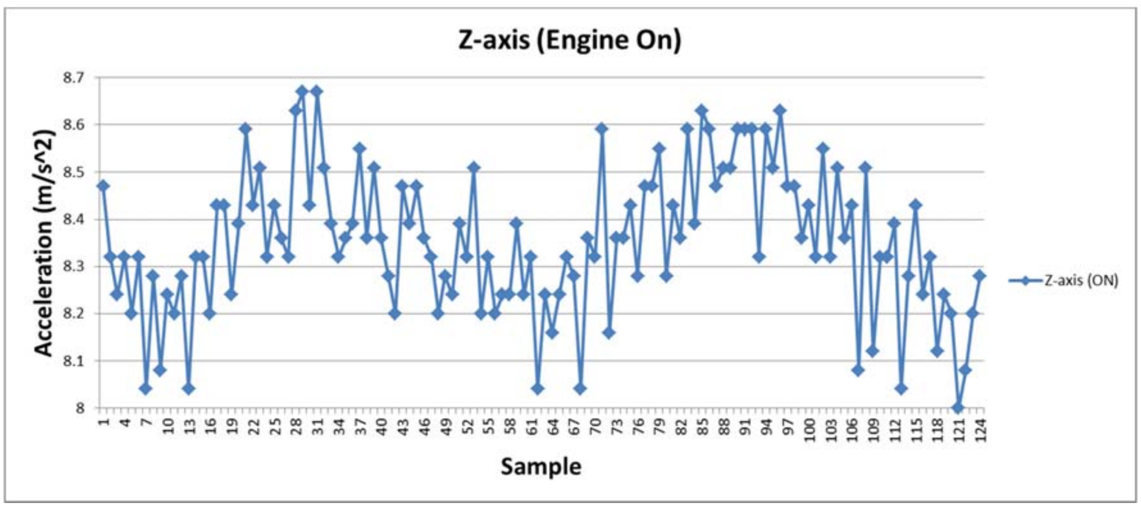

Figure 10: Accelerometer Readings with Vehicle Engine On

Table 3: Accelerometer Reading Comparison

	Engine Off	**Engine On**
Average (m/s^2)	8.367	8.351
Standard Deviation	0.06514	0.15234

Figures 9 and 10 represent data from the test. In each test, 124 acceleration samples were collected. The corresponding accelerations are shown on the y-axis. Based on this data, a threshold value could be set. When polling the accelerometer, the PIC can calculate the standard deviation of a set sample of values and determine if a vehicle is on if its standard deviation is above this threshold. It should be noted that most significant fluctuations occurred in the Z-axis parallel to the engine. Though the output was an acceleration value, the actual acceleration value is not important in determining whether a vehicle is moving. Examining the unscaled raw accelerometer data in integer form would show greater value variation that can also be used to determine a threshold.

2.5 Circuit Schematics/Drawings/Flow diagrams

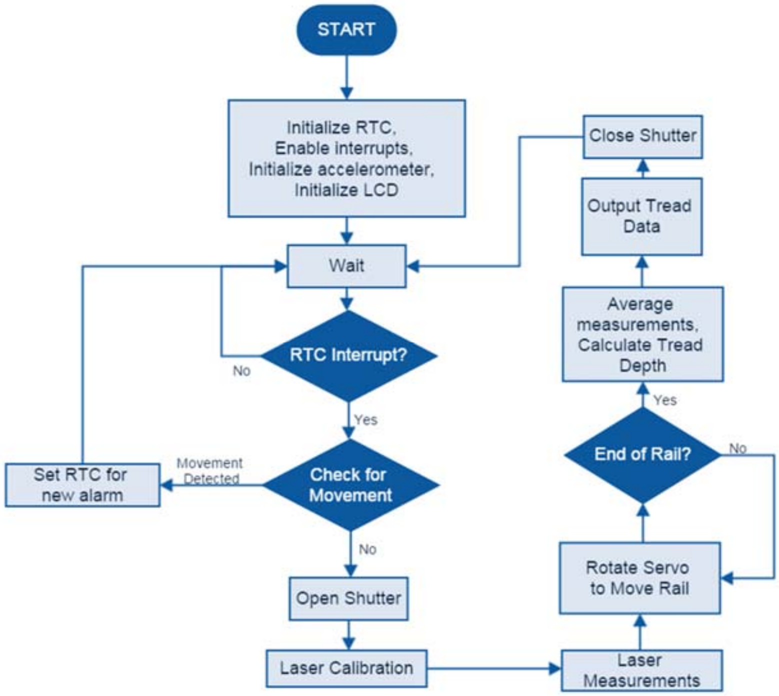

Figure 11: Flowchart of the Microcontroller Operation

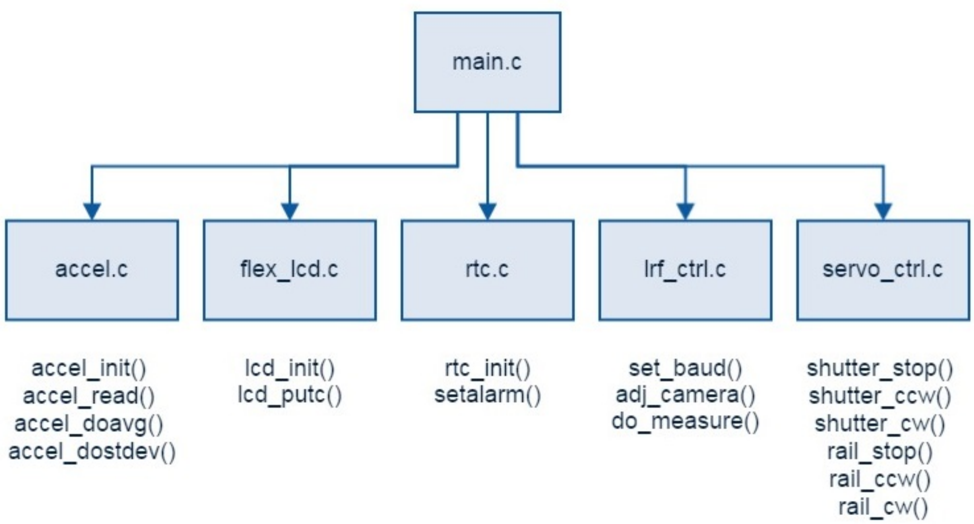

Figure 12: Code Structure for the Microcontroller

Figure 11 describes the operation of the microcontroller. This is the flowchart of the actual operation of our microcontroller for the final demonstration.

Once it is powered on, the PIC will initialize its components and enable interrupts. After initialization, the PIC will wait for the RTC interrupt to occur, which is preset to 1 minute for the demonstration. For the actual implementation, it can be adjusted to a day or a week. If the RTC interrupt occurs, it moves on to check the accelerometer for movements. If the accelerometer returns a value above threshold, it resets the RTC for a new alarm and returns to the wait status.

Otherwise, it will proceed to the measuring stage, where it opens the shutter, calibrates the camera on the laser sensor and makes measurements. After it measures each location, the servo motor will move the laser sensor to the next location. Once all locations are measured for distance, PIC will calculate the tread depth based on the data. The measurements are output to the LCD screen, shutter is closed and the PIC returns to the wait loop until the next RTC interrupt occurs.

Figure 12 shows the structure of the code for the microcontroller. The main.c file contains the above algorithm. The remaining C files contain the code necessary to interact with the peripheral blocks. Each required function from the devices are written into wrapper functions so that they can be easily accessed by the main.c file for readability.

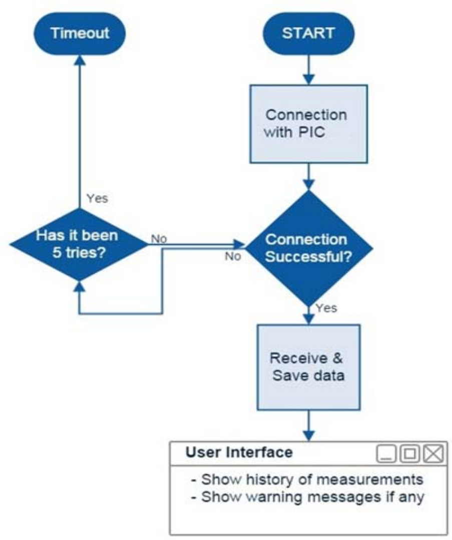

Figure 13: Flowchart of Android Application Operation

Figure 13 illustrates the proposed flow for the Android application. When the Android app is started, first it will try and connect to the microcontroller. Once it is connected, it will receive and save the data into the app's database. The user interface will then open up, which will provide the user with a history of measurements made by the system, and show any new warning messages. Warning messages will be of three kinds; a failed system boot warning, a corrupt data warning, and a low tread warning.

The failed system boot warning will occur when the PIC cannot initiate its components properly. The corrupt data warning will occur when the tread calculation comes to less than 1mm, which could mean that the tire tread is filled with alien material such as mud or snow. A low tread warning will alert the user if the tire tread is less than 3 mm.

3. Verification

3.1 Testing Procedure & Results

3.1.1 Laser Rangefinder

To verify its two requirements in Appendix A, the laser rangefinder was placed approximately 20 cm from the tire. It measured the distance to a tread location for both inner and outer surfaces, 20 times each. The same test was performed twice for repeatability.

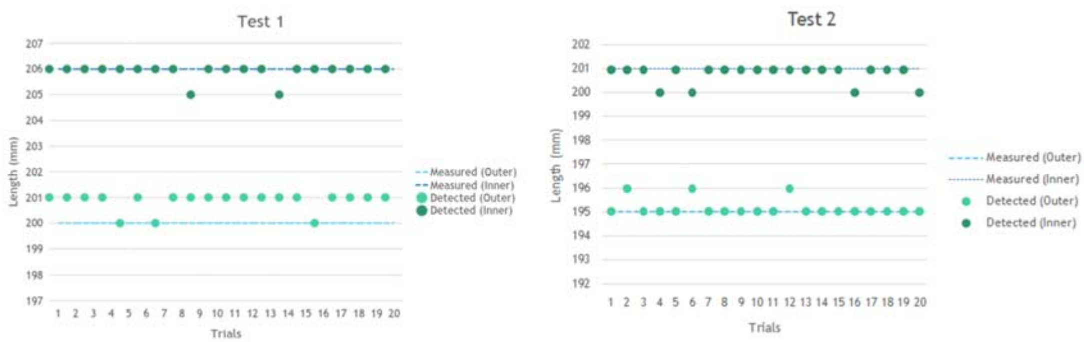

Figure 14: Laser Rangefinder Distance Measurements

Table 2: Laser Rangefinder Testing Results

	Test 1		Test 2	
	Outer Surface	Inner Surface	Outer Surface	Inner Surface
Actual Distance	200mm	206mm	195mm	201mm
Measured Distance	200.85mm	205.9mm	195.15mm	200.8mm

Figure 14 shows the distance measurements plotted into two separate plots, for each test. Table 2 shows the results calculated into averages. It shows that the laser rangefinder measured the distance to both the inner and outer surfaces within 1mm of accuracy. This verifies both the requirements of the laser rangefinder as mentioned in Appendix A.

3.1.2 Power Supply

The steps for validating the 5 V boost converter are shown in Appendix A. This validation included applying 3 V to the boost converter and achieving between 4.85 and 5.15 V, Disabling the converter and observing that the output voltage is 0 V, Sweeping the voltage and observing that the fluctuating voltage maintains regulation within the stated range. The voltage across the regulator was read from a multimeter. The 5V regulator passed these tests with the voltage equaling approximately 4.9 V. Due to the small size of the regulators, soldering was an issue. Tests performed attempting to mount the device to a breadboard where not successful. The device needed to be tested on a PCB and so for validation purposes, the device was mounted to a separate PCB for modular power supply testing. Unfortunately, we were unable to get the required voltage values from the 3.3 V boost converter.

3.1.3 Accelerometer & RTC

Though the system is designed to operate at a weekly interval, this would not be a practical testing period. Therefore, though the RTC is capable of weekly interrupts, we scaled the time back and instead validated this component with an interrupt period of 1 minute. After the components where connected within a breadboard, the test LCD displayed tread depth sensor measurement repeat every minute. Every time a new measurement began a stopwatch was used ensuring that the RTC was accurate. The RTC was accurate and met our expectations.

The Accelerometer was tested in conjunction with the RTC. After each interrupt, the PIC completes 20 samples of the accelerometer and outputs the standard deviation. When no vibration is detected, the system continued its measurement process by controlling the laser and servos. If vibration was detected, the system successfully stopped its measurement process and the RTC was reset to generate another interrupt in a minute. The source of vibration for validation was a cell phone controlled by a vibration application. During testing, the stationary standard deviation was 0.00. When vibrated by phone, the standard deviation was >= 0.01. Therefore, our threshold value for this particular test was 0.01. For our first test, the outputs are a scaled version of the acceleration in which the outputs represent G forces experience in the Z-direction (1 G = 9.8 m/s^2) perpendicular to the floor. The graph below shows stationary g-forces vs vibrational g-forces for one recorded trial with (STD = 0.01) and without (STD = 0.00) vibration. The tabulated data for this test is recorded in Appendix D.

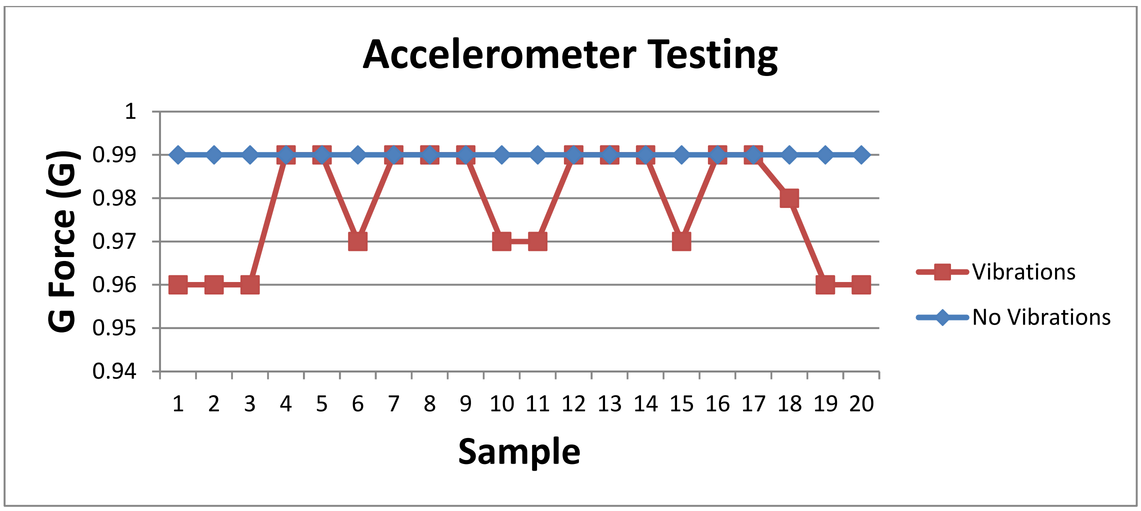

Figure 15: Accelerometer Testing Plot

3.1.4 Rail and Shutter System

Servo for the rail system required to move counter-clockwise, clockwise, and moves the rail in 5mm ±0.1mm units. We connected the servo motor to the PIC and rotated servo in both counter-clockwise and clockwise. Since we used short radius servo arm, we verified that the servo could move the laser sensor as we required.

Servo for the shutter system required to move from 0° to 180°. We connected the servo motor to the PIC and tested if it rotates as we required. We adjusted the pulse width and verified the servo rotates from 0° to 180°. It is important because shutter should fully open, while the laser sensor in the rail system takes the measurement.

3.1.5 Complete System

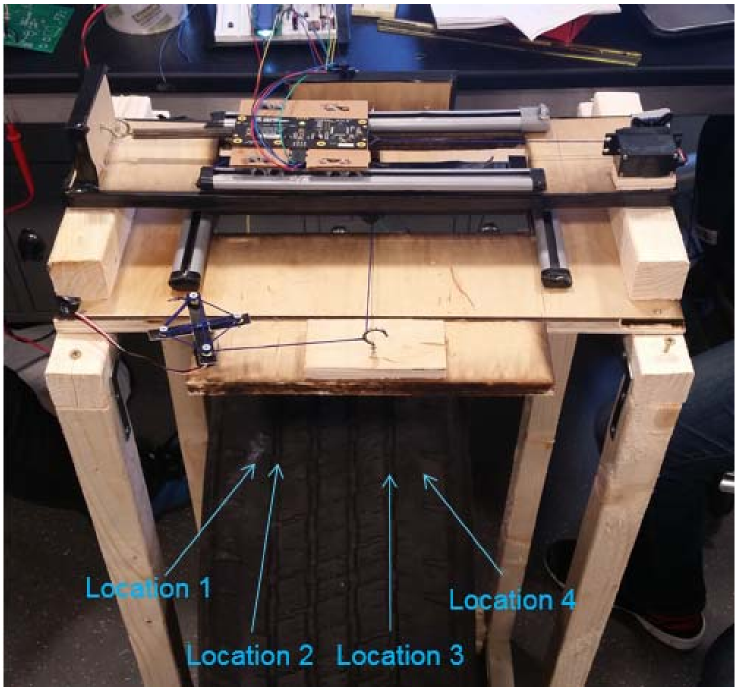

Figure 16: Laser Rangefinder Distance Measurements

The complete system we built measures in the range of 250mm (25cm). Since the laser sensor gives the most optimal result around 15 to 20cm, we did the testing on 20cm with new setup. Figure 16 indicates the locations that the laser sensor takes measurement. Location 1: Left Outer Surface, Location 2: Left Inner Surface, Location 3: Right Inner Surface, and Location 4: Right outer Surface

Table 3: Laser Rangefinder Testing Results

Distance Measured (avg from 20 measurements)	Location 1	Location 2	Location 3	Location 4
Original Test Setup	245.2 mm	247 mm	249.2 mm	248 mm
New Setup	193 mm	197.5 mm	194.8 mm	192 mm

Table 4: Laser Rangefinder Testing Results

Calculated Tread Depth	Left Side	Right Side
Original Test Setup	1.8 mm	1.2 mm
New Setup	4.5 mm	2.7 mm
Actual Tread Depth	4.5 mm	3.5 mm

In table 3 and 4, the result shows the significant improvement of the measurement accuracy. With original test setup, the left side tread depth was 1.8 mm while actual tread depth taken by the ruler is 4.5mm and the right side tread depth was 1.2 mm while actual tread is 3.5mm. As we changed in new setup, we got the result of 4.5 mm on the left side tread and 2.7mm on the right side tread.

3.2 Failed Verifications

Originally, we planned to incorporate a Wi-Fi module which would send the tread depth data remotely to the Android phone. We also planned to build an Android application which would provide a user interface for the system, allowing the user to interact with the system via an Android mobile phone. Due to schedule delays, we were unable to meet these deadlines. For the Wi-Fi module, we could not establish an ad-hoc connection which would have been the starting place to interact with the PIC remotely. The laptop computers that we worked with could not establish an ad-hoc connection because the manufacturer's guide states that 'The automatically assigned IP address must be on the 169.254.x.y subnet, otherwise the WiFly module will not be accessible' [8]. We tried to reconfigure using ipconfig and disabling the local networks on the laptop, but the laptops always connected on 10.0.x.y subnet. There was not an easy fix to this problem, and we could not get it solved before the final demonstration.

Due to issues with soldering, and a lack of additional boost converters, The 3.3 V regulator failed to meet our validation requirements. Soldering the boost convert to a surface mount to DIP test PCB proved ineffective. This is likely due to the devices susceptibility to electromagnetic interference and high frequency nature of the device. Soldering itself was difficult and multiple devices where soldered incorrectly. Air reflow soldering had to be used in conjunction with a copy of our PCB to separately test the converters.

4. Cost

4.1 Cost Analysis

Table 4: Labor Cost Calculation

Name	Work hours	Hourly rate	Total Labor Cost Estimation
Rene Lamb	20 hours/week	$35	20hrs/week * $35 * 16 weeks = $11200
Byungchan Ryu	20 hours/week	$35	20hrs/week * $35 * 16 weeks = $11200
Jongho You	20 hours/week	$35	20hrs/week * $35 * 16 weeks = $11200
Total	60 hours/week	$105	$33600

Table 5: Parts used and Costs

Item	Quantity	Manufacturer	Part Number	Cost/Unit	Total
Microcontroller	2	MicroChip	PIC16F877a	$5.08	$10.16
Laser Sensor	1	Parallax	28044	$109.08	$109.08
Servo Motor	1	Parallax	900-00008	$12.59	$12.59
Servo Motor	1	Arduino	T010050	$7.30	$7.30
Batteries (24 pack)	1	Energizer	B00451SSBI	$0.69	$16.50
Accelerometer	2	Analog Devices	ADXL345	$3.54	$7.08
Accelerometer Breakout	1	Analog Devices	ADXL345	$17.95	$17.95
Wi-Fi Module	1	Roving Networks	RN-XV	$34.95	$34.95
5 V Boost Converter	2	TI	TPS61230	$3.00	$6.00
Variable Boost Converter	2	TI	TPS61232	$3.00	$6.00
Real Time Clock Breakout	1	Maxim Integrated	DS3234	$19.95	$19.95
Real Time Clock	2	Maxim Integrated	DS3234	$8.58	$17.16
Arduino	1	Arduino	Uno-R3	$20	$20.00
22uF Capacitor	10	Yageo	CC0603KRX7R7BB103	$0.22	$0.22
10nF Capacitor	10	Murata	GRM21BR60J226ME39L	$0.135	$1.35
100KΩ Resistor	4	Vishay	CRCW0603100KFKEA	$0.08	$0.32
Custom PCB	1	Silver Circuits	-	$85	$85.00
1uH Inductor	4	ECE Store	-	$1	$4.00
Android Phone	1	Samsung	Galaxy Nexus	-	-
Test Setup	1	-	-	$50	$50

Table 6: Total Costs

Type	Cost
Labor	$33,600.00
Parts	$425.61
Total	$34,025.61

5. Conclusion

5.1 Accomplishments

By the final demonstration, we were able to show the most important component of our design working. We successfully measured the tread depth at two locations within 1 mm of accuracy. We were able to trigger the start of the measurement by the interrupt from the RTC, and the accelerometer checked whether there was movement or not before the measurement cycle continued. The rail system moved the laser to the correct position every time, and the shutter opened and closed, providing protection and the ability to measure at the same time.

Although we failed to implement the Wi-Fi module and the Android application, and could not demo on an actual vehicle, we successfully showed this proof-of-concept that our design, upon improvements, could serve people to provide a real-world solution.

5.2 Ethics Statements

We as a team recognize the importance of ethical and professional conduct throughout the course of this project. Here are the parts of IEEE Code of Ethics[1] that we feel are related to our project.

1. **To accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;**

We realize that our system provides a valuable information with regards to the safety of the driver and the passengers of the vehicle. However, we do not suggest to solely rely on our system for the safety of the vehicle, if it were actually used on a real vehicle since we cannot guarantee the safety of the tires and vehicle based on our system.

2. **To be honest and realistic in stating claims or estimates based on available data;**

Throughout the project we tried to only make claims that are realistic and have data or calculations to back our claims.

3. **To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;**

We gladly accept any professional criticism and advice to our design and correct our errors based on these criticisms.

4. **To assist colleagues and co-workers in their professional development and to support them in following this code of ethics.**

We worked as a team and try to help each other by sharing our strengths and expertise to create synergy as a team.

5.3 Future Works

Many new ideas came up during the length of the project that could be incorporated to further improve the design.

The most crucial improvement needs to come from the sensor itself. A sensor with a closer minimum measuring distance and an improved accuracy could make this project more realistic and dependable. Our sensor also only functions in the temperature range of 0 to 50 ºC, which is not suitable for vehicles that operate in cold regions. If a sensor with these improvements can be manufactured, our system would be one step closer to mass production.

Another idea for future works was a need for a sweeping algorithm that would detect the location of the treads automatically. Instead of having the user input the location of the treads to be measured, if the laser could measure in a small interval, map the data, and calculate the location of the treads to be used in later measurements, it would be more comfortable for the user.

Other ideas included the ability to stop the measuring cycle if the vehicle started moving after the measurement begins; a more accurate rail system; a better solution to detecting whether the car is in motion or in engine idle or completely powered off.

Appendix A Block Level Requirements and Verification

Item	Requirements	Verifications
Parallax Laser Rangefinder	Sensor measures distance to an object 20 cm away with a resolution of 1mm	1. Position the laser distance sensor 20 cm away from object. 2. Connect the sensor to the PIC microcontroller. 3. Make the laser output measurements continuously. 4. Move the laser towards the object by 1mm. Observe the measurement output.
	Sensor reports tread depth to ±1mm.	1. Place the sensor such that laser is pointing into the inner and outer surfaces of the tire (LT245/75R16) 2. Make measurements by interfacing with PIC microcontroller. 3. Compute tread depth by calculating the difference.
Accelerometer	Standard deviation of Z-axis acceleration samples produce STD = 0.00 when no vibration detected and STD >= 0.01 when vibration detected	1. Break out accelerometer on breadboard with test LCD Display to output collected data. 2. Perform 20 samples of Z-axis data. 3. Display standard deviation of sample on LCD and verify that STD = 0.00 4. Repeat step 2 with accelerometer breadboard resting above cellphone made to vibrate with vibration app 5. Display standard deviation of sample on LCD and verify that STD >= 0.01
Wi-Fi Module	Use Wi-Fi interface to send and receive Serial Data	1. Connect Wi-Fi module to the PIC board, set up an ad-hoc connection. 2. Use telnet from a laptop, send and receive commands
PIC Microcontroller	When powered on, controller draws < 1mA	1. Use ammeter to measure current draw from power supply with accelerometer, RTC, Wi-Fi module, laser distance sensor, and the servos connected to the PIC
	Can use flash memory to store data	1. Read off an integer value from ROM. 2. Increment by one and save it to the same address. 3. Power off the PIC, wait for a few seconds and power on again. 4. Access the integer and display in LCD. Repeat above steps.
	Measurement process begins after RTC interrupt	1. Program PIC to turn on flag after an RTC interrupt. 2. Measurement process begins after RTC interrupt.
	When accelerometer detects movement, PIC skips measurement.	1. Program PIC to wait if accelerometer detects movement.

	Executes correct sequence of steps of programming in order when powered up.	Observe that the system follows the following steps; 1. Upon power up, wait for RTC interrupt 2. After interrupt occurs, check accelerometer 3. If movement occurs, system waits for next RTC interrupt 4. If no movement occurs, initiates laser rangefinder 5. Shutter opens and Laser Rangefinder camera adjusts itself 6. Makes series of measurements, outputs tread data
Boost Converters	5V Converter provides constant output voltage within the range of 4.85V < 5V < 5.15, with and without current draw.	1. Supply power supply breakout PCB with 3 V. 2. Set EN pin high 3. Observe that output voltage is within range 4. Set EN pin low 5. Observe that output voltage is 0 V 6. Set EN pin High 7. Sweep DC voltage between 2.8 and 4.5 V 8. Observe that output voltage is within range
	3.3V converter provides output voltage range of 3.0V<3.3V<3.6V, with and without current draw.	1. Supply power supply breakout PCB with 3 V. 2. Set EN pin high 3. Observe that output voltage is within range 4. Set EN pin low 5. Observe that output voltage is 0 V 6. Set EN pin High 7. Sweep DC voltage between 2.8 and 3.2 V 8. Observe that output voltage is within range
RTC	RTC generates an interrupt signal after a preset amount of time in ± 10% accuracy	1. Connect RTC to PIC 2. Configure RTC to generate an interrupt every second 3. Program PIC to toggle an LED every interrupt 4. Time the LED toggles 5. Configure RTC to generate an interrupt every minute 6. Time the LED toggles
Android Phone	Receive strings from the Wi-Fi module of the PIC	1. Use the Phone to connect to the ad-hoc network created by the Wi-Fi module on the PIC. 2. Send and receive string data.
Servo (moves Rail)	Servo moves the Laser Rangefinder to the leftmost and rightmost treads	1. Connect the servo motor to the PIC 2. Let the servo move the rail so that the laser pointer points into the leftmost and rightmost treads
Servo (Arduino 180 Servo)	Servo opens and closes the shutter	1. Connect the servo motor to the PIC, send signal to the Digital I/O pin 2. Observe shutter movement

Appendix B Preliminary Testing

We tested the Parallax Laser Rangefinder to find out whether it is capable of measuring 1mm difference. Our requirement for the laser sensor is to have at most 1mm resolution. We used the Arduino example program from Parallax.

Code is as follows[2]:

```
/*
 Laser Range Finder Module: Basic Demonstration

 Author: Joe Grand [www.grandideastudio.com]
 Contact: support@parallax.com

 Program Description:

 This program provides a simple demonstration of the Laser Range Finder
 Module. The distance to the target object is displayed in the Arduino
 Serial Monitor.

 Please refer to the product manual for full details of system functionality
and capabilities.

 Revisions:

 1.0 (December 11, 2013): Initial release
 1.1 (April 29, 2014): Changed rxPin/txPin to use pins 10/11, respectively, for widest support across the
Arduino family (http://arduino.cc/en/Reference/SoftwareSerial)

 */

// include the SoftwareSerial library so we can use it to talk to the LRF
#include <SoftwareSerial.h>

#define rxPin    10  // Serial input (connects to the LRF's SOUT pin)
#define txPin    11  // Serial output (connects to the LRF's SIN pin)
#define ledPin   13  // Most Arduino boards have an on-board LED on this pin

#define BUFSIZE  16  // Size of buffer (in bytes) for incoming data from the LRF (this should be adjusted to be
larger than the expected response)

// set up a new serial port
SoftwareSeriallrfSerial =  SoftwareSerial(rxPin, txPin);

void setup()  // Set up code called once on start-up
{
  // define pin modes
pinMode(ledPin, OUTPUT);
pinMode(rxPin, INPUT);
pinMode(txPin, OUTPUT);
```

```
digitalWrite(ledPin, LOW);  // turn LED off

  // setup Arduino Serial Monitor
Serial.begin(9600);
  while (!Serial);   // Wait until ready
Serial.println("\n\nParallax Laser Range Finder");

  // set the baud rate for the SoftwareSerial port
lrfSerial.begin(9600);

  /*
    When the LRF powers on, it launches an auto-baud routine to determine the
host's baud rate. it will stay in this state until a "U" ($55) character
is sent by the host.
   */

delay(2000);                        // Delay to let LRF module start up
lrfSerial.print('U');               // Send character
while (lrfSerial.read() != ':');    // When the LRF has initialized and is ready, it will send a single ':' character, so
wait here until we receive it
  delay(10);                        // Short delay
lrfSerial.flush();                  // Flush the receive buffer
Serial.flush();                     // Wait for all bytes to be transmitted to the Serial Monitor
}

void loop()  // Main code, to run repeatedly
{
  /*
   When a single range (R) command is sent, the LRF returns the distance to the target
object in ASCII in millimeters. For example:

   D = 0123 mm
   */

lrfSerial.print('R');         // Send command
digitalWrite(ledPin, HIGH);   // Turn LED on while LRF is taking a measurement

  // Get response back from LRF
  // See Arduino readBytesUntil() as an alternative solution to read data from the LRF
  char lrfData[BUFSIZE];  // Buffer for incoming data
  char offset = 0;        // Offset into buffer
lrfData[0] = 0;         // Clear the buffer
  while(1)
  {
    if (lrfSerial.available() > 0) // If there are any bytes available to read, then the LRF must have responded
    {
lrfData[offset] = lrfSerial.read();  // Get the byte and store it in our buffer
      if (lrfData[offset] == ':')          // If a ":" character is received, all data has been sent and the LRF is ready to
accept the next command
      {
lrfData[offset] = 0; // Null terminate the string of bytes we just received
        break;              // Break out of the loop
      }
```

```
   offset++;  // Increment offset into array
   if (offset >= BUFSIZE) offset = 0; // If the incoming data string is longer than our buffer, wrap around to
avoid going out-of-bounds
  }
 }
Serial.println(lrfData);    // The lrfData string should now contain the data returned by the LRF, so display it on
the Serial Monitor
Serial.flush();            // Wait for all bytes to be transmitted to the Serial Monitor


digitalWrite(ledPin, LOW);  // Turn LED off
}
```

Figure 17: Preliminary Testing Setup

For our test, we tried to get an initial sense of whether the sensor is accurate, precise, and sensitive enough to detect the small displacement.

For accuracy, the sensor gives us value that's little off from the measurement that we take from the ruler. There is about 5mm to 1cm difference depending on the distance of the sensor to the wall. However, there are commands that can calibrate the sensor. Command 'E' adjusts camera for current lighting conditions and command 'X' calibrates camera system for range finding. We were unable to use these commands for preliminary testing but we believe that this will compensate for the offset we are getting. Also, since we are using two laser sensors, as long as we can optimize two laser sensors, we can measure the tread depths.

For precision, the sensor gives us the stable detection as long as it stays still. When we changed the distance from the wall to the sensor, it took about 5 to 10 seconds to stabilize. So, when we provide the measurement of the distance from the sensor to the tire, we will average the measurement that has taken for a few seconds after the stabilization.

For sensitivity, we tested if the sensor can measure the small displacement of change. The sensor detected the difference of the desired resolution, which is 1mm. It also required the time for stabilization of getting stable measurement.

Appendix C Schematic of the Printed Circuit Board

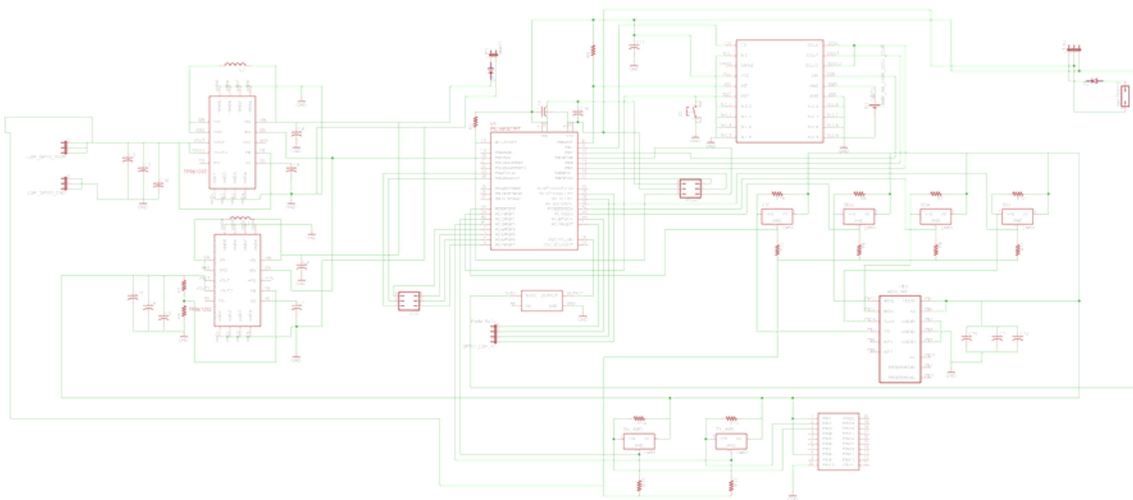

Figure 18: Schematic of the Printed Circuit Board

Appendix D Accelerometer Test Data

Table 7: Accelerometer Readings With/Without Vibration from Mobile Phone

Sample	Vibrations (G)	No Vibrations (G)
1	0.96	0.99
2	0.96	0.99
3	0.96	0.99
4	0.99	0.99
5	0.99	0.99
6	0.97	0.99
7	0.99	0.99
8	0.99	0.99
9	0.99	0.99
10	0.97	0.99
11	0.97	0.99
12	0.99	0.99
13	0.99	0.99
14	0.99	0.99
15	0.97	0.99
16	0.99	0.99
17	0.99	0.99
18	0.98	0.99
19	0.96	0.99
20	0.96	0.99
STD	0.01	0

Appendix E References

[1] IEEE. (2014). *7.8 IEEE Code of Ethics.*
http://www.ieee.org/about/corporate/governance/p7-8.html

[2] Arduino Example Program for the Parallax Laser Rangefinder
http://www.parallax.com/downloads/laser-range-finder-arduino-example-program

[3] Parallax Laser Rangefinder Datasheet
http://www.parallax.com/downloads/laser-range-finder-product-document

[4] Arduino Uno Data Page
http://arduino.cc/en/Main/arduinoBoardUno

[5] RN-XV Wi-Fi Module Datasheet
http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Wireless/WiFi/WiFly-RN-XV-DS.pdf

[6] Laser Safety Datasheet
http://www.rli.com/resources/articles/classification.aspx

[7] PIC Datasheet
http://ww1.microchip.com/downloads/en/DeviceDoc/30292D.pdf

[8] RN-XV Wi-Fly Module User Manual
http://ww1.microchip.com/downloads/en/DeviceDoc/rn-wiflycr-ug-v1.2r.pdf

[9] Boost Converter Datasheet
http://www.ti.com/product/TPS61230/datasheet