Laser Tag Droid

Jake Hamill, Martin Litwiller, Christian Topete
Group 81
ECE 445 Final Report

Teaching Assistant: Eric Clark
May 2017

Abstract

For our project, we wanted to create a toy which would provide a fun laser tag experience for anyone watching or playing. This product is similar to those of battle-bots except there is no damage involved. We created a laser tag droid which could be controlled either manually through an iOS app or autonomously. When in autonomous mode, it could identify and seek targets to engage in gameplay. This allows for player vs computer, player vs player, and computer vs. computer gameplay. In this report, we provide the hardware and software design in order to create a laser tag droid.

Table of Contents

1. Introduction

1.1 Background

Over the past 5 to 10 years, the market for remote control toys of all types has expanded dramatically. With the technology steadily becoming less expensive and more accessible, these toys have gotten a huge amount of exposure and have gained in popularity with consumers of almost every age. However, most products in this category are simple models of cars, planes, or helicopters that are only meant to be driven or flown around and nothing more. A relatively small number of RC toys take advantage of their potential to be used as (or in) games. Typically, competitions involving RC toys consists only of racing. However, many people are naturally competitive and would prefer something more dynamic than simply driving or flying. One example of a very popular type of RC competition is battle-bots. The drawback to these competitions is the inevitable and significant damage that the vehicles suffer. Therefore, we believe that an alternative non-destructive type of RC competition such as laser tag would be very popular with consumers. Laser tag bots have been implemented in several ways already. Some have been bipedal walking robots, and some have been built with a simple wheeled car design. Others only have been implemented using flying quadcopters [1]. Even with these, there are drawbacks. Flying vehicles are somewhat hard to control and slightly dangerous, and bipedal or wheeled designs are often rather slow and unexciting. Our design will attempt to remedy these negative factors and add unique functionality that would make for a successful product.

1.2 Objective

Our project was designed to build a remote control laser tag droid that allows users to engage in exciting physical and computer gameplay. When switched into autonomous mode, the droid is able to search for, identify, and engage opponent droids automatically. This will create a fun laser tag experience whether you are playing solo or with a group. The droid is a wheeled vehicle that has the mobility to move around quickly and maneuver easily. The droids are controllable wirelessly via an iOS app which allows users to play the game.

1.3 High level Requirements

- The droid has an autonomous target seeking mode, in which it can move around on its own looking for targets.
- The droid has a player-controllable mode, in which its movement and firing are controllable by a user with a mobile phone app.
- The droid is powered by a standard 7.4V LiPo hobby battery.

2. Design

2.1 Block Diagram

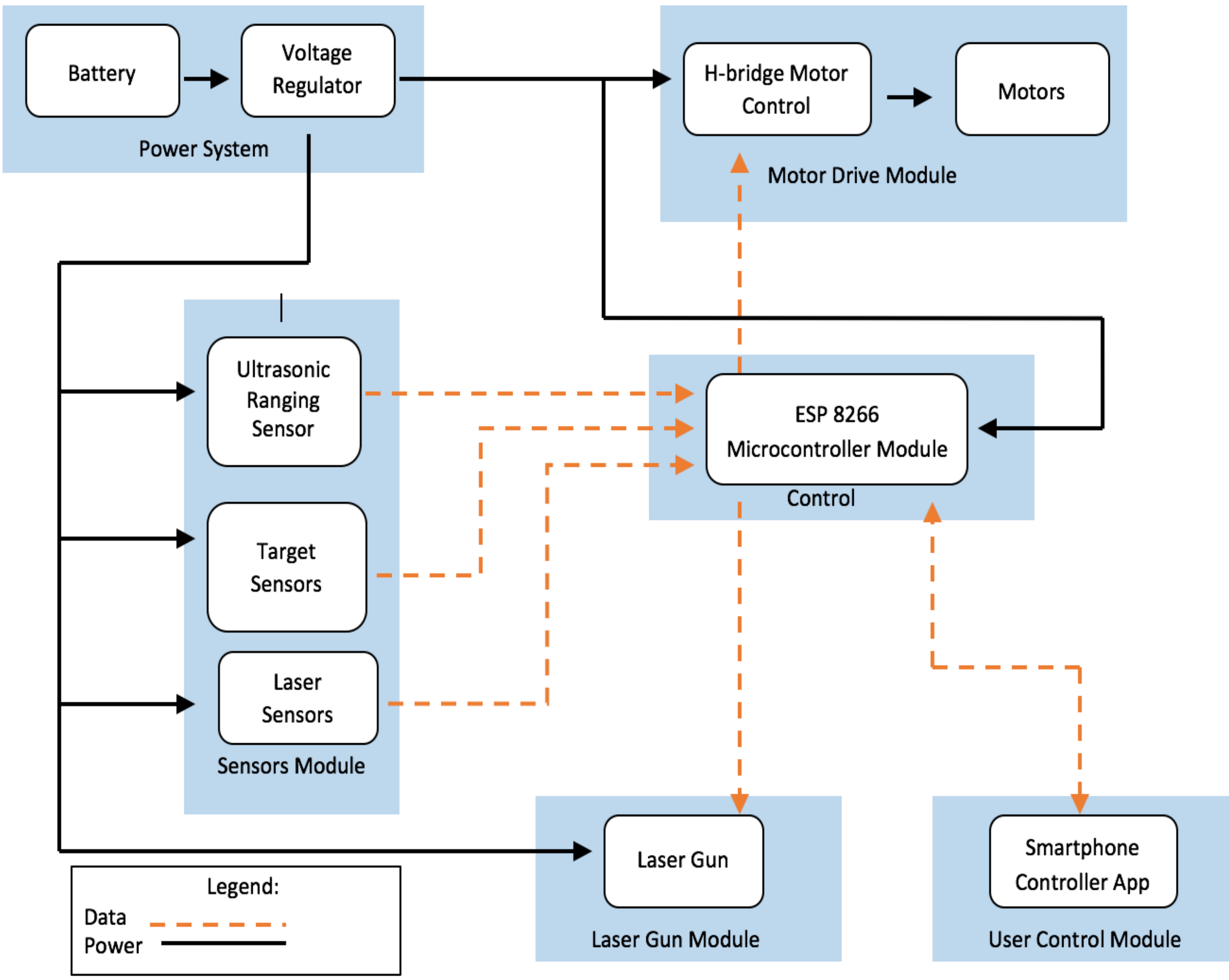

Figure 1: Top Level System Block Diagram

Power System: Battery, Voltage regulator
Motor Drive Module: H-bridge motor control, DC brushed motors
Control Module: ESP 8266 Microcontroller Module
Sensor Unit: Ultrasonic Range Sensors, Laser Sensor, Target Sensor
Laser Gun Module: 5mW Red Laser Pointer
User Control Module: Smartphone Controller App

2.2 Physical Design

The vehicle chassis we implemented is a Tamiya 70098 universal plate set with the corresponding track and wheel set. This works well with the Tamiya 70097 dual motor gearbox set that we used with our motors. We added standoffs with aluminum plating to place our Printed Circuit Board and sensor modules.

2.3 Power Supply

The power supply is essential in keeping all systems working efficiently and safely. The power supply consists of one KA378R05 low dropout voltage regulator. The voltage regulator is used to output 5 volts to each DC motor and to the sensor module. These low voltage dropout regulators work well with the 7.4 volt LiPo battery due to the low voltage drop from 7.4 to 5 volts. Additionally, the voltage regulator max current rating is 3 amps but we will limit it to 2.5 amps so we can prevent the regulator from overheating. We added a heat sink to the regulator so it can dissipate the heat more rapidly. The heat sink allows the regulator to dissipate a maximum of 15 watts.

2.4 Battery

The battery is a standard 25C 7.4 Volt 2500 mAh battery [2]. This is a 2 cell battery which outputs a nominal 3.7 volts per cell. Its peak voltage is 8.4 volts and minimum is 6.4 volts. The max current is 62.5 amps which bodes well with our circuitry. This battery powers the microcontroller, two DC brushed motors, and the sensor modules. Also, we created a battery protection circuit to ensure that the battery will not get damaged from over-discharging. The protection circuit involves 5 resistors, 1 TL431 adjustable shunt regulator, and a SFT 1350 P-Channel Power MOSFET. R8 is used to pull up the output when the TL431 is off and to turn off the SFT1350 p-channel MOSFET causing the circuit to cutoff. R9 and R10 are used to adjust the voltage at which we want the circuit to cutoff. R11 is used to eliminate hysteresis and R12 is used as a load. We note that the output of the battery cutoff circuit leads to the input of the disable pin at the voltage regulator. This causes the regulator to shut off if the voltage falls below 6.4 volts. We solve for the values R9 and R10 in the calculations section of this report.

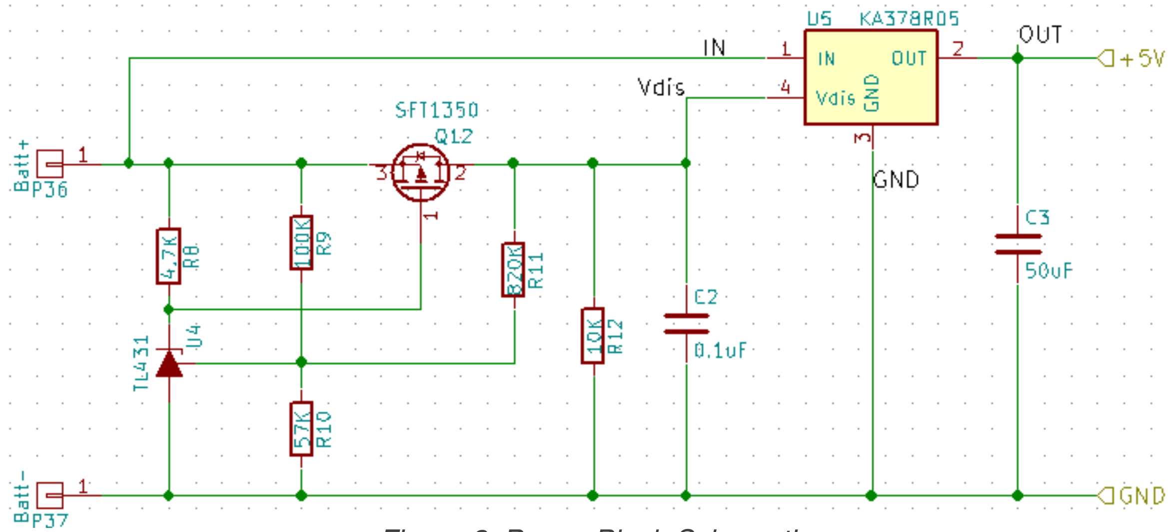

Figure 2: Power Block Schematic

2.5 Motors

The two motors we used are DC brushed motors. Each motor has a stall current of 800 mA. They will be powered by 5 volts. Additionally, we implemented PWM and created an H-bridge circuit in configuration with these motors. As stated in the physical design, we used the Tamiya 70097 motor gearbox kit to fit these specifications.

2.6 H-Bridge Motor Control/PWM

Originally, we planned on using the SN754410 quadruple half-h driver chip to implement the h-bridge control but we decided to design our own h-bridge instead. The circuit schematic is shown in figure 3. The circuit consists of 4 transistors (2 PNP and 2 NPN), 4 diodes, and 4 resistors. The resistors are used to prevent too much current from passing through the base of the transistors. The diodes allow for a safe dissipation path of energy for the motor. The transistors are used to control the direction of the motors. Table 1 shows the different ways of the controlling the H-bridge by applying power to certain terminals. R16, R15, R14, and R13 correspond to the resistive terminals on the left side of the h-bridge circuit. We apply this same logic to the right side of the H-bridge which consists of R17, R18, R19, and R20.

Table 1: Controlling the H-bridge Circuit				
Functionality	R16	R15	R14	R13
Brake/Slow down	High (5 volts)	High (5 volts)	High (5 volts)	High (5 volts)
Forward	Ground	Ground	High (5 volts)	High (5 volts)
Reverse	High (5 volts)	High (5 volts)	Ground	Ground
Coast	Ground	High (5 volts)	Ground	High (5 volts)

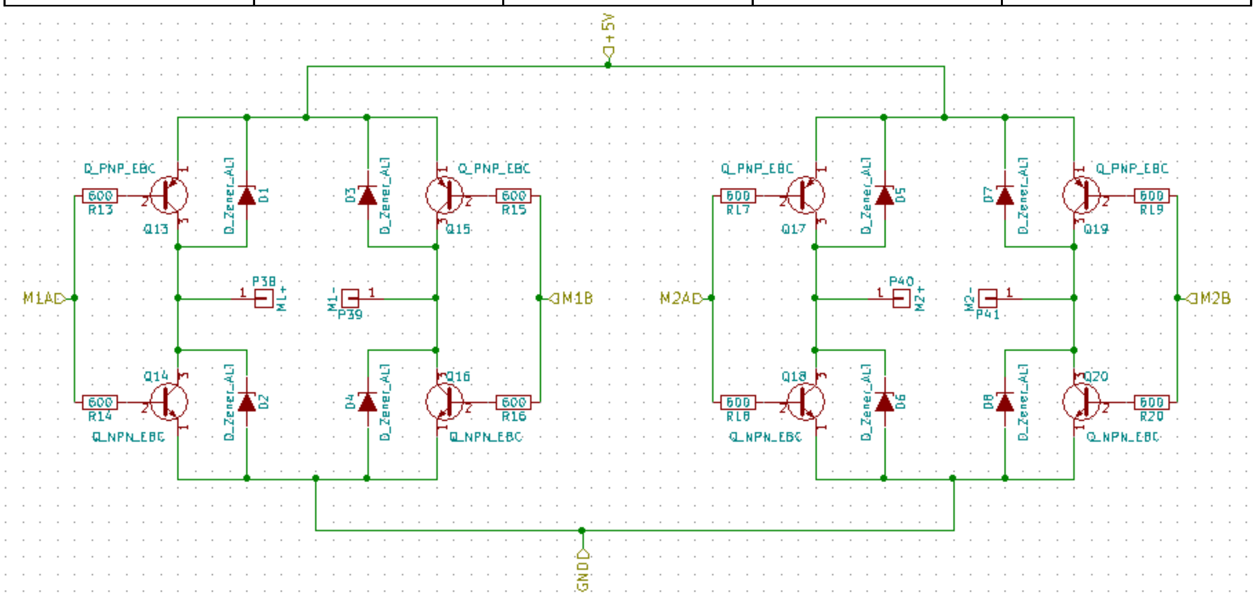

Figure 3: H-bridge Circuit for both motors

2.7 ESP8266 microcontroller/wifi module

This device is a combined microcontroller and wifi communications module. The microcontroller portion polls all sensors and controls all parts of the vehicle such as the motor control circuitry and the laser gun. It uses pins connected to the circuitry to control the motors, laser, etc. The ESP8266 acts as a server, and when a client (mobile app) connects to it, then it takes control commands from the client (see section 2.11). The ESP8266 is also programmed for controlling the droid in autonomous mode (High Level Req. 1). If the user sets the droid to manual mode via the app, then the ESP8266 listens to the steering commands sent by the app and sets the pins accordingly. Figure 4 shows a high level flow chart of the navigation logic.

The autonomous logic was developed through experimentation and testing over the course of our project. The goal of the logic is to enable the droid to roam around some area, aiming towards (and firing at) targets if it sees any, and avoiding obstacles (e.g. not running into walls, etc.). It uses the Ultrasonic Range Sensors (2.8) and the Target Sensor Module (2.10) to sense and avoid obstacles as well as seek targets. Figure 5 shows a high level diagram of the autonomous logic. I will elaborate on the lowermost block – turning away from an obstacle: The droid has three sonar sensors, one in front, one on the right, and one on the left. If there is an obstacle only in front of the droid, it will turn left (in place, not moving forward) until it no longer detects the obstacle and then proceed. If there is an obstacle in the front and on the left, it will turn right, and vice versa. This is a very simple algorithm and if we were going to improve upon the project we would add more sonar sensors and use a more advanced navigation algorithm.

Lastly, figure 6 shows the circuit schematic for the connections of the ESP8266. We implemented MIC4424CN non-inverting drivers on the ESP8266 because the microcontroller couldn't source enough current for the H-bridge module.

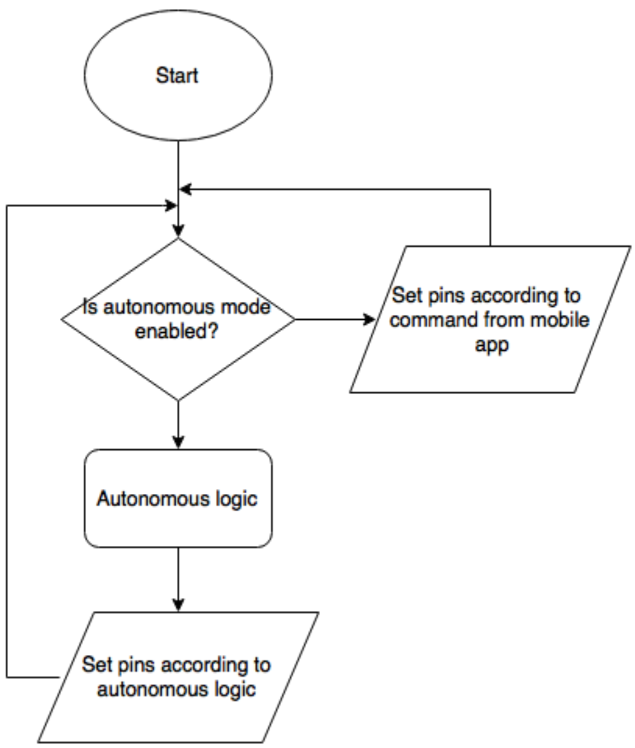

Figure 4: ESP8266 droid steering high level flow chart

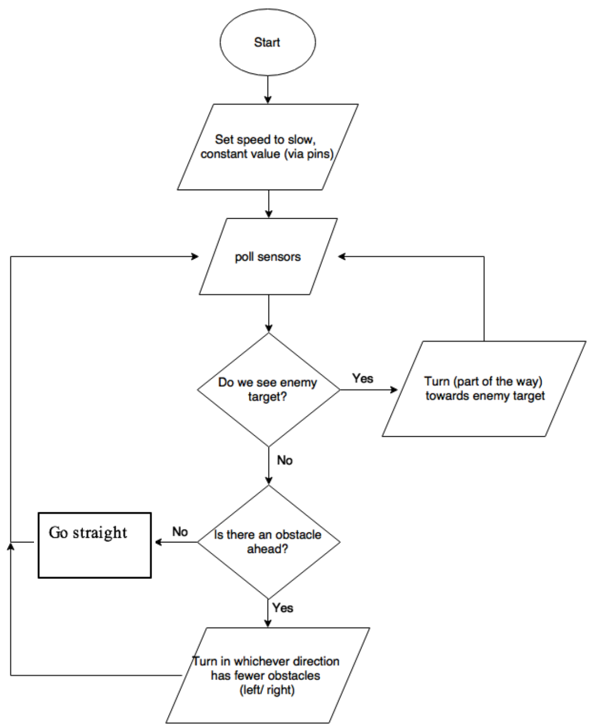

Figure 5: High level diagram of autonomous logic

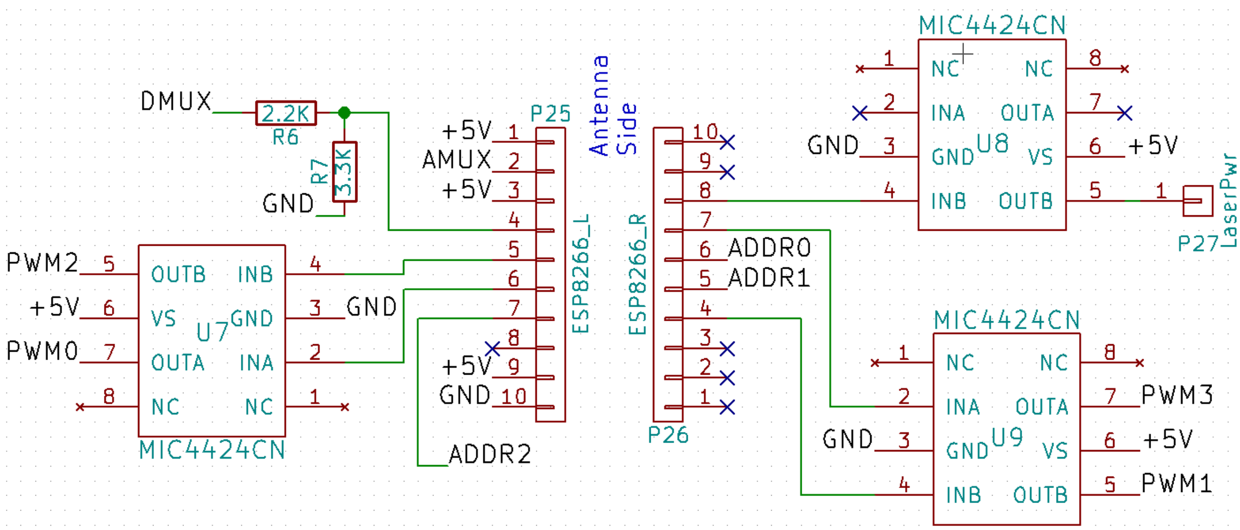

Figure 6: Circuit Schematic for ESP8266 signal connections

2.8 Ultrasonic Range Sensor Module

We used 3 HC SR-04 ultrasonic range sensors on the outside of our droid to help with the autonomy. The ultrasonic range sensor sends out a signal and then waits for the signal to determine how long it took. The sensor converts this time into a distance which helps us determine the proximity of any nearby objects. The sensor can determine obstacles within a range of 2cm to 400cm. We made our threshold at 40 cm, therefore, when an object was within 40 cm of the droid then it would turn away from it. This module is powered by 5 volts and controlled by an Arduino microcontroller. The circuit schematic for the ultrasonic sensor module is shown in figure 7.

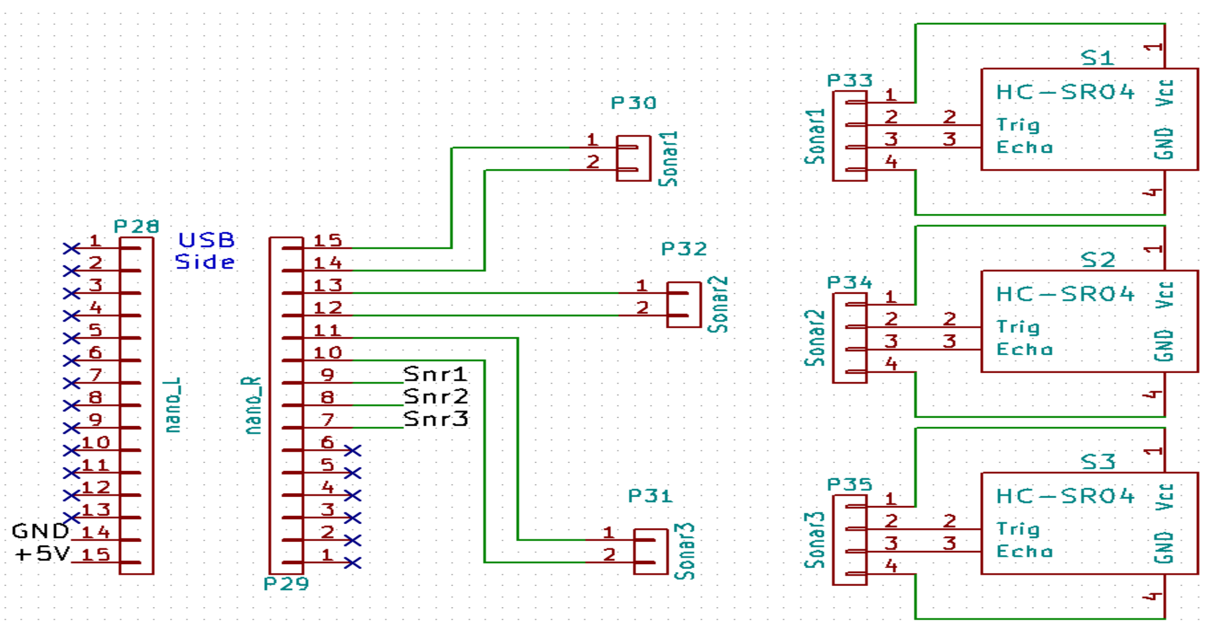

Figure 7: Circuit Schematic for ultrasonic sensor module

2.9 Laser Sensor Module

To detect incoming laser fire from opponent droids, we used a set of simple phototransistor circuits as shown in figure 4. A phototransistor generates a current through its emitter resistor if hit with a red laser beam, which will generate a 3.5-5V output signal. The outputs from the 4 phototransistor circuit copies will be passed through a 4-input OR gate whose output will be connected to the microcontroller. The microcontroller will detect the rise in output voltage as a digital signal and will deduce that the droid has been hit. Figure 8 shows the circuit schematic for the laser sensor module.

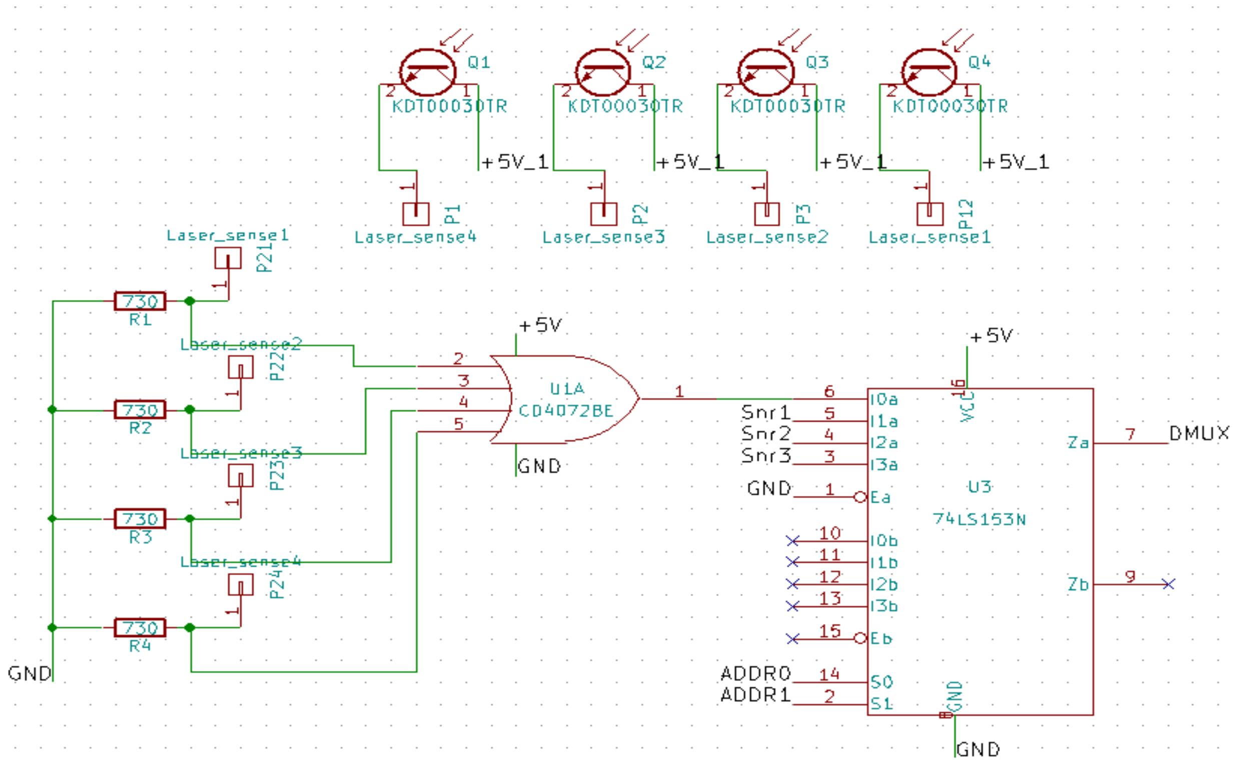

Figure 8: Circuit Schematic for laser sensor module

2.10 Target Sensor Module

The target sensor module is similar to the laser sensor module. It will use an array of 7 IR phototransistors to generate analog voltage readings, which will be passed to the microcontroller through an analog multiplexer chip. The microcontroller will translate the analog voltage level into relative levels of IR light in each direction. If a significant level of IR light is detected, the droid will assume that it is an opponent droid and will turn toward and follow it. Figure 9 shows the circuit schematic for the target sensor module.

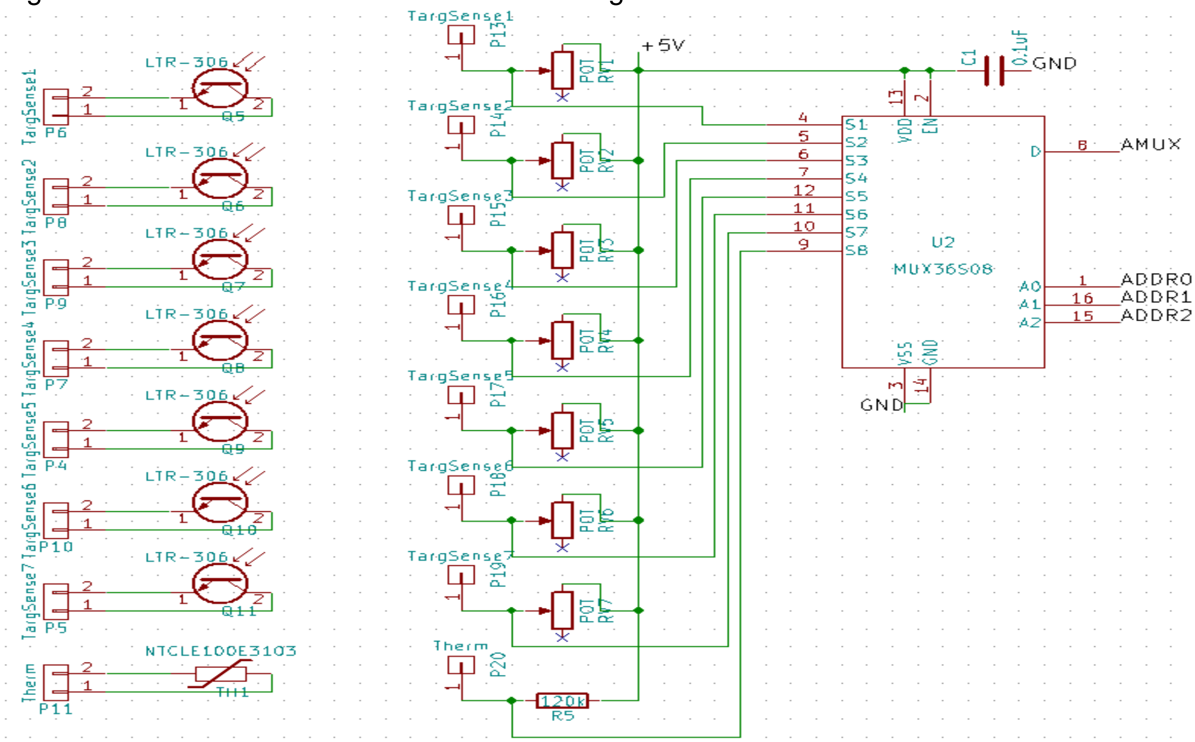

Figure 9: Circuit Schematic for target sensor module

2.11 Mobile Application Module

The mobile phone application enables a human user to control the droid. Through the app, the user is able to accelerate the droid forwards and backwards, as well as turn to the left and to the right (High Level Req. 2). The user can also fire the laser from the app. Furthermore, the app allows the user to toggle between autonomous driving mode and manual (driven by the user through the app) mode (High Level Req. 1, 2). More specifically, the mobile app communicates with the droid over Wifi. When the user taps anywhere on the screen where there is not a button, the app sends a “shoot” signal to the droid which causes the laser to be fired. The droid has two modes of operation: an autonomous mode, and a manual mode. There is a button in the app (labeled “Autonomous Mode” or “Manual Mode”, depending on the current state – we will refer to it as “toggleMode”) that allows the user to toggle between the two modes when pressed. The app also has a “Connect” / “Disconnect” button which connects or disconnects, respectively, the app to the droid. The connect, shoot, and toggleMode signals are sent to the droid using TCP.

In manual mode, the user can steer the droid by tilting the phone in the desired direction of motion. More specifically, there are 3 different measures we use to define its position: yaw, pitch, and roll (see figure 10).

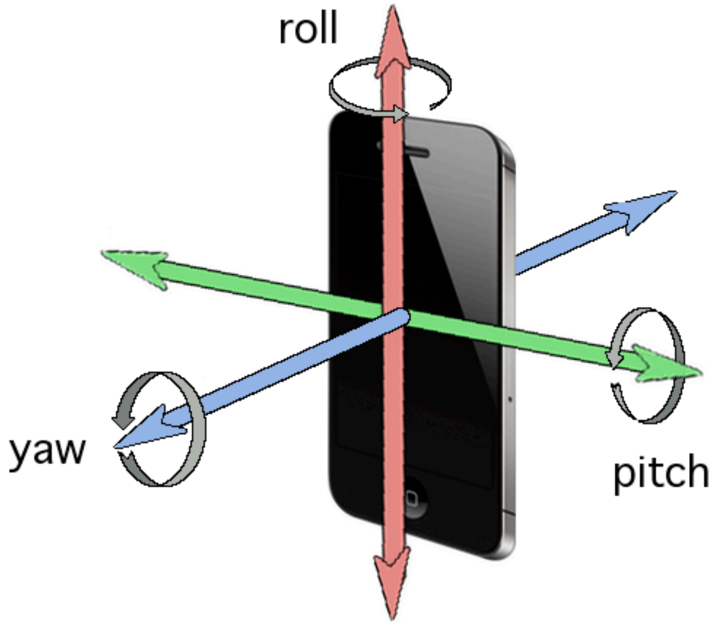

Figure 10: visual description of phone position in 3d space [3]

Define yaw, pitch and roll to all be zero when the user is holding the phone horizontally with the screen perpendicular to the ground and the home button on the user's right (Screen facing user). We will say that yaw, pitch, and roll are positive if the user rotates the phone slightly (less than 90 degrees) in the direction of the arrows in figure 10 and negative if the phone is rotated slightly in the opposite direction when starting from zero.

We care in particular about the roll and yaw. The roll defines the forward/ backward velocity, and the yaw defines the degree to which the droid is turned to the right or left. There is a tolerance region (ε) around zero for each value. If the angle is within ±ε of 0, it is considered zero, thus making it easier for the user to keep the droid stationary. Furthermore, rotations of more than 90˚ (positive or negative) are considered "undefined" — that is, there will be no speed value associated with these regions. All speeds (in the case of roll) or steering angles (yaw) are determined by discretizing the regions $(90 - \varepsilon)$ into evenly spaced intervals, where each interval represents one speed. For example, if there are r possible speeds (or angles of turning), then the size of each interval, s, is given by:

$$s = \frac{90-\varepsilon}{r} \tag{1}$$

and then the interval of the nth smallest value (e.g. speed) begins at $s(n-1)$ degrees and ends at sn degrees. So each of these n intervals will have a specific speed or direction value associated with it. Increasing the number of speed/ direction intervals gives the user a smoother steering feel. In the end, we decided on 800 possible speeds, an epsilon value of 10 for the roll, and an epsilon value of 5 for the yaw, which appears to be more sensitive on the iPhone. The phone position was measured every .01s, and sent to the ESP8266 using UDP. A flowchart describing the "steering" loop of the app is shown in figure 11 below. This loop can be interrupted by pushing the toggle mode button (thereby switching the app into autonomous mode).

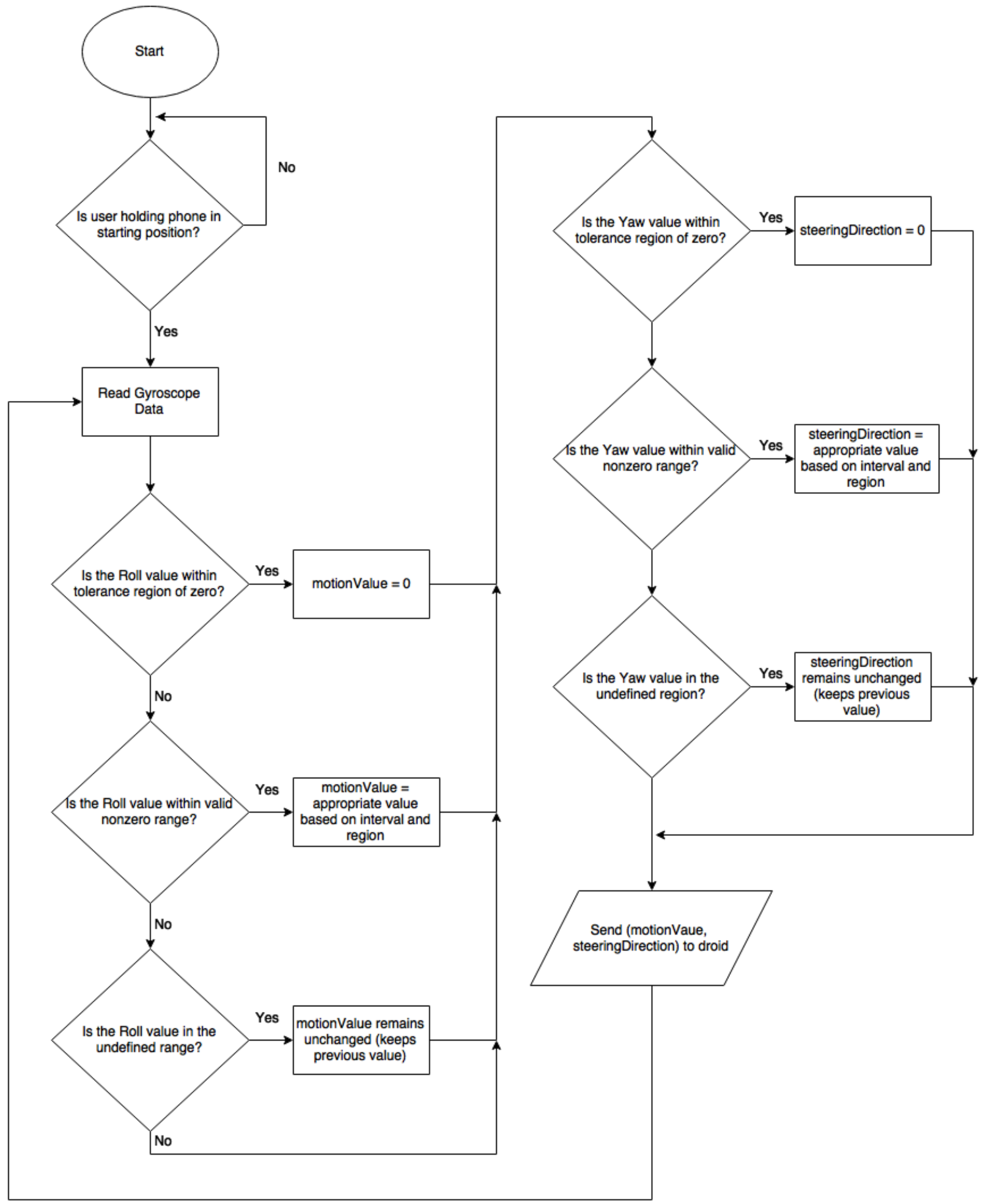

Figure 11: Mobile App steering logic flow chart

3. Calculations and Experimental Data

3.1 Cutoff Voltage Calculation and Experimental Data

In order to ensure our circuit cuts off the LiPo battery at 6.4 volts, we need to calculate resistance values for both R9 and R10. Our TL431 has a reference voltage of 2.5 volts and we calculate the cutoff voltage based on the equation below.

$$V_{cutoff}=V_{ref}\left(1+\frac{R_9}{R_{10}}\right) \tag{2}$$

We then determine the relationship between R2 and R3 by having a cutoff voltage of 6.4 volts and a reference voltage of 2.5 volts.

$$6.4 = 2.5\left(1+\frac{R_9}{R_{10}}\right) \tag{3}$$

$$1.56 * R_{10} = R_9 \tag{4}$$

We pick large arbitrary values for R9 and R10. We pick values in the kilo-ohms to ensure each resistor isn't dissipating a lot of power. The values we selected are R2=100 Kilo-Ohms and R3=64.1 Kilo-Ohms. This gave us our theoretical values for the LiPo cutoff circuit. When we tested the circuit, we had to change our R10 value to 57 Kilo-Ohms in order to ensure it cutoff at 6.4 volts. The experimental data of the cutoff is shown below.

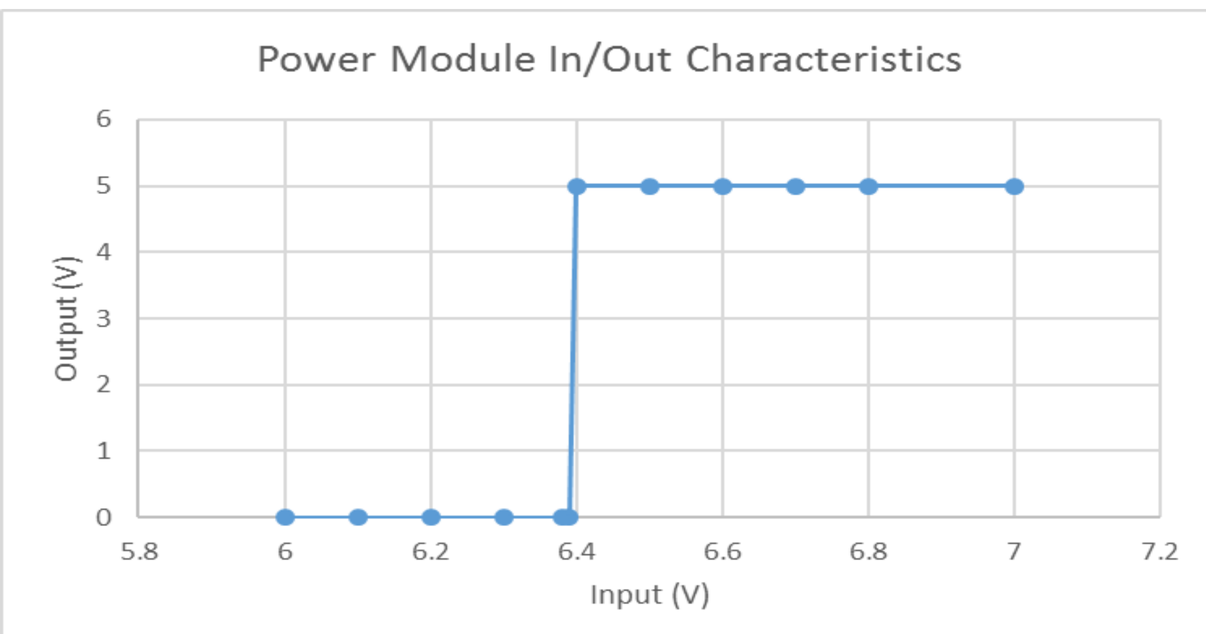

Figure 12: Experimental data for LiPo cutoff circuit

3.2 Minimum and Average Lifetime of Battery

Next, we calculate the average lifetime of our battery when our values are at average rating and max rating. Each component and its corresponding current rating are shown in the table below.

Table 2: Module current consumptions

Module	Average (mA)	Max (mA)
ESP 8266	250	500
DC motors (per)	400	800
Ultrasonic Sensor (per)	15	15
KA378R05 Voltage Regulator	10	10
Phototransistors	10	21.2
Laser	30	30

We first calculate the average lifetime of the battery. Then, we calculate the total average current and then divide it by the battery's capacity to determine the average run time.

$$I_{Total} = 2 * I_{per\ motor} + I_{ESP8266} + I_{laser} + 3 * I_{per\ range\ sensor} + I_{phototransistors} + I_{Voltage\ Regulator} \quad (5)$$

$$I_{Total,Avg} = 2(400) + 250 + 30 + 3(15) + 10 + 10 = 1145\ mA \quad (6)$$

$$Capacity = 2500\ mAh \quad (7)$$

$$Average\ Run\ time = \frac{Capacity}{I_{Total,Avg}} = \frac{(2500\ mAh)}{1145\ mA} = 2.18\ hours \approx 131\ minutes \quad (8)$$

We see that the average lifetime of our battery is approximately 131 minutes. Next, we determined the minimum lifetime of our battery by using our max ratings.

$$I_{Total} = 2 * I_{per\ motor} + I_{ESP8266} + I_{laser} + 3 * I_{per\ range\ sensor} + I_{photodiodes} + I_{Voltage\ Regulator} \quad (9)$$

$$I_{Total,Max} = 2(800) + 500 + 30 + 3(15) + 21.2 + 10 = 2206\ mA \quad (10)$$

$$Capacity = 2500\ mAh \quad (11)$$

$$Max\ Run\ time = \frac{Capacity}{I_{Total,Avg}} = \frac{(2500\ mAh)}{2206\ mA} = 1.13\ hours \approx 68\ minutes \tag{12}$$

We see that the time we can run the system at maximum ratings is approximately 62 minutes. This bodes well because it verifies our requirement theoretically.

3.3 Power dissipation of voltage regulator

Next, we would like to calculate the power dissipation of the KA378R05 voltage regulator to ensure that it can withstand the system's load.

$$V_{IN,Max} = 8.4 \qquad V_{out} = 5$$

$$P_{Dissipation,Max} = \left(V_{IN,Max} - V_{Out}\right) * I_{Total,Max} = (8.4 - 5) * 2435 \tag{13}$$

$$P_{Dissipation,Max} = 8.28\ Watts \tag{14}$$

We note that our regulator can handle the system load because it has a max rating of 15 watts with the heat sink.

4. Design Verifications

4.1 Power Supply

We verified that the power supply worked for the laser tag droid. We first verified that the output of the voltage regulator was between 4.9 to 5.2 volts when we used a 2-ohm power resistor as a load. Next, we verified that the Lipo cutoff circuit cutoff at 6.4 volts by taking data points. The data is shown in the calculations section for the power supply. Furthermore, we verified that once we fell below 6.4 volts, then the output of the voltage regulator fell to close to 0 volts. Lastly, we verified the battery life of the system by running it in autonomous mode for 1 hour.

4.2 Motor Control

We verified that the whole motor control worked by sending different voltage signals to the corresponding pins on the H-bridge. For example, we sent a high (3.3 volt) signal to pin PWM1 and low (0 volts) to pin PWM0 to verify that the motor was spinning in the forward direction. We applied the reverse logic to ensure that the motor was spinning in the reverse direction. In order to ensure variability, we sent out different duty cycle PWM signals in sequential manner to PWM1. We verified that the motor gets slower with the sequential increase in duty cycle. Lastly, we had to verify that the H-bridge could supply +/-800 mA to each motor. We attached a resistive load across it and verified that the voltage was steady between 4.9 and 5.2. Then we repeated this procedure for the other H-bridge circuit as well.

4.3 Sensors Module

Our laser sensors module worked great. When we shined the laser to the phototransistors on the laser module then the voltage would increase above 2.5 volts. The voltage would output less then .8 volts when the laser wasn't shined on it. Our ultrasonic sensors module also worked well. When we placed an object within 40 cm, then the vehicle would turn accordingly to avoid the object. Our target sensors would not output a range between 0-1 volts reliably when exposed to an IR light source. This was due to the fact that each room emitted a different amount of IR light which would cause our voltage reading to spike up to 3 volts.

4.4 Mobile App Module

Everything in the app worked as planned. Before testing on the droid, we verified that the ESP received the correct signals from the app by connecting the ESP to a laptop as specified in the R&V table. The Boolean experimental result values are therefore "true" (for success) in all cases.

4.5 Microcontroller Module

The only listed requirement for the Microcontroller is successful target detection, which did not work due to our issue with the infrared sensors. In retrospect, we should have had more requirements – such as controlling the wheels, firing the laser, etc. – which all worked.

5. Financial Analysis

5.1 Labor

Based on [4], a reasonable estimate for the salary (converted to hourly) of an ECE graduate from UIUC is $36 / hour. We estimate that it will take the three of us and average of 15 hours per week to complete this project. Following the equation given in the instructions, the labor cost is then:

$$\frac{\$36}{hr}(2.5)\left(\frac{15hr}{week}\right)(12\ weeks) = \frac{\$16{,}200}{person} \tag{15}$$

$$\left(\frac{\$16{,}200}{person}\right)(3\ people) = \$48{,}600 = total\ labor\ cost \tag{16}$$

5.2 Parts

Table 3: Cost of Parts

Part name	Part Number	Unit Cost ($)	Quantity	Total ($)
ESP 8266 Microcontroller	Adafruit Industries LLC #2471	9.95	1	9.95
TL431	Mouser Electronics #595-TL431ILP package TO-92-3	0.44	1	0.44
SN754410	Banana Robotics SN754410	1.75	2	3.50
KA378R05TU	Manufacturer: On semicoductor Digikey part# KA378R05TU-ND	1.25	1	1.25
7.4 Volt LiPo battery 25C 2500 mAh	Amazon Part # B01D9OTRVQ	11.59	1	11.59
Tamiya 70097 motor gear box kit	Pololu item #61	8.95	1	8.95
DC brushed motors	Pololu item #1117	1.79	2	3.58

Part name	Part Number	Unit Cost ($)	Quantity	Total ($)
SFT1350 P-channel MOSFET	Mouser electronics part #863-SFT1350-H	0.92	1	0.92
Passive Components	Resistors and Capcitors	$7.00	1	7.00
TO-220 Heatsink for Voltage Regulator	Jameco Electronics part #326596	0.29	1	0.29
Farhop 4.5 5 mW laser diode module	Amazon part#650-D-9-21-R-5	9.95	1	9.95
HC-SR04 Ultrasonic Range Sensor	Banana Robotics part# BR010020	2.99	3	8.97
Phototransistor Pk wavelength 940nm	LTR-306	0.21	7	1.47
Phototransistor Pk wavelength 630nm	KDT00030TR	0.46	4	1.84
Analog Multiplexer	MUX36S08IPWR	3.57	1	3.57
4 input OR gate	CD4072BM96	0.39	1	0.39
Tamiya 70098 Universal Plate set with 70100 track and wheel set	Pololu item #79 and #106	14.20	1	14.20
Total Cost	-	-	-	87.86

5.3 Grand Total

$$Grand\ Total = Parts + Labor \tag{17}$$

$$Grand\ Total = 87.86 + 48{,}600 \tag{18}$$

$$Grand\ Total = \$48{,}687.86 \tag{19}$$

6. Conclusions

6.1 Accomplishments

We designed a PCB with all the modules we wanted originally. We had to implement some minor changes on a solderless breadboard for the addition of the MIC4424CN non-inverting drivers but most of our design was utilized through the PCB we designed. We were able to make an iPhone app to control the different modes of the droid. Lastly, our system worked in unison with the exception of the unreliability of the target sensors module.

6.2 Uncertainties

One of the main uncertainties with this project was the target sensor module. It couldn't detect the IR light source reliably and we had to point the IR source directly to the phototransistors in order for there to be a reliable reading. Next time, we would implement stronger IR diodes and phototransistors that could detect the IR light source from all angles. Another possibility would be to pulse the IR LEDs and use a band pass filter for our target sensors so we could filter out the background IR. Furthermore, our LiPo cutoff circuit would behave slightly differently under load conditions. We would do more research on this to ensure the cutoff was accurate under load conditions as well.

6.3 Ethical Considerations

As a group of aspiring engineers, we have to abide by the IEEE ethics and ACM guidelines. First and foremost, every project involved has to follow #1 on IEEE code of ethics to ensure the safety of those who will be utilizing this droid which involves power sources and electronic components that could catch on fire or explode [5]. We will also be working with #7 on the IEEE code of ethics to ensure we build the most viable, efficient, and safe project. We will be taking constructive criticism from professors, teaching assistants, and ourselves as group members [5]. We have to keep ourselves honest and ensure that we acknowledge and correct our mistakes. For example, if there is a better way to electronically design our PCB to avoid overheating of components then we have to keep ourselves honest and make this revision. This project entails the aspects of #8 and #10 in IEEE code as well [5]. Whether it is the treatment of group members or other groups, we have to ensure that we don't engage in discrimination of any kind. This will also bode well with #10 in the IEEE code of ethics because it will create a great aura for group members to work in which in turn will enhance the development of our project [5]. We, aspiring engineers, endure #5 on the IEEE code of ethics on a daily basis regardless of whether we are working on a project or not. We have to understand the technology we are using and know the applications/consequences that the project entails. Number 3 on the IEEE code touches in with #5 in the aspect of finding the best application with the available data will enhance the development of our project [5]. We, aspiring engineers, endure #5 on the IEEE code of ethics on a daily basis regardless of whether we are working on a project or not. We have to understand the technology we are using and know the applications/consequences that the project entails. Number 3 on the IEEE code touches in with #5 in the aspect of finding the best application with the available data.

6.4 Future Work

As stated in “uncertainties”, we would do more research for our target sensor module and power module. Also, we would do more research on how to make this product more cost effective in order to make it commercially feasible. We would like to improve on the physical design as well by having a chassis with multiple layers of plates instead of implementing one ourselves. Finally, we would consider adding more features to our droid to make it more appealing.

References

[1] RC Flying Battle Robots, Hammacher Schlemmer, web page. Available at: http://www.hammacher.com/Product/86747?cm_cat=ProductSEM&cm_pla=AdWordsPLA&source=PRODSEM&gclid=CjwKEAiAoOvEBRDD25uyu9Lg9ycSJAD0cnByIKk0xZQLOUp9dgr-qBPlZkX-nBhmWhaOWV50icdiYhoC2PHw_wcB Accessed February 2017.

[2] Onyx battery manual, Hobbico Inc. Available at: http://manuals.hobbico.com/dtx/dtx-onyx-lipo-manual.pdf

[3] Nomtek, image. Available at: https://www.nomtek.com/wp-content/uploads/2014/11/8RHfT.png Accessed February 2017.

[4] ECE Average Salaries, web page. Available at: https://www.ece.illinois.edu/admissions/why-ece/salary-averages.asp Accessed February 2017.

[5] IEEE Code of Ethics, web page. Available at: http://www.ieee.org/about/corporate/governance/p7-8.html Accessed February 2017.

Appendix A: Requirement and Verification Table

Mobile Application Module

Requirement	Verification
1. Pressing of both the "toggleMode" and "shoot" buttons successfully send their respective control messages over wifi (4 points)	Equipment: Laptop, Phone, ESP, connector cables Procedure: A) connect both devices to laptop B) load code onto ESP and iPhone C) connect iPhone to ESP Wifi D) connect sockets and test each button, confirming that the correct signal is received by the ESP Presentation of Results: 2 booleans indicating successful reception the 2 signals
2. When in manual mode, the app correctly sends correct control signals motionValue and steeringDirection, which correspond to forward/ backward, left/ right (at varying intensities) based on the rotation of the device in the respective directions (4 points)	Equipment: Laptop, Phone, ESP, connector cables Procedure: A) connect both devices to laptop B) load code onto ESP and iPhone C) connect iPhone to ESP Wifi D) for each direction, place phone in start position and proceed to rotate phone slightly past 90 degrees in the given direction. E) confirm that the values seen start at 1023 and monotonically decrease to 0 for the respective pins Presentation of Results: 4 booleans indicating successful reception of forward, backward, left, right signals
3. Latency - droid must be able to initiate change in velocity within .25 seconds of phone rotation (4 points)	Equipment: phone, droid, stopwatch or slo-mo camera Procedure: A) position stationary droid at user's feet, with user holding phone in hands ready to steer B) user tilts phone forward, initiating the forward motion signal to be sent to the phone, and timer starts timing C) timer stops timing when car starts moving forward Note: this may be below human reaction time, so can also use slo-mo camera to measure time. Presentation of Results: A single real number indicating the time between phone rotation and droid movement

Microcontroller/wifi Module

1. Target acquisition - droid must be able to sense target, turn towards it, and shoot. (4 points)	Equipment: droid, target Procedure: A)place the droid on the ground, and place the target on the ground 10 feet behind the droid B) enable autonomous mode on the droid C) confirm that the droid turns around and fires at the target Presentation of Results: 2 boolean values, indicating that the droid successfully turned and successfully fired.

Power Supply Module

Requirement	Verification
1. Voltage regulator must be able to output 4.9 to 5.2 volts with a maximum current rating of 2.5 amps. (3 points)	Procedure: A) Attach 2 ohm power resistor as load B) Attach oscilloscope across load C) Verify that voltage is steady between 4.9 to 5.2 volts
2. Battery protection circuit must cut off power (<50mV) when the battery voltage falls below 6.4V. (4 points)	Procedure: A) Apply 8V to the battery connections with a DC power supply B) Measure voltage regulator output and ensure the voltage is between 6.4-6.6V. C) Decrease DC power supply voltage to 6.3V Measure voltage regulator output and verify that the voltage is <50mV
3. The overall system battery life should at least be 15 minutes (3 points)	Procedure: A) Upload system with proper software and let the vehicle run in autonomous mode B) Let the droid run for at least 15 minutes C) Verify that the droid is still running after 15 minutes D) Repeat this procedure 3 times to ensure consistency of verification

Motor Control Module

Requirement	Verification
1. H-bridge must be able to control the direction of each motor (4 points)	Procedure: A) From the microcontroller, output high (3.3 volts) to pin PWM1 on chip and low (0 volts) to PWM0 B) Verify motor is spinning in the forward direction C) From the microcontroller, output high (3.3 volts) to pin PWM0 on chip and low (0 volts) to PWM1 D) Verify motor is spinning in the reverse direction
2.H-bridge must control variability of speed (4 points)	Procedure: A) From the microcontroller, output low (0 volts) to PWM0 B) Output 30, 60, 90% duty cycle PWM signal in sequential manner to PWM1 C) Verify that the motor gets slower with the sequential increase in the duty cycle
3. H-bridge must be able to supply up to +/-800 mA to each motor (4 points)	Procedure: A) Attach 6.5 ohm resistor as load B) Attach oscilloscope across load C) Verify voltage is steady between 4.9 to 5.2 volts D) Repeat this procedure for other motor

Sensors Module

Requirement	Verification
1. Laser sensors must output <0.8V under normal operation and >2.5V when hit with a laser. (4 points)	A) Power circuit B) Use voltmeter to measure output with no laser C) Use voltmeter to measure output while exposing the sensor to the laser beam.
2. Target sensors must output a range of voltages between 0-1V corresponding to relative levels of light. (4 points)	A) Power circuit B) Use voltmeter to measure output under only ambient light. C) Use voltmeter to measure output while exposing phototransistors to IR light source in close proximity.

3. System should be able to detect the wall within 10 cm of it with an error of 1 cm based on the HC-SR04 readings. (4 points)	A) Place vehicle 12 cm away from a wall B) Ensure that the vehicle doesn't make an attempt to turn away from wall C) Place Vehicle 10 cm away from the wall D) Ensure that the vehicle turns away from wall