***University of Houston***

***Department of Electrical Engineering***

Michael Le

11706 Keswick Pines Ln

Houston, TX 77066

(713) 307 – 5771

December 14, 2015

John Glover

Cullen College of Engineering

University of Houston

4800 Calhoun Rd.

Houston, TX 77004

Dear Dr. Glover,

Attached is the 2016 IEEE Region 5 Robotics Competition Team 1’s final report for the 2015 fall semester. The report includes the technical details of the project, including the specifications and constraints and goal analysis, along with the goals accomplished, testing done to satisfy the accomplished goals, engineering standards, possible future risks, and the budget and expenditures to date.

We have completed all goals set for the fall semester. The Tiva-C microcontroller is able to send commands to the motorcontroller, as well as interpret signals from the distance sensors. The robot is able to follow the wall while avoiding it, perform turns and spins, and detect red and yellow victims. The securing platform has been built and is fully functional. The platform can lower and raise, and the arm can open and close. We are now ready to move on into the second phase of the project.

Should you have any questions or concerns about the report, please feel free to contact me at the phone number above or by email at [lemichael12@hotmail.com](mailto:lemichael12@hotmail.com).

Sincerely,

Michael Le

Enclosed

2016 IEEE Region 5 Robotics Competition

Team 1: 2015 Fall Final Report

Michael Le

Kain Dominguez

Trung Ngo

Phat Nguyen

Sponsored by: Dr. John Glover

**ABSTRACT**

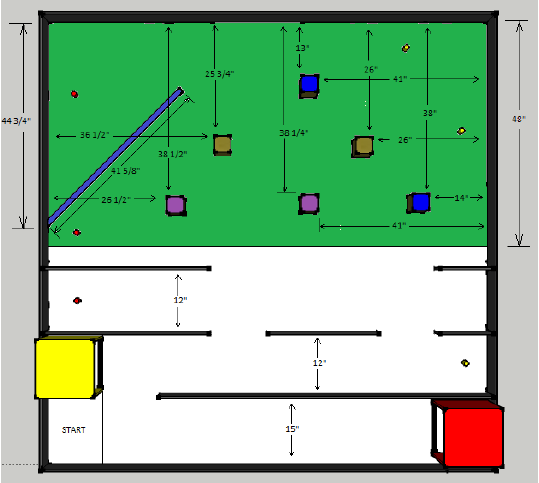
The purpose of this project is to bring recognition to the University of Houston’s IEEE Chapter and ECE Department by competing in the 2016 IEEE Region 5 Robotics Competition and winning first place. The theme of the competition is a search and rescue operation, where the competitors will have to traverse a course, pick up wooden dowels, and deliver them to their appropriate drop-off zone. To perform these tasks efficiently, our robot has three subsystems: embedded control system, image processing system, and securing platform. The goals of the fall semester were mainly focused on motor controls and image recognition. A Tiva-C microcontroller was used for the embedded control system, and a Raspberry Pi microcontroller was used for the image processing system. Extensive tests were done for each of the goals set for the fall semester, and all goals have been satisfied and completed. The robot is able to follow and avoid walls, perform turns, raise and lower its V-shaped platform, and recognize the dowels. The total expected cost for the fall semester is approximately $9,554.00, and we have expended $5,448.00 to date, which is well within the budget. We are ready to begin the final three goals needed to reach our 2016 target objective of having our robot deliver victims to their appropriate drop-off zone.

# Purpose

The purpose of this project is to bring recognition to the University of Houston’s IEEE Chapter and ECE Department. By designing an efficiently functional robot and winning first place, we intend to inspire and motivate engineering students to participate in IEEE.

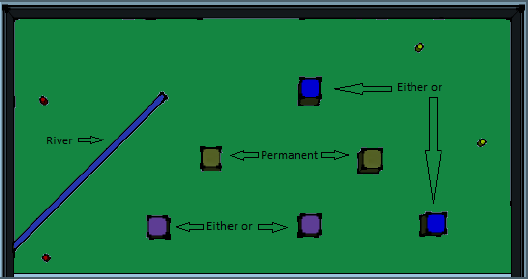
# Background

The theme of the competition is to locate and rescue injured victims in disaster zones using autonomous robots. Figure 1 shows the competition board that will simulate the disaster zone divided into a city section and an off-road section.



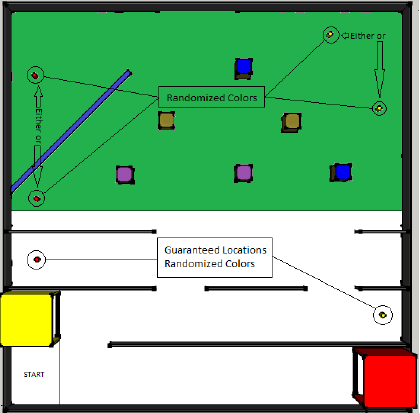
**Figure 1. Competition Board Simulating Disaster Zone with Dimensions Displayed[1].**

The board in Figure 1 is 8[ft2] and is divided evenly into two sections: the city section (white) and the off-road section (green). The city section is composed of a flat, smooth surface simulating paved roads, with walls that are 6[in] high and ½[in] thick. The off-road section’s surface is made of artificial grass to simulate rough terrain and has obstacles placed on it. A thin, wooden rectangular block, simulating a river, of 41.625[in] in length, and width and height of 1[in], will be placed at the top left of the board as seen in Figure 1. Wooden blocks are used to simulate obstacles of which the robot should avoid. The wooden blocks are 4[in] by 4[in] with a height of 6[in], and only four blocks will exist in six possible locations. Figure 2 shows the varying locations in which the obstacles will be placed.



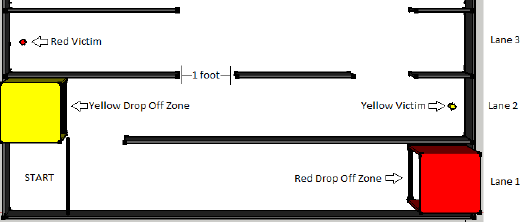
**Figure 2. Off-road Section Displaying Possible Locations of the Obstacles[1].**

The location of the wooden blocks in purple and blue will vary between two different locations, as seen in Figure 2, in each round of the competition. Each section will include two victims represented by cylindrical, wooden dowels. The wooden dowels are 1.5[in] in diameter and 2[in] in height, and will be colored red or yellow to indicate severity of injury. Figure 3 shows the location of the victims, where the victims are seen as small, round, red or yellow circles.



**Figure 3. Competition Board Showing Possible Locations of the Victims[1].**

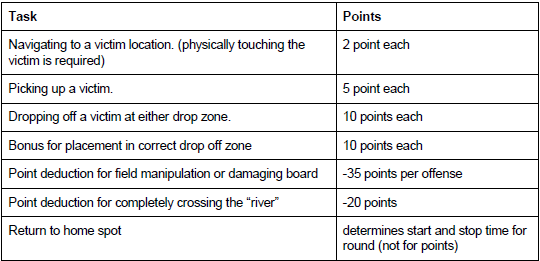
In each section, there will always be one red victim and one yellow victim. As seen in Figure 3, the city section will have guaranteed locations for the victims with randomized colors. The off-road section will have both randomized colors and locations for the victims. The victims will be secured and delivered to drop-off zones simulating care facilities. These drop-off zones will be located in the city section as seen in Figure 4.



**Figure 4. City Section Displaying Victims and Locations of Drop-off Zones[1].**

The drop-off zones are hollow cardboard boxes, measuring 1[ft3], with one open face. One box will be colored red and placed on the right side of Lane 1, as seen in Figure 4, and the other box will be colored yellow and placed on the left side of Lane 2. Victims are delivered to the drop-off zone that correspond to their color. The robot will have six minutes to earn points by successfully navigating to, securing, and delivering victims. Points will be deducted if the robot damages any walls or obstacles as well as if the robot crosses the river. In case of a tie, the deciding factor will be the length of time it took the robot to complete the tasks. Table 1 lists the details of the amount of points gained or lost in each round.

**Table 1. List of Scenarios and Their Corresponding Points Gained or Lost[1].**



As seen in Table 1, damaging the board or obstacles has a large point deduction of 35 points, while successfully delivering a victim has a total gain of only 27 points. The team with the highest score after three rounds will be declared the winner.

# Problem, Need, and Significance

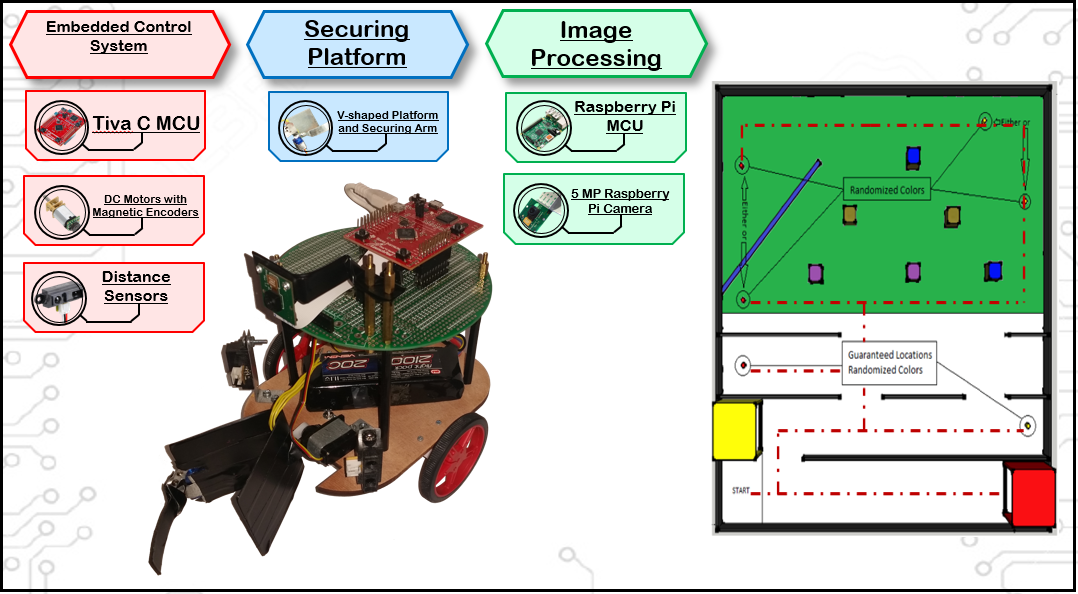
University of Houston’s engineering students are not actively participating in IEEE. A well-designed, efficiently functional robot can inspire and motivate students to participate in IEEE activities and increase growing interests in electrical engineering. IEEE, both the national organization and the UH branch, and the ECE Department will benefit from the publicity of the competition.

# User Analysis

Our robot will primarily be used by the IEEE Robotics Competition team and possibly the ECE Department for demonstration purposes to the students and staff. In order to use the robot for its intended purpose, a replica of the competition board is needed. The team will have expertise in embedded systems, microcontrollers, programming languages, and electronics, as well as experience in woodworking tools.

# Overview Diagram

Figure 5 below illustrates the overview diagram of the project, including the robot, its subsystems, and the competition board with the robot’s navigation path.



Embedded Control System

**Figure 5. Overview Diagram of the Robot, its Subsystems, and the Competition Board with the Robot’s Navigation Path.**

As seen in Figure 5, the robot comprises three primary systems: the embedded control system, securing platform, and image processing system. The embedded control system will be used for navigating through the board by using various distance sensors, DC motors with magnetic encoders, and a Tiva-C microcontroller. The image processing system will be used to detect the dowels as well as its color by using a Raspberry Pi microcontroller and 5-megapixel Raspberry Pi camera. The securing platform will be used to secure the dowels onto the robot by using a V-shaped platform, a securing arm, and two servo motors that will operate the securing arm and V-shaped platform. Figure 5 also shows the competition board with the predetermined path that the robot will take. This path was chosen to minimize the number of obstacles encountered by the robot and maximizing the amount of time the robot will be able to use the walls in order to align itself.

# Target Objective and Goal Analysis

The target objective for the 2015 fall semester is for the robot to be able to move, detect, and pick up a victim. The wheels and motors allow the robot to perform movement, the camera identifies and distinguishes between red and yellow victims, and the securing platform secures the victim in place without dropping it. Figure 6 depicts our goal analysis for both the fall and spring semesters.

Robot delivers victims to appropriate drop-off zone

Robot autonomously traverses board and finds victims

Robot arm sweeps and secures victim into V-shaped platform

Robot follows walls, performs left and right turns, and spins

Tiva-C sends commands to motorcontroller

Tiva-C interprets signal from distance sensors and determines distance

Robot lowers and raises V-shaped platform

Spring goals

Fall goals

Robot approaches victim and stops at ¼” from the victim

Robot distinguishes between red and yellow victims

Robot avoids walls

**Figure 6. Project Goal Analysis.**

The progress toward the goals are indicated by a colored circle. A completed goal is indicated by a green circle and a red circle indicates that progress on the goal has not been started yet.

# Specifications and Constraints

There are two primary constraints for the robot: the time limit to complete the course and its size. The rules of the competition states that the robot has a maximum of six minutes to complete the course. The rules also state that the robot must, initially, fit inside a box of one cubic foot (1,728[in3]) before running the course. Once the robot begins traversing the course, the robot may extend past the one cubic foot dimension.

Our robot has a length and width of 6[in] and a height of 8[in] to easily maneuver through the predetermined path seen in Figure 5. The estimated distance the robot will travel is 33.32[m], which was calculated by taking the longest route to complete the course. To complete the course in under six minutes, we estimated the robot, with a wheel diameter of 40[mm], should travel at 9.26[cm/s] to traverse the course under the time limit. However, the robot will travel at 13.88[cm/s] to traverse the course in four minutes to reserve the time needed to secure the victims. The RPM needed to travel at this velocity was calculated to be 66.3[rpm], and the minimal torque needed is 0.0883[N∙m]. A 100:1 Brushed DC Motor that has a maximum RPM of 320 and torque of 0.216[N∙m] is used, which more than satisfies the specification. The lanes in the city section of the competition board are at least 12[in] wide, and our robot will be centered between the walls of the lane when traveling through it. Therefore, there will be a gap of at least 3[in] from the sides of the robot to the walls. Infrared distance sensors with a range of 4 to 30[cm] (1.6 to 11[in]) are placed on the sides and front of the robot to avoid and follow the walls. An additional infrared distance sensor with a range of 0 to 10[cm] (0 to 4[in]) is placed inside the V-shaped platform to ensure the victim is inside.

The V-shaped platform will be used to hold the victim while an arm, attached to a servo motor with a maximum torque of 22.2[oz/in], will keep the victim inside. The platform is made of sheet metal and lined with electrical tape to improve frictional contact. The platform is 2[in] when standing vertically, and its opening is approximately 60 degrees such that the length from one edge of the opening to the other is 2.25[in]. The arm is also made of sheet metal and is 2.25[in] in length, curved slightly, and lined with electrical tape to improve grip and effectively secure the victim inside the platform. The total weight of the platform, victim, arm, and arm servo motor is 2.5[oz] or 70 grams. A servo motor with a maximum torque of 30.55[oz/in] is used to lift the securing platform and the victim, and it can lift several times the total weight of both the platform and victim. In addition to traversing the competition board in four minutes, the securing platform will deploy, secure the victim, and retract in four seconds to stay under the time limit.

Statement of Accomplishments

**Tiva-C Sends Commands to Motorcontroller**

For this goal to be completed, we tested the response of the motorcontroller to various inputs. In order to drive the robot, we connected two motors, which are driving two independent wheels, to a motorcontroller. A Bluetooth device was used to wirelessly send commands to a Tiva-C microcontroller over a serial port from a computer. From the Tiva-C, we send a pulse-width modulated signal (PWM) to set how fast the motors will turn. A PWM signal is a square wave with a frequency motors can respond to and the duty cycle of the pulse is varied to adjust the motor’s angular speed, in our case, the frequency is set from 50 to 60[Hz]. Our motors stop spinning at 5% duty cycle and maximum speed is reached at 100% duty cycle. From the microcontroller, signals are sent to the motorcontroller, which then drives the motors. The motors, at full speed, take in a voltage of 6[V], therefore the motorcontroller will have an output varying from 0 to 6[V] in order to drive the motors. The output voltage of the motorcontroller was recorded every 10% duty cycle (See Appendix A). Figure 7 shows the relationship between the voltage and the duty cycle percentage of the PWM signals.

**Figure 7. Graph of the Motorcontroller Voltage Output to Duty Cycle Percentage.**

As seen in Figure 7, the voltage in the motorcontroller varies as the duty cycle percentage increases, therefore proving that communication between the Tiva-C and motorcontroller exists.

**Tiva-C Interprets Signal from Distance Sensors and Determines Distance**

The goal is completed if the Tiva-C is able to receive the ADC value from the distance sensors. A driver for the distance sensors was written to use the ADC modules of the Tiva-C. Since the ADC module has a resolution of 12 bits, the ADC value will range from 0 to 4095, where the higher the value, the closer the detected object is. The Sharp GP2Y0A41SK0F Analog Sensor was chosen to be our desired 4 to 30[cm] distance sensor. The Bluetooth device was used to display the ADC value received from the distance sensors to a computer. The ADC value per centimeter was recorded by measuring the ADC value at specific distances (See Appendix B). Figure 8 shows the distance to ADC value of the sensor.

**Figure 8. Distance to ADC Value Graph of the Sharp GP2Y0A41SK0F Analog Distance Sensor.**

Each tick is separated by an interval of one centimeter from 4 to 20[cm] (1.6 to 7.9[in]). This range was chosen because it is unnecessary to measure the ADC value at a farther range. As seen, the difference in ADC value shortens as the sensor detects objects at farther distances. To calculate the distance accurately from the ADC value, the graph shown in Figure 8 is linearized. The sensor ADC value as a function of distance is approximated to

where *V* is the ADC value, *d* is the distance, and *k* is the linearization variable[2]. The inversion acts as a linearization function. Figure 9 shows the linearized graph with respect to the ADC value where *k* = 4.

**Figure 9. Linearized ADC Graph of the Sharp GP2Y0A41SK0F Analog Distance Sensor.**

The red line indicates the linear trend line of the graph. The value of *k* was chosen to be 4 because it provided the best linear trend line. Since our program will be using integer math, the equation used to find the distance with respect to the ADC value is

where *m* is the linearized slope and *b* is the y-intercept. These constants are pre-computed to provide integers for our code. The equation was implemented in our code and tested for accuracy. Table 2 below shows the measured distance with respect to actual distance.

**Table 2. Measured vs. Actual Distance with Linearized Equation.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Distance[cm] | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Measured[cm] | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 11 | 12 | 13 | 14 |

The measured distance is equal to the actual distance up to 11[cm], where the measured distance becomes offset by one centimeter thereafter. The values in Table 2 prove that the distance sensor can accurately measure distances from 4 to 11[cm]. These measurements are accurate enough for the linearized method to be reliable. Using this method, we can determine the reference ADC values that will be used to determine the distance that is needed.

**Robot Follows Walls, Performs Left and Right Turns, and Spins**

To test if the robot follows walls, perform left and right turns, and spins, a Bluetooth device was used to send “follow” and “turn” commands from a computer to the Tiva-C. A “follow” movement is satisfied if the robot can follow the wall while maintaining its distance from the wall. A PID control algorithm was developed, which would give feedback from the distance sensors and calculate an appropriate signal to command the motors. The PID controller was tuned to keep the robot moving parallel with a wall while maintaining a distance of approximately 4[in]. The distance between the robot and the wall, as the robot moved along the wall, was recorded and our results can be seen in Figure 10.

**Figure 10. Graph of the Robot’s Distance from the Wall with Respect to Time.**

As can be seen in Figure 10, it takes the robot approximately 2[sec] to align itself 4[in] away from the wall and continues to remain aligned as time continues to increase. Although there are minor oscillations in the steady state section of the graph, they are minor and do not affect the robots ability to perform its tasks. A “turn” and “spin” movement is satisfied if the robot can turn uniformly in place from its center. To have the robot turn, we needed to command one wheel to spin counterclockwise and the other clockwise. We tested the robot by applying different constant speeds and timed the robot as well as by visually inspecting it. Table 3 shows the time it took our robot to turn at various degrees while a duty cycle of 40% was sent to the motors.

**Table 3. Robot Turning Time at 40% Duty Cycle.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Time[ms] | 90 | 135 | 180 | 225 | 270 |
| Degree | 60 | 90 | 120 | 150 | 180 |

As can be seen from Table 3, our robot was able to perform turns varying from 60 to 180 therefore showing that our goal was accomplished.

**Robot Avoids Walls**

To test if the robot could avoid walls, the robot was placed on a path perpendicular to a wall. The test is considered successful if the robot can detect the wall and stop before colliding with it. The results of the test done by placing a wall perpendicular to the robot can be seen in Figure 11.

**Figure 11. Speed of the Robot vs. Frontal Wall Distance.**

As can be seen in Figure 11, the robot approached the wall at its maximum speed of 120[RPM], and at a distance of 4[in] away from the wall it stopped, therefore accomplishing its goal.

**Robot Lowers and Raises V-Shaped Platform**

The goal of lowering and raising the V-shaped platform was verified by visual inspection. The robot’s V-shaped platform was attached to the robot while maintaining a distance of approximately half an inch off the ground. The servo was connected to the Tiva-C microcontroller and a command was sent to turn the servo’s arm 90 backward and forward, which caused the platform to raise and lower. After many trials we could visually see that the platform did raise and lower at the appropriate angles necessary to pick up and secure the wooden dowels, therefore the goal was accomplished. In addition to the testing done on the V-shaped platform’s servo, the servo controlling the arm used to secure the victim was also tested. The arm was able to open approximately 120 from its closed position and when a wooden dowel was placed at the opening of the V-shaped platform it was able to successfully return to its closed position and secure the wooden dowel onto the V-shaped platform.

**Robot Distinguishes Between Red and Yellow Victims**

To detect and distinguish between red and yellow victims, a Raspberry Pi 2 microcontroller and a 5-megapixel camera were used. The Raspberry Pi uses the OpenCV library to realize our image processing system. The average time for an image to be processed by the Raspberry Pi is approximately 1.25 seconds, which will help to assure that the robot can finish each round under the allotted time. Testing for victim recognition was done in the Senior Design Lab and the Robotics Lab. Both labs have different lighting conditions, where the lighting condition in the Robotics Lab is brighter than in the Senior Design Lab. Since the lighting conditions at the competition are unknown, performing victim recognition tests in these labs are crucial. The red and yellow dowels were placed in various distances within 1 to 5[ft] of the camera, and the test was done ten times for each 1[ft] of distance. A victim’s detection is considered successful if more than 50% of the victim’s pixel area was detected. Figure 12 shows a graph of the average percentage of the area of the victim detected at various distances in the Senior Design Lab.

**Figure 12. Average Percentage of the Area of the Victim Detected at Various Distances in the Senior Design Lab.**

A percentage of 90%, for example, means that an average of 90% of the victim was detected. As seen in Figure 12, over 95% of the victim was detected in every trial, with the victim placed at 5[ft] having the best results at 99% for both red and yellow. Likewise, Figure 13 shows a graph of the average percentage of the area of the victim detected at various distances in the Robotics Lab.

**Figure 13. Average Percentage of the Area of the Victim Detected at Various Distances in the Robotics Lab.**

As seen in Figure 13, the results for red in the Robotics Lab were poorer than those in the Senior Design Lab. However, since none of the detections at the various distances were below 50%, the test proved successful.

# Engineering Standards

C programming language was used as one of our engineering standards in our software interface. In order for the team to collaborate on the different programming modules, a programming language that the entire team knows and understands was used. If team members were to use different programming languages, then it would be difficult to aid members in troubleshooting if the members helping are not familiar with a particular language.

Another standard used was wireless Bluetooth communication with the robot in order to test the robot’s various components. To allow the robot to move freely without having to have cables attached to it and to help prevent damage to the microcontrollers’ micro-USB ports, it was decided to use wireless communication with the robot. In particular, Bluetooth wireless communication was selected since all team members’ computers either had Bluetooth technology built into them or were able to be adapted. By using one type of wireless technology we were able to reduce the amount of components on the robot’s PC board as well as allowing all members to receive the same data sent back from the robot.

# Budget

Table 4 shows the team’s estimated budget for the fall semester and the amount expended to date.

**Table 4. Project Budget for the Fall Semester.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Total projected** |  | **Expended to date** |
| **Labor** |  | **Hourly rate** | **Total hours** |  |  |  |
|  | **Team Members** | **$ 10.00** | **420** | **$ 4,200.00** |  | **$ 3,600.00** |
|  | **Consultants** | **$ 180.00** | **28** | **$ 5,040.00** |  | **$ 1,500.00** |
| **Parts** | | | | | | |
|  | **Microcontrollers** | **$ 45.00** |  |  |  | **$ 45.00** |
|  | **Camera** | **$ 25.00** |  |  |  | **$ 25.00** |
|  | **DC Gearmotors** | **$ 34.00** |  |  |  | **$ 34.00** |
|  | **Servo Motors** | **$ 72.00** |  |  |  | **$ 72.00** |
|  | **Distance Sensors** | **$ 27.00** |  |  |  | **$ 27.00** |
|  | **Wheels** | **$ 5.00** |  |  |  | **$ 5.00** |
|  | **Ball Casters** | **$ 8.00** |  |  |  | **$ 8.00** |
|  | **Gearmotor Brackets** | **$ 5.00** |  |  |  | **$ 5.00** |
|  | **Motor Driver** | **$ 5.00** |  |  |  | **$ 15.00** |
|  | **Voltage Regulators** | **$ 15.00** |  |  |  | **$ 30.00** |
|  | **Prototyping PCB** | **$ 12.00** |  |  |  | **$ 12.00** |
|  | **Robot Chassis** | **$ 10.00** |  |  |  | **$ 10.00** |
|  | **Encoder For Motors** | **$ 9.00** |  |  |  | **$ 18.00** |
|  | **Shipping Fee** | **$ 42.00** |  |  |  | **$ 42.00** |
|  | | | | | | |
|  | **Total cost of Robot** |  |  | **$ 314.00** |  | **$ 348.00** |
|  |  |  |  |  |  |  |
| **Total** |  |  |  | **$ 9,554.00** |  | **$ 5,448.00** |

As seen in Table 4, we have gone over the estimated cost of the robot since some parts needed to be replaced due to damage. However, we are still under the budget for the total cost of the semester.

# Risks

A potential risk that we may face in the final implementation of the project is that the robot may not be able to approach and secure the victim as expected. The main purpose of the camera is to detect and determine the color of the victim. The camera will not be used as feedback to accurately determine how far the victim is. Magnetic encoders are used to help determine distance; however, it may not be very reliable 100% of the time. By visually inspecting the course in Figure 1, we will determine how far the robot will need to move before stopping at 8[cm] from the victim. This will let us use the 0 to 10[cm] sensor in front, while following the wall, to position the victim inside the platform. In the case that the above method fails to accomplish the task for the final project, we will attach a much longer range front sensor and use the feedback from the wall behind the victim. Another possible option is to research and implement feedback from the camera. This may include using a different image recognition system.

Another risk that we may face is that the camera may detect red or yellow outside of the course. In our design, we will angle the camera down such that it can see the victim at a range of 4[ft] while the camera will not see over the exterior walls of the course. As a contingency plan, we can have the camera perform shape recognition and determine the color pixel area as well. By using both the shape and area of the color detected, the Raspberry Pi can determine if it’s a victim or not. We are researching other camera and microcontroller options that may be faster as well as improving and refining the current code used to control the detection process.

# Conclusion

IEEE Team 1 is well on its way to inspiring and motivating engineering students to participate in IEEE by winning first place in the IEEE Region 5 Robotics Competition. To date, we have completed all of the goals proposed for the fall semester, which include sending commands to the motorcontroller using a Tiva-C, having the Tiva-C interpret signals from distance sensors, and programming the robot to follow walls, perform left and right turns, perform spins, avoid walls, lower and raise its V-shaped platform, and distinguish between red and yellow victims. There are three remaining goals that need to be accomplished in the spring semester so that we may reach our final goal of having our robot deliver victims to their appropriate drop off zone. We had a very successful fall semester and with this current momentum we know that we will be able to deliver a final product in time for the robotics competition.

# References

1. *2016 IEEE R5 Conference Student Robotics Competition Rules*. N.p.: IEEE, 2 Sept. 2015. Pdf.
2. Acroname, ‘Linearizing Sharp Ranger Data’, 2015. [Online]. Available: [2]. <https://www.acroname.com/articles/linearizing-sharp-ranger-data>. [Accessed 12- Dec- 2015].

# Appendix A

**Table 5. Data for the Voltage Output of the Motorcontroller vs. Duty Cycle.**

|  |  |
| --- | --- |
| Input/Duty Cycle | Voltage (RMS) |
| 0.00% | 0 |
| 10.00% | 1.8 |
| 20.00% | 4.2 |
| 30.00% | 4.9 |
| 40.00% | 5.3 |
| 50.00% | 5.58 |
| 60.00% | 5.73 |
| 70.00% | 5.83 |
| 80.00% | 5.9 |
| 90.00% | 5.95 |
| 100.00% | 6 |

The data in Table 5 was acquired by sending specific duty cycles to the motorcontroller while measuring the output voltage using a voltmeter.

# Appendix B

**Table 6. Data for the Linearization of the Distance Sensors.**

|  |  |  |
| --- | --- | --- |
| Distance[cm] | ADC | 1/(d+k) |
| 4 | 3250 | 0.125 |
| 5 | 2760 | 0.111 |
| 6 | 2470 | 0.1 |
| 7 | 2080 | 0.091 |
| 8 | 1860 | 0.083 |
| 9 | 1694 | 0.077 |
| 10 | 1508 | 0.071 |
| 11 | 1361 | 0.067 |
| 12 | 1283 | 0.063 |
| 13 | 1160 | 0.059 |
| 14 | 970 | 0.056 |
| 15 | 770 | 0.053 |
| 16 | 750 | 0.05 |
| 17 | 738 | 0.048 |
| 18 | 680 | 0.045 |
| 19 | 650 | 0.043 |
| 20 | 578 | 0.042 |

The data in Table 6 was acquired by placing a flat object at specific distances from the sensor and measuring the ADC value. The inversion of the data [1/(d+k)] provides the linearization of the distance and ADC value obtained. The value of *k* was chosen to be 4 as it provided the best linear trend line.