EGR 107 ROBOT PROJECT
FALL 2018

Collin Rogers
Renzo Garza Motta
Aziz Gram Sarhan Jr.

EGR 107 Section Number 02
Team Number 5
Professor Terry Stevens

6th December 2018

Final Report Grading Rubric

<table>
<tr><th>Item</th><th>Points</th><th>Score</th></tr>
<tr><td>Neatness and Organization
Cover Page – including team member names, Team #, Section #, professor name, date
Rubric included
Table of content with page numbers
Binding
Neatness
Report order follows the Rubric order
Final Report uploaded to BB
Misc</td><td>30</td><td></td></tr>
<tr><td>Introduction- An detailed overview of the entire project</td><td>10</td><td></td></tr>
<tr><td>SECTION 1 - Design Procedure

Description of the design process of your machine. Include Mechanical structural design, Electrical system design, and Software development. You may reuse (“improved”) documents from prior robot assignments.
Gantt Chart
Project constraints
Project specifications
Initial chassis design. Include sketches & drawings.
Sensor selection. Include process.
Function Structure Diagram (FSD)
Benchmarking - Include sketches, pics, documentation.
Brainstorming – Include sketches, pics, documentation.
Concepts generation and sketches
Concept selection matrices
Prototypes - pictures, comments, summary
Design Iteration 1
Design Iteration 2</td><td>75</td><td></td></tr>
</table>

SECTION 2 – Midterm Progress
 	
• A picture of your assembled robot at midterm competition
• Results of Midterm competition
    ○ Qualification score
    ○ Competition rank
    ○ Takeaways
• Software flow diagram (at Midterm)
• Wiring/schematic diagram of electrical system (at Midterm)
	50	

SECTION 3 – Finalized Design Summary • Results - specification list showing that each specification was verified for your robot. • Discussion of strength and weakness • Specific strategy & goals for Final Competition • Future Improvement • Conclusions	20	
Appendix A: Final Software Flow Chart/ State Machine Diagram	20	
Appendix B: Final programming code. All.	20	
Appendix C: Final electrical wiring diagram of robot	20	
Appendix D: Final budget for machine with all the receipts attached	20	
Appendix E: Complete Working Drawings of the finalized robot. Detailed orthographic drawings of all finalized components and an assembly drawing of the entire machine. SolidWorks models for all parts and assemblies. (You may simplify the modeling of the electrical parts. i.e. Use a rectangular prism to represent the Arduino board.)	75	
Appendix F: G Code. For all the CNC machined parts. G code for each team members initials. Design diagrams or sketches for G code development.	20	
Appendix G: Robot assembly instructions and any additional information. Recommended spare parts. Maintenance suggestions.	20	
Appendix H: Technical Data Sheets on all electro/mechanical parts	20	
TOTAL	400	

Robot Project Assignment 12 – Final Report
Submitted: 6th December 2018

The Final Report should be a well-organized Word or .pdf document that records **ALL** information for the design process.

This must include Mechanical structural design, Electrical system design, and Software development.

Document **ALL** revisions and iterations during the design process. Keep records of **ALL** work completed and include it in the Final Report. Take numerous pictures during the process for inclusion in the Final report. Keep notes, sketches, etc.

Include the grading rubric pages (shown below) in your report as the first pages following the cover sheet.

Follow the order of the Rubric exactly and include everything requested.

Submit the Final Report on BB AND submit the report in a BOUND hard copy.

<u>**Table of Contents**</u>

Design Procedure

Empathize

This step is to gather information on customer/audience in this case Dr. Sirkus. In this step you are to observe, empathize and immerse. In the case of the Robot Sockey competition our information is the Robot Sockey specification & rules. The team should fully understand what is allowed, prohibited and are allowed to assume anything stated is fair game.

Define

The defining step is to develop a short and sweet problem statement. The team original problem statement was and still is “To store ball and score goals autonomously”. At this point we need to have a complete understanding of the problem so that the team can focus on the solution. In this phase any information spoken by the audience should be taken in to consideration into the problem statement. The announcements from Dr. Sirkus would act as prompts to adjust our problem statements. The method to define is to develop a problem statement, set scope of the project and determine the specifications, constraints & critical functions.

Critical Functions:
Durability: Are Robot main goal is to survive. Due to the nature of the Competition it is highly frowned upon to build an attack Robot however we should build a Robot that can take a couple hits and be unaffected.

Speed: We need to build a Robot that is fast enough to collect majority of the ball but stable enough that it doesn't topple over or crash and not be able to recover.

LED we can call these Phase Lights. We need to know what our Robot is thinking at all times. It'll make it easier to spot malfunction in the prototype stage, it will show good design protocol (points)

So, to summarize our critical functions are: Durability, pick up Multiple balls, Speed, Robot - Human Feedback.

Ideate

In this stage we brainstormed all the possible idea by using a morphological matrix. This allowed us to try the variation of ideas we came up with and filter them through the critical functions which we had decided on. Since we had an abundance in ideas, we had to condense more by using criteria weighting matrices and concept selection matrix which allowed us to see what ideas were important using quantities rather than by a qualitive means which may have cause disagreements. By doing this we came to a decision on an idea that met all of our critical function and did so in the way the team felt most effective. This was the foundation in what our first chassis design prototype was derived from.

Prototype

The idea of our prototype shown in the pictures above was exploring the paddle design idea in which the robot will pick up and store balls.

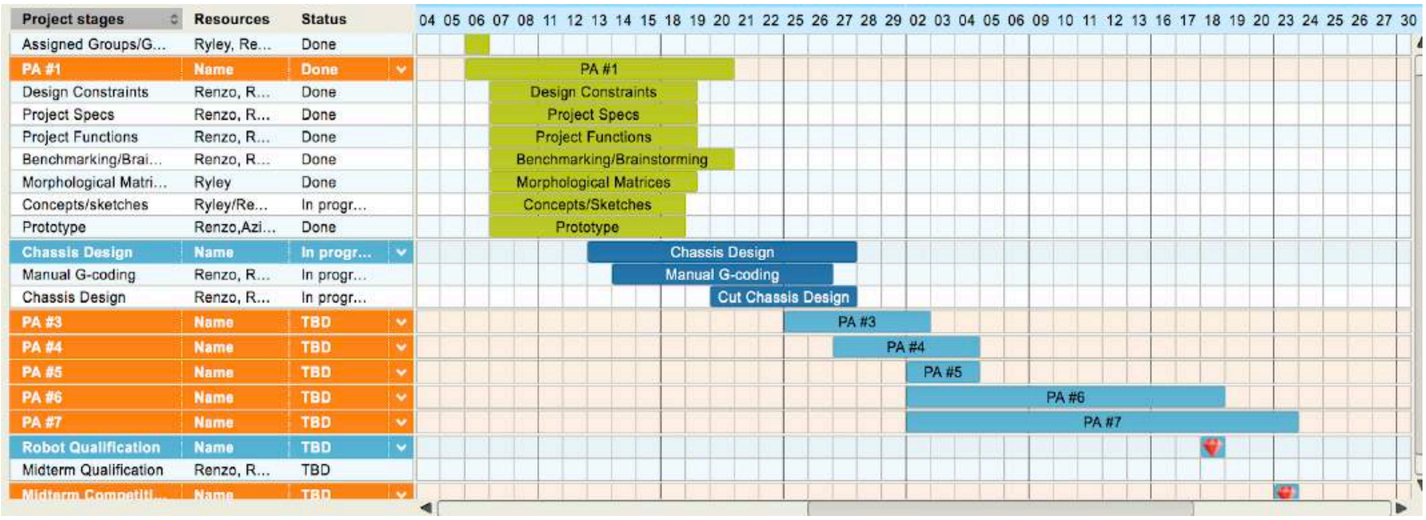
Introduction

The EGR 107 Robot Sockey Project began on the first day of class. Our first day of class was a class quiz on coding in order to evaluate our skill which we had retained from EGR 106, which would influence the 3 – 4-person groups. The semester was dedicated to learning how to power a totally autonomous robot that can sense its surrounding, maneuver around the Robot Sockey field and score goals. Every class in the EGR 107 either directly or indirectly applied to the end of semester goal of having a working robot to compete in the final competition. The assignments that focused on the progress of the overall robot were the project assignments which included becoming experienced with the Arduino programming, various sensors, motors & generating/ innovating ideas.

How the Robot Sockey works is an arena which takes up to 4 robots which has stationary obstacles with black tape which dissect the board into 4 which lead to the goals. Each of the robots are instructed to start in the middle, turn on and let them run completely autonomously. The robot should then be able sense the balls on the table; which range from golf ball, ping pong balls and raffle balls. After the ball has been sensed, the robot should store it and score the ball(s) in a controlled fashion. Goals can also be scored by unintentionally which will be judged accordingly and will be reward less points. A control goal is worth 5 points and a unintentional goal is worth 1 point.

The robot which scores the most point by the end of regulation is considered the winner, in the situation of a tie and there are no obvious distinguishing factors such as robot design, repeatability and face-off then a coin toss is also plausible. The robot that has the most point by the end of the final competition is the winner.

GNATT CHART



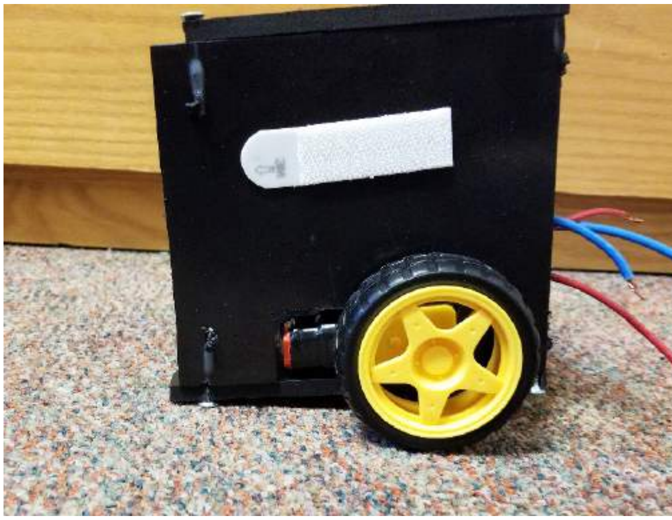
PROJECT CONSTRAINTS

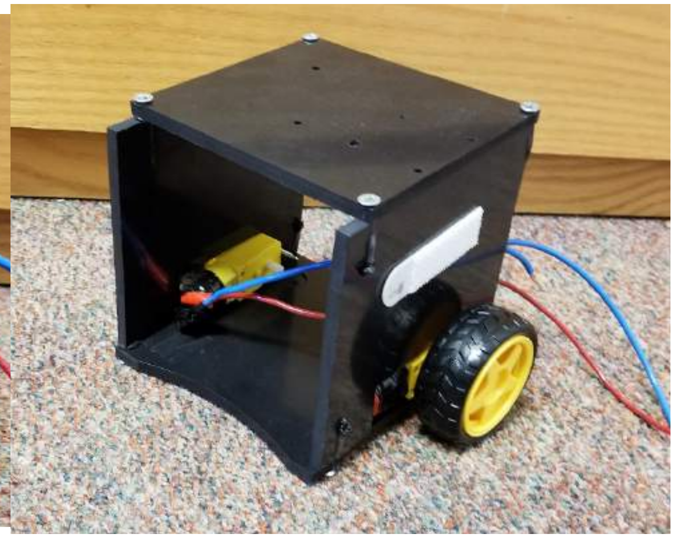
- The robot must be durable to sustain hits from obstacles, walls & other robots
- The robot should be able to detect the black lines
- The robot should be able to sense other robots and walls
- The robot must be able to differentiate between an obstacle and ball
- The robot should not hurt or put any of the other competitors at risk
- The robot should not destroy the field of play
- The robot must be able to be easily disassembled in case of repairs
- The robot must be able to easy assemble

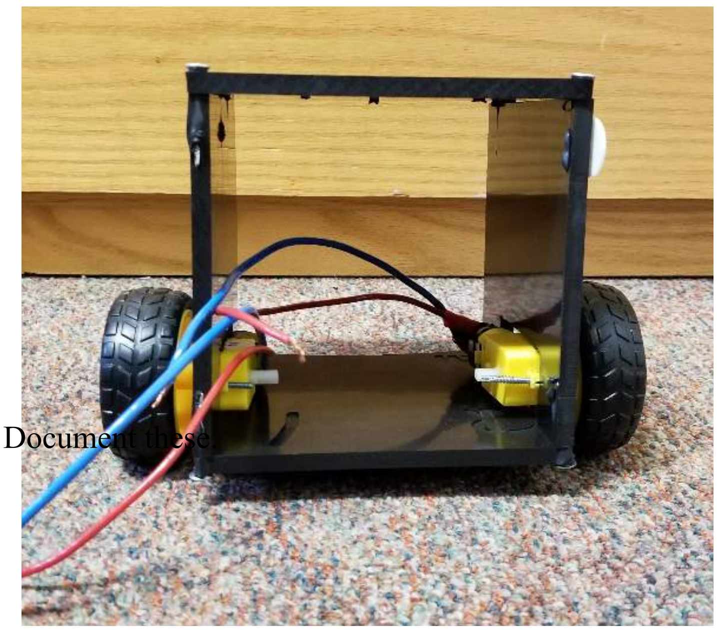
PROJECT SPECIFICATIONS

I. The Total cost of all parts cannot exceed $80
II. To qualify, the robot must not exceed the dimensions 7" x 7" x 7'
III. The robot must be less than 10 lbs.
IV. The battery powering the robot must provide at least 1V
V. The robot must be able to move at least ½ lbs. objects
VI. The robot must move at 1mph or faster
VII. The robot must withstand a 3-foot fall
VIII. The robot must be able to move spheres that are $2in^3$
IX. The robot must remain totally autonomous while in the field of play
X. The robot must be able to autonomous for at least 3 minutes

INITIAL CHASIS DESIGN







Identify, Strong & Weak Points

Are there any specifications you have not met? Document these.

Specification **not** met:

- Robot Numbering Rule
- Withstand 3-foot drop

How many specifications does your design meet so far?

- Currently out robot meets all the requirements for the midterm competition, however our robot lacks repeatability and simplicity, so it may run and met requirements it may not be able to perform critical function

Is there any specification that you need to update?

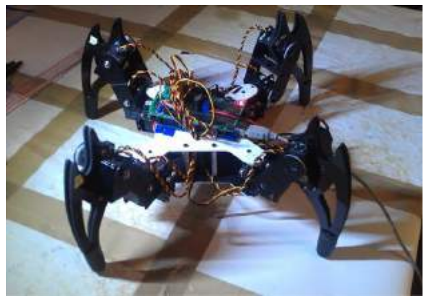
- Our robot has been instructed to have a button for the final competition
- Robot must be built more durably to withstand 3-foot drop
- We must incorporate the robot number rule.

<u>SENSOR SELECTION</u>

For our robot we opted for the Ultrasonic Sensor because of the distance it capable of sensing, it has a large range and was cost effective since it came with the original EGR 107 kit. One of the down falls of this sensor is that it has minimum sensing distance of about 2cm. Also, it seems that with less dense objects it can become unreliable. We decided to use 2 Ultrasonic sensors located on perpendicular sides on the robot which solved our initial problems, allows us to wall sense & sense objects on the field of play including other robots.

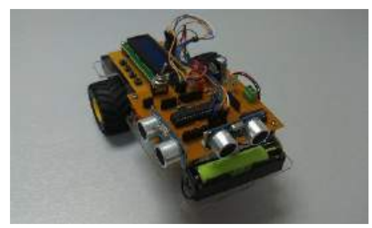
<u>Sensors:</u>

- IR Sensor
- Digital Reflectance Sensor
- Laser & Detector



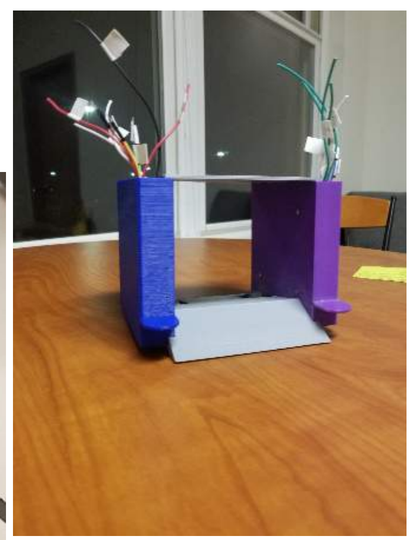
<u>Sensing:</u>

- Follow Black Line
- Follow wall
- Random Movement
- Straight Line movement
- Proximity Sensor



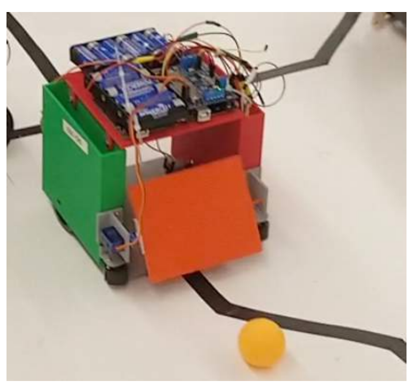
<u>Capture Balls:</u>

- Vacuum
- Funnel
- Bridge
- Paddle Wheel



<u>Release Balls:</u>

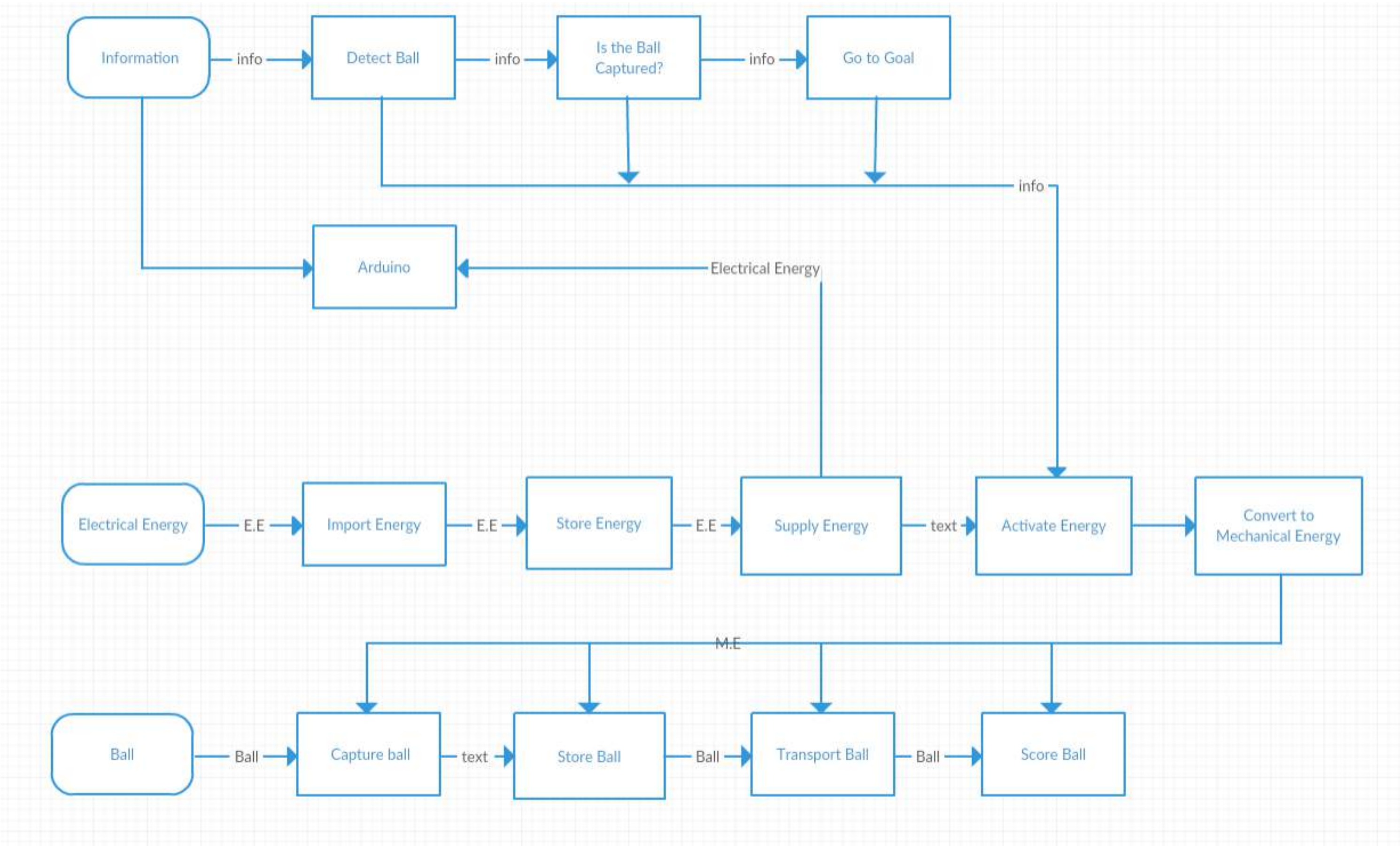
- Ramp
- Spring
- Piston
- Open Bridge
- Reverse Paddle Wheel



<u>Power:</u>

- AA Battery
- AAA Battery
- 9V Battery
- 6V Parallel Pack

Function Structure Diagram (FSD)



Brainstorming/Benchmarking

We want to survive and in order to do that we need to score goals but at the same time prevent others from doing so.

In business for you to own the company if you have 51% of the stakes. This is because at this point nobody can own more than you. The task is to own 51% of the ball on that table. So essentially ball hog and

Our robot must have pockets that allow us to hide our wires to prevent being disconnected on the field of play.

After storing the ball we can score by paddle, dropping the ball off like a dump truck using a bridge like contraption to guide the balls in.

CONCEPT SELECTION MATRICES

Table 1							
Criteria	Speed	Durability	Ease of Assembly	Siz e	Capacity	**Mean**	**Weight**
Speed	1	2	2	1	3	1.6437	0.3128
Durability	0.5	1	0.5	1	_	0.6597	0.1257
Ease of Assembly	0.5	2	1	0.5	2	1	0.1903
Size	1	1	2	1	1	1.1486	0.2185
Capacity	0.333	2	0.5	1	1	0.8027	0.1527
					Total	5.2547	1

Table 2							
_	_	Concept 1		Concept 2		Concept 3	
Criteria	Weight	Rank	Score	Rank	Score	Rank	Score
Speed	0.3128	1	0.3128	2	0.6256	2	0.6256
Durability	0.1257	5	0.6285	4	0.5028	5	0.6285
Ease of assembly	0.1903	5	0.9515	3	0.5709	4	0.7612
Size	0.2185	3	0.6555	1	0.2185	3	0.6555
Capacity	0.1527	4	0.6108	5	0.7635	3	0.4581
Totals		1	3.1591	3	2.6813	2	3.1289

DESIGN ITERATION 1

Ease of Fixing Problem			Importance of Fixing Problem	
Weakness	Rank		Weakness	Rank
Goal Avoidance	5		Goal Avoidance	4
Assembly	4		Assembly	3
Wiring Control	3		Wiring Control	1
Component Distribution	2		Component Distribution	2
Ball Collection	1		Ball Collection	5

Strong Points	**Weak Points**

Design - Keeps wires hidden	Ball collection - Paddle system needed to be more effective and receiving balls
Lightweight - allows agility on field	Battery location - In a vulnerable location which could lead to a disconnection of the battery in the field of play
Durability - Strong plastic that can handle impacts that the robot will be exposed to	Assembly - Takes time to assemble, this means if there is a hardware problem the robot will take a lot of time and effort to fix
Simplicity - Design is easy, low cost & effective	Controlled goal - Robot struggles to execute a controlled goal which is important to our success

Rank the Weaknesses

The weakness we found in our robot are the following goal avoidance, assembly, wiring control, component distribution, and ball control. Goal avoidance is in terms of not allowing the robot to drive itself into the goal falling to the ground and breaking. Assembly means the attachment of all the parts together because as of now everything was put together with hand drilled holes that don't always line up the best causing everything to not be square. Wiring needs to be managed and organized better so we don't have a lot of wires coming out of nowhere making problem solving very difficult. Component distribution is meant to mean how we organize the electrical components throughout the robot along with that a proper place to store the batteries. Then finally ball collection due to the intended collection was not working during the midterm competition so we had to hope we scored with the method we chose last minute.

Ideate: brainstorm ways you might fix the problem

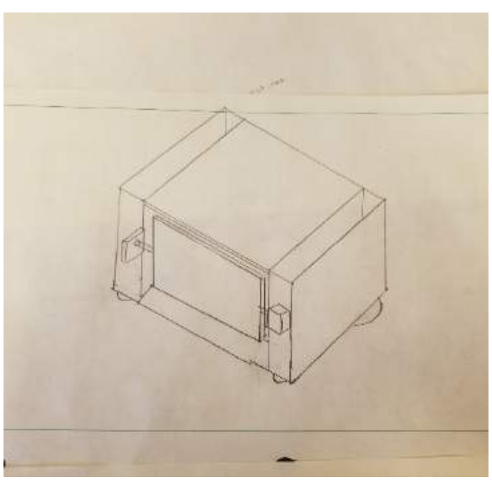
To fix the problem of goal avoidance we can go as simple as a bar that extends up so the robot is taller than the goal preventing it from driving through the goal and possibly damaging itself. On the other had we could use an IR sensor to detect the IR output above the goal so that if we are too close the robot will know to back up and change its direction.

Assembly is very simple as well because all we need to do is 3D print the parts again but this time having the holes already made and lined up in an assembly to make sure everything fits. We will also use uniform hardware throughout so if we do need to take it apart there is no need to switch tools or find a toll when we can use the same tool throughout.

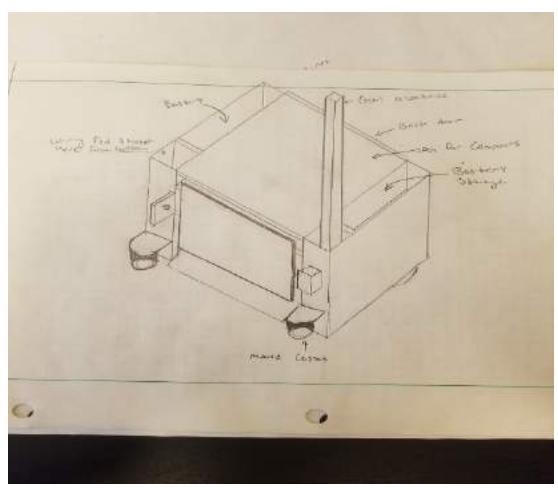
Wiring control will also be solved within the process of printing because we will include designated pathways for the wire to travel. This will be completed by including a larger area within the base for the wires to be ran. Along with that there will be wall to separate the wheel from the wires within the sides as well as a cutout that matches the one on the base for ease of feeding wire from the base up to the top.

Component distribution can be fixed by redesigning the base of the robot to allow for ease of access and wire management. The top where the Arduino is located would have to be sorted out better. The batteries will be located on the sides. The top needs to be laid out better so that the components all fit together better on top.

Ball collection will remain the same design as before, having the paddle scoop the balls into the robot and held there until the robot is aligned with the goal allowing the back door to open releasing the balls, we will just need to get it functioning correctly. The main problem came from not having proper coding for the back door and sensors to work together but that problem will be resolved. If needed, we can reverse the paddle in front to “shoot” the balls upon contact but that will also result in an uncontrolled goal which will result in earning less points.



Old

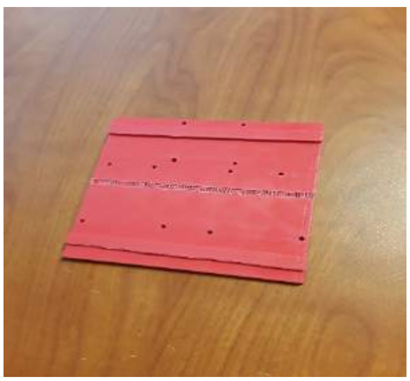


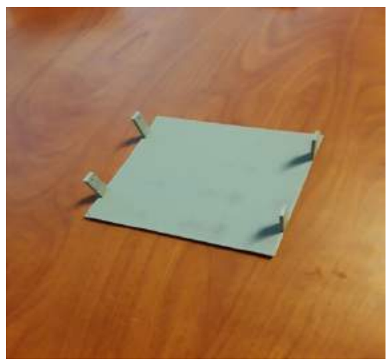
New

Prototype:

Below are some of the prototypes that we came up with as the design of the robot had to be modified. The changes occur from left to right, left being the first designs.

Top Mount:
This piece serves as a plate to which we attach our Arduino and shield to. Initially our first overall design was meant to fit its pieces into each other like puzzle pieces, but as we soon realized, the clearance was of type inference, for we were unable to fully assemble without breaking the pieces. We then changed our models to make the top mount have an L-bracket, so we could screw the piece unto the wall piece



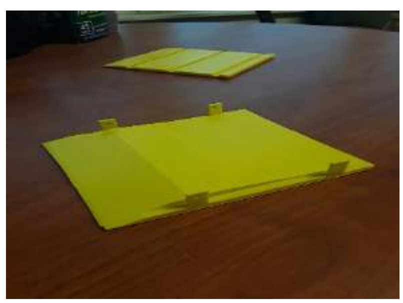
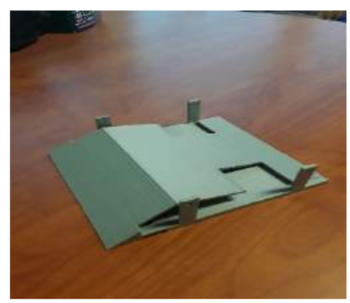


Wall piece:

This piece was originally designed to protect the wheel, and the belts running down from the top since the design was to have the motors top mounted and have belts run down to the motors. Just as the top mount, we originally had planned to have them fit like puzzle pieces, and we soon got rid of that design, and went for a more simplistic design. Finally, the latest design has the implementations of the latter, and in addition we added a slot to run cables from the base of the robot to the Arduino up top without them getting caught into the week or other moving components. Also, in the design we added a bracket in the front for our ball casters. In addition, the implementations made in the base allowed us to add a storage unit right above the wheel, that will allow us to store the robot's batteries.

Base:
The design for this has become quite important for our overall design because even though it was originally designed to have a small slope and serve to house the reflectance sensor array, we too realized that we needed much more space. Eventually, when we ran into problems with the belt (top mounted motors) we decided to bring the motors down to the base to make them more efficient, and to have less moving parts. Once we moved the motors down that opened space, space that were previously taken by the belt. This space was now used to store the batteries right above the wheels, which provide even weight balance for traction.

Test: Re-test the new implementations and re-check your specification table

All of our changes have yielded positive results, from one iteration to another. Every time we see aspects that can be improved, we make note of it, and we edit our part accordingly.

Although we originally designed our robot to meet all criteria from the start. There are plenty of improvements that can be made. Since all specification were kept in mind when designing the robot, there weren’t many changes that had to be made, except for our own criteria, which we have been able to slowly achieve.

Testing Specifications

Requirements	Our Robot
Cannot exceed 10lbs	0.425 Kg (1.07 lbs): Weighed with electronic scale
Maximum initial static size of 7in x 7in x 7ft	Size: [6.5 in x 6 in x 6 in] (Dimensioned with CAD & 3D printed)
Robot must withstand 3-foot drop	Our robot cannot withstand this drop
Autonomous while in the field of play	Robot has been programmed by an Arduino kit to be completely autonomous
Cost of robot may not exceed $80	Our total cost on the robot is $28.98
Robot Numbering Rule Section number, team number 02-5	We currently do not have a robot with our team number and section number.
The robot must be powered by electrical storage batteries	Our Robot is powered by 2 6-slot battery packs. Powered by 9V AA batteries
Must move at 1mph	Our robot can go 5mph
Must stay powered for 3 minutes	Our robot can stay powered for 3+ minutes
Weight less than 10 lbs	Our robot currently weighs 5.75lbs
1” clearance	Our robot exceeds the 1” clearance

DESIGN ITERATION 2

Re-Evaluation

Are there any specifications that you have not met?   Document these.

- All of the specifications have been followed and applied to the robot rigorously. We were and are very cautious when making any changes to the robot.

How many specifications does your design meet so far?

- All of specifications are met due to the simple fact if they are not met, we will lose points or even become disqualified. With that thought in the back of our heads we stuck to the specifications rigorously because we did not want to lose points which would result in failure of the course. Every thought or change was always compared to that of the specifications to make sure we were not overstepping and causing self-harm for later.

Are there any that you may need to update?

- We just remade the robot for the newest version and within the making of that version the specifications were revisited and checked. There were no specifications found that needed an update, so we did not need make any new updates for this version of the robot.

Team discussion

What did other robots do better than your robot in the Midterm Competition?

- When looking at the other robots during the midterm competition we noticed a few things. The first being that many other robots were scoring controlled goals which caused for a much greater score in comparison to that of our team. Our issue was we couldn't score a controlled goal, every time we scored it was from an uncontrolled goal which is worth less causing for our point deficit. Second the other robots could detect either the goal or other robots better than what we could do. We were running based on random motion and hoping we scored randomly. Another main problem was the goal itself the robot drove through it and nearly fell and hit the floor, but we caught it before that could happen.

What are your goals (such as max score) for your robot performance in the Final Competition?

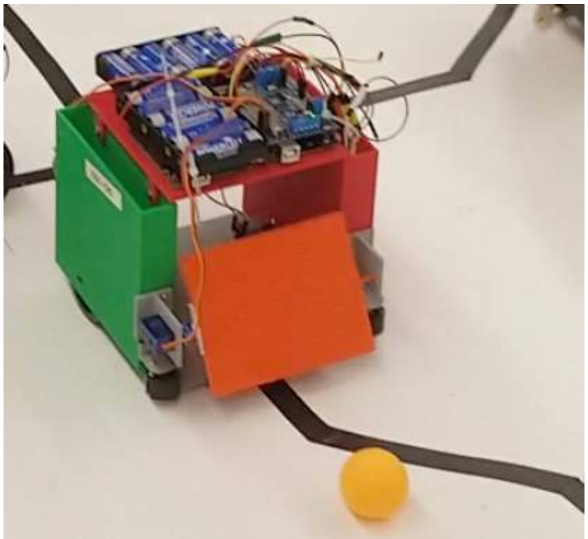
- We have a few goals, one being to score a controlled goal, we could not score a single controlled goal during the midterm competition all of our goals were a result from uncontrolled goals. Another goal is to place better than what we did prior, which was 5th out of 6 total teams, this caused a drop in our overall grade hurting us. We want to have at least 3 controlled rounds of competition due to all of our past competition rounds have been random and based on pure luck. Our score for the midterm competition came out to a total of 3, we hope to improve this by scoring at least 5 points per each round this will mainly come from scoring controlled goals.

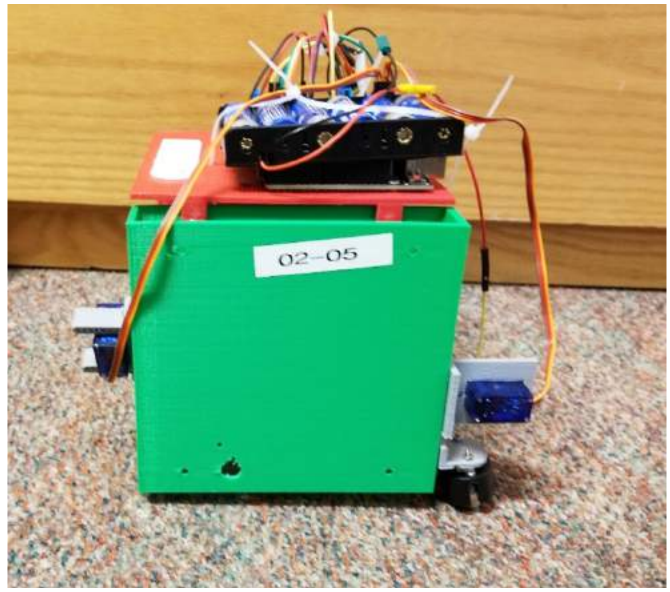
Is your current conceptual design good enough to meet your expectations in the Final Competition? Justify your answer. If your current design cannot meet your expectations for the final competition, do you need to make only small changes or try a different conceptual design?

- We feel as though the current state of the robot will work and allow for us to be competitive and place well in the competition. The reason for this is that we have met and change all of the problems we listed out in the prior assignment. The current state of the robot is much more refined and stands a better chance of scoring. Through multiple iterations of the robot we were able to make improvements making for a better robot and solving problems we had with prior robots.

Midterm Progress

Picture of Assembled Robot (Midterm Competition)





Results from Midterm Competition

Midterm qualification score: 25/100

Midterm Competition Rank: 5th

Midterm Competition score: 88/100

The Midterm Takeaway:

Our group agreed that after the qualifier there was a lot of problem both amongst the team and the robot which was affecting our overall success in the class. To begin our robot struggled to complete the triangle/straight line (Project Assignment 4) which we received an 50/100. This exposed that our robot had a problem with going in a simple straight-line task. After some more design iterations and improvements we entered the Midterm qualifier with the impression that these previous problems had been dealt with. Due to lack of resolution we needed many attempts to qualify for the midterm. The reason for this was the robot did not repeat the same function, so ideal conditions could not be created to optimize the robot performance. Therefore, it took multiple tries to actual score a goal. Nevertheless, we manage to set up conditions which allowed us to qualify and advance to the Midterm competition.

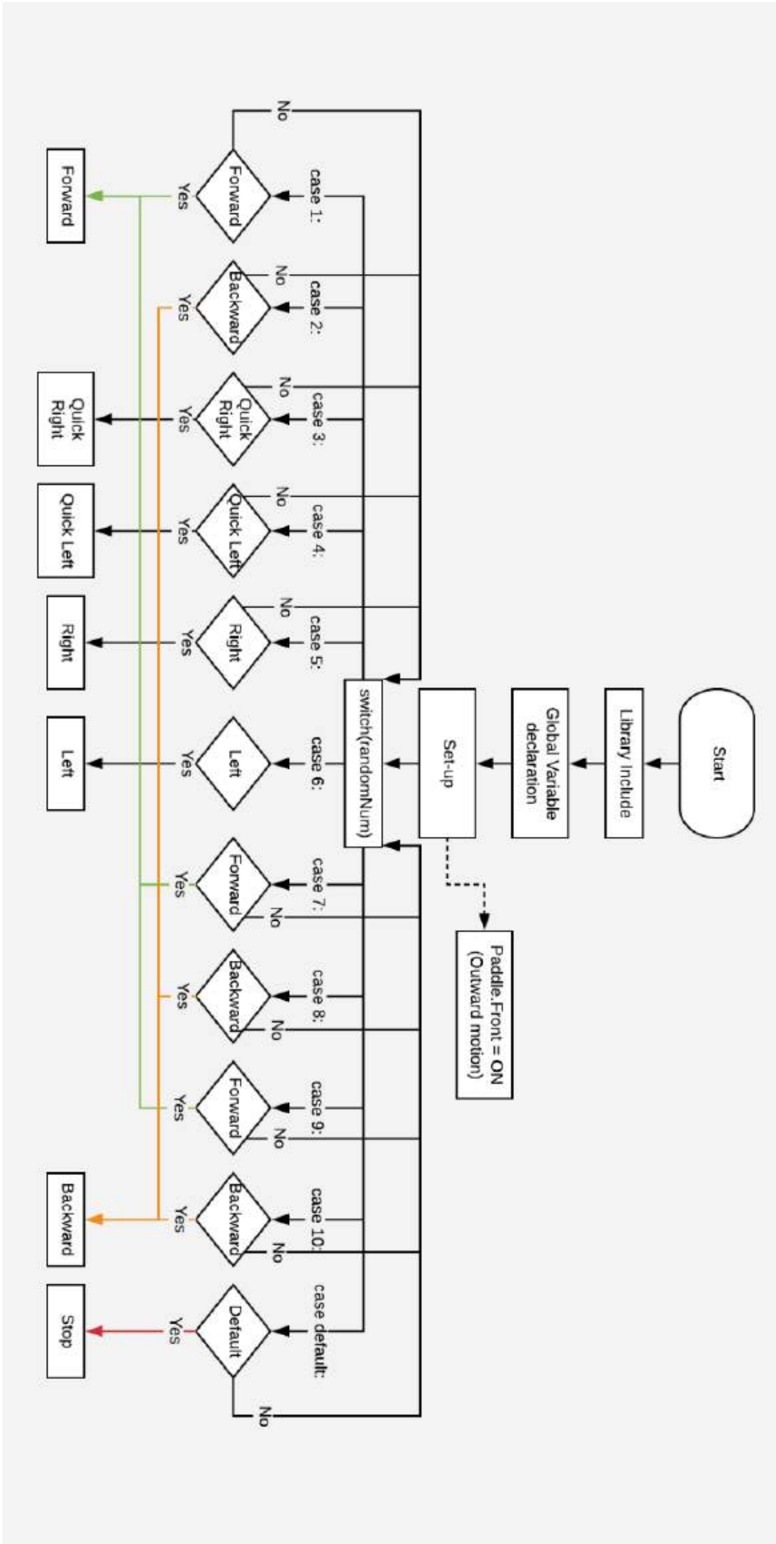
During the midterm competition we had made more adjustments and were able to compete amongst the other robots and even score an unintentional goal, while this was going on, one member would record all the matches and we extrapolated that ant the end of every round the balls would end up in the same general area. We deduced the either the table was not even or the other robots heavy activity in the middle of the field cause this to happen continuously.

To conclude the journey up to the midterm, exposed that we needed a robot that prioritize simplicity, our robot was trying to do to much and it couldn't even handle simple tasks such as going in a straight line and sensing a wall consistently. Which bring up the next point of repeatability. Our robot because of its random movement means that it would so random that it was unpredictable in terms of satisfying our critical functions.

Our main objective is durability to survive the bumps and hits from moving(robots) and static (obstacles) object. Then simplicity to execute small tasks such as sense the wall, obstacles and most importantly go in a straight line. Next the code must be repeatable in the sense that the robot will have behavioral pattern when it encounter the same situations.

Finally when the above has been incorporated then have a function that searches the perimeter of the course looking for the inevitable ball that trail off to the corners which we could collect and score using method to be discovered.

Software Flow Diagram (Midterm Competition)



Finalized Design Summary

Results

Requirements	Our Robot
Cannot exceed 10lbs	0.425 Kg (1.07 lbs): Weighed with electronic scale
Maximum initial static size of 7in x 7in x 7ft	Size: [6.5 in x 6 in x 6 in] (Dimensioned with CAD & 3D printed)
Robot must withstand 3-foot drop	Our robot cannot withstand this drop
Autonomous while in the field of play	Robot has been programmed by an Arduino kit to be completely autonomous
Cost of robot may not exceed $80	Our total cost on the robot is $28.98
Robot Numbering Rule Section number, team number 02-5	We currently do not have a robot with our team number and section number.
The robot must be powered by electrical storage batteries	Our Robot is powered by 2 6-slot battery packs. Powered by 9V AA batteries
Must move at 1mph	Our robot can go 5mph
Must stay powered for 3 minutes	Our robot can stay powered for 3+ minutes
Weight less than 10 lbs	Our robot currently weighs 5.75lbs
1” clearance	Our robot exceeds the 1” clearance

So just as the just like the midterm our robot has met all the specification for the Robo Sockey tournament. However, where we hope to improve is using an alternative code. This code will be scanning the perimeter instead of dueling the other robots in the middle will be following the wall using our 2 ultrasonic sensors which have been placed perpendicularly to each other to sense the wall and an Digital IR sensor which will be used to detect the balls.

Discussion of Strength and Weaknesses

Strength:

The current strengths of our robot is its ability to sense its surrounding and recover, a random movement code has been included if the robot comes to a complete stop or it detects the constant bumping into an object. This will mean we are less likely to get stuck in a corner with no way out. We want to remain in the flow of the competition as long as possible and attribute allows this.

Another advantage of our robot is its build means our robot can take a lot of heavy hits from other robots and continue going without too much interruption/ deviation from the original program.

Weaknesses:

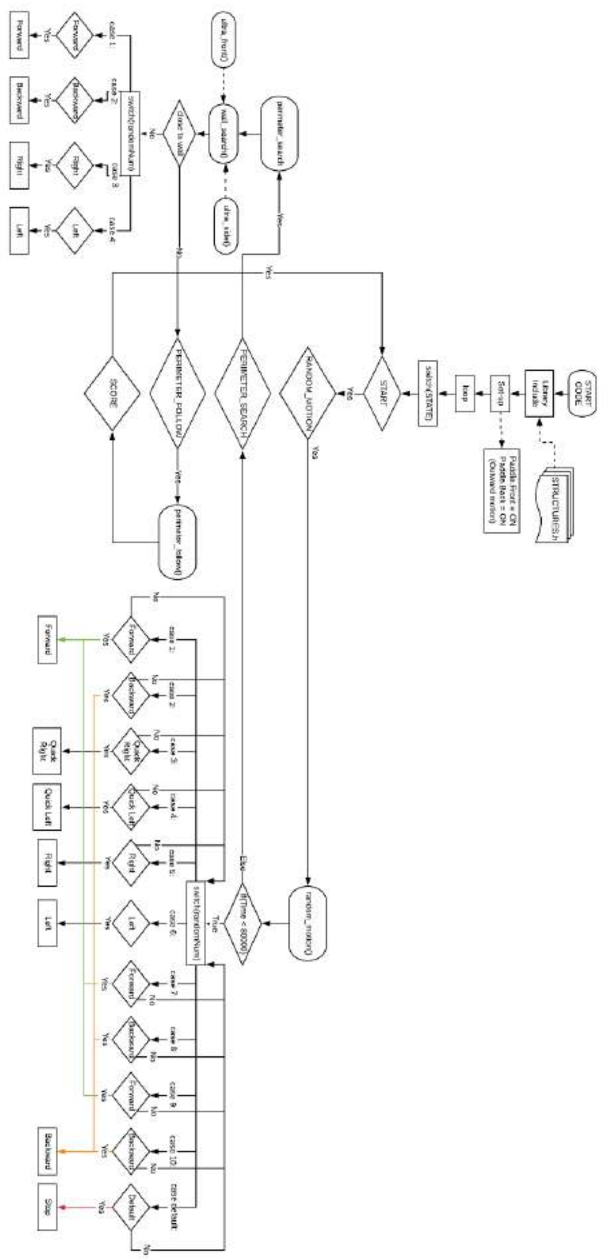
The weakness of the robot which has been a glaring weakness from the midterm is the ability to repeat the same behavior in similar situations. What is meant by this is our robot may make it out of a corner, but it does it is so many variations that the team don't know what to expect on majority of the runs. Which means that there isn't a real way to make effective changes because how do we differentiate the difference between a bug in the robot or the robot just being unpredictable as per usual.

The ultimately means that not many changes can be made because there is no space for growth due to the uncertainty in the robot's behavior.

To conclude this robot has unlimited potential to do what we needed as critical functions but by biting off more than we can chew too early on into the competition before really building a stable foundation on basic commands is what ultimately lead to our downfall.

It should be noted that the robot in many instances did perform well but due to the nature of the robot we can't repeat that behavior.

Appendix A – Final Software Flow Chart/ State Machine Diagram



Appendix B – Final Programming Code

Final Code w/ Structure

```
#include <Wire.h>

#include <Adafruit_MotorShield.h>

#include <Servo.h>

#include "utility/Adafruit_MS_PWMServoDriver.h"

#include "STRUCTURES.h"

Servo Paddle;

Servo Back;

void setup() {

  // put your setup code here, to run once:

  Serial.begin(9600);

  AFMS.begin();

  rightMotor -> setSpeed(170);

  leftMotor -> setSpeed(220);

  pinMode(buttonPin, INPUT);

  pinMode(frontEcho, INPUT);

  pinMode(frontTrig, OUTPUT);

  pinMode(sideEcho, INPUT);

  pinMode(sideTrig, OUTPUT);

//  pinMode(backEcho, INPUT);

//  pinMode(backTrig, OUTPUT);

  randomSeed(analogRead(A2));

  randomSeed(analogRead(A0));

  Paddle.attach(10);
```

```
  Paddle.write(45);
  Back.attach(9);
  Back.write(0);
  delay(1500);
  Start();
//  delay(1200);
  STATE = START;
}
void loop() {
  // put your main code here, to run repeatedly:
  switch (STATE){
    case START:
      Start();
      delay(800);
      STATE = RANDOM_MOTION;
      break;
    case RANDOM_MOTION:
      if(Time < 80000){
        Paddle.write(45);
        random_motion();
      }/*else if((Time > 30000) && (Time < 60000)){
        Paddle.write(120);
        random_motion();
      }*/else{
        STATE = PERIMETER_SEARCH;
      }
```

```
    break;
  case PERIMETER_SEARCH:
    perimeter_search();
    if((distanceSide < closeLimit)/* && (distanceBack < closeLimit)*/){
      STATE = PERIMETER_FOLLOW;
    }
    break;
  case PERIMETER_FOLLOW:
    perimeter_follow();
    ultraFront();
    ultraSide();
    if((distanceSide - lastsdist) > 6){
       STATE = SCORE;
    }
    break;
  case SCORE:
    break;
};
ultraFront();
ultraSide();

if((Time > 170000) && (Time < 180000)){
  Paddle.write(45);
  Back.write(90);
  quickRight();
  delay(1500);
```

```
  }
  if(Time == 180000){
    exit(0);
  }
}
void score(){
  Forward();
  delay(100);
  quickLeft();
  delay(300);
  Backward();
  delay(100);
  Back.write(60);
  delay(1000);
  Back.write(40);
  STATE = START;
}
```

Structures.h

```
Adafruit_MotorShield AFMS = Adafruit_MotorShield();

Adafruit_DCMotor *rightMotor = AFMS.getMotor(2);
Adafruit_DCMotor *leftMotor = AFMS.getMotor(1);


int ledPin = 13;
int buttonPin = 12;
int buttonState = 0;
int Time;
int lastfdist = 0;
int lastsdist = 0;
int closeLimit = 12;
int farLimit = 17;
int frontEcho = 2;
int frontTrig = 3;
int sideEcho = 4;
int sideTrig = 5;
//int backTrig = 6;
//int backEcho = 7;
long durationFront;
long distanceFront;
long durationSide;
long distanceSide;
//long durationBack;
```

```
//long distanceBack;

enum states{
  START,
  RANDOM_MOTION,
  PERIMETER_SEARCH,
  PERIMETER_FOLLOW,
  LINE_FINDER,
  LINE_FOLLOWING,
  GOAL_DETECT,
  SCORE
} STATE;

void ultraFront(){
  digitalWrite(frontTrig, HIGH);
  delayMicroseconds(1000);
  digitalWrite(frontTrig, LOW);
  durationFront = pulseIn(frontEcho, HIGH, 10000);
  distanceFront = (durationFront / 2) / 29.1;
  Serial.print (distanceFront);
  Serial.println("cm - Front");
  delay(500);
}

void ultraSide(){
  digitalWrite(sideTrig, HIGH);
```

```
  delayMicroseconds(1000);
  digitalWrite(sideTrig, LOW);
  durationSide = pulseIn(sideEcho, HIGH, 10000);
  distanceSide = (durationSide / 2) / 29.1;
  Serial.print (distanceSide);
  Serial.println("cm - Side");
  delay(500);
}

//void ultraBack(){
//  digitalWrite(backTrig, HIGH);
//  delayMicroseconds(1000);
//  digitalWrite(backTrig, LOW);
//  durationBack = pulseIn(backEcho, HIGH, 10000);
//  distanceBack = (durationBack / 2) / 29.1;
//  Serial.print (distanceBack);
//  Serial.println("cm");
//  delay(500);
//}

void Forward(){
  rightMotor -> run(FORWARD);
  leftMotor -> run(FORWARD);
}

void Backward(){
```

```
  rightMotor -> run(BACKWARD);
  leftMotor -> run(BACKWARD);
}

void quickRight(){
  rightMotor -> run(BACKWARD);
  leftMotor -> run(FORWARD);
}

void quickLeft(){
  rightMotor -> run(FORWARD);
  leftMotor -> run(BACKWARD);
}

void Right(){
  rightMotor -> run(RELEASE);
  leftMotor -> run(FORWARD);
}

void Left(){
  rightMotor -> run(FORWARD);
  leftMotor -> run(RELEASE);
}

void Stop(){
  rightMotor -> run(RELEASE);
```

```
  leftMotor -> run(RELEASE);
}

void Start(){
  rightMotor -> setSpeed(220);
  leftMotor -> setSpeed(170);
  rightMotor -> run(FORWARD);
  leftMotor -> run(FORWARD);
}

void random_motion(){
  long randomNum = random(1, 9);
  Serial.println(randomNum);
  long randomDelay = random(250, 3001);
  Serial.println(randomDelay);

  switch(randomNum){
    case 1:
      Forward();
      delay(randomDelay);
      break;
    case 2:
      Backward();
      delay(randomDelay);
      break;
    case 3:
```

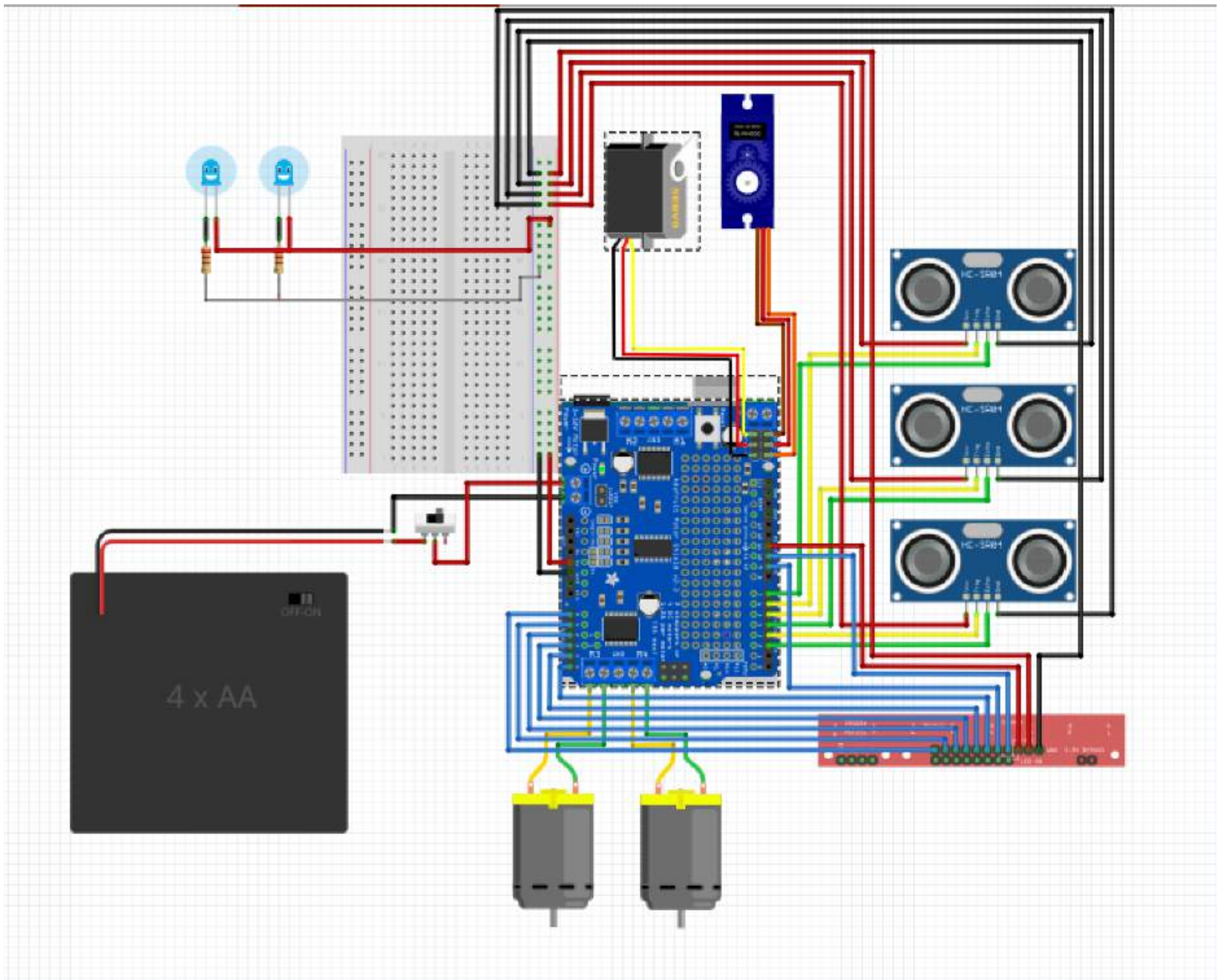
```
    quickRight();
    delay(randomDelay);
  case 4:
    quickLeft();
    delay(randomDelay);
    break;
  case 5:
    Right();
    delay(randomDelay);
    break;
  case 6:
    Left();
    delay(randomDelay);
    break;
  case 7:
    Forward();
    delay(randomDelay);
    break;
  case 8:
    Forward();
    delay(randomDelay);
    break;
  default:
    Forward();
    delay(randomDelay);
    break;
```

```
  }
}

void wall_search(){
  long randomNum = random(1, 5);
  Serial.println(randomNum);
  long randomDelay = random(250, 1001);
  Serial.println(randomDelay);

  switch(randomNum){
    case 1:
      Forward();
      delay(randomDelay);
      break;
    case 2:
      Backward();
      delay(randomDelay);
      break;
    case 3:
      Right();
      delay(randomDelay);
      break;
    case 4:
      Left();
      delay(randomDelay);
      break;
```

```
   default:
     break;
  }

  delay(200);
}

void perimeter_search(){
  wall_search();
  ultraFront();
  ultraSide();
  if((distanceSide > closeLimit)/* && (distanceBack > closeLimit)*/){
    perimeter_search();
  }
}

void perimeter_follow(){
  if(distanceFront > closeLimit && distanceSide < closeLimit){
    Forward();
    delay(500);
  }
  if (distanceFront < closeLimit){
    Backward();
    delay(500);
    Forward();
    quickLeft();
```

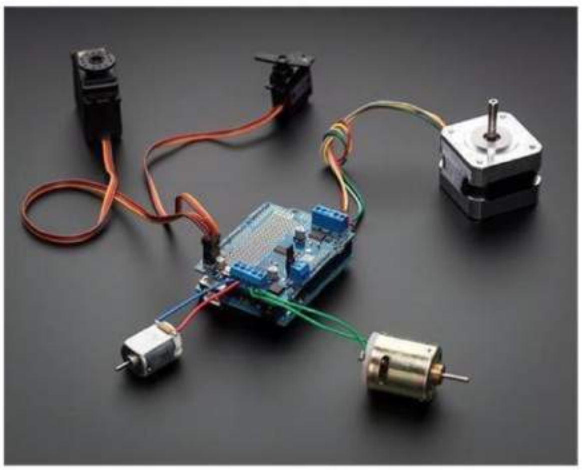
```
    delay(500);
  }
  if (distanceSide < closeLimit){
    quickLeft();
    delay(500);
    Forward();
    delay(500);
  }
  if (distanceSide > closeLimit){
    quickRight();
    delay(500);
    Forward();
    delay(500);
  }
  lastfdist = distanceFront;
  lastsdist = distanceSide;
}
```

Appendix C – Final Electrical Wiring Diagram of Robot



Appendix D – Final Budget for Machine

*Everything that was included in the EGR 107 Arduino kit will not be included in this final report.



Adafruit Motor/Stepper/Servo Shield for Arduino v2.3 Kit

by Adafruit

58 customer reviews | 36 answered questions

Amazon's Choice for "arduino motor shield"

Price: $18.90 prime

Your cost could be $8.90. Eligible customers get a $10 bonus when reloading $100.

Note: Available at a lower price from other sellers, potentially without free Prime shipping.

Arrives before Christmas.

Service: **Get professional installation** Details

Without expert installation | Include installation +$96.54 per unit

See more

Greartisan DC 6V 300RPM N20 High Torque Speed Reduction Motor with Metal Gearbox Motor for DIY RC Toys

by Greartisan

Be the first to review this item

Price: $11.99 prime

Coupon Save an extra 5% when you apply this coupon. Details

Your cost could be $1.99. Eligible customers get a $10 bonus when reloading $100.

Arrives before Christmas.

Size: **6 Volt**

3 Volt | 6 Volt | 12 Volt

Color: **6V-300RPM-50i**

We Brought x2 of this item.

You last purchased this item on October 6, 2018.
View this order

Skinny Wheel for TT DC Gearbox Motors

PRODUCT ID: 3757

$2.50
IN STOCK

1 ADD TO CART

QTY	DISCOUNT
1-9	$2.50
10-99	$2.25
100+	$2.00

ADD TO WISHLIST

DESCRIPTION
TECHNICAL DETAILS

We Brought x2 of this item.

You last purchased this item on October 6, 2018.
View this order

Ball Caster - 3/4" Metal Ball

PRODUCT ID: 1200

$4.95
IN STOCK

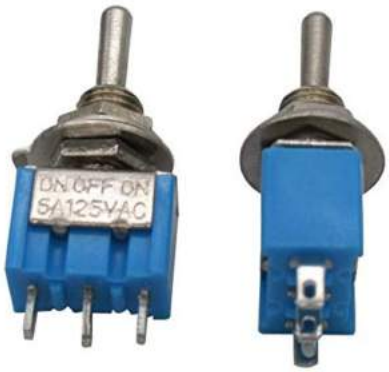
1 ADD TO CART

QTY	DISCOUNT
1-9	$4.95
10-99	$4.46
100+	$3.96

ADD TO WISHLIST

DESCRIPTION
TECHNICAL DETAILS

We Brought x2 of this item.



TWTADE / 10pcs MTS-103 On/Off/On Mini Miniature Toggle Switch Car Dash Dashboard 3 position SPDT 3Pin + 10pcs Waterproof Cap

by TWTADE

3 customer reviews

Price: $10.99 prime

Arrives before Christmas.

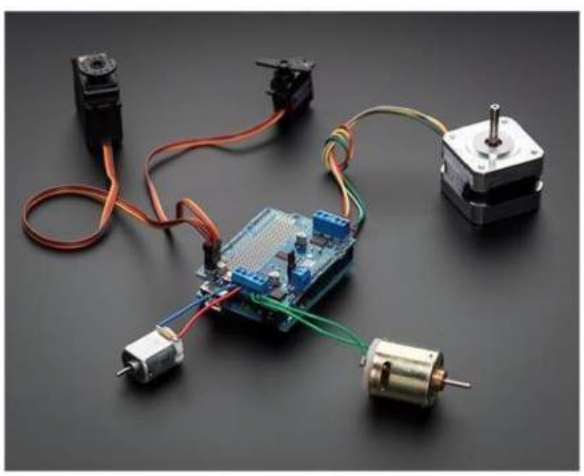
Size: **3 Pin ON/OFF/ON**

3 Pin NO/NO $10.99 prime	3 Pin ON/OFF $10.99 prime	**3 Pin ON/OFF/ON** $10.99 prime
6 Pin NO/NO $10.99 prime	6 Pin ON/OFF/ON $10.99 prime	

- Product name: mini Toggle switch; country of manufacture: China; material: plastic, metal
- Model: mts-103; main color: Blue, Silver Tone; rated Voltage: AC 125V
- Rated current: 6A; contact Type: SPDT; position number: 3 (On-OFF-ON) ;Screw Diameter : 6mm
- Pin number: 3; dimension: 13 x 8 x 33mm/0.5" x 0.3" x 1.3"(L*w*h); action Type: latching
- Net weight: 53G; package content: 10 x mini Toggle switches + 10 x Waterproof Cap

› See more product details

New (1) from $10.99 prime



Adafruit Motor/Stepper/Servo Shield for Arduino v2.3 Kit

by Adafruit

58 customer reviews | 36 answered questions

Amazon's Choice for "arduino motor shield"

Price: $18.90 prime

Your cost could be $8.90. Eligible customers get a $10 bonus when reloading $100.

Note: Available at a lower price from other sellers, potentially without free Prime shipping.

Arrives before Christmas.

Service: **Get professional installation** Details

Without expert installation	Include installation **+$96.54** per unit

See more

BEHIND EVERY PROJECT IS A

True Value

Allendale True Value
5425 Lake Michigan Drive
Allendale, MI 49401

Phone: (616)895-5400
Fax: (616)892-4777
Website: www.allendaletruevalue.com
Email: info@allendaletruevalue.com

INVOICE

Invoice No. 1994197 Date 11/03/18 16:16:21 Cashier Bella

TRANSACT # PC2

Payment Terms
Cash

Sold To: Customer # GAR033
Renzo Garza
46503 arboretum circle
Plymouth, MI 48170
(616)227-6920

Ship To:
Renzo Garza
46503 arboretum circle
Plymouth, MI 48170

Ln. #	Item # / Qty	Description / List	Price	Extended
001	HILLMAN	HILLMAN FASTENERS (Nuts, Bolts, Hardware)		
	10EA1	0.30	$0.30	3.00
002	HILLMAN	HILLMAN FASTENERS (Nuts, Bolts, Hardware)		
	10EA1	0.27	$0.27	2.70
003	HILLMAN	HILLMAN FASTENERS (Nuts, Bolts, Hardware)		
	8EA1	0.27	$0.27	2.16
004	GVSU	GVSU 10% OFF STUDENT DISCOUNT (Reg $)		
	-1EA1			

Payment Method: Amount

DEBI
Card: 2522 8.33
Name: MOTTA/RENZO GARZA
Auth: 461780

Non Taxable SubTot 0.00
Taxable SubTotal 7.86
Sales Tax 0.47
Total 8.33
Total Due

BEHIND EVERY PROJECT IS A

True Value

Allendale True Value
5425 Lake Michigan Drive
Allendale, MI 49401

Phone: (616)895-5400
Fax: (616)892-4777
Website: www.allendaletruevalue.com
Email: info@allendaletruevalue.com

INVOICE

Invoice No 1987467 Date 10/16/18 12:43:24 Cashier Aubree

TRANSACT # PC1

Payment Terms
Cash

Sold To: Customer # CASH
MI

Ship To:
MI

Ln. #	Item # / Qty	Description / List	Price	Extended
001	798004	3OZ 032Elec Lead Solder		
	1EA1	10.99	$9.89	9.89
002	175233	25W Soldering Iron Kit		
	1EA1	19.99	$17.99	17.99
003	117181	#1 LGT Hobby Knife		
	1EA1	4.29	$3.86	3.86
004	HILLMAN	HILLMAN FASTENERS (Nuts, Bolts, Hardware)		
	11EA1	0.35	$0.35	3.85
005	HILLMAN	HILLMAN FASTENERS (Nuts, Bolts, Hardware)		
	11EA1	0.30	$0.30	3.30
006	HILLMAN	HILLMAN FASTENERS (Nuts, Bolts, Hardware)		
	11EA1	0.37	$0.37	4.07
007	GVSU	GVSU 10% OFF STUDENT DISCOUNT (Reg $)		
	-1EA1			

Payment Method: Amount

DEBI
Card: 2522 45.54
Name: MOTTA/RENZO GARZA
Auth: 731860

Non Taxable SubTot 0.00
Taxable SubTotal 42.96
Sales Tax 2.58
Total 45.54
Total Due 0.00

meijer

315 Wilson Ave. NW
Grand Rapids, MI 49534 #216

The Meijer Team appreciates your business
10/10/18
Your checkout was provided by Fastlane104

MEIJER SAVINGS
SPECIALS .30
SAVINGS TOTAL .30

SALE
GENERAL MERCHANDISE
7079800158 DAP RAPID FUSE 3.99 CT
7287910040 PRYM DRSSMKR P 3.99 CT
7630849686 3M SANDBLASTER 3.99 CT
*3749583740 DORMAN
was 1.99 now 1.69 CT

TOTAL
MI 6% Sales Tax .82
TOTAL TAX .82
TOTAL 14.48

PAYMENTS
Primary Account - Debit
ATM/DEBIT CARD TENDER 14.48
XXXXXXXXXXXX2522 (C)
US DEBIT
AID A0000000980840
TC 8F286FAF23586F24
PIN VERIFIED ONLINE

NUMBER OF ITEMS 4

For information on Meijer return policy
visit meijer.com

A0216OMATI18RFFS

Tx:454 Op:555 Tm:104 St:216 21:15:07

We value your feedback.
Share your experience by emailing:
Customer.Feedback@meijer.com

meijer

315 Wilson Ave. NW
Grand Rapids, MI 49534 - #216

The Meijer Team appreciates your business
10/10/18
Your checkout was provided by Fastlane104

MEIJER SAVINGS
SPECIALS .30
SAVINGS TOTAL .30

SALE
GENERAL MERCHANDISE
7079800158 DAP RAPID FUSE 3.99 CT
7287910040 PRYM DRSSMKR P 3.99 CT
7630849686 3M SANDBLASTER 3.99 CT
*3749583740 DORMAN
was 1.99 now 1.69 CT

TOTAL
MI 6% Sales Tax .82
TOTAL TAX .82
TOTAL 14.48

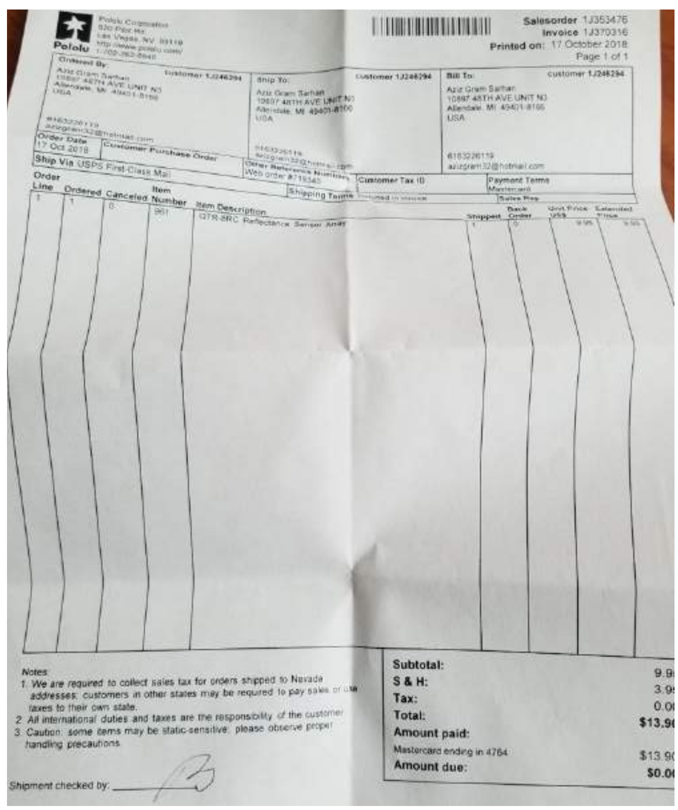
PAYMENTS
Primary Account - Debit
ATM/DEBIT CARD TENDER 14.48
XXXXXXXXXXXX2522 (C)
US DEBIT
AID A0000000980840
TC 8F286FAF23586F24
PIN VERIFIED ONLINE

NUMBER OF ITEMS 4

For information on Meijer return policy
visit meijer.com

A0216OMATI18RFFS

Tx:454 Op:555 Tm:104 St:216 21:15:07



Pololu Corporation
Las Vegas, NV 89119

Salesorder 1J353476
Invoice 1J370316
Printed on: 17 October 2018
Page 1 of 1

Ordered By: Customer 1J246294
Aziz Gram Sarhan
10897 48TH AVE UNIT N3
Allendale, MI 49401-8166
USA

Ship To: Customer 1J246294
Aziz Gram Sarhan
10897 48TH AVE UNIT N3
Allendale, MI 49401-8166
USA

Bill To: Customer 1J246294
Aziz Gram Sarhan
10897 48TH AVE UNIT N3
Allendale, MI 49401-8166
USA

Order Date: 17 Oct 2018
Ship Via: USPS First-Class Mail
Payment Terms: Mastercard

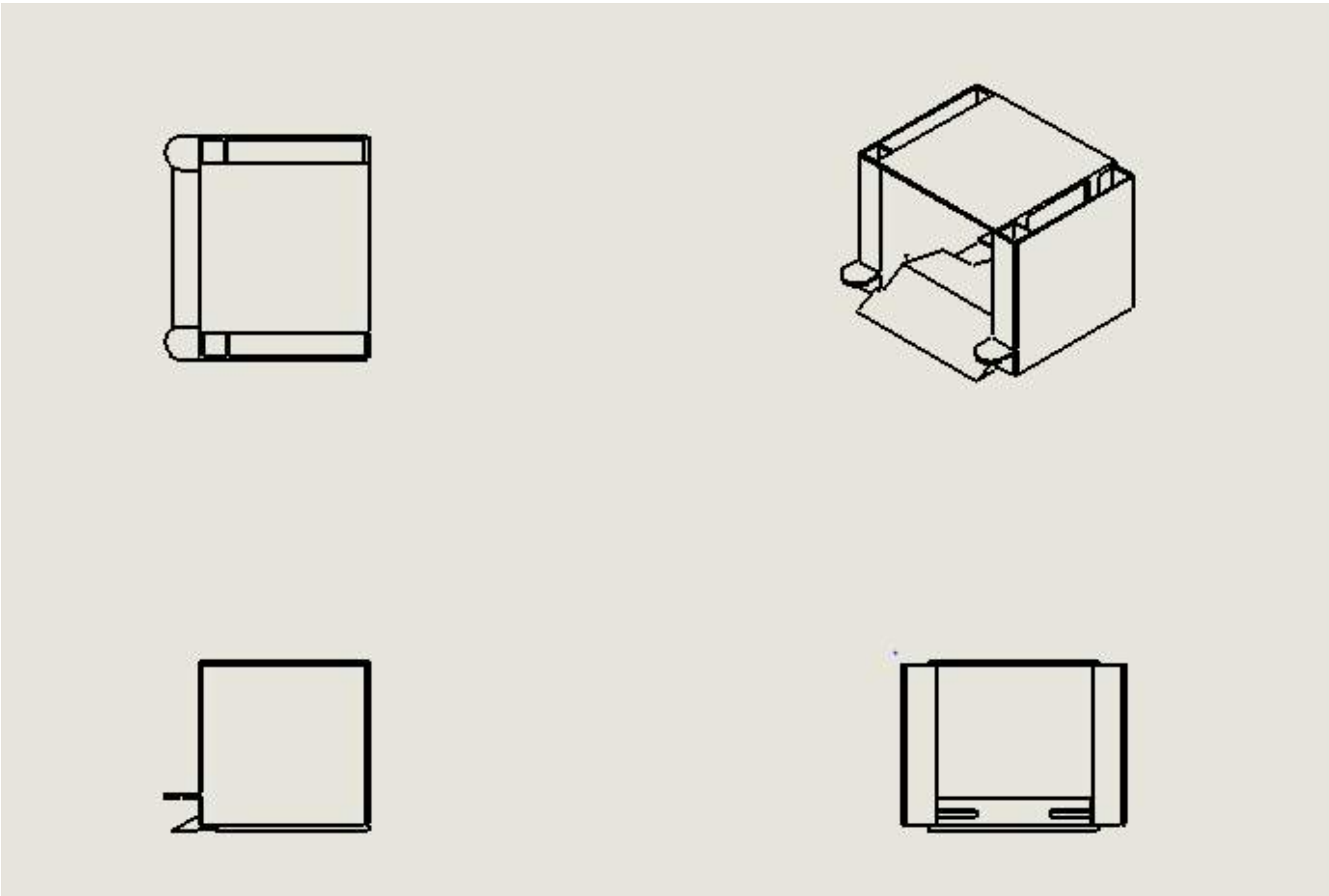
Order Line	Ordered	Canceled	Item Number	Item Description	Shipped	Back Order	Unit Price US$	Extended Price
1	1	0	961	QTR-8RC Reflectance Sensor Array	1	0	9.95	9.95

Notes:
1. We are required to collect sales tax for orders shipped to Nevada addresses; customers in other states may be required to pay sales or use taxes to their own state.
2. All international duties and taxes are the responsibility of the customer.
3. Caution: some items may be static-sensitive; please observe proper handling precautions.

Shipment checked by: ______

Subtotal: 9.9
S & H: 3.9
Tax: 0.0
Total: $13.9
Amount paid:
Mastercard ending in 4764 $13.9
Amount due: $0.0

Appendix E – Working Drawing of Finalized Robot



Appendix F – G Code (for all CNC Machined Parts)

G CODE

```
G20  G90 G17  G94  G40  G49  G80 G98
G54
M03
(CR)
(G00 X1.50 Y5.25 Z0.5                (MOVING TO THE TOP OF THE 'C')
(G01 X1.50 Y5.25 Z-0.015 F10         (CUTTING INTO THE MATERIAL)
(G03 X1.50 Y4.25 I1.50 J4.75 F30      (MAKING A COUNTER CLOCKWISE HALF CIRCLE FOR TEH 'C')
(G01 Z0.5 F10                        (EXITING THE MATERIAL)
(G00 X1.75 Y4.25                     (MOVING TO TEH START OF THE 'R')
(G01 X1.75 Y4.25 Z-0.015 F10         (ENTERING THE MATERIAL)
(G01 X1.75 Y5.25 F30                 (CREATING A STRAIGHT LINE)
(G02 X1.75 Y4.75 I1.75 J5.0 F30      (CLOCKWISE HALF CIRCLE)
(G01 X2.0 Y4.25                                (DIAGNIOAL LINE BEING CUT)
(G01 Z0.5                            (EXITING MATERIAL)

(RM)
(G00 X0.75 Y1.0 Z0.5
(G01 X0.75 Y1.0 Z-0.015 F10
(G01 X0.75 Y2.0 F30
(G02 X0.75 Y1.50 I0.75 J1.75
(G01 X1.25 Y1.0
(G01 Z0.5
(G00 X1.50 Y1.0
```

```
(G01 X1.50 Y1.0 Z-0.015 F10

(G01 X1.50 Y2.0 F30

(G01 X1.75 Y1.0

(G01 X2.0 Y2.0

(G01 X2.0 Y1.0

(G01 Z0.5


(AS)

(G00 X4.75 Y4.25                (GOING TO THE START OF THE 'A')

(G01 X4.75 Y4.25 Z-0.015 F10    (ENTERING THE MATERIAL AT THE START)

(G01 X4.75 Y4.75 F30            (STRAIGHT LINE UPWARDS OF THE 'A')

(G02 X5.50 Y4.75 I5.125 J4.75   (CLOCKWISE ROTATION OF THE TOP OF 'A')

(G01 X4.75 Y4.75                        (THE CROSS IS BEING MADE)

(G01 X4.75 Y4.75 Z0.5           (EXITING MATERIAL)

(G00 X5.50 Y4.75                        (MOVING BACK TO THE SIDE OF 'A')

(G01 X5.50 Y4.75 Z-0.015                (ENTERING THE MATERIAL)

(G01 X5.50 Y4.25                        (OTHERSIDE OF 'A' IS CUT)

(G01 Z0.5                       (EXITING THE MATERIAL)

(G00 X6.0 Y4.25                 (MOVING OVER TO THE START OF 'S')

(G01 X6.0 Y4.25 Z-0.015 F10     (ENTERING THE MATERIAL)

(G03 X6.0 Y4.75 I6.0 J4.50 F30  (COUNTER CLOCKWISE CIRCLE FOR BOTTOM OF 'S')

(G02 X6.0 Y5.25 I6.0 J5.0 F30   (CLOCKWISE CIRCLE TOP OF 'S')

(G01 X6.25 Y5.25 Z0.5)          (EXITING THE MATERIAL)
```

(CUTTING)

(G00 X0 Y0.1 Z0.5 (GOING TO ORIGIN TO START)

(G01 Z-0.05 F10 (ENTERING THE MATERIAL)

(G01 X0 Y2.25 F30 (STRAIGHTLINE UPWARDS)

(G03 X0 Y3.25 I0 J2.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G03 X0 Y4.25 I0 J3.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G01 X0 Y5.75 (STRAIGHT LINE UPWARD)

(G01 X7.0 Y5.75 (LINE ACROSS MATERIAL)

(G01 X7.0 Y4.25 (LINE DOWNWARD)

(G03 X7.0 Y3.25 I7.0 J3.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G03 X7.0 Y2.25 I7.0 J2.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G01 X7.0 Y0.1 (LINE TO CORNER)

(G01 X5.5 Y0.1 (LINE TO START OF ARC)

(G03 X1.5 Y0.1 I3.5 J-5.0 (COUNTER CLOCKWISE MOTION ARC)

(G01 X0 Y0.1 (LINE BACK TO ORIGIN)

(CUTTING)

(G00 X0 Y0.1 Z0.5 (GOING TO ORIGIN TO START)

(G01 Z-0.10 F10 (ENTERING THE MATERIAL)

(G01 X0 Y2.25 F30 (STRAIGHTLINE UPWARDS)

(G03 X0 Y3.25 I0 J2.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G03 X0 Y4.25 I0 J3.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G01 X0 Y5.75 (STRAIGHT LINE UPWARD)

(G01 X7.0 Y5.75 (LINE ACROSS MATERIAL)

(G01 X7.0 Y4.25 (LINE DOWNWARD)

(G03 X7.0 Y3.25 I7.0 J3.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G03 X7.0 Y2.25 I7.0 J2.75 (COUNTER CLOCKWISE HALF CIRCLE)

(G01 X7.0 Y0.1 (LINE TO CORNER)

```
(G01 X5.5 Y0.1                  (LINE TO START OF ARC)
(G03 X1.5 Y0.1 I3.5 J-5.0       (COUNTER CLOCKWISE MOTION ARC)
(G01 X0 Y0.1                    (LINE BACK TO ORIGIN)

(CUTTING)
G00 X0 Y0.1 Z0.5                        (GOING TO ORIGIN TO START)
G01 Z-0.15 F10                  (ENTERING THE MATERIAL)
G01 X0 Y2.25 F30                (STRAIGHTLINE UPWARDS)
G03 X0 Y3.25 I0 J2.75           (COUNTER CLOCKWISE HALF CIRCLE)
G03 X0 Y4.25 I0 J3.75           (COUNTER CLOCKWISE HALF CIRCLE)
G01 X0 Y5.75                    (STRAIGHT LINE UPWARD)
G01 X7.0 Y5.75                  (LINE ACROSS MATERIAL)
G01 X7.0 Y4.25                  (LINE DOWNWARD)
G03 X7.0 Y3.25 I7.0 J3.75       (COUNTER CLOCKWISE HALF CIRCLE)
G03 X7.0 Y2.25 I7.0 J2.75       (COUNTER CLOCKWISE HALF CIRCLE)
G01 X7.0 Y0.1                   (LINE TO CORNER)
G01 X5.5 Y0.1                   (LINE TO START OF ARC)
G03 X1.5 Y0.1 I3.5 J-5.0        (COUNTER CLOCKWISE MOTION ARC)
G01 X0 Y0.1                     (LINE BACK TO ORIGIN)

(CUTTING)
G00 X0 Y0.1 Z0.5                        (GOING TO ORIGIN TO START)
G01 Z-0.20 F10             (ENTERING THE MATERIAL)
G01 X0 Y2.25 F30                (STRAIGHTLINE UPWARDS)
G03 X0 Y3.25 I0 J2.75           (COUNTER CLOCKWISE HALF CIRCLE)
G03 X0 Y4.25 I0 J3.75           (COUNTER CLOCKWISE HALF CIRCLE)
G01 X0 Y5.75                    (STRAIGHT LINE UPWARD)
G01 X7.0 Y5.75                  (LINE ACROSS MATERIAL)
G01 X7.0 Y4.25                  (LINE DOWNWARD)
```

```
G03 X7.0 Y3.25 I7.0 J3.75          (COUNTER CLOCKWISE HALF CIRCLE)
G03 X7.0 Y2.25 I7.0 J2.75          (COUNTER CLOCKWISE HALF CIRCLE)
G01 X7.0 Y0.1                      (LINE TO CORNER)
G01 X5.5 Y0.1                      (LINE TO START OF ARC)
G03 X1.5 Y0.1 I3.5 J-5.0           (COUNTER CLOCKWISE MOTION ARC)
G01 X0 Y0.1                        (LINE BACK TO ORIGIN)

(CUTTING)
G00 X0 Y0.1 Z0.5                             (GOING TO ORIGIN TO START)
G01 Z-0.25 F10                (ENTERING THE MATERIAL)
G01 X0 Y2.25 F30                   (STRAIGHTLINE UPWARDS)
G03 X0 Y3.25 I0 J2.75              (COUNTER CLOCKWISE HALF CIRCLE)
G03 X0 Y4.25 I0 J3.75              (COUNTER CLOCKWISE HALF CIRCLE)
G01 X0 Y5.75                       (STRAIGHT LINE UPWARD)
G01 X7.0 Y5.75                     (LINE ACROSS MATERIAL)
G01 X7.0 Y4.25                     (LINE DOWNWARD)
G03 X7.0 Y3.25 I7.0 J3.75          (COUNTER CLOCKWISE HALF CIRCLE)
G03 X7.0 Y2.25 I7.0 J2.75          (COUNTER CLOCKWISE HALF CIRCLE)
G01 X7.0 Y0.1                      (LINE TO CORNER)
G01 X5.5 Y0.1                      (LINE TO START OF ARC)
G03 X1.5 Y0.1 I3.5 J-5.0           (COUNTER CLOCKWISE MOTION ARC)
G01 X0 Y0.1                        (LINE BACK TO ORIGIN)
```

Appendix G – Robot Assembly

Appendix H – Technical Data Sheets (Electrical / Mechanical Parts)