The Engineering Design Report

TER3M1 – 02

8 June 2018

A picture containing water, outdoor

Description generated with high confidence

By: Anthony Tedja

**Summary**

**Research** - Page 3

(Brainstorming ideas, identify project and steps, usefulness, comparison to industry, skill required or additional learning)

**Planning** - Page 4 - 6

(List of supplies, schedule, how program will interact with hardware, schematic)

**Building / Prototyping** - Page 7 - 11

(Timeline, photos, design changes and debugging, testing)

**Conclusion –** Page 11

(Summary and conclusion)

**Research**

For this project, I first didn’t know what I wanted to do for is but I knew that I wanted to make something to top last years project, which was a batmobile. At first, I was going to reuse the parts from last year to build a batmobile robot with sensors on it, but when I learned that I couldn’t reuse the parts, I started looking for different ideas of thing to make. I settled on my first choice of the R2D2 robot from the Star Wars series. I knew that it would be a lot of work to make in the small timespan given for this project so I was going to make a R2D2 inspired robot. The goal of this assignment is to create a robot that follows any black line given to it, using programming and the circuitry that communicates with the motors and sensors that adheres to the goal of having a sort of “autonomous robot” through a course. The assignment is to follow a black line, stop at intersections, and spin around at the end of the course identified by two intersection lines and follow the course again back. The flow chart and schematic outline provides guidelines and specifications on how the robot and code is made and its processes. The project theoretical knowledge can also be applied to program based electronics into practical use. This can be used by many commercial robots use the same basic concept to either move or interact with and around objects like using sensors for transportation or minimal invasion surgeries. For this project to be completed, knowledge in basic coding in C language, circuit design and wiring, and wood work skills would be needed, which I have lots of personal experience and understanding in, except in woodworking where I have basic experience so I would need a little more practice.

**Planning**

ATMega 328P-20PU Controller – 28960-A x 1 $5.85 The microcontroller that connects to everything and runs the code.

Gearhead Motor – GM2 x 2 $14.96 The motor that holds the wheel to turn the wheel that makes the robot either forwards or backwards.

Wheel – GMPW x 2 $8.46 The plastic and rubber wheel that attaches to the motor.

9v Battery Snap Connector – Bhold9v x 1 $0.98 The connector which connects the 9v battery to the breadboard.

Breadboard – 21020 x 1 $6.44 The circuit board which connects all the components non-permanently.

L293D Motor Driver – L293D x 1 $4.55 The motor controller which can control up to two motors simultaneously going forwards, backwards or turning.

7805 Voltage Regulator – L7805CV x 1 $0.52 The voltage regulator which allows the constant 5v power supply to be given throughout the circuit.

SPST Pushbutton – SWT10 x 3 $2.34 The push to make buttons that allows the current to flow while being pushed.

16MHz Crystal – 17090 x 1 $1.24 The crystal which allows the microcontroller to keep track of and to count time.

Male Strip Pins - 6 – Mpin6RA x 1 $0.46 The connector for the programmer and the board to communicate.

FTDI Programmer – 50512 x 1 $19.44 The programmer allows code to be programmed on the microcontroller as it converts high level coding language to binary.

IR Reflector sensor with lead – 17092 x2 $2.60 The sensors which can differentiate between shades like white and black that communicates with the chip so it can tell the motors to turn accordingly.

Wood body and arms – I use these pieces to make the robot look more like R2D2 and they serve as anchors to hold down everything and the main components like the breadboard, motors, and sensors. These were gathered from the wood shop class, two 1.75” x 7” x 0.75” blocks, one 4.75” x 1.75” x 0.75” block, and a 3.75” x 0.75” x 1.75” block. The two exterior wood planks are 6” x 3” x 0.12” that covers the front body. Most of these pieces were from the woodshop class while the others were scrap wood lying somewhere in the basement.

Switch – I had this switch from home that I put on the head of the robot connecting to the robot. I used this switch for aesthetic and practical use. This switch is connected to the circuit of the robot so when the battery is inside, it wouldn’t instantly start the program until I hit the switch. Even though the motors and sensors don’t start running till I press the button, the switch still looks cool as it is more for just aesthetics.

Revolving front wheel – I use this piece to match the look of R2D2 with the single wheel under the body. This is used to move the robot more easily since without it, the friction from the body would cause the robot to be very slow or not move if the body was too heavy. This wheel can move 360 degrees and has a top that has 4 screw holes for rigidity and can be found at a RONA store for a little over a dollar.

SCHEDULE

Step 1 – Figure out what to make

Step 2 – Acquire parts and materials needed

Step 3 – Have all circuit components ready (soldering, striping, wiring)

Step 4 – Do the schematics for the circuit

Step 5 – Do the flow chart for the code

Step 6 – Adjust circuit board if needed and work on body of robot

Step 7 – Finish body and adjust if needed

Step 8 – Do the code and put in on the robot

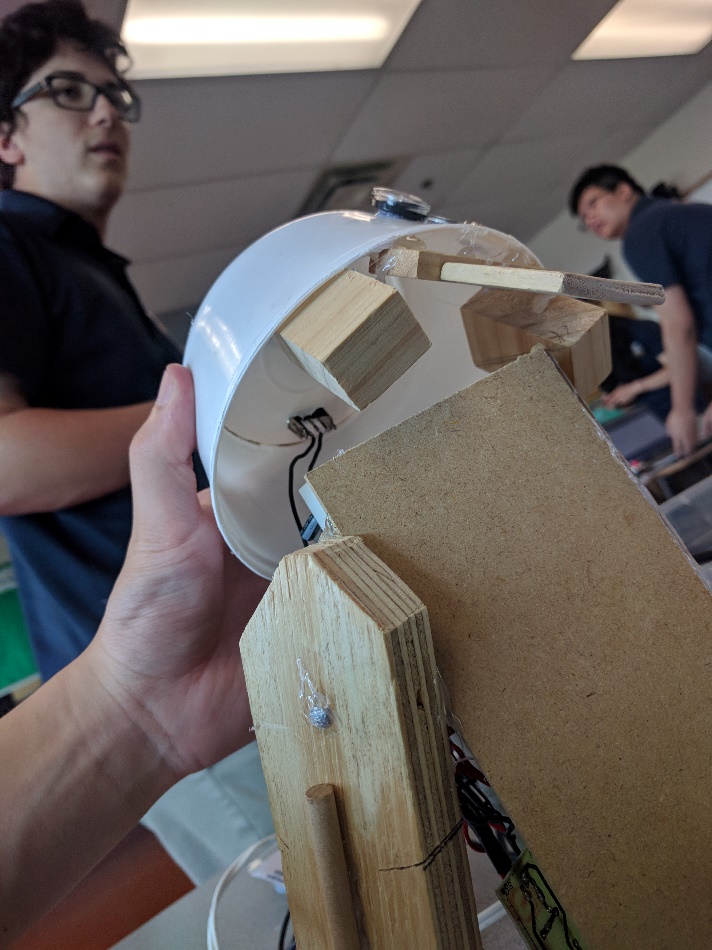
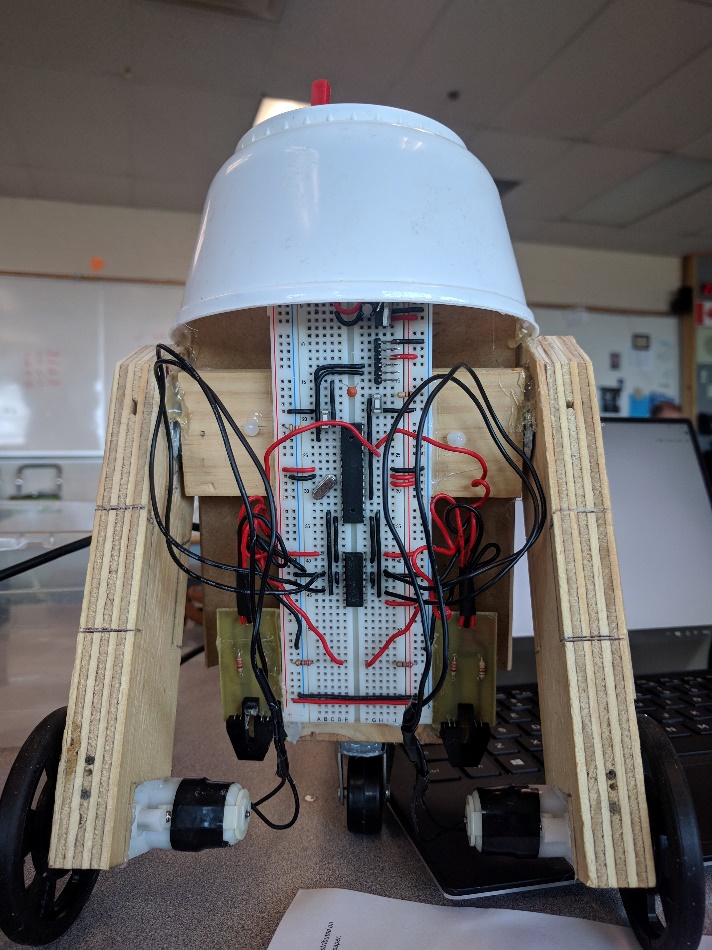
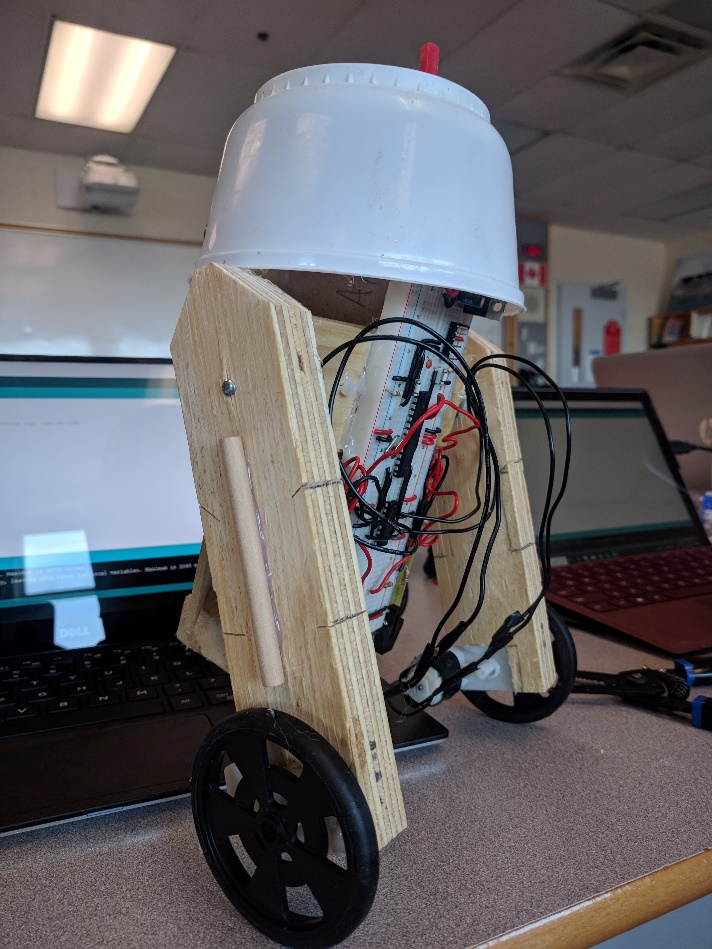
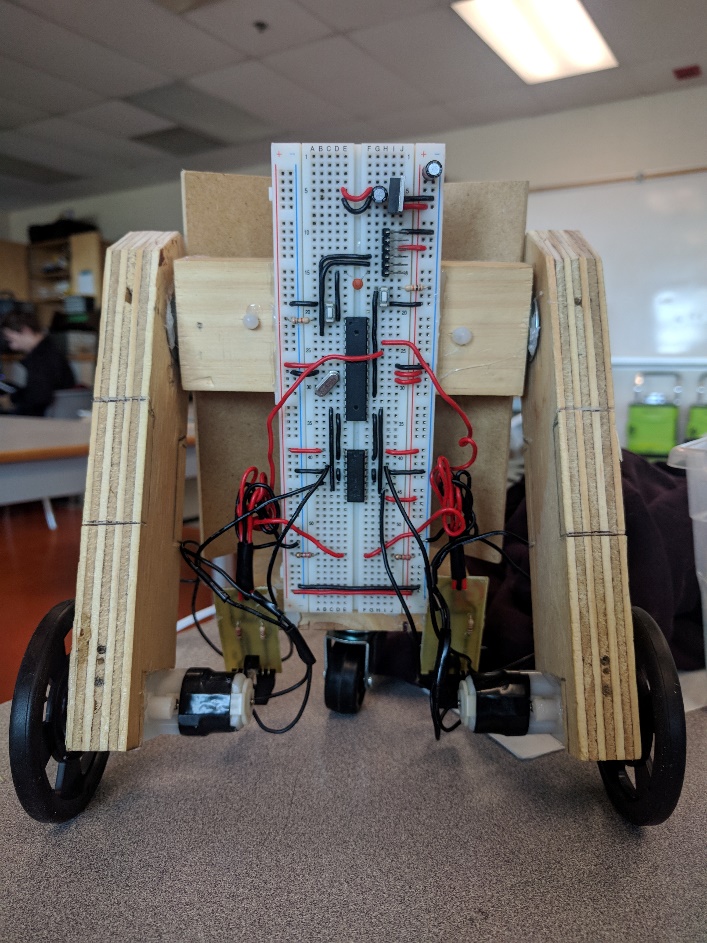
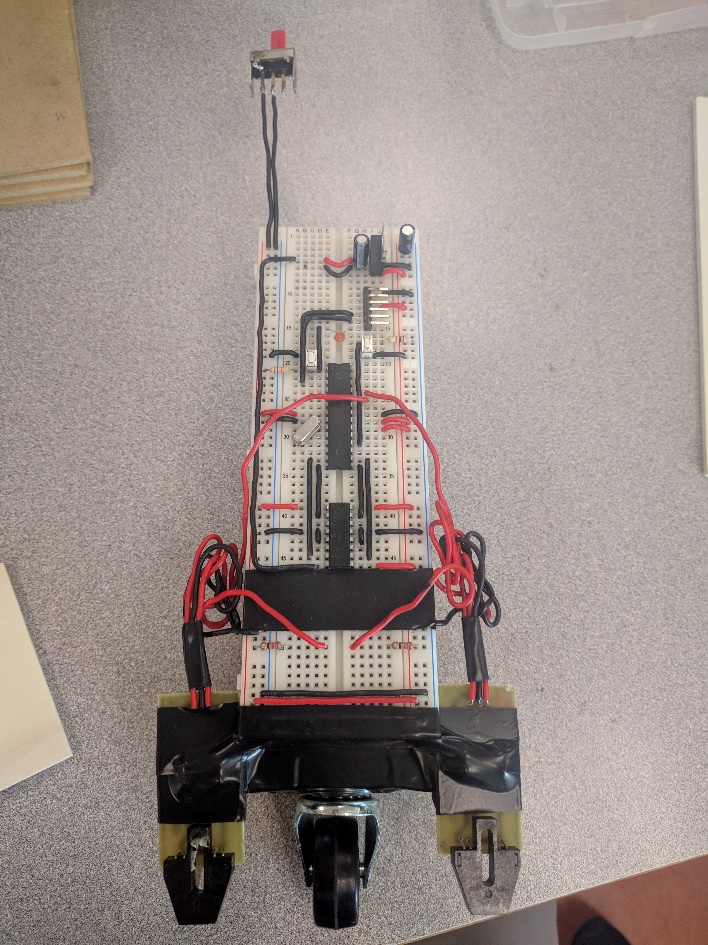
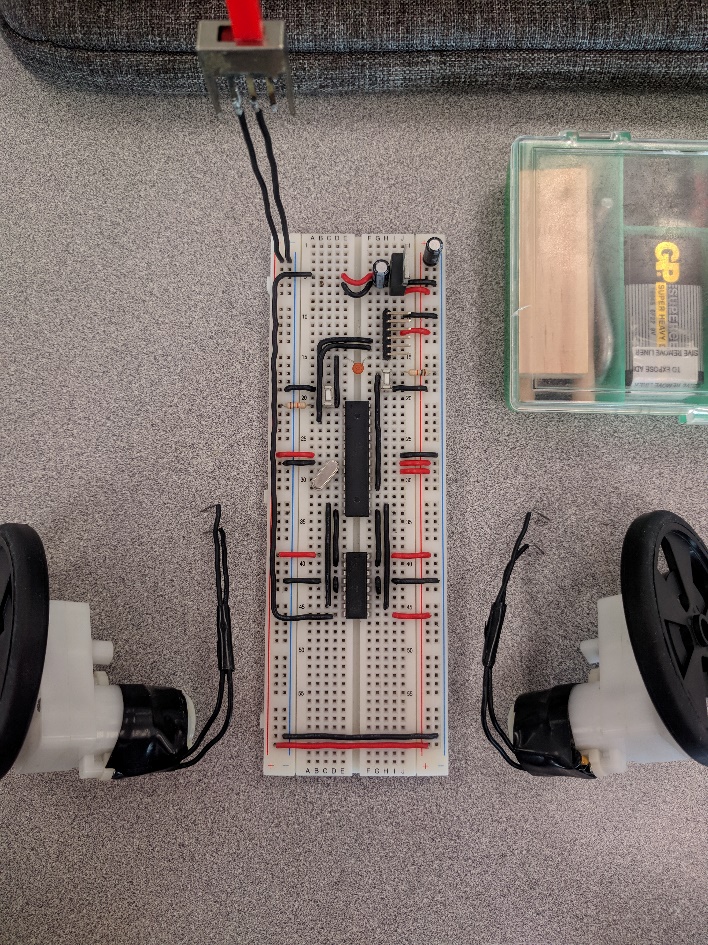
Step 9 – Test out the runs on the track and make programming adjustments to the hardware

Step 10 – Finalize any adjustments and tweaks

**Building / Prototyping**

TIMELINE

Wednesday May 22 - Get Assignment and research what culminating project to make - r2d2 and think bout what parts to use for it  
Thursday May 23 - Relearn how to solder and about the motor chip  
Friday May 24 - Start schematics for the bread board  
Saturday May 25 - Started report for ideas  
Tuesday May 29 - Soldered the switch and added it to the board and schematic. Coded a bit more of the assignment  
Wednesday May 30 - Worked on report   
Thursday May 31 - Coded about half of the code  
Friday June 1 - Finished most of the code, added the button and sensors to the physical board and the schematic  
Monday June 4 - Hammered in nails for wheels to base and taped breadboard to base  
Tuesday June 5 - Building robot and cut wood in workshop  
Wednesday June 6 - Hot glue everywhere that has tape and hammered in nails on arms. Connected arms to middle piece w nails and sawed some wood pieces for body and found head piece  
Thursday June 7 - Hot glued the rest of the pieces and tested code on track and adjusted accordingly. Finished the report and schematics and flow chart  
Friday June 8 - Made final adjustments to code and report



PROBLEMS AND ADJUSTMENTS

Most of the problems were not with the software section, but rather with the hardware with this project. The first problem before even I started to build the project was how I was going to build this project, since R2D2 is a round cylindrical robot, and with wood, I wouldn’t be able to bend it to its shape. I ended up using two pieces of wood planks and hot glued them side by side so it was more of a v shape rather than a circle. I also couldn’t put anything on the back of the robot because that was where all the circuitry was located and since the requirements for the project was to start with pushing a button, I had to have the back exposed for easy access when wanting to run the code every time. I also realized that the switch was kind of useless since the program wouldn’t start without physically pushing the button but I kept it in the design because of the aesthetics. The motors of the arms of my robot isn’t parallel so I had to adjust my code for this problem, since it was affecting the delay times. Some changes that I had to make to the code were the counter time for the black line and the timing for the delays and sensor value numbers. At first, I was going to use a counter for times the robot hits the black line so when it reaches the fourth black line stop, it would simulate a stop and go back again. I found a more efficient and more applicable to any type of track way of coding, which was to use a timer that times how long it takes to get to the next black line and if it’s in a short duration of time like after 5 seconds it goes to the new black line, it will indicate that that is the stop. The timing for the delays is the most changed part of this project since there are so many complications and variables so everyday might be different, like depending on how bright the sunlight is when it comes into the class, since the sensors must read the color values. The delays were changed based on how new the battery was, the weight of the robot, the positioning of the motors and many things like that. I also recently found a problem where I didn’t know why the sensors wouldn’t work no matter what value I put for it but I realized they wouldn’t work because the sensors were too high off the ground so it wouldn’t be able to read the color values. I moved the sensors lower and it solved my problem.

**Final Testing and Demo**

In conclusion, this project improved my practical learning and allowed me to apply what I already know and a little bit of what I learned to create a functioning robot. The various parts of this project took me through the stages of research, planning, and building of a robot, where each stage plays an important part of the overall success of this project which will be a basic for future activities that use the same principles of design.